

NUMERICAL SIMULATION OF THE HORIZONTAL BRIDGMAN GROWTH. PART III: CALCULATION OF THE INTERFACE

P. WOUTERS, J. J. VAN SCHAFTINGEN, M. J. CROCHET

Université Catholique de Louvain, Louvain-la-Neuve, Belgium

AND

F. T. GEYLING

Bell Laboratories, Murray Hill, New Jersey, U.S.A.

SUMMARY

We study the transient motion of the solidification front during the growth of semiconductor crystals in the horizontal Bridgman geometry. The calculation is based on a two-dimensional flow. We use finite elements which deform with the motion of the interface. The energy equation is coupled with the isothermal constraint of the interface in an implicit transient algorithm. Several examples show the oscillatory motion of the interface caused by the periodic flow of the melt, and they reveal the importance of the growth rate on the shape of the interface.

KEY WORDS Finite Elements Transient Flows Three-dimensional Flows Natural Convection Interfaces Oscillatory Flows Crystal Growth Semiconductors Gallium Arsenide.

1. INTRODUCTION

Although the general title of the papers in this series bears on the numerical simulation of semiconductor crystal growth, Parts I and II^{1,2} have dealt with a single liquid phase in geometries similar to those of the crucibles in the actual Bridgman growth. We have concluded from Part I that the flow of the melt is typically transient; thus, any attempt to simulate a realistic problem should include the possibility of calculating oscillatory motions of the interface. Such oscillatory motions are important; they explain the occurrence of striation in the growth crystal.³ We have found in Part II that, despite the presence of oscillatory flow mechanisms in a two-dimensional flow, the actual crucibles used in the horizontal Bridgman growth would in general induce truly three-dimensional flows. Although long term developments will certainly require the simultaneous calculation of two phases in a three-dimensional geometry, there is no doubt that much can be learnt from the analysis of two-dimensional situations. Moreover, techniques can then be developed at a much lower cost. More information on the physics of crystal growth may be found in References 4 and 5.

In the present paper, we wish to present an algorithm for studying the simultaneous two-dimensional transient flow and change of phase which occur during the horizontal Bridgman growth. From a fluid mechanics point of view, we wish to demonstrate the important geometrical

differences which occur when one calculates a true two-phase situation as compared to a single liquid phase. The method which is proposed here is not limited to the calculation of the horizontal Bridgman growth. It can be adapted without difficulty to the calculation of the vertical Bridgman growth and the Czochralski growth.

In this problem, the unknowns are the velocity components and the pressure in the liquid phase, the temperature in the liquid and in the solid phases, and the location of the interface. The available equations are as usual the Navier–Stokes equations (with the Boussinesq approximation) and the energy equation. For locating the interface, and in the absence of supercooling, we know that the interface coincides with a melting point isotherm and that its motion releases or recovers the latent heat of fusion. When one uses the finite element method for simulating crystal growth, it is usually convenient to identify the liquid–solid interface with interelement boundaries. Thus, the location of the nodes lying on the interface is one of the unknown fields. In earlier work, Lynch⁶ and Yoo and Rubinsky⁷ have chosen to impose the melting temperature at the interfacial nodes while solving the energy equation; the location of the interface is determined on the basis of a local heat balance which includes the latent heat of fusion. Here, we will rather follow the approach used by Ettouney and Brown⁸ for steady-state problems and by Crochet, Geyling and Van Schaftingen⁹ for transient problems, where the release of latent heat of fusion is included as a source term in the energy equation and the interface is identified as the melting point isotherm.

The cost of transient calculations is drastically reduced when one is able to decouple the set of partial differential equations into several separate groups. For calculating low Prandtl number flows, we have seen in Part I that the equations of motion can safely be decoupled from the energy equation. Decoupling was further carried on by Crochet, Geyling and Van Schaftingen⁹ in calculating the location of the interface. However, the resulting algorithm was not robust in the sense that very small time steps were required for calculating an interface during rapid transients while the location of the interface was sometimes exhibiting spurious numerical oscillations.

In the present paper, the governing equations are separated into two groups. On the one hand, we solve the Navier–Stokes equations whereas on the other we solve the energy equation together with the constraint on the location of the interface. The time-stepping technique for both sets of equations is based on the classical predictor–corrector scheme developed by Gresho, Lee and Sani.¹⁰

The good behaviour of the algorithm is exhibited by means of two theoretical examples and the actual simulation of the horizontal Bridgman growth. One finds that the actual motion of the interface during growth affects its shape in a dramatic way, and that the fluid motion induces melting–solidification cycles near the interface.

2. BASIC EQUATIONS

We wish to solve transient non-isothermal two-phase problems schematized in Figure 1, which is representative of plane as well as axisymmetric geometries. The problem domain Ω is the union of two subdomains Ω_L and Ω_S separated by an interface Σ ; Ω_L is filled with the liquid phase of the material, whereas Ω_S stands for the solid phase. The boundary of Ω_L consists of Σ and its interface $\partial\Omega_L$ with the outside world; similarly, the boundary of Ω_S is the union of Σ and $\partial\Omega_S$.

The growth problem is essentially transient for the following reasons: (i) boundary conditions imposed on $\partial\Omega_L$ and $\partial\Omega_S$ vary with time and affect in particular the location of Σ ; (ii) even when the boundary conditions do not vary with time, the solid–liquid system may not have reached its final state in view of its thermal inertia; (iii) in Part I of the present series of papers,¹ Crochet, Geyling and Van Schaftingen have shown that buoyancy-driven oscillations occur in the liquid phase beyond a critical value of the Grashof number, and generate an intrinsically transient problem.

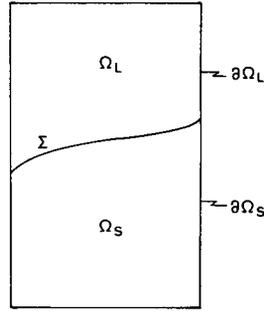


Figure 1. Geometry of the two-phase problem. Ω_L and Ω_S denote the liquid and solid domains, respectively

Physical situations obeying this general description are found in crystal growth based on the horizontal and vertical Bridgman processes and on the Czochralski technique.

For the sake of simplicity, let us review the basic equations describing the state of the system in rectangular Cartesian co-ordinates. In Ω_L , the motion of the fluid is governed by the Navier-Stokes equations given as follows:

$$\rho_L \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) + \nabla p - \mu \Delta \mathbf{v} + \rho_L \alpha (T - T_0) \mathbf{g} = \mathbf{0},$$

$$\nabla \cdot \mathbf{v} = 0, \quad (1)$$

where \mathbf{v} is the velocity field, p is the pressure, T and T_0 stand for the actual and a reference temperature, respectively, ρ_L is a reference density of the liquid phase, μ its shear viscosity, α the thermal dilatation coefficient and \mathbf{g} the acceleration due to gravity. We note that the Boussinesq approximation is used for generating buoyancy forces.

The temperature field in Ω_L is governed by the energy equation which has the following form:

$$\rho_L c_L \left(\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T \right) - k_L \Delta T = 0, \quad (2)$$

where c_L and k_L stand for the heat capacity and the thermal conductivity in the liquid phase; heat generation by viscous dissipation is neglected. There is no flow in the solid phase Ω_S ; the energy equation is given by

$$\rho_S c_S \frac{\partial T}{\partial t} - k_S \Delta T = 0, \quad (3)$$

where ρ_S , c_S and k_S are the density, heat capacity and thermal conductivity in the solid phase. In what follows, we will assume that $\rho_L = \rho_S = \rho$.

Let us now consider the equations needed for determining the location of the interface Σ . We will assume that the interface coincides with the melting point isotherm:

$$T = T_m \text{ on } \Sigma. \quad (4)$$

Moreover, the motion of Σ produces a release of latent heat of fusion which is taken into account as a line heat source on Σ for the simultaneous solution of (2) and (3). The heat produced per unit length of Σ and per unit time is given by

$$r_\Sigma = -\rho \Delta H_f v_\Sigma, \quad (5)$$

where v_z is the normal component of the interface velocity, which is taken as positive when directed towards the solid phase Ω_S , and ΔH_f is the latent heat of fusion.

Equations (1)–(5) are sufficient for calculating the unknowns of the problem. The explicit unknowns are the temperatures in Ω_L and Ω_S , and the velocity and pressure fields in Ω_L . The location of Σ is an implicit unknown of the system (1)–(5) which must be calculated together with the other fields.

Time-dependent boundary conditions are applied on $\partial\Omega_S$ and $\partial\Omega_L$. As far as the kinematic boundary conditions are concerned, we will assume that the melt does not slip along the wall $\partial\Omega_L$ and along Σ . A variety of boundary conditions may be chosen for calculating the temperature field: they include heat convection, radiation and an imposed temperature field.

3. SPACE DISCRETIZATION

One of the intrinsic difficulties of the present problem is the spatial discretization of the domains Ω_L and Ω_S , separated by a time-dependent interface Σ . The location of Σ is *a priori* unknown; when the temperature is not imposed as a boundary condition, one does not even know the location of a single point of Σ . With the present state of the art, one should not expect to calculate in a single run the full crystallization process in which Ω_S is empty at the outset and eventually replaces Ω_L . Rather, we will calculate limited phases of the growth in which Σ moves within a domain of limited extension.

The domain is discretized by means of finite elements and we will always impose the condition that the interface Σ coincides with interelement boundaries. Figure 2 shows a typical (very coarse) finite element mesh covering the domain of Figure 1. Although the position of Σ is time-dependent, we will base our finite element technique on the hypothesis that the topology of the mesh remains invariant with respect to time, in such a way that the location of the nodes is entirely determined by the position of Σ and the topology of the mesh.

By way of example, let us assume on Figure 2 that the nodes of the interface move along vertical lines which have been chosen *a priori*. The location of the interface is then fully determined by the set of vertical co-ordinates S_1, S_2, \dots which are time-dependent. Pre-established rules allow one to draw the finite element mesh on the basis of these vertical co-ordinates. In Figure 2, the vertical segments in Ω_L and Ω_S are divided into smaller segments of equal size (they do not need to be equal)

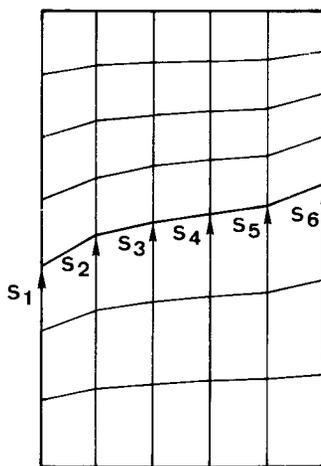


Figure 2. Typical finite element mesh covering the domain of Figure 1

identified as element sides. The topology selected for the mesh depends upon the problem under consideration; the basic idea of Figure 2 is in general applicable to the Bridgman and Czochralski processes, but we might select other lines for the motion of the nodes on the interface with an appropriate topology.

To summarize, the domains Ω_L and Ω_S are covered by a finite element mesh endowed with a fixed topology and which is entirely characterized by a set of scalar variables S_k describing the location of Σ .

A finite element interpolation is chosen for the unknown fields; the shape functions are invariant with respect to the co-ordinates of the parent element, but their values at a given spatial location depend upon the co-ordinates of the nodes through the isoparametric transformation and thus upon the variables S_k , which are time dependent. In our work, the shape functions will be the same for the velocity components and the temperature, whereas they are different for the pressure. The finite element interpolation is then given as follows:

$$\begin{aligned}\tilde{T} &= \sum_j T_j(t) \psi_j(S_k), \\ \tilde{\mathbf{v}} &= \sum_j \mathbf{V}_j(t) \psi_j(S_k), \\ \tilde{p} &= \sum_j P_j(t) \phi_j(S_k).\end{aligned}\quad (6)$$

where T_j , \mathbf{V}_j , P_j denote (generalized) nodal values for the temperature, the velocity and the pressure, respectively. The finite element meshes for the present work contain quadrilateral elements only; the ψ_j s are biquadratic Lagrangian shape functions, and the ϕ_j s stand for complete piecewise linear polynomials. There are three P_j unknowns per element, which are not necessarily associated with the nodes; the representation for the pressure is piecewise continuous.

The spatial discretization is obtained by means of the Galerkin procedure applied to the system (1)–(3). We will however consider that (2) and (3) coalesce into a single equation

$$\rho c \left(\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T \right) - k \Delta T = 0, \quad (7)$$

where the physical parameters ρ , c and k depend upon the location within the mesh, and \mathbf{v} vanishes in Ω_S . Equation (5) is implicitly contained in (7) because the latent heat of fusion on Σ acts as a line heat source within the domain.

The Galerkin procedure, followed by the application of the divergence theorem on the second-order derivatives, produces the following set of algebraic equations:

$$\begin{aligned}\left\langle \psi_i; \rho_L \left(\frac{\partial \tilde{\mathbf{v}}}{\partial t} + \tilde{\mathbf{v}} \cdot \nabla \tilde{\mathbf{v}} \right) \right\rangle + \langle \nabla \psi_i^T; -\tilde{p} \mathbf{I} + \mu (\nabla \tilde{\mathbf{v}} + \nabla \tilde{\mathbf{v}}^T) \rangle + \langle \psi_i; \rho_L \alpha (\tilde{T} - T_0) \mathbf{g} \rangle - \mathbf{F}_i = \mathbf{0}, \quad \text{in } \Omega_L, \\ \langle \phi_i; \nabla \cdot \tilde{\mathbf{v}} \rangle = 0, \quad \text{in } \Omega_L, \\ \left\langle \psi_i; \rho c \left(\frac{\partial \tilde{T}}{\partial t} + \tilde{\mathbf{v}} \cdot \nabla \tilde{T} \right) \right\rangle + \langle \nabla \psi_i^T; k \nabla \tilde{T} \rangle - Q_i = 0, \quad \text{in } \Omega_L \cup \Omega_S.\end{aligned}\quad (8)$$

where $\langle ; \rangle$ denotes the L^2 scalar product. The symbols \mathbf{F}_i and Q_i stand for generalized nodal forces; Q_i in particular contains the contributions from the boundary conditions and the latent heat of fusion. The discretized equivalent of (4) is

$$\tilde{T} = T_m \quad \text{on } \Sigma. \quad (9)$$

The system (8) contains the Eulerian time derivatives $\partial\tilde{\mathbf{v}}/\partial t$ and $\partial\tilde{\mathbf{T}}/\partial t$ which, in view of (6), are given by

$$\begin{aligned}\frac{\partial\tilde{\mathbf{T}}}{\partial t}\Big|_{\mathbf{x}} &= \sum_j \left(\dot{T}_j \psi_j + T_j \frac{\partial\psi_j}{\partial t}\Big|_{\mathbf{x}} \right), \\ \frac{\partial\tilde{\mathbf{v}}}{\partial t}\Big|_{\mathbf{x}} &= \sum_j \left(\dot{V}_j \psi_j + V_j \frac{\partial\psi_j}{\partial t}\Big|_{\mathbf{x}} \right),\end{aligned}\quad (10)$$

since the shape functions are time-dependent through their relation to interface co-ordinates S_k . For evaluating the time derivatives $\partial\psi_j/\partial t$, we use the procedure introduced by Lynch and Gray,¹¹ which we briefly recall. Let us consider on Figure 3 a time-dependent isoparametric element and its parent element expressed in the standard co-ordinates ξ and η . For a fixed value of ξ and η , the velocity of the corresponding point in the isoparametric element is defined as the velocity of the moving mesh and identified as \mathbf{v}_M . Since the value of ψ_i is invariant with respect to ξ and η , we have

$$\frac{\partial\psi_i}{\partial t}\Big|_{\xi,\eta} = 0, \quad (11)$$

and by means of the co-ordinate transformation from (ξ, η) to \mathbf{x} we obtain

$$\frac{\partial\psi_i}{\partial t}\Big|_{\mathbf{x}} + \frac{\partial\mathbf{x}}{\partial t}\Big|_{\xi,\eta} \cdot \nabla\psi_i|_t = 0. \quad (12)$$

In view of our definition of \mathbf{v}_M we have

$$\mathbf{v}_M = \frac{\partial\mathbf{x}}{\partial t}\Big|_{\xi,\eta} \quad (13)$$

and thus

$$\frac{\partial\psi_i}{\partial t}\Big|_{\mathbf{x}} = -\mathbf{v}_M \cdot \nabla\psi_i|_t. \quad (14)$$

Thus, the time derivatives in (8) will be replaced by the following expressions:

$$\begin{aligned}\frac{\partial\tilde{\mathbf{T}}}{\partial t}\Big|_{\mathbf{x}} &= \sum_j \left(\dot{T}_j \psi_j - T_j \mathbf{v}_M \cdot \nabla\psi_j \right), \\ \frac{\partial\tilde{\mathbf{v}}}{\partial t}\Big|_{\mathbf{x}} &= \sum_j \left(\dot{V}_j \psi_j - V_j \mathbf{v}_M \cdot \nabla\psi_j \right),\end{aligned}\quad (15)$$

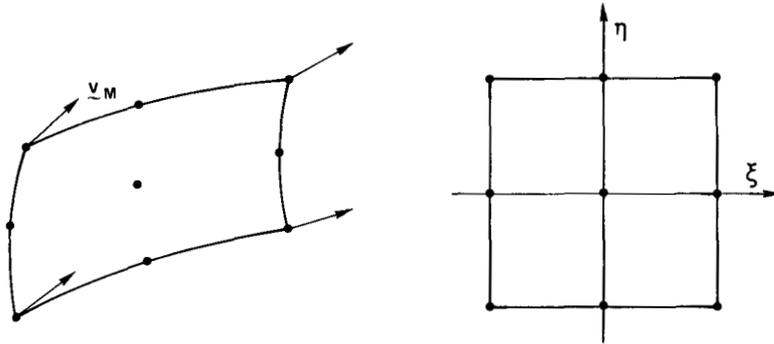


Figure 3. Time-dependent isoparametric element and its parent element

where dot superscripts denote the time derivatives of the nodal variables. We note that the mesh velocity v_M and the normal velocity of the interface v_Σ in (5) depend linearly upon the time derivatives \dot{S}_k of the interface co-ordinates.

For obtaining the final form of the discretized equations, we substitute (6) and (10) for $\tilde{\mathbf{T}}, \tilde{\mathbf{v}}, \tilde{p}$ and their time-derivatives in (8). Let $\mathbf{V}^*, \mathbf{P}^*, \mathbf{T}^*, \mathbf{S}^*$ denote the vectors of nodal velocity components, nodal pressures, nodal temperatures and nodal interface co-ordinates, respectively. The discretized form of (8) may then be written as follows:

$$\mathbf{M}(\mathbf{S}^*)\dot{\mathbf{V}}^* + \mathbf{K}(\mathbf{V}^*, \mathbf{S}^*, \dot{\mathbf{S}}^*)\mathbf{V}^* + \mathbf{C}(\mathbf{S}^*)\mathbf{P}^* - \mathbf{F}(\mathbf{T}^*, \mathbf{S}^*) = \mathbf{0}, \quad (16a)$$

$$\mathbf{C}^T(\mathbf{S}^*)\mathbf{V}^* = \mathbf{0}, \quad (16b)$$

$$\mathbf{M}'(\mathbf{S}^*)\dot{\mathbf{T}}^* + \mathbf{K}'(\mathbf{V}^*, \mathbf{S}^*, \dot{\mathbf{S}}^*)\mathbf{T}^* - \mathbf{Q}(\mathbf{T}^*, \mathbf{S}^*, \dot{\mathbf{S}}^*) = \mathbf{0}, \quad (16c)$$

where \mathbf{M}, \mathbf{M}' are the mass matrices and \mathbf{K}, \mathbf{K}' include the diffusive and convective contributions in the momentum and energy equations. The term $\mathbf{F}(\mathbf{T}^*, \mathbf{S}^*)$ contains the buoyancy forces and the surface forces, while $\mathbf{Q}(\mathbf{T}^*, \mathbf{S}^*, \dot{\mathbf{S}}^*)$ contains the contribution of the heat of fusion and the heat flux on the boundary. Finally, (9) takes the general form

$$\mathbf{D}\mathbf{T}^* = \mathbf{T}_m \quad (17)$$

where \mathbf{T}_m is a vector with all its components equal to T_m .

4. TIME-DISCRETIZATION

Let \mathbf{Z} denote a vector containing the n nodal unknowns of the problem. The system of ordinary non-linear differential equations (16), (17) may be summarized as follows:

$$\mathbf{A}(\mathbf{Z})\dot{\mathbf{Z}} = \mathbf{B}(\mathbf{Z}), \quad (18)$$

with a set of initial conditions at time t_0

$$\mathbf{Z}(t_0) = \mathbf{Z}_0. \quad (19)$$

When a steady-state solution exists, it is the solution of

$$\mathbf{B}(\mathbf{Z}) = \mathbf{0}. \quad (20)$$

The time is discretized into a set of instants $t_0 < t_1 < \dots < t_N$; let Δt_{n+1} denote the difference ($t_{n+1} - t_n$). We assume that \mathbf{Z} and $\dot{\mathbf{Z}}$ are known up to time t_n . We need to specify an algorithm for calculating $\mathbf{Z}_{n+1} = \mathbf{Z}(t_{n+1})$.

An explicit algorithm is such that \mathbf{Z}_{n+1} may be calculated entirely in terms of the values of \mathbf{Z} and $\dot{\mathbf{Z}}$ at earlier instants; it is summarized by

$$\mathbf{Z}_{n+1} = \mathbf{Z}_r, \quad (21)$$

where \mathbf{Z}_r is a linear combination of \mathbf{Z}_i and $\dot{\mathbf{Z}}_i$, $i \leq n$. With the Euler explicit method, we have

$$\mathbf{Z}_r = \mathbf{Z}_n + \Delta t_{n+1} \dot{\mathbf{Z}}_n, \quad (22)$$

whereas the second-order Adams–Bashforth method gives us

$$\mathbf{Z}_r = \mathbf{Z}_n + \frac{1}{2} \Delta t_{n+1} [2\dot{\mathbf{Z}}_n + \Delta t_{n+1} / \Delta t_n (\dot{\mathbf{Z}}_n - \dot{\mathbf{Z}}_{n-1})]. \quad (23)$$

An implicit algorithm is such that \mathbf{Z}_{n+1} is calculated as a linear combination of $\dot{\mathbf{Z}}_{n+1}$, formally obtained on the basis of (18), and a vector \mathbf{Z}_r which is itself a linear combination of earlier values of \mathbf{Z} and $\dot{\mathbf{Z}}$, i.e.

$$\mathbf{Z}_{n+1} = \tau \dot{\mathbf{Z}}_{n+1} + \mathbf{Z}_r, \quad (24)$$

where τ is a scalar parameter. Multiplying (24) on the left by $\mathbf{A}(\mathbf{Z}_{n+1})$ and substituting $\mathbf{B}(\mathbf{Z}_{n+1})$ for $\mathbf{A}(\mathbf{Z}_{n+1})\dot{\mathbf{Z}}_{n+1}$ on the basis of (18) we obtain

$$\mathbf{A}(\mathbf{Z}_{n+1})(\mathbf{Z}_{n+1} - \mathbf{Z}_r) - \tau \mathbf{B}(\mathbf{Z}_{n+1}) = \mathbf{0} \quad (25)$$

which constitutes a non-linear algebraic system of equations in \mathbf{Z}_{n+1} . Classical one-step implicit formulae are given by the equation

$$\mathbf{Z}_{n+1} = \mathbf{Z}_n + \Delta t_{n+1} [\theta \dot{\mathbf{Z}}_{n+1} + (1 - \theta) \dot{\mathbf{Z}}_n]; \quad (26)$$

respective values of θ equal to 1/2, 2/3 and 1 produce the Crank–Nicolson, Galerkin and implicit Euler algorithms. Comparing (24) and (26) we obtain

$$\tau = \theta \Delta t_{n+1}, \quad \mathbf{Z}_r = \mathbf{Z}_n + (1 - \theta) \Delta t_{n+1} \dot{\mathbf{Z}}_n. \quad (27)$$

An explicit method requires a reduced amount of computer time as compared to an implicit method; however, the explicit method is conditionally stable, and (26) is A-stable¹² whenever $1/2 \leq \theta \leq 1$. Moreover, with the finite element method, the use of an explicit method for every time step requires further approximations such as mass-lumping, and the pressure needs to be calculated as the solution of a linear system. Finally, the calculation of the interface location is difficult with an explicit algorithm.

Here, we follow the method suggested by Gresho, Lee and Sani¹⁰ which is summarized as follows:

- (i) A predicted value \mathbf{Z}_{n+1}^p is calculated on the basis of an explicit formula such as (21).
- (ii) The non-linear system (25) is solved with the use of \mathbf{Z}_{n+1}^p as an initial guess. The method of solution is the Newton–Raphson method. A small enough time step allows the use of a single iteration.
- (iii) The value of $\dot{\mathbf{Z}}_{n+1}$ is calculated from \mathbf{Z}_r and \mathbf{Z}_{n+1} on the basis of (24), which is consistent with (18).

For obtaining the results shown in sections 6 and 7 we have selected the Adams–Bashforth formula (23) for the explicit scheme and the Crank–Nicolson formula (26) for the implicit scheme with a view to a lower discretization error. Since the procedure requires the knowledge of $\dot{\mathbf{Z}}$ at earlier times, the first few values are calculated on the basis of Euler’s explicit and implicit formulae.

An important advantage of the predictor–corrector scheme described above is the possibility of monitoring the time step for reducing the local discretization error. Let us assume that the solution at t_n is exact, and let \mathbf{Z}_{n+1}^c denote the exact solution at t_{n+1} . We define the discretization error \mathbf{d}_{n+1} as follows:

$$\mathbf{d}_{n+1} = \mathbf{Z}_{n+1} - \mathbf{Z}_{n+1}^c. \quad (28)$$

It was shown by Gresho, Lee and Sani¹⁰ that

$$\mathbf{d}_{n+1} = \frac{1}{3}(1 + \Delta t_n / \Delta t_{n+1})^{-1} (\mathbf{Z}_{n+1} - \mathbf{Z}_{n+1}^p) + O(\Delta t_{n+1}^4) \quad (29)$$

for a combination of the Adams–Bashforth and Crank–Nicolson formulae. Thus, (29) provides us with an easy evaluation of the local discretization error. Let a superscript L on the vectors \mathbf{d} and \mathbf{Z} , $1 \leq L \leq 3$, denote those partial vectors which contain the velocity, temperature and interface variables, respectively. By means of a Taylor series argument, it is then possible to show that

$$\frac{\|\mathbf{d}_{n+2}^L\|_\infty}{\|\mathbf{d}_{n+1}^L\|_\infty} = \left(\frac{\Delta t_{n+2}}{\Delta t_{n+1}} \right)^3 \frac{\|\dot{\dot{\mathbf{Z}}}_{n+1}^L\|_\infty}{\|\dot{\mathbf{Z}}_n^L\|_\infty} \simeq \left(\frac{\Delta t_{n+2}}{\Delta t_{n+1}} \right)^3, \quad (30)$$

where we have assumed that $\|\ddot{\mathbf{Z}}_{n+1}^L\|_\infty \simeq \|\ddot{\mathbf{Z}}_n^L\|_\infty$. Finally, we need to impose a criterion on the magnitude of the error \mathbf{d}_{n+2} . For each type of variable, we select a relative tolerance ε^L and require that

$$\|\mathbf{d}_{n+2}^L\|_\infty < \varepsilon^L \|\mathbf{Z}_{n+1}^L\|_\infty, \quad (31)$$

and we obtain from (30) and (31) that

$$\Delta t_{n+2} < \min_{L=1,3} (\varepsilon^L \|\mathbf{Z}_{n+1}^L\|_\infty / \|\mathbf{d}_{n+1}^L\|_\infty)^{1/3} \Delta t_{n+1}. \quad (32)$$

Typical values of ε^L in the present work lie between 5×10^{-5} and 5×10^{-3} .

In practice, however, the time step is not modified at every iteration. Maximum and minimum values Δt_{\max} and Δt_{\min} are selected at the outset. When Δt_{n+2} deviates from Δt_{n+1} by less than 10 per cent, the time step remains unchanged. When Δt_{n+2} is less than $0.8 \Delta t_{n+1}$, the current step is recalculated with a lower value equal to Δt_{n+2} . When Δt_{n+2} is less than Δt_{\min} , the time marching procedure is interrupted.

The predictor–corrector algorithm which has just been described for the general equation (18) may now be applied to the solution of (16), (17). We note however that there is no predictor for the pressure variables which do not appear through their time derivatives in (16). The system (16), (17) is fully coupled and leads to a large number of simultaneous algebraic equations. The cost of the calculation is roughly divided by two when the equations of motion are decoupled from the energy and the interface equations. At each time step, the algorithm is decomposed as follows:

- (i) The predicted values of the velocity components, the temperature and the interface co-ordinates are evaluated by means of an explicit scheme (22) or (23).
- (ii) The energy equation (16c) and the interface equation (17) are simultaneously solved for calculating \mathbf{T}_{n+1}^* and \mathbf{S}_{n+1}^* , with \mathbf{T}_{n+1}^{*p} and \mathbf{S}_{n+1}^{*p} as the initial guess, and the convective terms in \mathbf{K}' are calculated on the basis of the predicted velocity field \mathbf{V}_{n+1}^p .
- (iii) The equations of motion and incompressibility (16a, b) are solved for calculating \mathbf{V}_{n+1}^* and \mathbf{P}_{n+1}^* with \mathbf{V}_{n+1}^{*p} and \mathbf{P}_n^* as the initial guess. The solution is calculated on the (moving) updated mesh, and the buoyancy forces are evaluated on the basis of the new temperature field \mathbf{T}_{n+1}^* .

One must realize that the coupling between the energy equation (16c) and the interface equation (17) requires a considerable amount of calculation and programming for evaluating the Jacobian matrix in the Newton–Raphson algorithm. Indeed, the geometry of each finite element will depend in general upon the components of \mathbf{S}^* . In earlier work⁹ we had avoided such difficulties through further decoupling the energy and the interface equations. Instead of step (ii) above, we would first solve the energy equation with \mathbf{S}_{n+1}^{*p} substituted for \mathbf{S}^* , and then select the interface as the melting point isotherm. However, it has been found that the coupling between the energy equation and the interface condition is very strong, up to the point where an actual decoupling of the calculation usually requires very small time steps and numerous iterations on the location of the interface. With a decoupled scheme, we have encountered dramatic oscillations of the interface co-ordinates when time proceeds, and an erratic behaviour is frequent when the latent heat of fusion is assigned its correct physical value. We have now forsaken the decoupling of the interface calculation from the energy equation, and we will find in sections 6 and 7 that our new algorithm is very robust.

The boundary conditions during crystal growth are time dependent. The natural boundary conditions are easy to implement; surface forces and heat fluxes are indeed contained in the vectors \mathbf{F} and \mathbf{Q} in (16). However, one needs to remember that the location of the boundary nodes moves with respect to the outer world together with the finite element mesh and the interface. *Nodal* fluxes

may very well be time-dependent although their values are fixed, but non-uniform, with respect to the frame of reference of the calculations.

A spurious oscillatory behaviour of the time-dependent solution is sometimes detected when time-varying essential boundary conditions are imposed at some nodes. Let y be a nodal value which is assigned the value $f(t)$ as time proceeds. The Galerkin equation corresponding to the nodal value y is then replaced by the equation

$$y^{n+1} = f(t_{n+1}). \quad (33)$$

Oscillations may occur when a time-discontinuous behaviour is imposed on y . It has been found useful to rewrite essential boundary conditions as differential equations of the form

$$\dot{y} = \lambda[y - f(t)] + \dot{f}(t), \quad (34)$$

where λ is fixed real negative scalar. Equation (34) is equivalent to $y = f(t)$ when $f(t)$ is continuous. When $f(t)$ presents a discontinuity, y nevertheless remains continuous, and tends to the behaviour of $f(t)$ when $|\lambda|$ is large. Another advantage of choosing (34) is that the essential boundary condition is not dissociated from the other equations during the time-stepping procedure.

5. CALCULATION OF THE COEFFICIENTS OF THE ALGEBRAIC SYSTEM

We have seen in section 3 that the coefficients of the algebraic system (16) depend upon the co-ordinates of the interface which are represented by the vector \mathbf{S}^* . Such a dependence results from the shape of the element which varies with the location of the interface, and from the values of the shape functions. One of the main features of our work is the simultaneous calculation of the temperature and of the interface in the transient process; we have seen that the method requires the use of the Newton–Raphson procedure at each time step for linearizing the equations with respect to the nodal temperatures and the components of \mathbf{S}^* .

In order to calculate the derivatives of the coefficients of (16) with respect to the components of \mathbf{S}^* , we use the chain rule of differentiation as follows. Let I be the contribution of an element to a term of the system (16) and let x_i, y_i be the nodal co-ordinates of the element. We will write

$$\frac{\partial I}{\partial S_k} = \sum_i \left(\frac{\partial I}{\partial x_i} \frac{\partial x_i}{\partial S_k} + \frac{\partial I}{\partial y_i} \frac{\partial y_i}{\partial S_k} \right), \quad (35)$$

where S_k is a typical component of \mathbf{S}^* . The partial derivatives of x_i and y_i with respect to S_k are easily obtained from the transformation law of the mesh with the motion of the interface. However, for each term of (16) we need to find an easy way of calculating such derivatives as $\partial I/\partial x_i$ and $\partial I/\partial y_i$.

There exists an interesting transformation which has allowed us to simplify our calculations and which is worth reviewing in the present context by means of a specific example. A typical diffusion term of \mathbf{K} in (16) has the following form:

$$k_{ij}^e = \int_{\Omega^e} k \left(\frac{\partial \psi_i}{\partial x} \frac{\partial \psi_j}{\partial x} + \frac{\partial \psi_i}{\partial y} \frac{\partial \psi_j}{\partial y} \right) d\Omega, \quad (36)$$

where we have limited the integration to the finite element Ω^e ; the domain Ω^e and the shape functions ψ_i depend upon \mathbf{S}^* through the nodal co-ordinates. Let us show how one can easily formulate the derivatives $\partial k_{ij}^e/\partial x_m$ or $\partial k_{ij}^e/\partial y_m$, where x_m, y_m are the nodal co-ordinates of the element.

Let ω be the parent element in a ξ – η co-ordinate system to which Ω^e corresponds through an isoparametric transformation (Figure 4). The Jacobian J of the transformation from (ξ, η) to (x, y) is given by

$$J = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi}. \quad (37)$$

Knowing that

$$x = \sum x_i \psi_i, \quad y = \sum y_i \psi_i \quad (38)$$

through the isoparametric transformation, we find

$$\begin{aligned} \frac{\partial J}{\partial x_i} &= \frac{\partial \psi_i}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial \psi_i}{\partial \eta} \frac{\partial y}{\partial \xi}, \\ \frac{\partial J}{\partial y_i} &= \frac{\partial x}{\partial \xi} \frac{\partial \psi_i}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial \psi_i}{\partial \xi}. \end{aligned} \quad (39)$$

It is then easy to see that

$$\frac{\partial \psi_i}{\partial x} = \frac{\partial \psi_i}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \psi_i}{\partial \eta} \frac{\partial \eta}{\partial x} = \frac{1}{J} \left(\frac{\partial \psi_i}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial \psi_i}{\partial \eta} \frac{\partial y}{\partial \xi} \right) = \frac{1}{J} \frac{\partial J}{\partial x_i}, \quad (40)$$

and similarly

$$\frac{\partial \psi_i}{\partial y} = \frac{1}{J} \frac{\partial J}{\partial y_i}. \quad (41)$$

In view of (36) and (40) we obtain,

$$k_{ij}^e = \int_{\omega} k J^{-1} \left(\frac{\partial J}{\partial x_i} \frac{\partial J}{\partial x_j} + \frac{\partial J}{\partial y_i} \frac{\partial J}{\partial y_j} \right) d\omega, \quad (42)$$

where the integral is now evaluated over the parent element.

On the basis of (37) and (38) one can verify that it is easy to calculate the first and second derivatives of J with respect to the nodal co-ordinates x_k and y_k within the parent element. Similarly, calculating the derivatives of k_{ij}^e with respect to x_k and y_k reduces to a simple numerical integration in the parent element. Although the transformation (41) has facilitated our calculations considerably, obtaining the linearizing form of (16) remains a formidable task. Analytical test problems are essential for ascertaining the correctness of the numerical code.

6. TEST PROBLEMS

The transient algorithm coupled with the remeshing procedure has been tested on two simple one-dimensional problems solved on a single row of finite elements.

First, we solve the heat diffusion equation

$$0 \leq x \leq L: \frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial x^2}, \quad (43)$$

where κ is the heat diffusivity, with the following boundary conditions:

$$\begin{aligned} T(0, t) &= 12.5, \\ T(L, t) &= 7.5 + \sin \omega t \end{aligned} \quad (44)$$

and the initial condition

$$T(x, 0) = 12.5 - 5x/L. \quad (45)$$

The analytical solution to that problem is given by

$$\begin{aligned}
T(x,t) = & 12.5 - 5x/L + \sum_{n=1}^{\infty} A_n \sin(k_n x) \exp(-\kappa k_n^2 t) \\
& + \{ \sin \omega t [a \cos(\gamma x) \text{sh}(\gamma x) + b \sin(\gamma x) \text{ch}(\gamma x)] \\
& + \cos \omega t [a \sin(\gamma x) \text{ch}(\gamma x) - b \cos(\gamma x) \text{sh}(\gamma x)] \} (a^2 + b^2)^{-1},
\end{aligned} \tag{46}$$

with

$$\begin{aligned}
k_n = n\pi/L, \quad \gamma = \left(\frac{\omega}{2\kappa} \right)^{1/2}, \quad a = \text{sh}(\gamma L) \cos(\gamma L), \quad b = \text{ch}(\gamma L) \sin(\gamma L), \\
A_n = \frac{1}{L} (-1)^{n+1} 4k_n \gamma^2 [\gamma^2 + (\gamma - k_n)^2]^{-1} [\gamma^2 + (\gamma + k_n)^2]^{-1}.
\end{aligned} \tag{47}$$

The right-hand side consists of a time-independent response which is linear in x , a transient response and the oscillatory regime. The interface coincides with the isotherm $T = 10$. For the numerical calculation, we have selected $\omega = 2\pi$, $\kappa = 1$, $L = 1$. The mesh is composed of 20 elements of equal size at $t = 0$ and it deforms continuously with the interface.

With a tolerance $\varepsilon = 10^{-3}$, one reaches a time $t = 0.1114$ after 10 time steps. At that time, one cannot distinguish between the analytical and the numerical plots given in Figure 4. In Figure 5 we show a graph of the absolute error; its maximum value is 1.6×10^{-3} despite the relatively permissive value of the tolerance. With $\varepsilon = 10^{-5}$, one obtains a maximum error of 2×10^{-4} , but 21 time steps are now needed to reach the same value of time. The graph of the error for $\varepsilon = 10^{-5}$ is shown in Figure 6 when the temperature at $x = L$ is imposed by means of an algebraic equation (curve (a)) or by a differential constraint (curve (b)). The order of magnitude of the error is the same with both techniques.

As a second test, we wish to study a problem where the latent heat of fusion is taken into account. We consider in Figure 7 a circular cylindrical volume filled with gallium arsenide (its physical

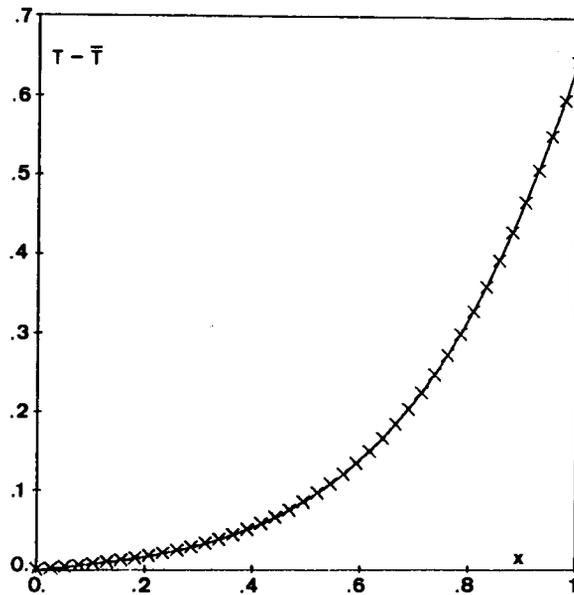


Figure 4. Temperature at time $t = 0.1114$. The analytical solution is represented by a continuous curve, whereas the crosses correspond to the numerical solution

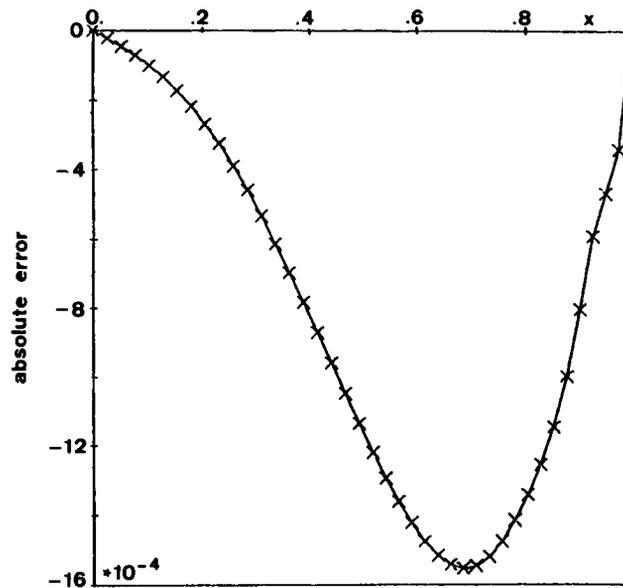


Figure 5. Graph of the local error of the temperature at time $t = 0.1114$ with a tolerance parameter $\varepsilon = 10^{-3}$

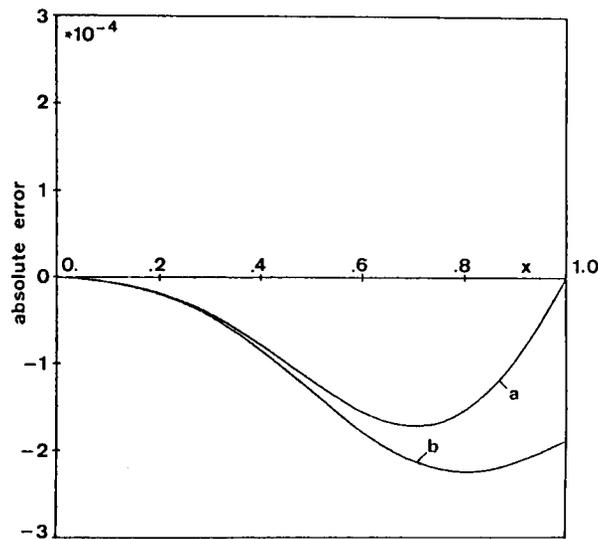


Figure 6. Graph of the local error of the temperature at time $t = 0.1114$ with a tolerance parameter $\varepsilon = 10^{-5}$. The temperature on the right is imposed by means of an algebraic equation for curve (a) and by means of a differential constraint for curve (b)

properties are given in the Appendix). The outer boundary is thermally insulated. The temperature on the lower boundary oscillates in time:

$$T(0, t) = 1491 + 5 \sin(2\pi t) \text{ (in K)}, \quad (48)$$

and $T = 1531$ K on the upper boundary. The interface is located at $T = 1511$ K, which is the melting point isotherm. As an initial condition we select the steady-state solution with $T = 1491$ K on the lower boundary. The initial mesh contains four elements of equal size.

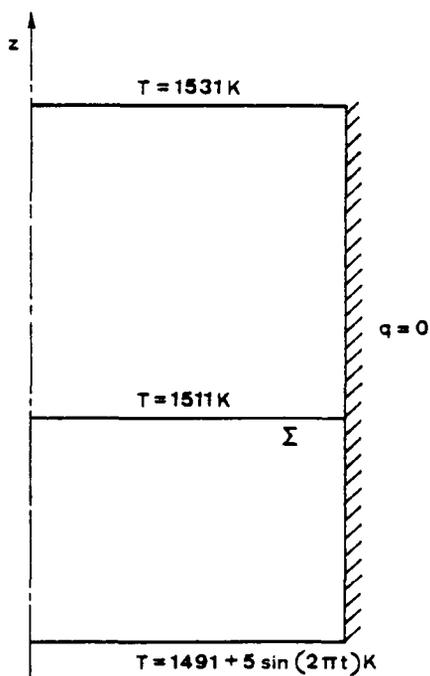


Figure 7. Geometry and boundary conditions of the test problem

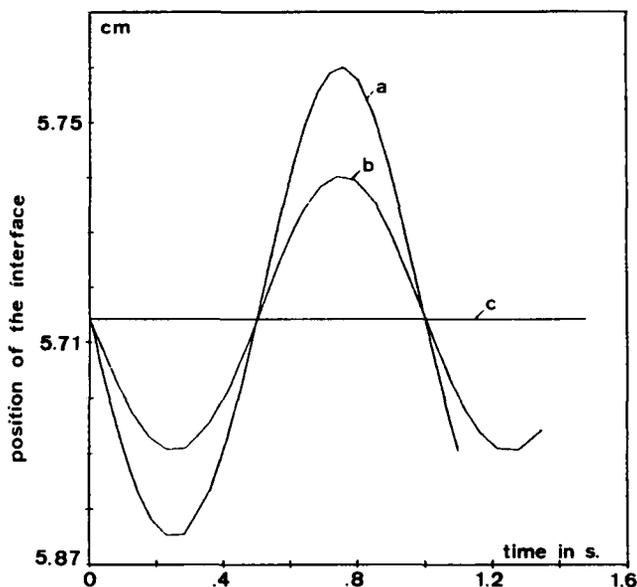


Figure 8. Position of the interface as a function of time for various values of the latent heat of fusion: (a) $\Delta H_f = 0$; (b) $\Delta H_f = 7.2 \times 10^2$; (c) $\Delta H_f = 7.2 \times 10^5 \text{ J kg}^{-1}$

In order to test the interface algorithm we have considered three different values of the latent heat of fusion, i. e. $\Delta H_f = 0$, 7.2×10^2 and $7.2 \times 10^5 \text{ J/kg}$; the latter is the correct physical value. Figure 8 shows the calculated position of the interface as a function of time for these three values.

Each curve corresponds to 30 time steps with a tolerance $\varepsilon = 10^{-5}$. As one might expect, the amplitude of the oscillations decreases when the thermal inertia of the interface increases. The magnitude of the time steps is also increasing with the latent heat of fusion; after 30 time steps, a time lapse of 11 s has been obtained with a vanishing value of ΔH_f and 15 s has been reached for the maximum value. The reason is that, for high values of ΔH_f , the motion of the interface is small, and the corrected location of the interface is almost equal to the predicted value.

7. NUMERICAL SIMULATION OF THE HORIZONTAL GROWTH

We wish to apply our numerical algorithm to predict the motion of the melt and the location of the interface during the horizontal growth of a semi conductor crystal. The geometry of the problem is given in Figure 9. The melt is contained in a four-to-one rectangular cavity. The upper boundary is a free surface, and the fluid does not slip along the walls or along the interface. Here we assume that the temperature is imposed on the boundary and varies linearly with the horizontal co-ordinate. The algorithm is also applicable to more realistic types of boundary conditions.

The data of the problem are given in a non-dimensional form. We select the depth h of the cavity as the characteristic length and the temperature difference δT between the end walls as the characteristic temperature. The Grashof number of the flow is given by

$$Gr = \alpha g \delta T h^3 / \nu^2 \quad (49)$$

where ν is the kinematic viscosity, and the Prandtl number is

$$Pr = \nu / \kappa_1, \quad (50)$$

where κ_1 is the heat diffusivity in the liquid phase. Finally, the Stefan number St is defined by

$$St = \Delta H_f / c_L \delta T. \quad (51)$$

We will consider a case where $Gr = 316,666$ and $Pr = 0.015$, and we will use different values of St . We will assume in this example that the thermal diffusivity is the same in the liquid and in the solid phases. Plots of time-dependent variables will be presented as functions of a non-dimensional time, which we need to multiply by 1250 for a crucible which is 2.5 cm deep and a kinematic viscosity of the melt equal to 0.5 cs. In the sequel, we will refer to dimensional times based on such a scaling.

The initial graded undeformed mesh contains 12×22 finite elements; 22×17 are located in the liquid phase, and 12×5 in the solid phase; the mesh is shown in Figure 10. In a first calculation we assume that the temperature profile on the boundary is time-independent, and that the interface has no thermal inertia, i.e. ΔH_f vanishes ($St = 0$). At that stage, we are not properly calculating crystal growth since the heating elements are not moving, despite the coexistence of solid and liquid

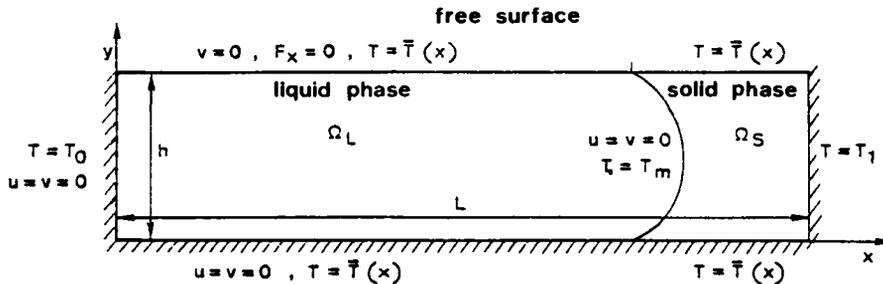


Figure 9. Geometry and boundary conditions of the crystal growth problem



Figure 10. Finite element mesh for calculating the crystal growth

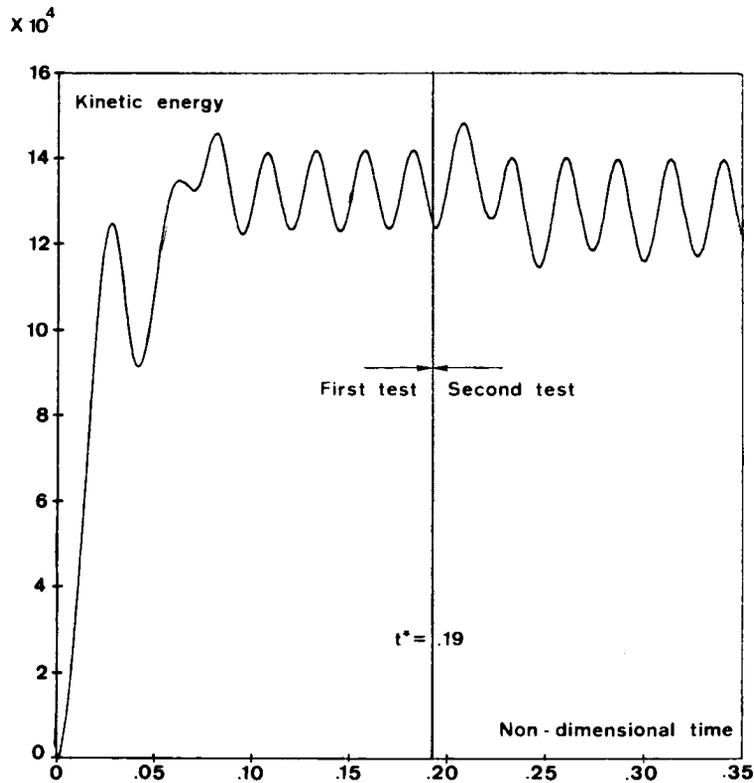


Figure 11. Plot of the non-dimensional kinetic energy as a function of time

phases. However, an oscillatory motion of the melt should induce an important motion of the interface since there is no latent heat of fusion.

The non-dimensional time for this first test goes from 0 to 0.19 (i.e. 0 to 237.5 s). Figure 11 shows the growth of the non-dimensional kinetic energy in the liquid phase. The kinetic energy reaches a periodic regime with a period of 31 s. From our earlier calculations⁹ we know that the Grashof number is slightly above the onset of periodic motion; this is confirmed by the sinusoidal aspect of the kinetic energy as a function of time, and by the Fourier analysis of the kinetic energy shown in Figure 12, where a single peak dominates the others. The location of the interface at mid-distance between the upper and the lower boundaries is shown in Figure 13, and a zoom of the last three periods is given in Figure 14. As we might expect from Figure 12, we find a period of 31 s, which is however clearly subdivided into two subperiods of 15.5 s. This is confirmed by Figure 15 which shows the Fourier analysis of the curve of Figure 14.

The same results were obtained earlier with a decoupled method⁹ where the location of the

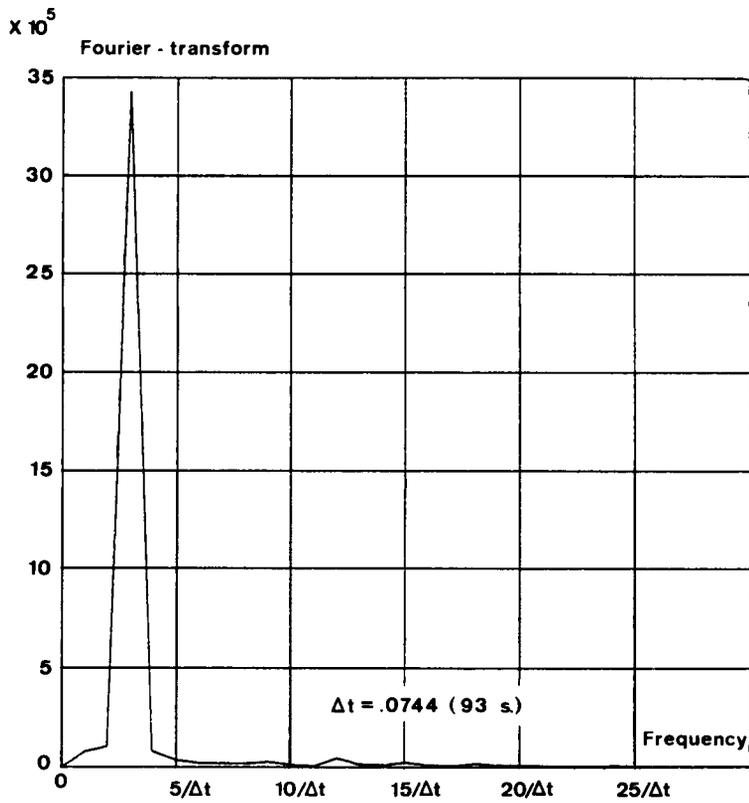


Figure 12. Fourier analysis of the curve of Figure 11

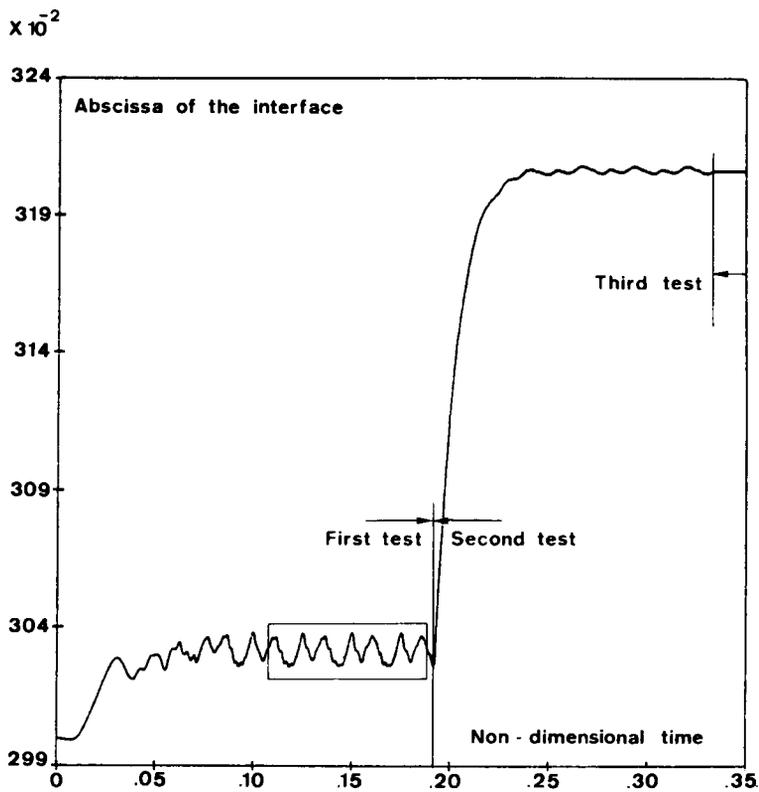


Figure 13. Plot of the location of the interface at mid-distance between the upper and lower boundaries as a function of time

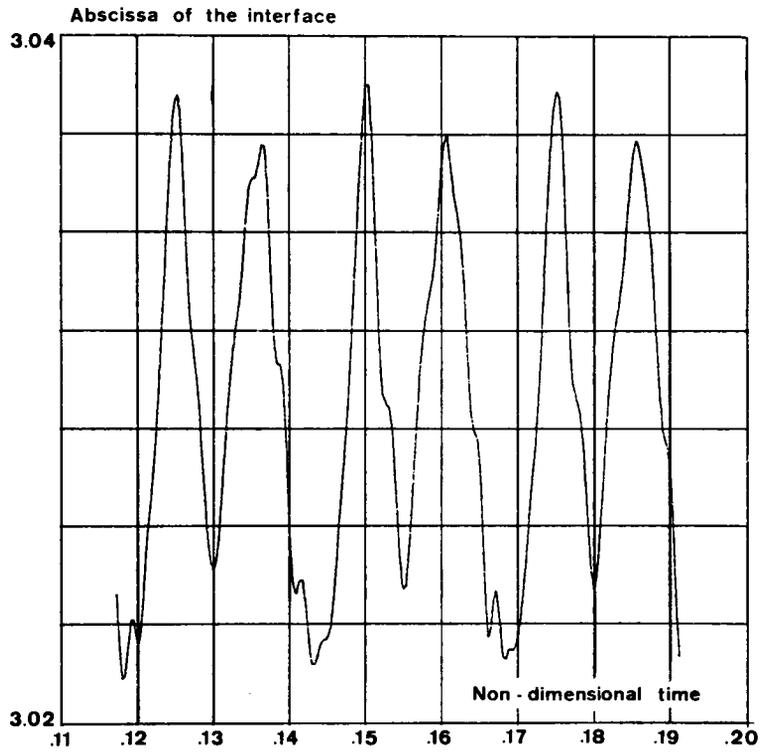


Figure 14. Enlarged view of the framed section of the curve in Figure 13

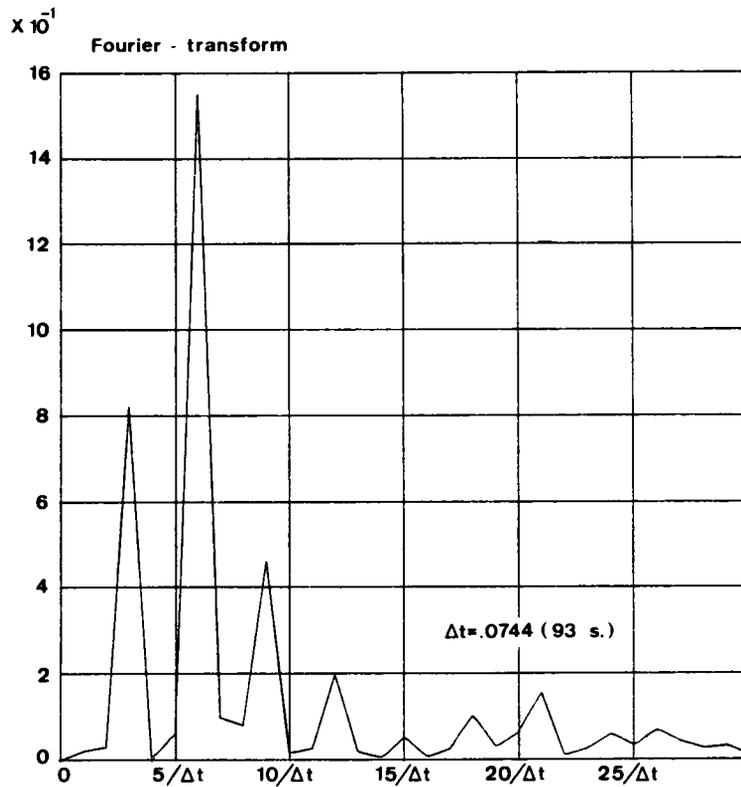


Figure 15. Fourier analysis of the kinetic energy shown in Figure 14

interface was not simultaneously calculated with the temperature field. It was then found that the magnitude of the time step was depending upon the tolerance on the interface location and would often decrease down to small unpractical values. With the present coupled method, the calculation of the velocity field in the Navier–Stokes equations is solely responsible for adjusting the time step. The robustness of the algorithm is quite apparent when one jumps from one type of problem definition to our next calculation at time 0.19 s.

In this second calculation, we wish to calculate the shape of the interface during crystal growth, i.e. when the interface moves across the crucible together with the heating elements, with the simultaneous release of solidification heat. We expect that the latter will bring dramatic changes to the shape of the interface. A typical value of the solidification rate is 1 cm/h, or a non-dimensional velocity of 0.139. Within the time lapse of the first test (0.19 non-dimensional units), the interface would then move across a distance less than 0.03 times the depth of the crucible. Rather than taking into account time dependent boundary conditions, we may as well consider time independent ones and calculate the velocity and temperature distribution when the heat produced per unit length of the interface and per unit time is given by

$$r_{\Sigma} = -\rho \Delta H_f v_{\Sigma} - \rho \Delta H_f \mathbf{v}_c \cdot \mathbf{n} \quad (52)$$

instead of (5). Here, $\mathbf{v}_c \cdot \mathbf{n}$ is the scalar product of the translational velocity of the heating elements and the normal to the interface directed towards the liquid phase, and v_{Σ} is the transient normal velocity of the interface, which should vanish once a steady-state interface has been reached. To obtain the correct shape of the interface during crystallization, we keep the product $\Delta H_f \mathbf{v}_c$ fixed when time proceeds with fixed boundary conditions.

In actual calculations, we often give different values to the factors ΔH_f in the two terms on the right of (52). The second term is fixed and corresponds to the release of latent heat in the actual process, whereas the first one opposes thermal inertia to the transient motion of the interface. Thus, while we keep $\Delta H_f \mathbf{v}_c$ fixed, it is advantageous to reduce ΔH_f in the group $\Delta H_f v_{\Sigma}$ in order to obtain the steady (or periodic) regime within a shorter time. Of course, the transient response obtained with such a procedure has no physical meaning.

For our second example, starting from the configuration at time 0.19 as an initial condition, we have used $St = 120$ for the second term on the right of (43) and $St = 1$ for the first term. Figure 11 shows the behaviour of the kinetic energy as a function of time. The main change is a larger amplitude of the oscillation with a slightly longer period of 33 s. The reason is that the interface is now convex and leaves more room for the periodic motion of the eddies. Figure 13 shows the location of the interface at mid-distance from the plane boundaries; we find that it moves quickly to its new position. No numerical difficulties are associated with this transient motion of the interface, whereas our earlier uncoupled technique was often unstable under transient conditions. The interface is still slightly oscillating with an amplitude of the order of 10^{-3} times the depth of the crucible. At time 0.33, we let the Stefan number in the first term of (52) reach its final value, and we find that the amplitude of the motion of the interface cannot be noticed in Figure 13. At that stage, the amplitude is of the order of 3×10^{-5} times the depth of the crucible, or approximately $1 \mu\text{m}$ with a depth of 2.5 cm. However, the periodic velocity of the interface is small as compared to the velocity of the heating elements, and the periodic motion may just result in a modulation of the growth velocity.

Let us now examine the nature of the oscillatory motion taking place in the crucible. Figure 16 shows a period of the kinetic energy as a function of time, on which we have indicated six specific times at which we want to inspect the streamlines and the isotherms. They are shown in Figure 17 where we find an oscillatory mechanism which is very similar to what was shown in the first paper of this series. Here, a solid phase coexists with a liquid phase and the shape and the interface is strongly affected by the release of the latent heat of fusion.

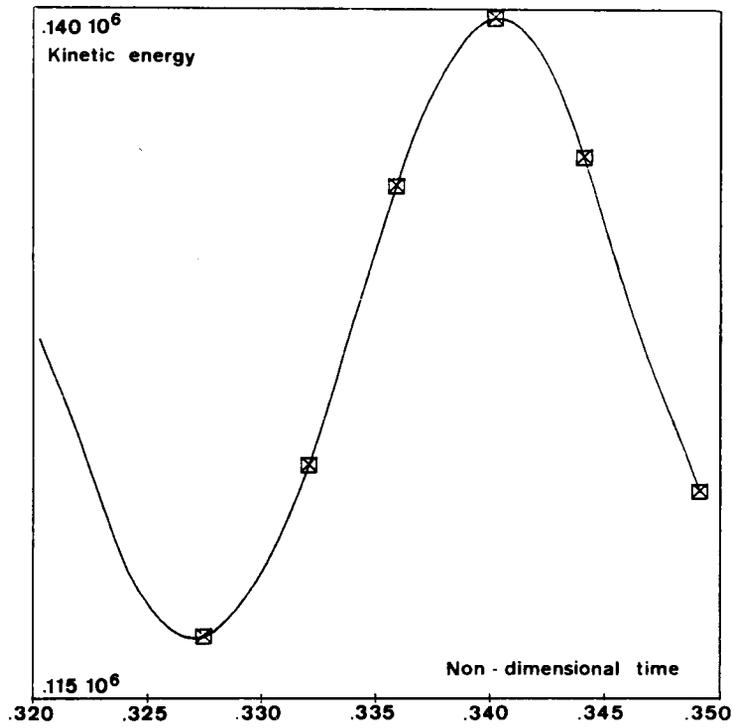


Figure 16. Period of the kinetic energy and instants at which the streamlines and the isotherms are shown in Figure 17

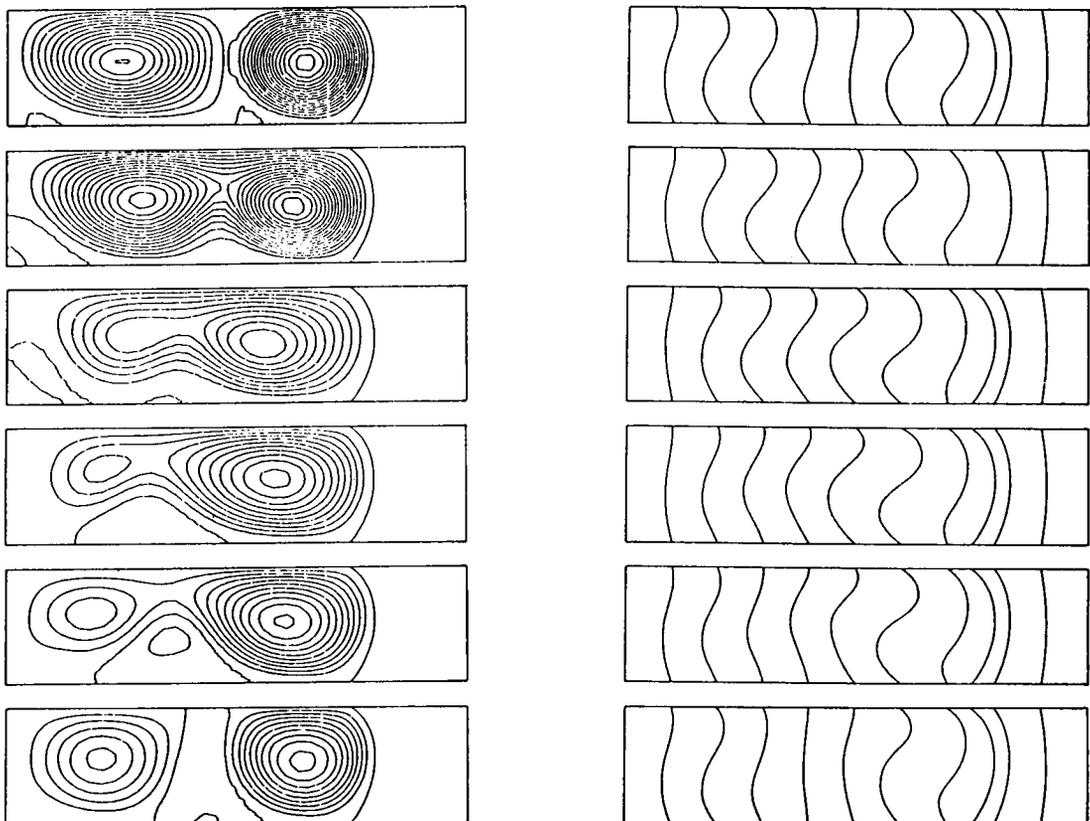


Figure 17. Streamlines and isotherms during the period of the kinetic energy shown in Figure 16

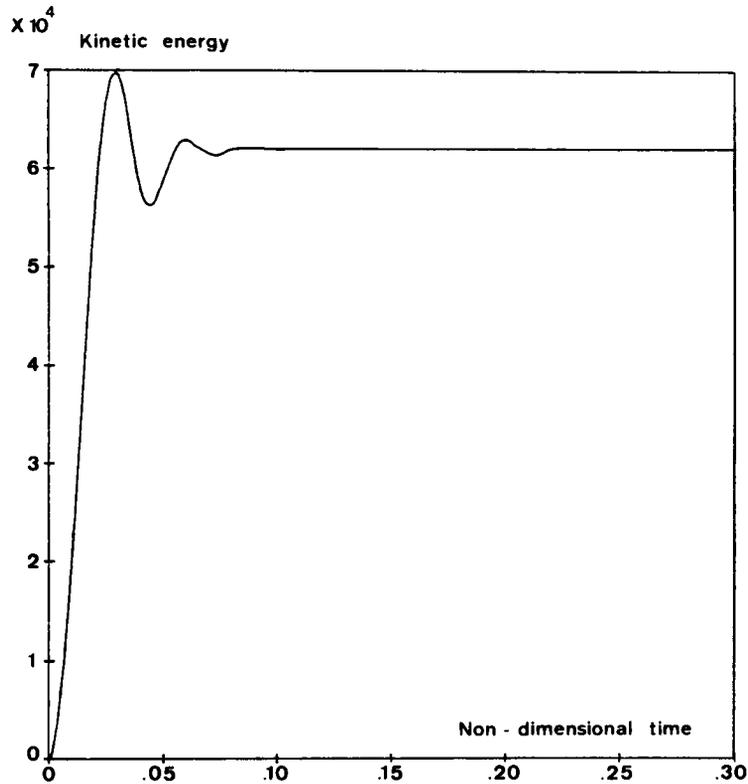


Figure 18. Plot of the non-dimensional kinetic energy as a function of time; the melting isotherm is anchored on the boundary at mid-distance from the end-walls

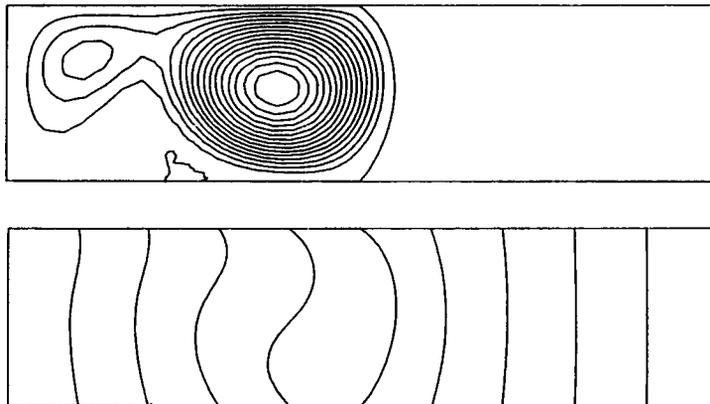


Figure 19. Streamlines and isotherms corresponding to the situation of Figure 18

For our third example, we have selected 0.5 as the non-dimensional melting temperature. With the boundary conditions of Figure 9, the interface should thus be anchored at mid-distance between the end-walls. We have used the same group $\Delta H_f v_c \cdot n$ as in our previous example with a lower value of ΔH_f for the first term on the right of (52). Figure 18 shows the behaviour of the kinetic energy as a function of time. The transient periodic behaviour is damped down very quickly. A steady state is reached within a non-dimensional interval of 0.1, or 125 s. The streamlines and the isotherms are shown in Figure 19. The geometrical ratio of the melt is such that an oscillatory

motion cannot occur at the present value of the Grashof number. The interface has also reached a steady-state position.

8. CONCLUSIONS

In Parts I and II of the present series, we have found that, despite the existence of a two-dimensional mechanism for explaining the oscillatory motion of the melt, three-dimensional effects might dominate the flow. Whatever be the future developments in the simulation of the Bridgman growth, it will be necessary to select an appropriate algorithm for locating the interface. The algorithm should also be applicable to the simulation of the vertical Bridgman growth and the Czochralski growth.

Our earlier investigations has shown that locating an interface in a transient problem is a delicate task, in particular when the release of latent heat of fusion is taken into account. In the present paper, we have shown that the coupling between the energy equation and the interface condition is such that, in our implicit time-marching technique, the corresponding discretized equations need to be solved simultaneously. Our scheme has been found accurate and robust.

We found that the release of latent heat of fusion has a major influence upon the shape of the interface. However, in order to obtain a snapshot of the isotherms, one can accelerate the transient phenomena by separating the release of heat at the freezing front into transient and steady contributions.

Our examples show the necessity of imposing realistic boundary conditions in our future developments, which will take into account such phenomena as radiation and heat convection on the walls of the crucible and at the free surface.

ACKNOWLEDGEMENTS

P. Wouters and J. J. Van Schaftingen wish to acknowledge a scholarship from the Belgian 'Fonds National de la Recherche Scientifique'.

APPENDIX: ESTIMATED PHYSICAL PROPERTIES OF GALLIUM ARSENIDE

Solid phase:	thermal conductivity	$k_s = 7.12 \text{ Wm}^{-1} \text{ K}^{-1}$
	specific heat	$c_s = 424 \text{ JK}^{-1} \text{ kg}^{-1}$
	specific mass	$\rho_s = 5.17 \times 10^3 \text{ kgm}^{-3}$
Liquid phase:	thermal conductivity	$k_l = 17.8 \text{ Wm}^{-1} \text{ K}^{-1}$
	specific heat	$c_l = 434 \text{ JK}^{-1} \text{ kg}^{-1}$
	specific mass	$\rho_l = 5.71 \times 10^3 \text{ kgm}^{-3}$
	thermal dilatation	$\alpha = 1.87 \times 10^{-4} \text{ K}^{-1}$
	kinematic viscosity	$\nu = 5 \times 10^7 \text{ m}^2 \text{ s}^{-1}$
Interface:	melting temperature	$T_m = 1511 \text{ K}$
	latent heat of fusion	$\Delta H_f = 7.17 \times 10^5 \text{ Jkg}^{-1}$

REFERENCES

1. M. J. Crochet, F. T. Geyling and J. J. van Schaftingen, 'Numerical simulation of the horizontal Bridgman growth. Part I: two-dimensional flow', *Int. j. numer. methods fluids*, **7**, 29–47 (1987).
2. S. Dupont, J. M. Marchal, M. J. Crochet and F. T. Geyling, 'Numerical simulation of the horizontal Bridgman growth. Part II: three-dimensional flow', *Int. j. numer. methods fluids*, **7**, 49–67 (1987).

3. F. Rosenberger, *Fundamentals of Crystal Growth I*, Springer Series in Solid-State Sciences 5, Springer Verlag, 1979.
4. D. T. J. Hurle, 'Current growth techniques', in W. Bardsley, D. T. J. Hurle, and J. B. Mullin, (eds), *Crystal Growth: a Tutorial Approach*, North Holland, 1979.
5. D. T. J. Hurle and E. Jakeman, 'Introduction to the techniques of crystal growth', *Physicochem. Hydrodyn.*, **2**, 263 (1981).
6. D. R. Lynch, 'Unified approach to simulation on deforming elements with application to phase change problems', *J. Comput. Phys.*, **47**, 387 (1982).
7. J. Yoo and B. Rubinsky, 'Numerical computation using finite elements for the moving interface in heat transfer problems with phase transformation', *Num. Heat Transfer*, **6**, 209 (1983).
8. H. M. Ettouney and R. A. Brown, 'Finite element methods for steady solidification problems', *J. Comput. Phys.*, **49**, 118 (1983).
9. M. J. Crochet, F. T. Geyling and J. J. Van Schaftingen, 'Finite element method for calculating the horizontal Bridgman growth of semi-conductor crystals', in G. F. Carey and J. T. Oden (eds), in *Finite elements in Fluids VI*, Wiley, 321 (1985).
10. P. M. Gresho, R. L. Lee and R. L. Sani, 'On the time-dependent solution of the incompressible Navier–Stokes equations in two and three dimensions', in *Recent Advances in Numerical Methods in Fluids*, Pineridge Press, 1980.
11. D. R. Lynch and W. G. Gray, 'Finite element simulation of flow in deforming regions', *J. Comput. Phys.*, **36**, 135 (1980).
12. G. Dahlquist, 'A special stability problem for linear multistep methods', *BIT*, **3**, 27 (1963).