

## Numerical Modeling of One-Dimensional Binary Solidification with a Mushy Layer Evolution

Daniel Lee<sup>1,\*</sup>, Dmitri Alexandrov<sup>2</sup> and H.-N. Huang<sup>1</sup>

<sup>1</sup> Department of Applied Mathematics, Tunghai University, Taichung, Taiwan 40704.

<sup>2</sup> Ural Federal University, Department of Mathematical Physics, Lenin ave. 51, Ekaterinburg 620083, Russian Federation.

Received 16 September 2010; Accepted (in revised version) 13 May 2011

Available online 27 March 2012

---

**Abstract.** The numerical modeling of a binary solidification with a mushy layer mechanism is considered in this manuscript. The nonlinear coupled system of equations describes the heat and mass diffusions of a one-dimensional spatial variable in the semi-infinite interval. Also formulated is a transformed system in a finite interval. We propose numerical methods for solving the nonlinear system using a threshold strategy based on fixed computation-domain approach. Our calculated results and those from the LeadEx field experiment are well-matched in their tendencies.

**AMS subject classifications:** 35K05, 65-05, 65M06

**Key words:** Direct numerical simulation, Heat and mass transfer, Mushy layer, Solidification, Stefan problem.

---

### 1. Introduction

Binary solidification of liquids has received considerable attention in the literature due to its wide applicability. If the liquid is an alloy (a mixture of two or more components), its crystallization process completely differs from the solidification process of a pure liquid. In particular, various distributions of impurity in the solid phase lead to different mechanical and physical properties of ingots. This phenomenon arises due to the impurity displacements in the melting process by the moving front of the solidification. If the impurity displacement is rather large, the constitutional supercooling originates ahead of the planar solid-liquid interface [23] and, generally speaking, the two-phase zone (mushy region) appears. Moreover, solid nuclei in the form of newly born crystals may evolve in this zone.

Mathematical descriptions of crystallization processes play a very important role in crystal growth [10,26], engineering [33], oceanography [22] and metallurgy [12]. The mathematical models allow for the accurate predictions of the many properties of solids

---

\*Corresponding author. *Email addresses:* danlee@thu.edu.tw (D. Lee), Dmitri.Alexandrov@usu.ru (D. Alexandrov), nhuang@thu.edu.tw (H.-N. Huang)

produced by the melt cooling process. The authors of Ref. [21] developed a full set of thermodynamic equations for a mushy zone, and approximately solved a mush-reduced set of them for the constrained growth of a binary alloy. A more complete solution has since been given in [1] for the steady-state solidification conditions. Nevertheless, solidification with a constant rate is the specific regime.

Generally speaking, the rate of solidification is a function of all operating and physical parameters and also it is a time-dependent function. First of all, the structure of this mushy region depends upon a relation between the kinetics of both the solid-phase formation and the front motion. When the former is much slower than the latter and, thus, the mushy region is almost free from solid elements, the classical Stefan problem provides for a sufficiently acceptable approximation. In the opposite limiting case, the two-phase zone structure is nearly at equilibrium so that its local temperature coincides with the phase transition temperature at a given point. In that situation the solidification is described with the help of Borisov's quasiequilibrium mushy region model [8]. The scenario suggested by Borisov is described by nonlinear heat and mass transfer equations and boundary conditions accounting for moving boundaries [8, 21].

Our previous study [1–5] considerably extended the range of the theoretical application, by solving approximately the Stefan problem in terms of analytical formulae in the special case of simple mushy region for the modeling of moving boundary processes frequently met in geophysics and metallurgy [12, 30, 38]. The solidification problem can be considered as a generalization of the classical two-phase Stefan problem [20]. Nonetheless, much of previous theoretical works have involved substantial approximations to facilitate the derivation of analytical solutions.

There exist various theoretical approaches to the classical Stefan problem in different forms with variable coefficients. The analysis can be based on generalized functions [29], or Green's function via the method of images [34], or an integral formulation and Schauder's fixed point theorem [25] or contraction mapping theory [16]. The arguments of existence of a solution, in many works, rely on functional iterations by finite differencing in the temporal variable. However, many of these assume that the temperatures are kept at zero on the free boundary and the maximum principle applies. This is not true in a mushy layer model, in which undercooling (or supercooling) is allowed. Furthermore, heat and mass diffusions are coupled. To the author's knowledge, there are no published results in a journal regarding computer simulations of this real-life problem [30]. We refer to [19, 37] for further theoretical references in the classical Stefan problem.

Very recently, the advent of numerical methods such as level set methods [28, 31, 35] and more powerful computers have provided for the opportunity to use DNS (direct numerical simulation) of binary solidification. Due to the high nonlinearity shown in the field experiments during the solidification process, we propose in this work numerical procedures to simulate the dynamical evolution of the process based on the mushy layer model [3]. The numerical computation of the Stefan problem has been studied by researchers over decades, e.g., [13, 18, 24] and references therein. Even so, the sheer size of the computational time-step remains the primary difficulty encountered in numerical evolution. As depicted by many researchers, the size of a time-step is around  $10^{-5}$  second

and the time duration from the LeadEx experiment [30] is 3776 minutes. Therefore, we need a very stable algorithm for longer time-frame simulation.

A classical two-phase Stefan problem with a similarity solution [20] is used as a prototype problem to verify our methods and routines. More than five computational models are thoroughly tested and compared in a preliminary study [27]. Finite-difference scheme is adopted, accordingly, for final runs in our DNS described in the current work. Based on the mushy layer model, our computation shows dynamic evolution which is qualitatively consistent with the field experiment [30, 38].

The idea of a model with a mushy layer is used by many authors in different applications from geophysics to metallurgy. For example, the Earth inner core represents a mushy layer ([14, 17]). The model we proposed and the numerical procedures we applied may be helpful in similar applications.

## 2. The mushy layer model for physical problems

We consider the modeling of a binary solidification with a mushy layer mechanism. The system of equations describes the heat and mass diffusions of one-dimensional spatial variable in the semi-infinite interval in the subsection below, and in a finite interval in a later subsection. The physical meanings of variables in the governing equations are described in the nomenclature section.

### 2.1. Mushy layer model in the semi-infinite interval

Continued from the previous work [3], we propose a formulation of a mushy layer model for the Stefan problem which consists of two phases with a mushy layer. The mathematical formulation here is new and suitable for theoretical analysis and computational purpose.

*Unknowns* :  $T(x, t)$ ,  $C(x, t)$ ,  $\varphi(x, t)$ ,  $\varphi_a(t)$ ,  $\varphi_b(t)$ ,  $a(t)$ ,  $b(t)$

*Solid Phase* ,  $0 \leq x < a(t)$

$$\text{PDE: } T_t(x, t) = \alpha_l T_{xx}(x, t). \quad (2.1)$$

$$\text{B.C.: } T(x = 0, t) = T_{\text{air}}(t). \quad (2.2)$$

*Solid-Mush Boundary* ,  $x = a(t)$

$$\text{ODE : } L_v(1 - \varphi_a(t))a_t = k_l T_x(a^-, t) - \left( k_l \varphi_a(t) + k_w(1 - \varphi_a(t)) \right) T_x(a^+, t) \quad (2.3)$$

$$C(a^+)(1 - \varphi_a(t))a_t = -D_w(1 - \varphi_a(t))C_x(a^+). \quad (2.4)$$

$$\text{B.C. : } \varphi(x = a(t), t) = \varphi_a(t). \quad (2.5)$$

*Mushy Layer* ,  $a(t) < x < b(t)$

$$\text{PDE : } \left( \rho_l c_l \varphi + \rho_w c_w (1 - \varphi) \right) T_t = \frac{\partial}{\partial x} \left( \left( k_l \varphi + k_w (1 - \varphi) \right) T_x \right) + L_v \varphi_t, \quad (2.6)$$

$$\frac{\partial}{\partial t} \left( (1 - \varphi) C \right) = \frac{\partial}{\partial x} \left( D_w (1 - \varphi) C_x \right), \quad (2.7)$$

$$T = -mC. \quad (2.8)$$

*Liquid-Mush Boundary* ,  $x = b(t)$

$$\text{ODE : } L_v \varphi_b(t) b_t = \left( k_l \varphi_b(t) + k_w (1 - \varphi_b(t)) \right) T_x(b^-, t) - k_w T_x(b^+, t), \quad (2.9)$$

$$C(b^-) \varphi_b(t) b_t = D_w (1 - \varphi_b(t)) C_x(b^-) - D_w C_x(b^+), \quad (2.10)$$

$$T_x = -mC_x. \quad (2.11)$$

$$\text{B.C. : } \varphi(x = b(t), t) = \varphi_b(t). \quad (2.12)$$

*Liquid Phase* ,  $b(t) < x < +\infty$

$$\text{PDE : } T_t = \alpha_w T_{xx}, \quad (2.13)$$

$$C_t = D_w C_{xx}. \quad (2.14)$$

$$\text{B.C. : } T(x = +\infty, t) = T_{\text{Ocean}}(t), \quad (2.15)$$

$$C(x = +\infty, t) = C_{\text{Ocean}}(t). \quad (2.16)$$

Including the two interface frontals, the above two-phase solidification problem with a mushy layer consists of five subsystems of equations. We note the classical Stefan problem involves the temperature only. While in the proposed mushy layer model, both the heat and mass transfers are considered. The temperature and the salinity variables,  $T$  and  $C$ , are coupled through the two frontals ( $a(t)$ ,  $b(t)$ ) and solid fractions ( $\varphi_a(t)$ ,  $\varphi_b(t)$ ). In a degenerate situation where exactly one of the frontals ( $a(t)$  or  $b(t)$ ) is absent by assuming, say,  $\varphi(x, t) = 1$  or  $0$ , the model is reduced to the classical Stefan two-phase problem. The temperature  $T(x, t)$  and the interface  $a(t)$  (or  $b(t)$ ) are uniquely solvable. The mass diffusion, (2.14), can be determined thereafter with the aid of (2.16) and (2.8) (or (2.11)). Therefore the salinity has no effect on the temperature at all. The mushy layer model then becomes a *significant* extension of the classical Stefan two-phase problem.

Before we discuss numerical methods and experiments in later sections, an alternative is worthy to mention.

## 2.2. Mushy layer model in a finite interval

Many numerical approximations and methods are designed for problems on finite domains. Reusability of routines is very important in practical problem solving. Therefore we convert the previous mushy layer model to one in a finite interval, as driven by our previous investigation into the classical Stefan problem.

We change the variable to convert the governing equations to a system defined in the finite interval  $[-1, 1)$ , resulting in the coordinate transformation and its inversion as described by

$$\xi = \xi(x), \quad x = x(\xi),$$

where  $x$  denotes the spatial variable in the semi-infinite interval  $[0, +\infty)$  and  $\xi$  in the finite interval  $[-1, 1)$ . The temperature in the new system is denoted by  $V(\xi, t)$ , i.e., the relation  $V(\xi, t) = T(x, t)$  holds. An application of chain rule yields

$$T_t = V_t, \quad T_x = \xi_x V_\xi, \quad T_{xx} = (\xi_x)^2 V_{\xi\xi} + \xi_{xx} V_\xi.$$

One such coordinate transform will be discussed in details below.

With  $x_{\text{ref}}$  being a real positive parameter, the transformation and its inversion are given by

$$\begin{aligned} \xi(x) &= 1 - \frac{2x_{\text{ref}}}{x + x_{\text{ref}}} = \frac{x - x_{\text{ref}}}{x + x_{\text{ref}}}, \\ x(\xi) &= x_{\text{ref}} \frac{1 + \xi}{1 - \xi}. \end{aligned}$$

This maps the triple  $x \in \{0, x_{\text{ref}}, +\infty\}$  to  $\xi \in \{-1, 0, 1\}$ . We note for a uniform grid in the finite interval  $[-1, 1)$  with mesh size  $\varepsilon = 2/n$ , the right-most interior node,  $\xi = 1 - \varepsilon$ , corresponds to  $x = (n - 1)x_{\text{ref}}$  in the semi-infinite interval.

Using the relation  $V(\xi, t) = T(x, t)$  and noting that

$$\begin{aligned} T_x &= \frac{(1 - \xi)^2}{2x_{\text{ref}}} V_\xi, \\ T_{xx} &= \frac{(1 - \xi)^4}{4x_{\text{ref}}^2} V_{\xi\xi} - \frac{(1 - \xi)^3}{2x_{\text{ref}}^2} V_\xi, \end{aligned}$$

we obtain the transformed problem with the temporal variable  $\tau$  instead of  $t$ .

*Unknowns* :  $V(\xi, \tau)$ ,  $C(\xi, \tau)$ ,  $\psi(\xi, \tau)$ ,  $\psi_A(\tau)$ ,  $\psi_B(\tau)$ ,  $A(\tau)$ ,  $B(\tau)$ .

*Solid Phase*,  $-1 \leq \xi < A(\tau)$

$$\text{PDE :} \quad V_\tau(\xi, \tau) = \alpha_l \left( \frac{(1 - \xi)^4}{4} V_{\xi\xi} - \frac{(1 - \xi)^3}{2} V_\xi \right), \quad (2.17)$$

$$\text{B.C. :} \quad V(\xi = -1, \tau) = V_{\text{Air}}(\tau). \quad (2.18)$$

*Solid-Mush Boundary*,  $\xi = A(\tau)$

$$\text{ODE : } \quad L_v(1 - \psi_A(\tau))A_\tau = \frac{(1 - \xi)^2}{2} \left( k_i V_\xi(A^-(\tau), \tau) - (k_i \psi_A(\tau) + k_w(1 - \psi_A(\tau))) V_\xi(A^+(\tau), \tau) \right), \quad (2.19)$$

$$C(1 - \psi_A(\tau))A_\tau = -D_w(1 - \psi_A(\tau)) \frac{(1 - \xi)^2}{2} C_\xi(\xi, \tau). \quad (2.20)$$

$$\text{B.C. : } \quad \psi(\xi = A(\tau), \tau) = \psi_A(\tau). \quad (2.21)$$

*Mushy Layer*,  $A(\tau) < \xi < B(\tau)$

$$\begin{aligned} \text{PDE : } \quad & \left( \rho_i c_i \psi + \rho_w c_w (1 - \psi) \right) V_\tau \\ & = \frac{(1 - \xi)^2}{2} \frac{\partial}{\partial \xi} \left( [k_i \psi + k_w(1 - \psi)] \frac{(1 - \xi)^2}{2} V_\xi \right) + L_v \psi_\tau, \end{aligned} \quad (2.22)$$

$$\frac{\partial}{\partial \tau} [(1 - \psi) C] = \frac{(1 - \xi)^2}{2} \frac{\partial}{\partial \xi} \left( D_w (1 - \psi) \frac{(1 - \xi)^2}{2} C_\xi \right), \quad (2.23)$$

$$V = -mC. \quad (2.24)$$

*Liquid-Mush Boundary*,  $\xi = B(\tau)$

$$\text{ODE : } \quad L_v \psi_B(\tau) B_\tau = \frac{(1 - \xi)^2}{2} \left( (k_i \psi_B(\tau) + k_w(1 - \psi_B(\tau))) V_\xi(B^-, \tau) - k_w V_\xi(B^+, \tau) \right), \quad (2.25)$$

$$C \psi_B(\tau) B_\tau = D_w(1 - \psi_B(\tau)) \frac{(1 - \xi)^2}{2} C_\xi(B^-, \tau) - D_w \frac{(1 - \xi)^2}{2} C_\xi(B^+, \tau), \quad (2.26)$$

$$V_\xi = -mC_\xi. \quad (2.27)$$

$$\text{B.C. : } \quad \psi(\xi = B(\tau), \tau) = \psi_B(\tau). \quad (2.28)$$

*Liquid Phase*,  $B(\tau) < \xi < 1$

$$\text{PDE : } \quad V_\tau = \alpha_w \left( \frac{(1 - \xi)^4}{4} V_{\xi\xi} - \frac{(1 - \xi)^3}{2} V_\xi \right), \quad (2.29)$$

$$C_\tau = D_w \left( \frac{(1 - \xi)^4}{4} C_{\xi\xi} - \frac{(1 - \xi)^3}{2} C_\xi \right). \quad (2.30)$$

$$\text{B.C. : } \quad V(\xi = 1, \tau) = V_{\text{Ocean}}(\tau), \quad (2.31)$$

$$C(\xi = 1, \tau) = C_{\text{Ocean}}(\tau). \quad (2.32)$$

Here,  $A(\tau)$  and  $B(\tau)$  denote the frontal positions corresponding to the solid-mush boundary  $a(t)$  and liquid-mush boundary  $b(t)$ , respectively. The associated solid fractions are  $\psi_A(\tau)$  and  $\psi_B(\tau)$ .

More detailed information regarding transformations can be found in [9, 11, 36]. We

Table 1: Physical Properties : (unit in cm/day)

Parameters	Values
$T_{Ocean}$	$-1.7^{\circ}\text{C}$
$T_b$	$-1.97^{\circ}\text{C}$
$L_v$	$3.5556 \cdot 10^{-3} \text{ W} \cdot \text{day} \cdot \text{cm}^{-3}$
$k$	0.68
$k_l$	$2.026 \cdot 10^{-2} \text{ W} \cdot \text{cm}^{-1} \cdot ^{\circ}\text{C}^{-1}$
$k_w$	$5.596682 \cdot 10^{-3} \text{ W} \cdot \text{cm}^{-1} \cdot ^{\circ}\text{C}^{-1}$
$D_w$	$1.0368 \text{ cm}^2 \cdot \text{day}^{-1}$
$m$	$0.05236 ^{\circ}\text{C}^{-1} \cdot \text{psu}^{-1}$
$\rho_l$	$9.17 \cdot 10^{-4} \text{ kg} \cdot \text{cm}^{-3}$
$\rho_w$	$1.0 \cdot 10^{-3} \text{ kg} \cdot \text{cm}^{-3}$
$C_{Ocean} = C_{\infty}$	34.5 psu
$c_l$	$0.023 \text{ W} \cdot \text{day} \cdot \text{kg}^{-1} \cdot ^{\circ}\text{C}^{-1}$
$c_w$	$0.048 \text{ W} \cdot \text{day} \cdot \text{kg}^{-1} \cdot ^{\circ}\text{C}^{-1}$

have assumed constant boundary values at the far field (2.15), (2.16), (2.31) and (2.32) for our current work. Other expressions may be used in different applications.

### 3. Computational models and numerical procedures

For a clear picture of data flow in the computations, we summarize the subsystems and variables in Table 2. Many details are implied therein and will be explained in the following.

#### 3.1. Variants of computational models

In computation of the model in a semi-infinite interval, we choose to truncate the physical domain to a finite interval and apply *approximate* boundary values at the far end. In contrast, the right-end computational boundary in the finite interval model represents a regular singular point (Eqs. (2.29) and (2.30)). In addition to choosing between the semi-infinite or finite interval (continuous) models, the discretized system can be based on either a fixed or moving coordinate. The former could be further divided into static or dynamic grid subcases. All of these were thoroughly tested and compared in [27] for the prototype problem using FD (finite difference) or DM (differentiation matrix). In Table 3, we summarize the computational models and discretizations in our previous work on the prototype problem.

Once the discretizations of the geometry and the equations are specified, we consider the following issues in solving the resulting discrete algebraic system:

1. *Block-iterative methods*. This can be performed in a forward, backward or parallel manner.

Table 2: Five subsystems, equations, primitive and derived variables. A derived variable may be needed as input or updated as output.

Subsystem	D.E.	Primitive	Derived	B.V. or I.V.
solid	Eq. (2.1)	$T$	$T_a$	Eq. (2.2) and $T_a$
solid-mush	Eqs. (2.3-4)	$a, \varphi_a$	$T_x(a^-), T_x(a^+), C_{a^+}, C_x(a^+)$	Eq. (2.5)
mush	Eqs. (2.6-8)	$T, C, \varphi$	$T_a, T_b, \varphi_{a^+}, \varphi_{b^-}, C_{a^+}, C_{b^-}$	the six derived
liquid-mush	Eqs. (2.9-10)	$b, \varphi_b$	$T_x(b^-), T_x(b^+), C_x(b^-), C_x(b^+), C_{b^-}$	Eqs. (2.11-12)
liquid	Eqs. (2.13-14)	$T, C$	$T_b, C_{b^+}$	Eqs. (2.15-16) and two derived

2. *Types of spatial finite differences.* Both explicit and implicit differences (or spectral differentiations) are allowed.
3. *The linear solver of each block.* The options are tridiagonal Gaussian Elimination and tridiagonal iterative methods. In the spectral DM approach, the *BCGstab* method in dense matrix version is taken.

Further details and trade-offs about various orders of interpolations and extrapolations can be found in [27].

We note more variants were implemented and tested than listed in Table 3, but only the more successful ones are reported. This cited reference discussed the first five cases ( in semi-infinite interval ) in detail. The computations in the finite interval models are less accurate than computations in the semi-infinite interval models. Although, the DM approach yields more accurate results than other models in the finite interval.

It was observed in [27] that, temporal and spatial FD method on fixed coordinates and static grid with a proposed threshold strategy, in the semi-infinite interval, out-performed all the other combinations. Indeed, we were successful in obtaining stable long-term simulation results in the current mushy layer evolution using this combination of computational model only. All of the other combinations failed.

However, the design and implementation of the same idea in the current study is more complicated than those in the prototype problem.

Table 3: Discretization of the geometry and equations of experiments in a prototype study.

Domain	Coordinate	Grid	Discretization of Eqs	Spatial Differentiation
semi-infinite	fixed	static, uniform	FD-FD	implicit
semi-infinite	fixed	dynamic, sub-uniform	FD-FD	implicit
semi-infinite	fixed	dynamic, sub-uniform	FD-FD	explicit
semi-infinite	moving	uniform	FD-FD	explicit
semi-infinite	fixed	dynamic, sub-uniform	FD-DM	implicit
finite	fixed	dynamic, sub-uniform	FD-DM	implicit
finite	fixed	dynamic, sub-uniform	FD-FD	implicit
finite	fixed	dynamic, sub-uniform	FD-FD	explicit

Next we describe the discretizations of the geometry and equations in the current study followed by methods of solving the discrete system later.

### 3.2. The grid arrangement

The computational domain is denoted by  $[z_{min}, z_{max}]$ . Piecewise-uniform nodes are deployed in the time-march procedure. The following relations are maintained.

$$\begin{aligned} z_{min} = z_0 &< z_1 < \dots < z_{n_1} < a(t), \\ a(t) &< z_{n_1+1} < \dots < z_{n_1+n_2} < b(t), \\ b(t) &< z_{n_1+n_2+1} < \dots < z_{n_1+n_2+n_3} < z_{n+1} = z_{max}, \end{aligned}$$

in the solid zone, the mush layer and the liquid zone, respectively. These, including the two frontals, are the computational nodes in the FD or the DM approach. The three subsystems of PDEs are each treated as a two-point boundary value problem with Dirichlet data. The FD nodes are almost uniform in each region, except near the two frontals  $a(t)$  and  $b(t)$ . The number of interior nodes are  $n_1$ ,  $n_2$  and  $n_3$  for the three regions, respectively. The total number of interior nodes is kept a constant,  $n = n_1 + n_2 + n_3$ , through the whole simulation. The memory for the two frontals are allocated separately. However, the size of each subsystem may vary as one of the frontals crosses a node. Another two subsystems of a DAE type govern the evolutions of the positions and temperatures at the two frontals. As time marches, the frontal(s) may move too close to a node. This yields a loss of significant digits in the computations of some interpolated values. The proposed threshold strategy is to re-set the frontal position properly in such a critical situation.

The continuous system, in the semi-infinite or finite interval, can be solved by numerical procedures in an algorithmically unified manner. See the following subsection for derivation.

### 3.3. Numerical methods and procedures

The application consists of five subsystems (Table 2) : the solid (ice) zone, the solid-mush interface, the mushy layer, the liquid-mush interface and the liquid (water) zone.

We propose three solution methods. The time marching technique is adopted with a two-level explicit or implicit FD in the temporal variable. The spatial derivatives can be resolved by either finite differences or spectral differentiations. At each time step, the coupled nonlinear system of differential equations yields a nonlinear algebraic system through either continuous linear spline model via finite differencing or spectral differentiation at Chebyshev nodes. Three solution methods are possible for general use. A delineation of the algorithms follows:

**Algorithm 1:**1. *Forward shoot.*

Solve the coupled subsystems {solid, interface  $a$ } and {mush, interface  $b$ }, and the liquid zone in order.

The two coupled subsystems are solved iteratively up to a specified tolerance or maximum iteration count.

The liquid zone equations are solved once.

2. *Backward shoot.*

Solve the coupled subsystems {liquid, interface  $b$ }, {mush, interface  $a$ }, and the solid zone in order.

The two coupled subsystems are solved iteratively up to a specified tolerance or maximum iteration count.

The solid zone equation is solved once.

3. *Parallel shoot.*

Solve {solid, mush, liquid} and {interfaces  $a, b$ } iteratively.

More details of the calculation at a single time step are described below.

**3.3.1. Parallel shoot**

This is similar to a block version of Jacobi iterations.

**First stage.** Solve in data-parallel fashion three out of the five subsystems.

- **Solid phase.** Here we have a linear diffusion, Eq. (2.1), with constant coefficients and Dirichlet data. New temperatures are obtained at interior nodes and the quantity  $T_x(a^-)$  is calculated as a post-processing.
- **Liquid phase.** The subsystem consists of two decoupled linear diffusion equations, (2.13) and (2.14), representing the heat and mass transfers respectively. New values of the two primitive variables are obtained by solving two tri-diagonal systems. Post-process to update  $T_x(b^+)$  and  $C_x(b^+)$ .
- **Mush layer.** We assume the problem is non-degenerate such that the relation

$$0 < \varphi(x, t) < 1 \text{ in } a(t) < x < b(t)$$

is valid at all times. This implies

$$0 < \min(\rho_I c_I, \rho_W c_W) \leq \rho_I c_I \varphi + \rho_W c_W (1 - \varphi) \leq \max(\rho_I c_I, \rho_W c_W)$$

Therefore Eqs. (2.6) and (2.7) are explicit in the temporal derivatives and we can apply spatial central differences with modifications near the two frontals. The three primitive variables are thus updated at interior nodes. Post-processings are carried out to derive approximate values of  $\varphi_{a^+}$ ,  $\varphi_{b^-}$ ,  $C_{a^+}$  and  $C_{b^-}$  as tabulated in Table 2.

At the end of the first stage, new values of the temperatures  $T_a$  and  $T_b$  are calculated by two-sided interpolations.

**Second stage.** The two frontals and the solid fractions ( $a(t)$ ,  $b(t)$ ,  $\varphi_a$ ,  $\varphi_b$ ) are to be determined.

- **Evolution at the solid-mush boundary  $a(t)$ .** First note  $0 < 1 - \varphi_a < 1$  and that the subsystem equations (2.3) and (2.4) can be treated as algebraic equations with all the needed derived variables already updated. Actually, the quantities  $a_t$  and  $\varphi_a(t)$  can be obtained in closed form expressions.
- **Evolution at the liquid-mush boundary  $b(t)$ .** The situation here is similar to the above. With the required derived variables ready, a linear combination of the ODEs (see (2.10) and (2.11)) eliminates the  $b_t$  term and yields a closed form expression of  $\varphi_b(t)$ . The variable  $b_t$  is also obtained explicitly.

At the end of the second stage, we calculate the new frontals  $a(t + \Delta t)$  and  $b(t + \Delta t)$ . Methods other than the Euler's may be applied as well. One may optionally repeat the two stages until tolerance is achieved. However, full implicitness is very costly.

At this moment, it is of the utmost importance to decide whether or not to call for the threshold strategy, which resembles at the very least a linear-order continuation of the continuous process.

At the end of a time step, we update all the relevant geometric and physical variables. An expensive reset of the whole grid is necessary if at least one of the frontals crosses a node as implied by the threshold strategy.

We give brief descriptions of the other two solution methods below.

### 3.3.2. Forward shoot

The five components are grouped and solved in a *forward* order.

- Step 1: Use possibly time-dependent boundary data at the sea level and freeze the boundary values on the solid-mush interface, perform the following until conditions are satisfied.

Step 1a: Solve the solid zone Eq. (2.1). Post-process to calculate one-sided gradients of the temperatures and concentrations.

Step 1b: Solve Eqs. (2.3) and (2.4) in the solid-mush interface to update  $a_t$  and  $\varphi_a$ . Post-process to update the frontal  $a$ .

- Step 2: With the existing boundary values on the liquid-mush interface and the calculated boundary values on the solid-mush interface from Step 1b, we go on to do the following.
  - Step 2a : Solve the mushy layer Eqs. (2.6)– (2.8).
  - Step 2b : Solve the liquid-mush interface Eqs. (2.9)– (2.11).
- Step 3: Solve the liquid zone Eqs. (2.13) and (2.14) using the boundary values which can be derived after Step 2b.

### 3.3.3. Backward shoot

The five components are grouped and solved in a *backward* order.

- Step 1: Use possibly time-dependent boundary data at the far field and freeze the boundary values on the liquid-mush interface, perform the following until satisfaction:
  - Step 1a: Solve, iteratively, the liquid zone Eqs. (2.13) and (2.14). Post-process to calculate one-sided gradients of the temperatures and concentrations at frontal  $b$ .
  - Step 1b: Solve the liquid-mush interface Eqs. (2.9), (2.10), and (2.11) , to update  $b_t$  and  $\varphi_b$ . Post-process to update the frontal  $b$ .
- Step 2: With the existing boundary values on the solid-mush interface and the calculated boundary values on the liquid-mush interface from Step 1b, we go on to do the following.
  - Step 2a : Solve the mushy layer Eqs. (2.6), (2.7) and (2.8) .
  - Step 2b : Solve the solid-mush interface Eqs. (2.3) and (2.4).
- Step 3: Solve the solid zone Eq. (2.1) using the boundary values which can be derived after Step 2b.

### 3.3.4. Pros and cons of the three solution methods

If full implicitness is implemented, the three methods seem to produce identical results up to truncation errors. However, we observed in a trial run that the *wall-clock cpu-time* took as many as fifty times more than using the simple one-sweep shooting. In view of the computational cost, the authors conducted the simulations with one sweep of each method for the final run. The (incomplete) parallel shoot was preferred for a theoretical reason. In the situation of the prototype problem, the one-point free boundary can be expressed as a solution to an equivalent integral equation in terms of the temperature distribution (together with the initial and boundary data), while the temperature itself is derived by a limit process with refined temporal meshes [15]. In our numerical study, the one-sweep parallel shoot also shows better stability than the forward and backward shoot.

We note the different groupings of combinations of the five subsystems are natural with regard to data (boundary values) communications in the context of domain decomposition and parallel computing. The boundary conditions, i.e., Eqs. ((2.2), (2.5), (2.12), (2.8), (2.11), (2.15), and (2.16)) are applied when needed and updated when appropriate.

### 3.3.5. A threshold strategy

The usual computation via static grid in a fixed coordinate approach is prone to error [13]. When an interface front approaches a computational node, some approximations (finite differences or interpolations) become close to being singular and are sensitive to truncation errors. The propagation of errors can be fatal in a task of long-term simulation. To handle this problem, we designed a threshold strategy in which a frontal position, when approaching a computational node and getting close within a threshold, is re-assigned across the node appropriately in the direction of current frontal velocity. House-keepings are carried out accordingly, i.e., all relevant variables including the clock time are modified properly after a time march. This may yield, at the time step, a linearly compatible continuation of the physical process, provided that the threshold is roughly at the scale of the motion at one computational time step. We mention that the implementation of the threshold strategy is much more involved than that of the classical Stefan problem [27], and that the scheme is a simple variant of the general level set method [28].

### 3.3.6. Some other issues

Here we mention several things which may also be important.

- **Extrapolation** When solving the two frontal dynamics, we need one-sided derivatives in the Stefan condition at the two interfaces. Various one-sided schemes are designed for this purpose. However, it was observed in the prototype study that a four-point third-order scheme was preferred for the semi-infinite interval equations, while linear order extrapolation works best for the current mush layer model. This indicates that numerical stability is, above all, the most critical issue.
- **Interpolation** In the moving and the dynamic fixed grid approaches, interpolations are needed at the end of each time step. It is also needed in case of a node-cross in a static fixed grid approach. Both the *natural* and the *not-a-knot* cubic spline interpolations are designed for the current second-order PDE system. The latter are fourth-order accurate in theory, as proved by de Boor [6, 7]. Although, we found no significant differences in choosing either of the spline interpolants in long-term simulations with fine spatial meshes deployed. Numerical data discussed later were obtained via *not-a-knot* cubic spline.

We mention that high order *ENO* or *WENO* type scheme [32] may be worth a try, at the trade-offs of code efficiency. Based on our preliminary study, we focus below on spatial FD in the semi-infinite interval with a fixed coordinate and static grid.

## 4. Simulations and Discussions

Twenty-one DNS', as described in Table 4, are carried out using the semi-infinite interval model in a fixed coordinate via a static grid with the proposed threshold strategy. Results from these simulations are compared in various ways, as shown in subsequent tables. Input data can be found in Figs. 1 and 2, while calculated evolutions of the two frontal positions and temperatures in Figs. 3 and 4, and evolutions of temperatures at various depths in Fig. 5, which match qualitatively the previous experimental report [30, 38]. Final time profiles of the temperature, the concentration and the solid-fraction are shown in Fig. 6. The threshold used in simulations reported here is  $5.0e-08$ . The details are laid out in the subsections.

### 4.1. The computation

We truncated the semi-infinite interval and worked on a finite interval of length 30 centimeters. Various simulations were done with spatial resolution of 0.1 and 0.05 centime-

Table 4: Direct numerical simulations and parameters. Here UD (CD) denotes upwind (central) difference, and method F (B,P) denotes forward (backward,parallel) shoot.

Run	$n_z$	$n_t$	FD	Initial solid fraction	Extrapolation order	Solution Method
1	300	2e+07	UD	profile 1	1	F, B, P
2	300	2e+07	UD	profile 2	1	F, B, P
3	300	2e+07	UD	profile 3	1	F, B, P
4	300	2e+07	UD	profile 4	1	F, B, P
5	300	2e+07	UD	profile 3	2	F, B, P
6	600	8e+07	UD	profile 3	1	F, B, P
7	300	2e+07	CD	profile 3	1	F, B, P

Table 5: Max deviations of the evolutions of first frontal position with four different initializations and three methods.

$Init_1$			$Init_2$			$Init_3$			$Init_4$		
1F	1B	1P	2F	2B	2P	3F	3B	3P	4F	4B	4P
0.0e-0	1.4e-8	1.3e-9	3.3e-3			4.4e-3			9.2e-3		
1.4e-8	0.0e-0	1.5e-8	3.3e-3			4.4e-3			9.2e-3		
1.3e-9	1.5e-8	0.0e-0	3.3e-3			4.4e-3			9.2e-3		
3.3e-3			0.0e-0	1.4e-8	1.3e-9	1.1e-3			5.8e-3		
3.3e-3			1.4e-8	0.0e-0	1.5e-8	1.1e-3			5.8e-3		
3.3e-3			1.3e-9	1.5e-8	0.0e-0	1.1e-3			5.8e-3		
4.4e-3			1.1e-3			0.0e-0	1.5e-8	1.3e-9	5.8e-3		
4.4e-3			1.1e-3			1.5e-8	0.0e-0	1.6e-8	5.8e-3		
4.4e-3			1.1e-3			1.3e-9	1.6e-8	0.0e-0	5.8e-3		
9.2e-3			5.8e-3			5.8e-3			0.0e-00	1.0e-08	9.0e-10
9.2e-3			5.8e-3			5.8e-3			1.0e-08	0.0e-00	1.1e-08
9.2e-3			5.8e-3			5.8e-3			9.0e-10	1.1e-08	0.0e-00

Table 6: Max deviations of the evolutions of second frontal position with four different initializations and three methods.

<i>Init<sub>1</sub></i>			<i>Init<sub>2</sub></i>			<i>Init<sub>3</sub></i>			<i>Init<sub>4</sub></i>		
1F	1B	1P	2F	2B	2P	3F	3B	3P	4F	4B	4P
0.0e-0	1.9e-7	6.2e-8	4.9e-3			8.7e-3			6.6e-3		
1.9e-7	0.0e-0	1.3e-7	4.9e-3			8.7e-3			6.6e-3		
6.2e-8	1.3e-7	0.0e-0	4.9e-3			8.7e-3			6.6e-3		
4.9e-3			0.0e-0	2.0e-7	6.8e-8	4.1e-3			6.1e-3		
4.9e-3			2.0e-7	0.0e-0	1.4e-7	4.1e-3			6.1e-3		
4.9e-3			6.8e-8	1.4e-7	0.0e-0	4.1e-3			6.1e-3		
8.7e-3			4.1e-3			0.0e-0	2.1e-7	7.0e-8	7.1e-3		
8.7e-3			4.1e-3			2.1e-7	0.0e-0	1.4e-7	7.1e-3		
8.7e-3			4.1e-3			7.0e-8	1.4e-7	0.0e-0	7.1e-3		
6.6e-3			6.1e-3			7.1e-3			0.0e-0	1.9e-7	6.3e-8
6.6e-3			6.1e-3			7.1e-3			1.9e-7	0.0e-0	1.3e-7
6.6e-3			6.1e-3			7.1e-3			6.3e-8	1.3e-7	0.0e-0

ters. The number of computational spatial nodes, denoted by  $n_z$  in tables, were roughly 300 and 600, and the total time steps in marching, denoted by  $n_t$ , were twenty- and eighty-million, respectively. The corresponding results are reported in columns labeled as 1F, 1B, 1P, ..., 7F, 7B, and 7P. The boundary temperature at the sea level, from [30, 38], are shown in Fig. 2. Initial data are assumed linear or piecewise linear as shown in Fig. 1. We have enforced obvious algebraic consistencies derived from the associated differential relations.

At the interfaces, the computed frontal positions and the frontal temperatures are compared among the different simulations. Various discrete norms of deviations in the com-

Table 7: Max deviations of the evolutions of first frontal temperature with four different initializations and three methods.

<i>Init<sub>1</sub></i>			<i>Init<sub>2</sub></i>			<i>Init<sub>3</sub></i>			<i>Init<sub>4</sub></i>		
1F	1B	1P	2F	2B	2P	3F	3B	3P	4F	4B	4P
0.0e-0	8.2e-6	2.5e-7	2.5e-2			4.5e-2			8.7e-2		
8.2e-6	0.0e-0	8.2e-6	2.5e-2			4.5e-2			8.7e-2		
2.5e-7	8.2e-6	0.0e-0	2.5e-2			4.5e-2			8.7e-2		
2.5e-2			0.0e-0	8.6e-6	5.0e-7	2.2e-2			8.2e-2		
2.5e-2			8.6e-6	0.0e-0	8.3e-6	2.2e-2			8.2e-2		
2.5e-2			5.0e-7	8.3e-6	0.0e-0	2.2e-2			8.2e-2		
4.5e-2			2.2e-2			0.0e-0	8.3e-6	2.3e-6	9.5e-2		
4.5e-2			2.2e-2			8.3e-6	0.0e-0	8.3e-6	9.5e-2		
4.5e-2			2.2e-2			2.3e-6	8.3e-6	0.0e-0	9.5e-2		
8.7e-2			8.2e-2			9.5e-2			0.0e-0	8.3e-6	9.6e-7
8.7e-2			8.2e-2			9.5e-2			8.3e-6	0.0e-0	8.3e-6
8.7e-2			8.2e-2			9.5e-2			9.6e-7	8.3e-6	0.0e-0

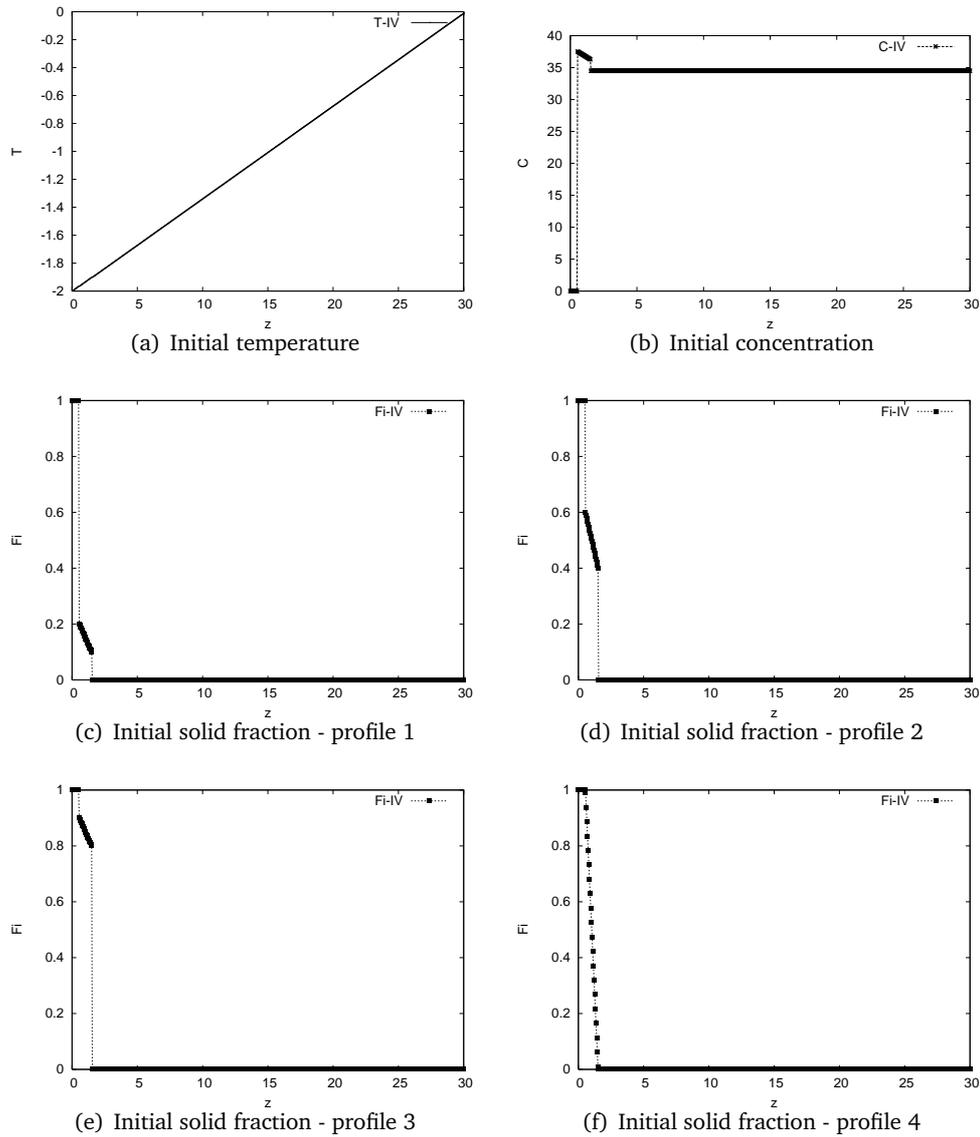


Figure 1: Input data of the direct numerical simulations.

puted results, at two-hundred intermediate time steps, are recorded with some selected data shown in tables. The Figs. 3 to 6 are results from test run 6 with *one-sweep parallel shoot*. We note there are no wiggles near the frontals with our usage of upwind differences. In case of central differencing, wiggles did appear in our previous experiment. Fig. 5 shows the data of computational results and LeadEx experiment at various depths. Although the deviation grows as the depth increases, their tendencies are well-matched.

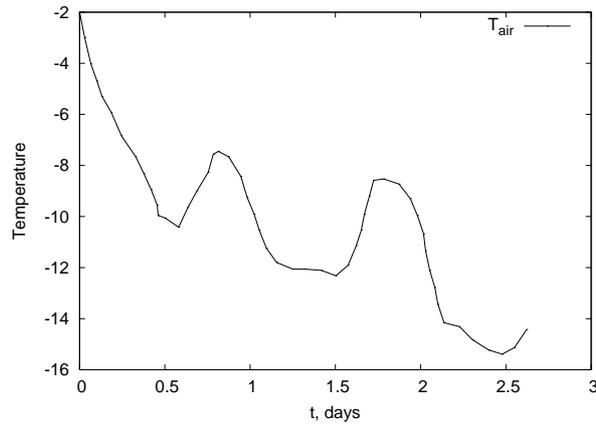


Figure 2: Sea level temperature.

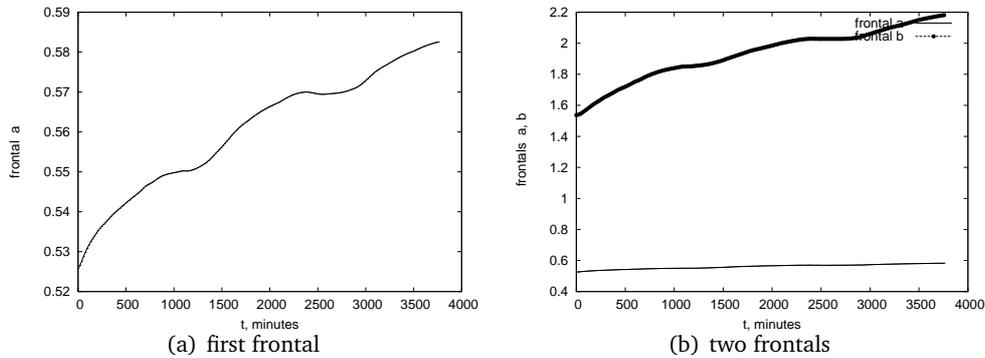


Figure 3: Evolution of the two frontal positions.

### 4.2. Observations

Results of the first twelve simulations in Table 4, with four different initializations and three solution methods, are compared pairwise. The time march procedure takes twenty-million time steps in a single run. Calculated results including the two frontal positions and temperatures are recorded at two-hundred intermediate time steps. Deviations of these quantities are then measured in the *discrete maximum, normalized 1- and 2-norms*. Some are shown in Tables 5 to 8 with rounding at the first digit in float precision format. These matrices are twelve-by-twelve, and better viewed as four-by-four block matrices with each block an ordinary three-by-three matrix. These main-diagonal blocks reveal the following.

Table 8: Max deviations of the evolutions of second frontal temperature with four different initializations and three methods.

$Init_1$			$Init_2$			$Init_3$			$Init_4$		
1F	1B	1P	2F	2B	2P	3F	3B	3P	4F	4B	4P
0.0e-0	1.6e-5	5.2e-6	1.9e-1			2.2e-1			2.2e-1		
1.6e-5	0.0e-0	1.0e-5	1.9e-1			2.2e-1			2.2e-1		
5.2e-6	1.0e-5	0.0e-0	1.9e-1			2.2e-1			2.2e-1		
1.9e-1			0.0e-0	1.6e-5	5.5e-6	1.4e-1			2.1e-1		
1.9e-1			1.6e-5	0.0e-0	1.5e-5	1.4e-1			2.1e-1		
1.9e-1			5.5e-6	1.5e-5	0.0e-0	1.4e-1			2.1e-1		
2.2e-1			1.4e-1			0.0e-0	1.6e-5	5.8e-6	2.4e-1		
2.2e-1			1.4e-1			1.6e-5	0.0e-0	1.1e-5	2.4e-1		
2.2e-1			1.4e-1			5.8e-6	1.1e-5	0.0e-0	2.4e-1		
2.2e-1			2.1e-1			2.4e-1			0.0e-0	1.6e-5	5.3e-6
2.2e-1			2.1e-1			2.4e-1			1.6e-5	0.0e-0	1.1e-5
2.2e-1			2.1e-1			2.4e-1			5.3e-6	1.1e-5	0.0e-0

Table 9: Convergence of 2-norm deviations of the evolution of first frontal position with initialization 3.

dist <sub>2</sub>	$n_z = 300$			$n_z = 600$		
	3F	3B	3P	6F	6B	6P
3F	0.0e-00	3.6e-09	5.9e-10	9.8e-03		
3B	3.6e-09	0.0e-00	4.0e-09	9.8e-03		
3P	5.9e-10	4.0e-09	0.0e-00	9.8e-03		
6F	9.8e-03			0.0e-00	5.2e-10	5.6e-10
6B	9.8e-03			5.2e-10	0.0e-00	8.1e-10
6P	9.8e-03			5.6e-10	8.1e-10	0.0e-00

#### 4.2.1. Computational accuracy

The evolution of the first frontal match among the three methods at about eight digits (Table 5), and seven digits (Table 6) for the second frontal. It is six and five digits with the frontal temperatures (Tables 7 and 8). We note the  $1$ -norm and  $2$ -norm results are about one magnitude smaller than the *maximum* one. These two are recorded but not shown because of limitation of space. It can be concluded at this point that, if further knowledge about the initial data is known, our continuous and computational models can provide with results accurate in about six or seven digits for simulation purposes. These include predictions, parameter sensitivity analysis and some type of inverse problems.

#### 4.2.2. Grid-independence

Comparisons between test runs 3 and 6 ( Table 4 ) indicate a grid-independence in the computed results. The matrices in Tables 9 to 12 are better viewed as two-by-two block

Table 10: Convergence of 2-norm deviations of the evolution of second frontal position with initialization 3.

	$n_z = 300$			$n_z = 600$		
dist <sub>2</sub>	3F	3B	3P	6F	6B	6P
3F	0.0e-0	9.9e-8	3.4e-8	1.3e-2		
3B	9.9e-8	0.0e-0	6.6e-8			
3P	3.4e-8	6.6e-8	0.0e-0			
6F	1.3e-2			0.0e-0	2.3e-8	7.0e-9
6B				2.3e-8	0.0e-0	1.7e-8
6P				7.0e-9	1.7e-8	0.0e-0

Table 11: Convergence of 2-norm deviations of the evolution of first frontal temperature with initialization 3.

	$n_z = 300$			$n_z = 600$		
dist <sub>2</sub>	3F	3B	3P	6F	6B	6P
3F	0.00e-0	1.97e-6	3.65e-7	5.05e-2		
3B	1.97e-6	0.00e-0	1.75e-6			
3P	3.65e-7	1.75e-6	0.00e-0			
6F	5.05e-2			0.00e-0	4.47e-7	1.31e-7
6B				4.47e-7	0.00e-0	5.37e-7
6P				1.31e-7	5.37e-7	0.00e-0

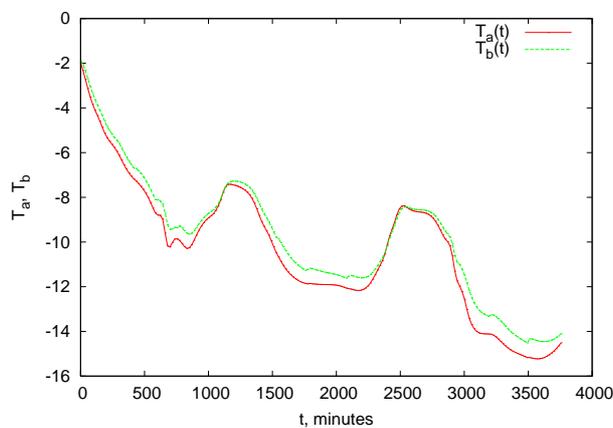


Figure 4: Evolution of the two frontal temperatures.

matrices. With regards to the calculation of the first frontal position, comparison of blocks (1,1) and (2,2) in Table 9 suggests a second order (Cauchy-) convergence in the spatial

Table 12: Convergence of 2-norm deviations of the evolution of second frontal temperature with initialization 3.

dist <sub>2</sub>	n <sub>z</sub> = 300			n <sub>z</sub> = 600		
	3F	3B	3P	6F	6B	6P
3F	0.00e-0	4.79e-6	1.77e-6	1.68e-1		
3B	4.79e-6	0.00e-0	3.05e-6			
3P	1.77e-6	3.05e-6	0.00e-0			
6F	1.68e-1			0.00e-0	1.22e-6	3.46e-7
6B				1.22e-6	0.00e-0	9.00e-7
6P				3.46e-7	9.00e-7	0.00e-0

Table 13: Order of convergence of calculations with grid sizes 300 and 600. Here dist<sub>∞</sub> (dist<sub>1</sub>, dist<sub>2</sub>) refers to the discrete *maximum* (normalized 1-, 2-) *norm* of computed results on all spatial nodes. For dist<sub>2</sub> case, the denominator and numerator of each ratio are from main-diagonals in Tables 9-12.

Ratio	Frontal a	Frontal b	Temperature at a	Temperature at b
dist <sub>1</sub> (6F,6B)/dist <sub>1</sub> (3F,3B)	$\frac{34}{220} = 0.154$	$\frac{19}{83} = 0.228$	$\frac{32}{146} = 0.219$	$\frac{95}{378} = 0.251$
dist <sub>2</sub> (6F,6B)/dist <sub>2</sub> (3F,3B)	$\frac{52}{360} = 0.144$	$\frac{23}{99} = 0.232$	$\frac{447}{1970} = 0.226$	$\frac{122}{479} = 0.254$
dist <sub>∞</sub> (6F,6B)/dist <sub>∞</sub> (3F,3B)	$\frac{24}{150} = 0.160$	$\frac{67}{210} = 0.319$	$\frac{218}{829} = 0.262$	$\frac{469}{1630} = 0.287$
dist <sub>1</sub> (6F,6P)/dist <sub>1</sub> (3F,3P)	$\frac{50}{51} = 0.980$	$\frac{48}{290} = 0.165$	$\frac{835}{1670} = 0.500$	$\frac{253}{1380} = 0.183$
dist <sub>2</sub> (6F,6P)/dist <sub>2</sub> (3F,3P)	$\frac{56}{59} = 0.949$	$\frac{7}{34} = 0.205$	$\frac{131}{365} = 0.358$	$\frac{346}{1770} = 0.195$
dist <sub>∞</sub> (6F,6P)/dist <sub>∞</sub> (3F,3P)	$\frac{11}{13} = 0.846$	$\frac{22}{70} = 0.314$	$\frac{46}{228} = 0.201$	$\frac{157}{577} = 0.272$
dist <sub>1</sub> (6B,6P)/dist <sub>1</sub> (3B,3P)	$\frac{68}{240} = 0.283$	$\frac{15}{55} = 0.272$	$\frac{399}{1300} = 0.306$	$\frac{703}{2390} = 0.294$
dist <sub>2</sub> (6B,6P)/dist <sub>2</sub> (3B,3P)	$\frac{81}{400} = 0.202$	$\frac{17}{66} = 0.257$	$\frac{537}{1750} = 0.306$	$\frac{90}{305} = 0.295$
dist <sub>∞</sub> (6B,6P)/dist <sub>∞</sub> (3B,3P)	$\frac{27}{160} = 0.168$	$\frac{45}{140} = 0.321$	$\frac{216}{829} = 0.260$	$\frac{312}{1090} = 0.286$

variable for each of the three solution methods, except for cases with a possible saturation of accuracy. Actually, this holds for all the two frontal positions and temperatures, measured by *normalized 2-norm* (Tables 9 to 12 ). Similar results in the *maximum* and *normalized 1-norm* are not shown because of limitation of space. These numbers are demonstrated collectively in Table 13.

#### 4.2.3. Characteristics of incomplete shootings

Some comments may be mentioned on the three shooting methods. For practical reasons, we carried out only one sweep of the shooting for each time step in the time march for the final runs (Table 4). This makes the three less ideal. In a different application, the incomplete forward (or backward) shoot may be advantageous in the case where the sea

Table 14: First-order vs second-order extrapolation on max deviations of the evolution of first frontal position.

$n_z = 300, extra_1$			$n_z = 300, extra_2$		
3F	3B	3P	5F	5B	5P
0.0e-0	1.5e-8	1.3e-9	9.6e-4		
1.5e-8	0.0e-0	1.6e-8			
1.3e-9	1.6e-8	0.0e-0			
9.6e-4			0.0e-0	1.5e-8	1.4e-9
			1.5e-8	0.0e-0	1.6e-8
			1.4e-9	1.6e-8	0.0e-0

level (or deep ocean) temperature plays a dominant role. However, we can confirm that the computation involving the parallel shoot is more stable in the current application because of smaller residuals. In fact, the current problem seems to be diffusion-dominant.

4.2.4. Other factors

Choice of an extrapolation scheme is critical in our pre-study in the classical Stefan problem. With the mushy layer mechanism, the current system is much more sensitive and we found that linear extrapolation is better suited in numerical experiments than second order extrapolation. Results in Tables 14 through 17 indicate that the accuracies are about the same. Nevertheless linear extrapolation is favored for stability reason. The abbreviations  $extra_1$  and  $extra_2$  in these tables stand for linear and second order extrapolation, respectively. The results with central and upwind differences are comparable, see Tables

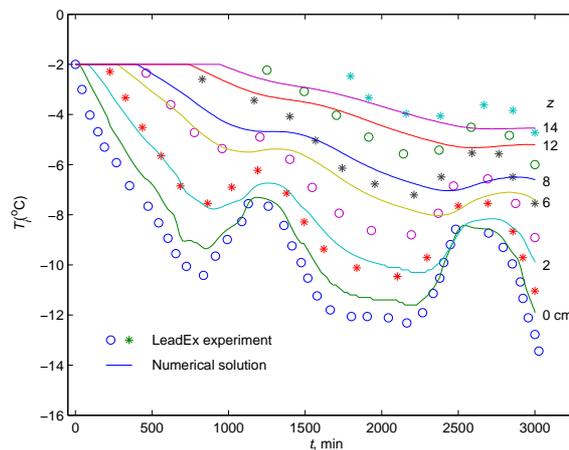


Figure 5: Temperatures at various depths.

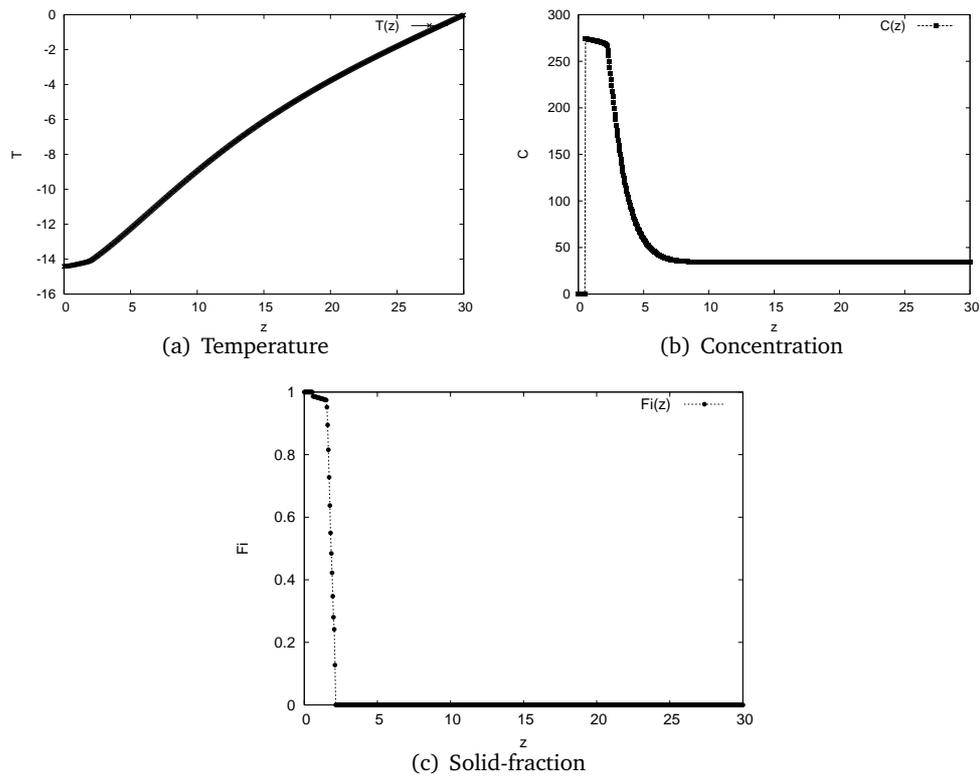


Figure 6: Profiles at final time.

18 through 21. The latter is preferred, due to a possible occurrence of a boundary layer near an interface.

### 4.3. Other computational models

We make some comments on the trade-offs between choosing static grid or dynamic grid. The latter is much easier to implement than the former. However, computation with the dynamic grid demands many of interpolations at each time step. This may deteriorate the stability in a long-term simulation. We encountered this difficulty in our previous experiments. On the other hand, static grid approach is difficult to implement and prone to errors. After careful design, it works fine and produces results comparable to the field data in long-term simulations, based on the semi-infinite interval model with the proposed threshold strategy.

As far as the finite interval model is concerned, it consists of nonlinear equations and the execution time required is much more than that of the model in the semi-infinite interval. This is because the spatial resolution suitable for the current application is about one magnitude finer, and several billions of time steps are required. Numerical stability is

Table 15: First-order vs second-order extrapolation on max deviations of the evolution of second frontal position.

$n_z = 300, extra_1$			$n_z = 300, extra_2$		
3F	3B	3P	5F	5B	5P
0.0e-0	2.1e-7	7.0e-8	6.6e-3		
2.1e-7	0.0e-0	1.4e-7			
7.0e-8	1.4e-7	0.0e-0			
6.6e-3			0.0e-0	2.2e-7	7.1e-8
			2.2e-7	0.0e-0	1.5e-7
			7.1e-8	1.5e-7	0.0e-0

Table 16: First-order vs second-order extrapolation on max deviations of the evolution of first frontal temperature.

$n_z = 300, extra_1$			$n_z = 300, extra_2$		
3F	3B	3P	5F	5B	5P
0.00e-0	8.29e-6	2.28e-6	2.23e-2		
8.29e-6	0.00e-0	8.29e-6			
2.28e-6	8.29e-6	0.00e-0			
2.23e-2			0.00e-0	8.72e-6	1.73e-6
			8.72e-6	0.00e-0	8.30e-6
			1.73e-6	8.30e-6	0.00e-0

therefore extremely critical.

#### 4.4. Things that may go wrong

The practical salinity unit ( *psu* ) is a dimensionless ratio. It describes the concentration of dissolved salts in the water. Open ocean salinity is generally in the range from 32 to 37. The medium value (  $C_{Ocean} = 34.5$  ) is used in our computation. This constantly causes error in the boundary data. The relative error is 7.2% in the worst case. Actually, the relative accuracies in the physical parameters ( Table 1 ) are only around  $10^{-5}$  to  $10^{-3}$ . One other uncertainty is regarding the initial profiles of all primitive variables. Linear initial profiles are assumed for the solid and liquid zones, while four different piecewise linear distributions of the primitive variables with possible jump discontinuities are initially deployed in the mush layer. We noted two things. Firstly, the calculated results were compared on a coarse and a fine grid, all of the four cases yielded second order convergences at the level of  $10^{-8}$  as observed in Table 13. Secondly, the deviations among the results of the four cases are at the level of  $10^{-3}$  in the calculated frontal positions. These indicate that, our continuous model and problem-solving procedures ( and software implementation )

Table 17: First-order vs second-order extrapolation on max deviations of the evolution of second frontal temperature.

$n_z = 300, extra_1$			$n_z = 300, extra_2$		
3F	3B	3P	5F	5B	5P
0.00e-0	1.63e-5	5.77e-6	1.07e-1		
1.63e-5	0.00e-0	1.09e-5			
5.77e-6	1.09e-5	0.00e-0			
1.07e-1			0.00e-0	1.64e-5	5.47e-6
			1.64e-5	0.00e-0	1.10e-5
			5.47e-6	1.10e-5	0.00e-0

Table 18: Max deviations of the evolution of first frontal position with UD or CD.

Upwind Difference			Central Difference		
3F	3B	3P	7F	7B	7P
0.0e-00	1.5e-08	1.3e-09	5.9e-03		
1.5e-08	0.0e-00	1.6e-08			
1.3e-09	1.6e-08	0.0e-00			
5.9e-03			0.0e-00	1.4e-08	1.4e-09
			1.4e-08	0.0e-00	1.6e-08
			1.4e-09	1.6e-08	0.0e-00

are reliable, and that, more complete input data are essential for achieving more accurate calculated results.

## 5. Conclusion

**Physical insights about the model.** These are connected with the nonlinear heat and mass transfer in a mushy layer representing a region of a liquid and solid material in the form of dendrites and growing solid particles (like a porous medium). The model under consideration describes the processes when the latent heat of crystallization practically compensates the constitutional supercooling. This is the main feature of the model equations detailed in the paper. A mushy layer model (e.g. the model under consideration) with moving boundaries of the phase transition cannot be solved by conventional numerical methods. This is due to nonlinearities in the model equations and boundary conditions imposed at the moving phase transition boundaries with unknown velocities. Therefore, numerical approaches under consideration are very useful and important, not only from the point of view of new developed procedures and numerical schemes, but also from the development of new ideas for calculation of such problems to model different problems with moving phase transition domains arising in applied science.

Table 19: Max deviations of the evolution of second frontal position with UD or CD.

Upwind Difference			Central Difference		
3F	3B	3P	7F	7B	7P
0.0e-00	2.1e-07	7.0e-08	2.9e-02		
2.1e-07	0.0e-00	1.4e-07			
7.0e-08	1.4e-07	0.0e-00			
2.9e-02			0.0e-00	2.2e-07	7.1e-08
			2.2e-07	0.0e-00	1.4e-07
			7.1e-08	1.4e-07	0.0e-00

Table 20: Max deviations of the evolution of first frontal temperature with UD or CD.

Upwind Difference			Central Difference		
3F	3B	3P	7F	7B	7P
0.00e-00	8.29e-06	2.28e-06	1.11e-01		
8.29e-06	0.00e-00	8.29e-06			
2.28e-06	8.29e-06	0.00e-00			
1.11e-01			0.00e-00	8.32e-06	9.20e-07
			8.32e-06	0.00e-00	8.32e-06
			9.20e-07	8.32e-06	0.00e-00

**Impact of these simulations on the model.** The present simulations describe the problems of the sea ice growth, crystallization in magma chambers, solidification at the inner core boundary of the Earth, crystal growth of dendritic structures, and solidification in metallurgy, where the phase transition takes place in the framework of the mushy layer model. The observed accuracy and (Cauchy-) convergence justifies our continuous and computational models, the numerical problem-solving procedures and software implementation.

In summary, we consider in this work a two-phase Stefan problem with a mushy layer. Formulations for the semi-infinite and a finite interval are both presented. The continuous system consists of five nonlinear coupled subsystems involving two moving interfaces. Three shooting type solution methods are proposed for numerical modeling of the continuous system based on finite difference discretization. They are designed on a static grid with a threshold strategy. The computation with our proposed numerical methods can yield values of the primitive variables accurate up to eight digits, provided that exact initial data are known. This is justified by calculations using four different initial data and three solution methods. Computed results reported here favor the *parallel shoot* method in stability, and are consistent with the results from previous LeadEx experiment [30, 38].

**Acknowledgments** The authors express their sincere gratitude to the Institute of Mathematics Academia Sinica, Taiwan, for support during summer 2008. D.V. Alexandrov thanks

Table 21: Max deviations of the evolution of second frontal temperature with UD or CD.

Upwind Difference			Central Difference		
3F	3B	3P	7F	7B	7P
0.00e-00	8.29e-06	2.28e-06	1.11e-01		
8.29e-06	0.00e-00	8.29e-06			
2.28e-06	8.29e-06	0.00e-00			
1.11e-01			0.00e-00	8.32e-06	9.20e-07
			8.32e-06	0.00e-00	8.32e-06
			9.20e-07	8.32e-06	0.00e-00

the following grants and programs for financial support: the Federal Target Program "Scientific and Academic - Teaching Staff of Innovative Russia" in 2009 - 2013 and Russian Foundation for Basic Research (project No 11-01-00137).

### Nomenclature

- $a$  – solid-mush boundary, m  
 $b$  – liquid-mush boundary, m  
 $A$  – transformed image in finite interval of  $a$ , m  
 $B$  – transformed image in finite interval of  $b$ , m  
 $c_l$  – specific heat of the solid phase (ice),  $W \cdot s \cdot kg^{-1} \cdot ^\circ C^{-1}$   
 $c_w$  – specific heat of the liquid phase (water),  $W \cdot s \cdot kg^{-1} \cdot ^\circ C^{-1}$   
 $C$  – salinity distribution, psu ( practical salinity unit )  
 $C_{Ocean}$  – salinity at the far field (ocean), psu  
 $D_w$  – diffusion coefficient of water,  $m^2 \cdot s^{-1}$   
 $k_l$  – thermal conductivity of the solid phase (sea ice),  $W \cdot m^{-1} \cdot ^\circ C^{-1}$   
 $k_w$  – thermal conductivity of the liquid phase (sea water),  $W \cdot m^{-1} \cdot ^\circ C^{-1}$   
 $L_v$  – latent heat of solidification,  $W \cdot s \cdot m^{-3}$   
 $m$  – liquidus slope,  $^\circ C \cdot psu^{-1}$   
 $n_t$  – number of computational divisions in time  
 $n_z$  – number of computational divisions in the spatial coordinate  
 $t$  – time, s  
 $T$  – temperature distribution,  $^\circ C$   
 $T_a$  – temperature at the solid-mush boundary,  $^\circ C$   
 $T_b$  – temperature at the liquid-mush boundary,  $^\circ C$   
 $T_{Air}$  – temperature at the solid wall (atmosphere - sea ice boundary),  $^\circ C$   
 $T_{Ocean}$  – boundary temperature at far field (of the isothermal ocean),  $^\circ C$   
 $V$  – transformed image in finite interval of  $T$ ,  $^\circ C$   
 $V_A$  – transformed image in finite interval of  $T_a$ ,  $^\circ C$   
 $V_B$  – transformed image in finite interval of  $T_b$ ,  $^\circ C$

$V_{\text{Air}}$  – transformed image in finite interval of  $T_{\text{Air}}, ^\circ\text{C}$   
 $V_{\text{Ocean}}$  – transformed image in finite interval of  $T_{\text{Ocean}}, ^\circ\text{C}$   
 $x$  – spatial coordinate, m  
 $x_{\text{ref}}$  – the reference point in the spatial coordinate, m

#### Greek symbols

$\alpha_I$  – thermal diffusivity of ice,  $\text{m}^2 \cdot \text{s}^{-1}$  ( $= \frac{k_I}{\rho_I c_I}$ )  
 $\alpha_W$  – thermal diffusivity of water,  $\text{m}^2 \cdot \text{s}^{-1}$  ( $= \frac{k_W}{\rho_W c_W}$ )  
 $\rho_I$  – density of the solid phase (ice),  $\text{kg} \cdot \text{m}^{-3}$   
 $\rho_W$  – density of the liquid phase (water),  $\text{kg} \cdot \text{m}^{-3}$   
 $\varphi$  – solid fraction  
 $\varphi_a$  – solid fraction at the solid-mush boundary  
 $\varphi_b$  – solid fraction at the liquid-mush boundary  
 $\psi$  – transformed image in finite interval of  $\varphi$   
 $\psi_A$  – transformed image in finite interval of  $\varphi_a$   
 $\psi_B$  – transformed image in finite interval of  $\varphi_b$

#### Subscripts

$a$  – solid-mush boundary in semi-infinite interval  
 $A$  – solid-mush boundary in finite interval  
 $b$  – liquid-mush boundary in semi-infinite interval  
 $B$  – liquid-mush boundary in finite interval  
 $I$  – solid zone (ice)  
 $m$  – mushy layer  
 $W$  – liquid zone (water)  
 $t$  – temporal derivatives in semi-infinite interval  
 $\tau$  – temporal derivatives in finite interval  
 $x$  – spatial derivatives in semi-infinite interval  
 $\xi$  – spatial derivatives in finite interval

## References

- [1] D.V. ALEXANDROV, *Solidification with a quasiequilibrium two-phase zone*, Acta Mater. 49 (2001) 759-764.
- [2] D.V. ALEXANDROV, D.L. ASEEV, *One-dimensional solidification of an alloy with a mushy zone: thermodiffusion and temperature-dependent diffusivity*, JFM 527 (2005) 57-66.
- [3] D.V. ALEXANDROV, D.L. ASEEV, I.G. NIZOVTSOVA, H.-N. HUANG, D. LEE, *Nonlinear dynamics of directional solidification with a mushy layer. Analytic solutions of the problem*. Int.J.Heat Mass Transfer, 50 (2007) 3616-3623.
- [4] D.V. ALEXANDROV, I.G. NIZOVTSOVA, A.P. MALYGIN, H-N HUANG AND D. LEE, *Unidirectional solidification of binary melts from a cooled boundary: analytical solutions of a nonlinear diffusion-limited problem*, JPCM vol 20, Issue 11, pp. 114105 (2008).

- [5] D.L. ASEEV, D.V. ALEXANDROV, *Unirectional solidification with a mushy layer*, The influence of weak convection, *Acta Mater.* 54 (2006) 2401-2406.
- [6] CARL DE BOOR, *A Practical Guide to Splines*, Revised Edition Springer-Verlog, NewYork, 2001.
- [7] CARL DE BOOR, *Convergence of cubic spline interpolation with the not-a-knot condition*; MRC 2876 ; University of Wisconsin at Madison, 1985.
- [8] V.T. BORISOV, *Theory of the Two-Phase Zone of a Metal Ingot*, Moscow: Metallurgiya Publishing House, 1987.
- [9] J.P. BOYD, *Chebyshev and Fourier-Spectral Methods*, 2nd ed. Dover, Mineola, New York, 2001
- [10] J.C. BRICE, *The Growth of Crystals from Liquid*, North Holland, Amsterdam, 1973.
- [11] C. CANUTO, M.Y. HUSSAINI, A. QUARTERONI, T.A. ZANG, *Spectral Methods, Fundamentals in Single Domains*, Springer New York, 2006
- [12] B. CHALMERS, *Principles of Solidification*, Wiley, New York, 1964.
- [13] J. CRANK, *Free and Moving Boundary Problems*, Clarendon Press, Oxford, England, 1984.
- [14] R. DEGUEN, T. ALBOUSSIERE AND D. BRITO, *On the existence and structure of a mush at the inner core boundary of the Earth*, *Physics of the Earth and Planetary Interiors*, Vol. 164, 2007, pp. 36-49.
- [15] A. FASANO AND M. PRIMICERIO, *General free boundary problems for the heat equation III*, *J. Math. Anal. Appl.*, Vol. 59, 1977, pp. 1-14.
- [16] A. FRIEDMAN, *Free boundary problems for parabolic equations I. Melting of solids*, *J. Math. Mechanics*, 8, 1959, pp. 499-516.
- [17] A.C. FOWLER, *The formation of freckles in binary alloys*, *IMA J. Appl. Math.*, Vol. 35, 1985, pp. 159-174.
- [18] F. GIBOU AND R. FEDKIW, *A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the stefan problem*, *J. Com. Phy.* 202 (2005) 577-601.
- [19] S.C. GUPTA, *The Classical Stefan Problem, Basic Concepts, Modelling and Analysis*, North-Holland series in Appl. Math. and Mechanics, Vol. 45. 2003.
- [20] J.M. HILL, *One-Dimensional Stefan Problems: An Introduction*, Longman Scientific Technical, Harlow, 1987.
- [21] R. N. HILLS, D. E. LOPER, P. H. ROBERTS, *A Thermodynamically Consistent Model of a Mushy Zone*, *Quarterly Jnl. of Mechanics & App. Maths.* 36 (1983) 505-540.
- [22] P.V. HOBBS, *Ice Physics*, Clarendon, Oxford, England, 1974.
- [23] G.P. IVANTSOV, *Diffusive Supercooling in Binary Alloy Solidification*, *Dokl. Akad. Nauk SSSR* 81 (1951) 179-182.
- [24] E. JAVIERRE, C. VUIK, E.J. VERMOLEN, S. VAN DER ZWAAG, *A comparison of numerical models for one-dimensional Stefan problem*, *J. Com. Appl. Math.* 192 (2006) 445-459.
- [25] W.T. KYNER, *An existence and uniqueness theorem for a nonlinear Stefan problem*, *J. Math. Mechanics*, 8, 1959, pp. 483-498.
- [26] R.A. LAUDISE, *The Growth of Single Crystals*, Prentice Hall, New York 1972.
- [27] DANIEL LEE AND D.V. ALEXANDROV, *Numerical Modeling of One-Dimensional Binary Solidification - the Classical Two-Phase Stefan Problem*, *IJPAM* V.58, No.4, 2010, 381-416.
- [28] ZHILIN LI AND KAZUFUMI ITO, *The Immersed Interface Method, Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*, SIAM, Philadelphia, 2006.
- [29] ANVARBEK M. MEIRMANOV, translated from the Russian by Marek Niezgodka and Anna Crowley, *The Stefan Problem*, de Gruyter Expositions in Mathematics 3, 1992. Walter de Gruyter , Berlin.
- [30] J. MORISON, M. MCPHEE, R. MUENCH, *et al.*, *The LeadEx Experiment*, *EOS. Trans. AGU* 74 (1993) 393-397.

- [31] S. OSHER, R. FEDKIW, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, New York, 2003.
- [32] S. OSHER AND C.W. SHU, *High-Order Essentially Nonoscillatory Schemes for Hamilton-Jacobi Equations*, SIAM J. Numer. Anal. vol 28, Issue 4, pp 907-922, 1991.
- [33] F. ROSENBERG, *Fundamentals of Crystal Growth*, Vol. 1, Springer, Berlin, 1979.
- [34] L.I. RUBINSTEIN, *The Stefan Problem*, American Mathematical Society, Providence, 1971.
- [35] J.A. SETHIAN, *Level Set Methods and Fast Marching Methods*, Cambridge University press, New York, 1999.
- [36] JIE SHEN AND TAO TANG, *Spectral and High-order Methods with Applications* Science Press, Beijing, China 2006.
- [37] C. VUIK, *Some historical notes about the Stefan problem*, Nieuw Archief voor Wiskunde, 4Ed, Serie 157, 1993.
- [38] J.S. WETTLAUFER, M.G. WORSTER, H.E. HUPPERT, *Solidification of Leads: Theory, Experiment and Field Observations*, J. Geophys. Res. 105 (2000) 1123-1134.