Journal of Computational Physics **161,** 537–557 (2000)
doi:10.1006/jcph.2000.6511, available online at http://www.idealibrary.com on IDEAL®

The Numerical Solution of One-Dimensional Phase Change Problems Using an Adaptive Moving Mesh Method

J. A. Mackenzie and M. L. Robertson

*Department of Mathematics, University of Strathclyde, Livingstone Tower, 26 Richmond Street,
Glasgow, Scotland G1 1XH*
E-mail: {jam, ta.mrob}@maths.strath.ac.uk

Received August 3, 1999; revised March 22, 2000

An adaptive moving mesh method is developed for the numerical solution of an enthalpy formulation of heat conduction problems with a phase change. The algorithm is based on a very simple mesh modification strategy that allows the smooth evolution of mesh nodes to track interfaces. At each time step the nonlinear enthalpy equation is solved using a novel semi-implicit moving mesh discretisation which is shown to possess a unique solution. Numerical examples are given for a two-phase freezing problem, a model of a spot-welding process, and a three-phase problem with a varying number of interfaces. These test cases demonstrate the accuracy and effectiveness of the overall strategy.

1. INTRODUCTION

A large number of important physical processes involve heat conduction and materials undergoing a change of phase. Examples include semiconductor design, geophysics, cryosurgery, and industrial applications involving metals, oil, and plastics [22]. These problems are often collectively called Stefan or moving boundary value problems. Unfortunately, analytical solutions are only available for a limited number of model examples and hence the solution of most practical cases requires the use of numerical techniques.

What makes these problems difficult to solve is the presence of the moving boundary at which the material is changing phase. A number of numerical methods have been proposed which essentially fall into two categories: front-tracking methods and enthalpy methods (see [8, 15]).

Front-tracking techniques is the term usually applied to methods that explicitly require the Stefan, or equivalent jump, condition to be satisfied on the moving boundary while solving the heat conduction equations in either phase. The location of the boundary is therefore

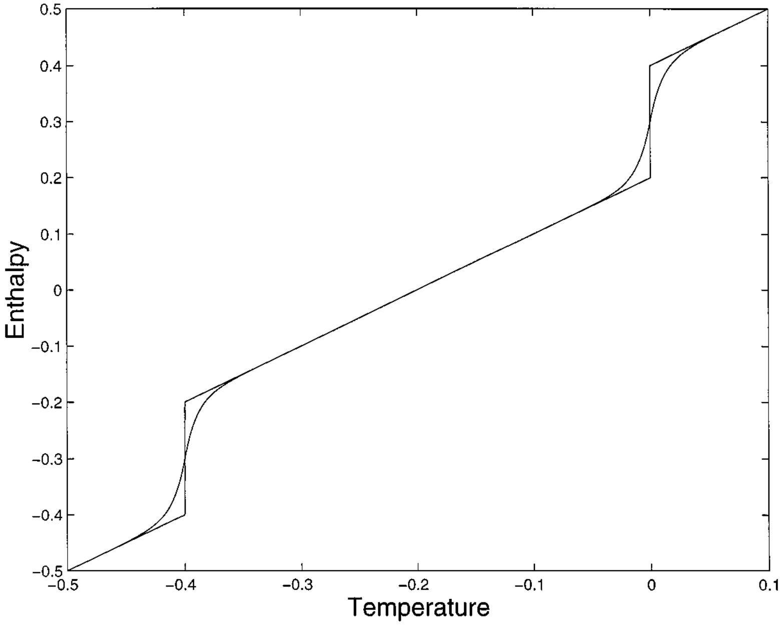

FIG. 1. Smooth and discontinuous temperature-enthalpy functions for a three-phase problem; $H_{ref} = 0$, $C_1 = C_2 = 1$, $T_{m_1} = -0.4$, $T_{m_2} = 0$, and $\varepsilon_1 = \varepsilon_2 = 1E{-}2$.

central to the accuracy and applicability of this approach. However, in multidimensions it is not uncommon for the phase boundary to develop cusps and to double back on itself and generally become difficult to track.

In an attempt to avoid the need to know the location of the phase boundary, the heat conduction equations can be reformulated in terms of the enthalpy which is the sum of sensible and latent heats. Using this formulation the energy balance at the phase boundary is satisfied automatically and if the location of the boundary is required then it can be determined a posteriori. Enthalpy methods therefore appear to avoid some of the difficulties of front-tracking methods.

However, if the material in question changes phase at a specified temperature then the temperature–enthalpy relationship has a jump discontinuity at the melting temperature (see Fig. 1). For these materials a naive discretisation of the enthalpy equation on a uniform grid is well known to predict non-physical features such as a step-like movement of the phase boundary and spurious temperature plateaux [8]. Various ways of eliminating these undesirable features have been proposed including specialised post-processing techniques [24] and the judicious choice of time steps. These methods work reasonably well for one-dimensional problems but their application to multidimensional examples seems less clear.

A second approach is to smooth the temperature–enthalpy relationship so that it is at least continuous (see, for example, [19] and Section 2). A smoothed temperature–enthalpy relationship can also be used to model materials that change phase over a temperature range rather than at a specified temperature [12]. If one uses a stationary grid then smoothing the enthalpy function has to be done carefully as it has been observed that if the amount of smoothing is too large then the numerical results can become inaccurate [23]. Reducing the level of smoothing improves the accuracy but eventually the step-like behaviour of the movement of the phase boundary reappears.

The simplest way of avoiding non-physical behaviour using the enthalpy formulation is to reduce the spatial step size. However, if this is done uniformly over the whole domain then the overall method would be computationally expensive. The mesh spacing need only be refined around the position of the moving phase change boundary which suggests some form of adaptive mesh algorithm would be useful. Within a finite element context this is usually achieved using the *h*-method of adaptation, where the mesh is locally refined or coarsened by adding or deleting points [20, 21]. A less popular approach is to use the so-called *r*-refinement method where mesh points are moved throughout the domain while the connectivity of the mesh is kept fixed. The main reason for the lack of popularity of this approach is the difficulty involved in controlling the geometry of the mesh elements. If this is not done with care then mesh tangling and elements with negative areas can easily arise. However, the development of a robust *r*-adaptive method is attractive in that it intuitively should be able to accurately resolve and follow important solution features. The coding involved in a moving mesh method is also simpler than an *h*-method, which requires a considerably more complicated data structure.

There has been much recent interest in the development of moving mesh methods for the solution of problems with steep solution fronts such as travelling wave solutions in reaction-diffusion systems and boundary and shear layers in fluid dynamics calculations [1, 3, 16]. At the heart of these methods is the grid movement strategy which is usually based on the idea of mesh equidistribution where a positive monitor function is evenly distributed between the available mesh nodes.

The first aim of this paper is to show how a very simple moving mesh method can be used to solve a smoothed enthalpy formulation of the heat conduction equations. The mesh movement algorithm is based on the equidistribution of an analytically integrable monitor function which avoids the need to discretise the equidistribution principle and automatically leads to grids that evolve smoothly in time. The approach is similar to that used by Farrell and Drury [14] to solve nonlinear hyperbolic problems.

As the moving grid method aims to cluster mesh points around the phase change interface, it is clear that we require some form of an implicit discretisation. Even on a stationary grid one has to be careful that unique solutions exist of the resulting nonlinear algebraic systems. Using a moving grid introduces convection-like terms from the semi-Lagrangian formulation of the original problem. A second aim of this paper is to consider a novel semi-implicit discretisation of these equations and to prove that the resulting nonlinear algebraic systems arising at each time step have unique solutions.

The layout of the rest of this paper is as follows. In the next section we present a smoothed enthalpy formulation of the heat conduction equations. In Section 3 we describe the semi-implicit discretisation of the enthalpy equation on a moving mesh. In Section 4 we describe the mesh movement strategy. Finally, in Section 5 we apply the moving mesh method to the solution of a two-phase freezing problem, to a model of a spot-welding process, and to a three-phase problem with a varying number of interfaces.

2. THE GOVERNING EQUATIONS

The governing equations for multiphase one-dimensional heat conduction are

$$C_i(T_i)\frac{\partial T_i}{\partial t} = \frac{\partial}{\partial x}\left(k_i(T_i)\frac{\partial T_i}{\partial x}\right) + \varphi_i(T_i), \tag{1}$$

where the index $i = 1, \ldots, NP$ corresponds to the NP separate phases. Here $C_i(T_i) = \rho c_i$, and $\rho, c_i, k_i(T_i)$, and $T_i(x,t)$ denote the volumetric heat capacities, the density (assumed the same in each phase), the specific heats, the thermal conductivities, and the temperatures, respectively. Here, φ_i represents possible body heating or cooling terms. If a phase change occurs between phase i and phase $i+1$ at a specific temperature $T = T_{m_i}$, and we denote the position of the phase change boundary by $x = s_i(t)$, then an energy balance gives rise to the conditions

$$T_i|_{s_i(t)} = T_{i+1}|_{s_i(t)} = T_{m_i} \tag{2}$$

$$k_{i+1}\frac{\partial T_{i+1}}{\partial x}\bigg|_{s_i(t)} - k_i\frac{\partial T_i}{\partial x}\bigg|_{s_i(t)} = \lambda_i \frac{\mathrm{d}s_i(t)}{\mathrm{d}t}, \tag{3}$$

where λ_i is the latent heat per unit volume involved in the phase change.

To reformulate this problem we introduce an enthalpy function which represents the sum of sensible and latent heats and is given by

$$H(T) = \begin{cases} \int_{T_{ref}}^{T} C(\xi)\,d\xi, & T < T_{m_1} \\ H\left(T_{m_i}^-\right) + \lambda_i + \int_{T_{m_i}}^{T} C(\xi)\,d\xi, & T_{m_i} < T < T_{m_{i+1}}, i = 1, \ldots, NP-2, \\ H\left(T_{m_{NP-1}}^-\right) + \lambda_{NP-1} + \int_{T_{m_{NP-1}}}^{T} C(\xi)\,d\xi, & T_{m_{NP-1}} < T, \end{cases} \tag{4}$$

where $H(T_{m_i}) = \lim_{\delta\to 0^-} H(T_{m_i} + \delta)$ and T_{ref} is any reference temperature below T_{m_1}. Equations (1) and conditions (2) and (3) can then be written as the one equation

$$\frac{\partial H}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \varphi, \qquad \text{where } \left.\begin{aligned} k(T) &= k_i(T) \\ \phi(T) &= \phi_i(T) \\ C(T) &= C_i(T) \end{aligned}\right\} T_{m_{i-1}} < T < T_{m_i}. \tag{5}$$

If the thermal conductivities are temperature dependent then (5) can be linearised by applying the Kirchoff transformation

$$u(T) = \int_{T_{ref}}^{T} k(\xi)\,d\xi. \tag{6}$$

In terms of this new variable Eq. (5) is simplified to

$$\frac{\partial H}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \varphi(u). \tag{7}$$

This equation can then be solved for u and transformed using (6) to find the temperature.

If C_i is constant in each phase then we see from (4) that $H(T)$ is a linear function with jump discontinuities at the phase change temperatures (see Fig. 1). For the reasons outlined in the Introduction, various attempts have been made to smooth out these discontinuities in H. Based on a continuously differentiable relationship suggested by Egolf and Manz [12] for two-phase problems, we consider the function

$$H(T) = \begin{cases} H_{ref} + C_1\left(T - T_{m_1}\right) + \frac{\lambda_1}{2}\exp\left(-\frac{|T-T_{m_1}|}{\varepsilon_1^-}\right), & T \le T_{m_1} \\ H\left(T_{m_{i-1}}\right) + \frac{\lambda_{i-1}}{2} + C_i\left(T - T_{m_{i-1}}\right) & \\ \quad - \frac{\lambda_{i-1}}{2}\exp\left(-\frac{|T-T_{m_{i-1}}|}{\varepsilon_{i-1}^+}\right) + \frac{\lambda_i}{2}\exp\left(-\frac{|T-T_{m_i}|}{\varepsilon_i^-}\right) & T_{m_{i-1}} \le T \le T_{m_i}, \\ & \quad i = 2, \ldots, NP-1 \\ H\left(T_{m_{NP-1}}\right) + \frac{\lambda_{NP-1}}{2} + C_{NP}\left(T - T_{m_{NP-1}}\right) & \\ \quad - \frac{\lambda_{NP-1}}{2}\exp\left(-\frac{|T-T_{m_{NP-1}}|}{\varepsilon_{NP-1}^+}\right), & T \ge T_{m_{NP-1}}, \end{cases} \tag{8}$$

where ε_i^- and ε_i^+ determine the rates at which the temperature-enthalpy function asymptotes to the linear relationship away from the phase change temperature T_{m_i} (see Fig. 1). We will assume that $\varepsilon_i^- \ll (T_{m_i} - T_{m_{i-1}})$ and $\varepsilon_i^+ \ll (T_{m_{i+1}} - T_{m_i})$, and hence for $H(T)$ to essentially be continuously differentiable at T_{m_i} we require that

$$C_i - C_{i+1} = \frac{\lambda_i}{2}\left(\frac{1}{\varepsilon_i^+} - \frac{1}{\varepsilon_i^-}\right). \tag{9}$$

If $\varepsilon_i = \varepsilon_i^- + \varepsilon_i^+$ then we can define a modified Stefan number for phase change i as

$$St_i^* = \frac{(C_{i+1} - C_i)\varepsilon_i}{\lambda_i}. \tag{10}$$

The simultaneous satisfaction of (9) and (10) gives rise to quadratic equations for ε_i^- and ε_i^+ which have physically relevant solutions given by

$$\varepsilon_i^- = \frac{\varepsilon_i}{2St_i^*}\left(1 + St_i^* - \sqrt{1 + St_i^*}\right) \tag{11}$$

and

$$\varepsilon_i^+ = \frac{\varepsilon_i}{2St_i^*}\left(St_i^* - 1 + \sqrt{1 + St_i^*}\right). \tag{12}$$

In the limit that $C_{i+1} \to C_i$ we have $\varepsilon_i^+ \to \varepsilon_i^- = \varepsilon_i/2$. Figure 1 shows the smoothed enthalpy function of a three-phase problem considered in Section 5.

The original motivation for this model was to describe mixtures and glassy substances that have a continuous enthalpy transition as a function of temperature from a pure solid phase to a pure liquid phase.

3. AN SEMI-IMPLICIT MOVING MESH DISCRETISATION

We now consider the numerical solution of (7) for $(x, t) \in \Omega = (x_L, x_R) \times (0, T)$. We assume that the domain is partitioned into strips such that

$$\Omega = \bigcup_{0 \le n \le N_t - 1} (x_L, x_R) \times [t^n, t^{n+1}).$$

Each strip is made up of two spatial grids

$$\mathbf{x}_\Delta^n = \left\{x_L = x_0^n < x_1^n < \cdots < x_{N-1}^n < x_N^n = x_R\right\}$$

and

$$\mathbf{x}_\Delta^{n+1} = \left\{x_L = x_0^{n+1} < x_1^{n+1} < \cdots < x_{N-1}^{n+1} < x_N^{n+1} = x_R\right\}.$$

In the next section we describe how the grid is generated at time level t^{n+1}. For the moment we will assume that it is given.

In order to incorporate the movement of the grid we require a discretisation of the semi-Lagrangian formulation of (7) which takes the form

$$\frac{\partial H}{\partial t} - \frac{dx}{dt}\frac{\partial H}{\partial x} = \frac{\partial^2 u}{\partial x^2} + \varphi(u). \tag{13}$$

To describe the discretisation we first introduce some notation. Let $h_j^n = x_j^n - x_{j-1}^n$, $\overline{h}_j^n = (h_{j+1}^n + h_j^n)/2$, and $\Delta t^n = t^n - t^{n-1}$. We will also denote

$$\mathbf{u}_\Delta^n = \left(u_0^n, u_1^n, \ldots, u_N^n\right)^T, \tag{14}$$

where u_j^n represents the approximation of $u(x_j^n,\ t^n)$. Similarly, let

$$\mathbf{H}_\Delta^n = \left(H_0^n, H_1^n, \ldots, H_N^n\right)^T = \left(H\left(u_0^n\right), H\left(u_1^n\right), \ldots, H\left(u_N^n\right)\right)^T \tag{15}$$

and

$$\varphi_\Delta^n = \left(\varphi_0^n, \varphi_1^n, \ldots, \varphi_N^n\right)^T = \left(\varphi\left(u_0^n\right), \varphi\left(u_1^n\right), \ldots, \varphi\left(u_N^n\right)\right)^T. \tag{16}$$

We consider the following discretisation of (13)

$$\begin{aligned}\frac{H_j^{n+1} - H_j^n}{\Delta t^{n+1}} &- \frac{x_j^{n+1} - x_j^n}{\Delta t^{n+1}}\left(\frac{H_{j+1}^n - H_{j-1}^n}{h_{j+1}^n + h_{j+1}^n}\right)\\ &= \frac{2}{h_{j+1}^{n+1} + h_j^{n+1}}\left(\frac{u_{j+1}^{n+1} - u_j^{n+1}}{h_{j+1}^{n+1}} - \frac{u_j^{n+1} - u_{j-1}^{n+1}}{h_j^{n+1}}\right) + \varphi_j^{n+1}.\end{aligned} \tag{17}$$

If Dirichlet conditions are given at $x = x_L$ and $x = x_R$ then (17) is applied for $j = 1, 2 \ldots, N-1$. If a physically relevant derivative boundary condition of the form

$$\frac{\partial u}{\partial x} = \alpha(t)u + g(t), \qquad \alpha(t) \geq 0, \tag{18}$$

is specified at $x = x_L$ then this is discretised at $t = t^{n+1}$ by introducing the fictitious unknown u_{-1}^{n+1} outside the domain at $x = x_L - h_1^{n+1}$ and a central difference is used to write

$$\frac{u_1^{n+1} - u_{-1}^{n+1}}{2h_1^{n+1}} = \alpha^{n+1}u_0^{n+1} + g^{n+1}. \tag{19}$$

We then apply the difference scheme (17) at $j = 0$ so that u_{-1}^{n+1} can be eliminated. A similar procedure can be carried out if a derivative condition of the form

$$\frac{\partial u}{\partial x} = \alpha(t)u + g(t), \qquad \alpha(t) \leq 0, \tag{20}$$

is specified at $x = x_R$.

Note that the terms on the righthand side of (17) are treated implicitly, whereas the term introduced from the mesh movement it treated explicitly. Since the grid will be clustered around the moving front we require an implicit discretisation of the heat conduction and source terms to allow the use of reasonably large time steps. We will see below that the explicit treatment of the $\dot{x}H_x$ term allows us to establish the existence and uniqueness of a solution of Eqs. (17).

3.1. Iterative solution of the nonlinear system of equations. The calculation of $\mathbf{u}_\Delta^{n+1}$ requires the solution of the nonlinear algebraic equations (17) which after multiplying through by Δt^{n+1} can be written in the form

$$\mathbf{F}\big(\mathbf{u}_\Delta^{n+1}\big) \equiv A\mathbf{u}_\Delta^{n+1} + \mathbf{H}_\Delta^{n+1} - \Delta t^{n+1}\varphi_\Delta^{n+1} + \mathbf{r}^n = \mathbf{0}, \tag{21}$$

where $\mathbf{r}^n$ is a vector that is independent of $\mathbf{u}_\Delta^{n+1}$. The tridiagonal matrix A has positive diagonal elements and negative off-diagonal elements and can easily be shown to be an irreducibly diagonally dominant M-matrix. An immediate question is whether a unique solution of (21) exists. If $\varphi(u) = 0$ we can write (21) in the form

$$\mathbf{F}\big(\mathbf{u}_\Delta^{n+1}\big) = A\mathbf{u}_\Delta^{n+1} + \phi\big(\mathbf{u}_\Delta^{n+1}\big) = 0, \tag{22}$$

where ϕ is continuous, diagonal, and monotone in each component. Existence and uniqueness is given in the following theorem [13].

THEOREM 3.1. *If*

$$G(v_j) \equiv a_{jj}v_j + \phi_j(v_j) + \sum_{i=1}^{j-1} a_{ji}u_i^{[n+1,s+1]} + \sum_{i=j+1}^{N-1} a_{ji}u_i^{[n+1,s]} = 0 \tag{23}$$

then the nonlinear SOR sequence $\{\mathbf{u}_\Delta^{[n+1,s]}\}$ *given by*

$$u_j^{[n+1,s+1]} = u_j^{[n+1,s]} + \omega\big(v_j - u_j^{[n+1,s]}\big),$$

if

$$\big\{\big(u_j^{[n+1,s]} + \omega(v_j - u_j^{[n+1,s]}\big)\big\}u_j^{[n+1,s]} > 0$$

and

$$u_j^{[n+1,s+1]} = v_j, \qquad \textit{otherwise},$$

converges globally to the unique solution of (22) *for all* $\omega \in (0, 2)$.

Note that an implicit discretisation of the $\dot{x}H_x$ term would lead to a system similar to (22) but the mapping ϕ would not be diagonal and we could not use the above theorem.

In practice, we use Newton's method to solve (21). Since the smoothed enthalpy function (8) is continuously differentiable we have no difficulty in defining the Newton iteration. If Newton's method fails to converge then we apply the nonlinear SOR iteration which requires the solution of Eqs. (23). The next question is whether unique solutions exist to these scalar nonlinear equations. This is clearly the case since ϕ_j is a monotonically increasing function and $a_{jj} > 0$ and hence $G \to \pm\infty$ as $v_j \to \pm\infty$ and so a solution exists. These scalar problems are solved using Newton's method which again is well defined due to the smoothness properties of H. If Newton's method fails to converge then we use a bisection procedure to provide an adequate initial guess.

4. MOVING THE MESH

4.1. Grid equidistribution. The discretisation described in the previous section can be used when grids $\mathbf{x}_\Delta^n$ and $\mathbf{x}_\Delta^{n+1}$ are available. Assuming that $\mathbf{x}_\Delta^n$ has already been determined it remains to describe how to calculate $\mathbf{x}_\Delta^{n+1}$. At each time step a new grid is generated based on the idea of mesh equidistribution. A computational grid is said to be equidistributing if

$$\int_{x_{j-1}}^{x_j} M(x)\,\mathrm{d}x = \int_{x_j}^{x_{j+1}} M(x)\,\mathrm{d}x = \frac{1}{N}\int_{x_L}^{x_R} M(x)\,\mathrm{d}x, \qquad j = 1, \ldots, N, \tag{24}$$

where $M(x) > 0$ is a monitor function which should be related to the local difficulty in solving the problem. The theoretical basis of mesh equidistribution has been established for a number of approximation problems such as optimal knot placements for spline collocation approximations of two-point boundary value problems [9] and the characterisation of optimal grids for piecewise polynomial interpolation [6].

In practice the monitor function is based on the numerical solution and the equidistribution conditions are discretised. For example, use of the mid-point rule to discretise (24) gives rise to the set of equations

$$M_{j+\frac{1}{2}}(x_{j+1} - x_j) = M_{j-\frac{1}{2}}(x_j - x_{j-1}), \qquad j = 1, 2, \ldots, N-1, \tag{25}$$

where $M_{j+1/2}$ is an approximation of $M(x_{j+1/2})$. The coupled set of Eqs. (21) and (25) then have to be solved simultaneously for $\{u_j, x_j\}_{j=1}^{N-1}$.

The quality of the adaptive grid depends crucially on the monitor function. A popular choice for problems with moving steep fronts is

$$M(x) = \sqrt{1 + \alpha\left(\frac{\partial H}{\partial x}\right)^2}, \qquad \alpha > 0, \tag{26}$$

which represents a scaled solution arc-length. Through numerical experimentation, Duncan [11] concludes that the grids obtained by equidistributing this monitor function are not sufficiently clustered at the phase change boundary and that it is unsuitable for multi-phase problems with different latent heat jumps. Instead, Duncan proposes the monitor function

$$M(x) = 1 + \eta\left|\frac{\partial F(H)}{\partial x}\right|, \tag{27}$$

where the parameter $\eta > 0$ controls the number of mesh points in the phase transition regions. The function $F(H)$ is chosen to give a uniform grid in pure phase regions and to give equal weighting to each phase transition region, regardless of the latent heat jumps.

The meshes obtained by equidistributing (26) or (27) will have very large ratios between adjacent grid cells. These can lead to a deterioration in solution accuracy and to very stiff systems of ordinary differential equations if a method of lines approach is used. A common strategy to alleviate this problem is to smooth the discrete values of the monitor function before attempting to equidistribute them [10]. This additional step can become computationally expensive if a smooth grid is required. Furthermore, it is far from clear what the correct balance should be between the competing effects of adaptivity provided by the monitor function, and the smoothing process.

4.2. An integrable monitor function. Discretisation of the equidistribution principle can be avoided if the monitor function is analytically integrable. If the monitor function is chosen carefully then the resulting grid will also be automatically smooth. There are many possible integrable monitor functions that will give rise to a smoothly clustered grid. For problems involving the propagation of a single phase change boundary we consider equidistributing the function

$$M(x) = 1 + \frac{\mu_1}{\sqrt{\mu_2^2(x - x_*)^2 + 1}}, \tag{28}$$

which was originally proposed by Farrell and Drury [14] to solve problems with steep solution fronts. The parameters μ_1 and μ_2 are positive constants that affect the smoothness and clustering of the grid around the point x_* which is an estimate of the position of the phase change boundary. The positions of the mesh points that exactly equidistribute (28) are the solutions of the scalar nonlinear equations

$$\begin{aligned} x_j &+ \frac{\mu_1}{\mu_2}\sinh^{-1}(\mu_2(x_j - x_*)) - \frac{\mu_1}{\mu_2}\left(1 - \frac{j}{N}\right)\sinh^{-1}(\mu_2(x_L - x_*)) \\ &- \frac{j}{N}\left((x_R - x_L) + \frac{\mu_1}{\mu_2}(\sinh^{-1}(\mu_2(x_R - x_*)))\right) = 0, \end{aligned} \tag{29}$$

for $j = 1, 2, \ldots, N - 1$. These can be easily solved using Newton's method and the resulting grid is clearly non-overlapping. Figures 3 and 4 demonstrate the influence of the choice of

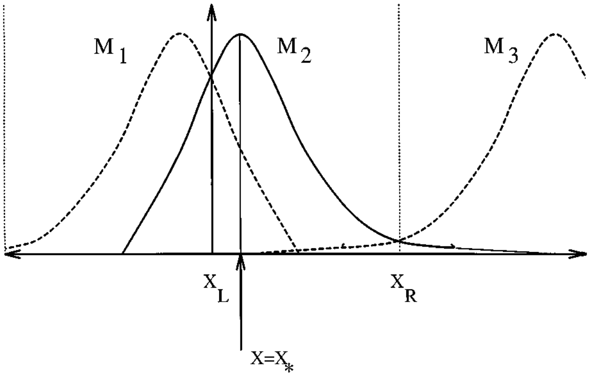

FIG. 2. Mirror grids.

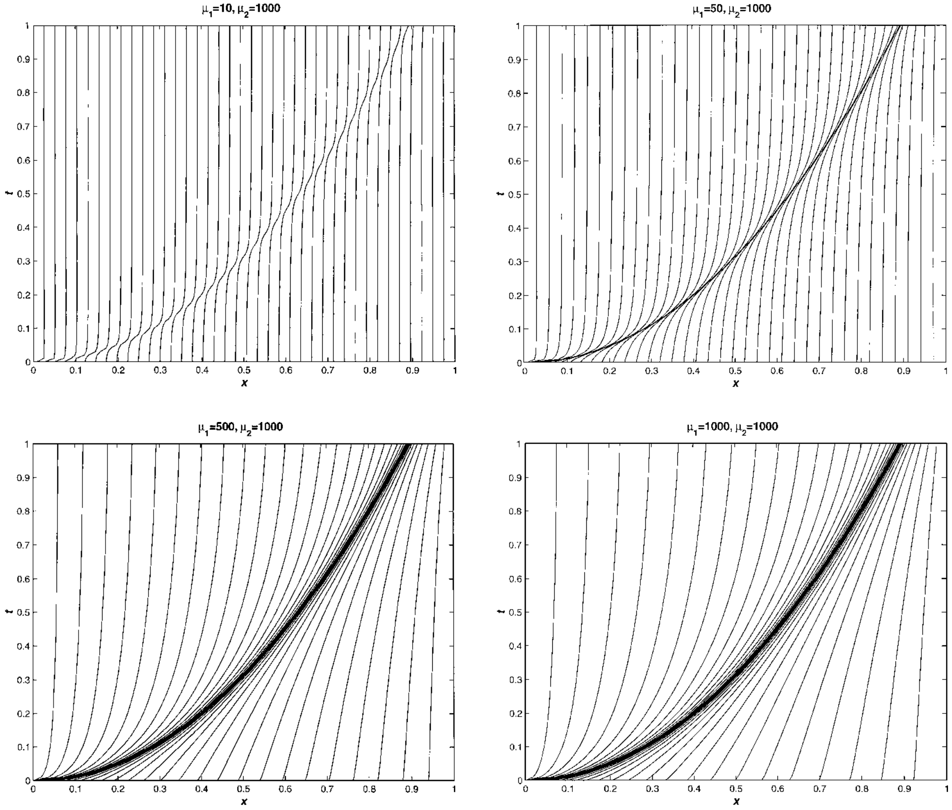

FIG. 3. Effect of varying μ_1 with μ_2 being fixed with $N = 40$.

parameters μ_1 and μ_2 on a grid that tracks a moving front where $x_*(t) = 2\sqrt{t/5}$. We can see clearly that the effect of increasing μ_1 while keeping μ_2 fixed is to reduce the mesh spacing around x_* and to widen the mesh spacing away from x_*. We also note that the main effect of increasing μ_2 is to reduce the extent over which the mesh clustering occurs around the front position. In all of these graphs we observe a very smooth evolution of the grid nodes.

It is often useful to have some estimate of the minimum mesh spacing using the monitor function (28). A lower bound is obtained by setting $x = x_*$ and using the equidistribution principle (24) we get

$$h_{min} \approx \frac{1}{\mu_1 N}\left[(x_R - x_L) + \frac{\mu_1}{\mu_2}\sinh^{-1}(\mu_2(x_R - x_L))\right]. \tag{30}$$

To ensure that Newton's method for the solution of (21) converges and to avoid oscillations it is important that μ_1 and μ_2 are chosen so that at least two mesh points are contained within the steep enthalpy layer at the phase front. The interfacial thickness is related to ε and λ in a non-trivial way so it is difficult to say exactly how μ_1 and μ_2 should be chosen for a particular value of ε. In the numerical results section we have chosen $\varepsilon/\lambda \approx 10^{-3}$ which results in a small perturbation of the original problem. By experimentation we have then chosen μ_1 and μ_2 large enough to ensure that we can resolve the enthalpy layer.

Often a phase will appear or disappear during the lifetime of a simulation. Figure 5a shows the behaviour of the mesh trajectories as a phase front exits the domain. Note the

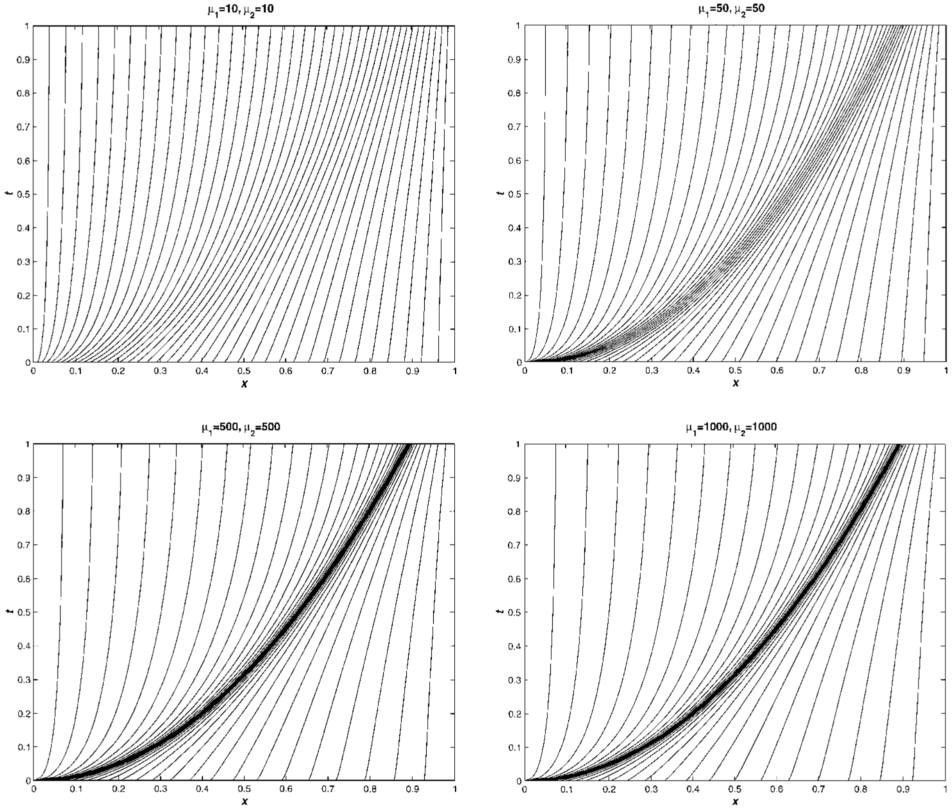

FIG. 4. Effect of varying both μ_1 and μ_2 simultaneously with $N = 40$.

rapid redistribution of the mesh points within the interior of the domain when the front leaves the domain. This behaviour can potentially lead to inaccuracies close to the boundary. To avoid this problem, when a front exits the domain the redistribution of the grid nodes is done smoothly by exponentially decreasing the value of μ_1 to zero while keeping the value of μ_2 fixed. As we have seen in Fig. 3, this allows a smooth transition from a significant amount of mesh clustering to a uniform grid. Figure 5b shows the effect of setting $\mu_1^* = \mu_1 e^{-\sigma(t-t^*)^2}$, $\sigma > 0$, where t^* is the time when the front reaches the boundary. We can see clearly that the mesh points are reallocated in a very smooth manner. Figure 5c shows that this process can also be run in reverse to smoothly introduce a phase front into the domain.

When a front appears from a boundary then $\int_0^1 M\,dx$ rapidly increase in time. This causes distant grid points to move non-smoothly since the local value of the monitor function away from the front position is almost constant. This effect can be seen clearly in Fig. 5a where grid points at the right of the domain move sharply to the left to increase the local mesh spacing as the front enters the domain at $t = 0$. To solve this problem Farrell and Drury [14] suggest the use of so-called mirror grids. To be precise, if a front is estimated to be at x_* then the grid is generated by equidistributing the monitor function $M(x) = M_1(x) + M_2(x) + M_3(x)$ where

$$M_1(x) = 1 + \frac{\mu_1}{\sqrt{\mu_2^2(x + x_* - 2x_L)^2 + 1}}, \qquad M_2(x) = 1 + \frac{\mu_1}{\sqrt{\mu_2^2(x - x_*)^2 + 1}},$$

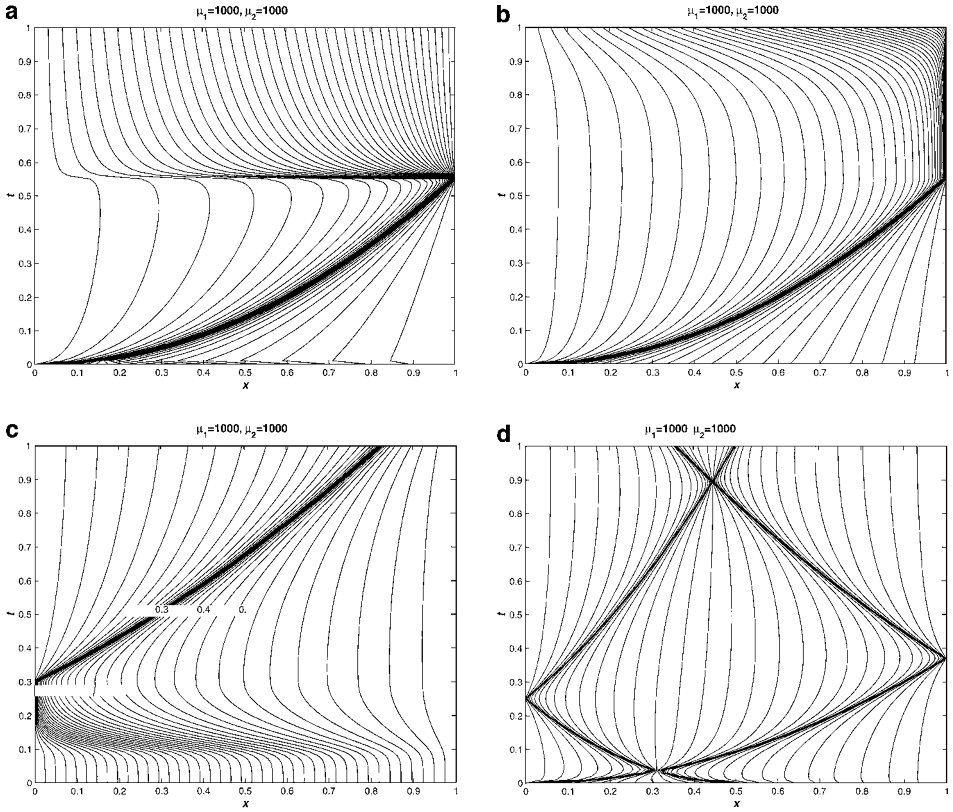

FIG. 5. Grids obtained using the analytically integrable monitor function with $N = 40$.

and

$$M_3(x) = 1 + \frac{\mu_1}{\sqrt{\mu_2^2(x + x_* - 2x_R)^2 + 1}}.$$

It is clear from Fig. 2 that these are just mirror images of the original monitor function reflected over the respective boundaries. Therefore, whatever amount of monitor function is lost from the domain, an equal amount is reflected back in. This composite monitor function can again be analytically integrated and the grid points found by Newton's method. The grid shown in Fig. 5b was obtained using this approach and we can see that the aforementioned problem close to the initial time has now been completely avoided. Finally, Fig. 5d shows that the mirror grid procedure can be used to adapt to the position of fronts reflecting off the boundary and also to fronts that interact. In the numerical examples that follow in the next section we will use the mirror grid modification to the basic equidistribution procedure outlined above.

For multiphase problems the monitor function is

$$M(x) = \sum_{i=1}^{NP-1} M^{(i)}(x), \tag{31}$$

where $M^{(i)}$ is the monitor function associated with the phase front connecting phase i and $i + 1$.

4.3. The complete algorithm. Each time step of the adaptive algorithm requires the solution of (29) and (21). One could solve these simultaneously as one large nonlinear algebraic system. The alternative is to decouple the calculation of the grid points from the solution. There are two advantages of decoupling. First the size of the algebraic systems that arise at each time step are smaller. This is of great importance for the extension to multidimensional problems. The second advantage is that decoupling allows flexibility in the choice of interative methods used to calculate the grid and the solution of the moving mesh equations. In particular, by decoupling it is possible to use iterative methods with different tolerances when determining the grid and the solution. The numerical results in Section 5 were obtained using the following algorithm:

(1) Perform the simple prediction

$$x_{(*,0)}^{n+1} = x_*^n + \Delta t^{n+1}\left(\frac{x_*^n - x_*^{n-1}}{\Delta t^n}\right). \tag{32}$$

Set $s = 0$.

(2) Let $x_* = x_{(*,0)}^{n+1}$ and solve (29) to give $\mathbf{x}_{\Delta,s}^{n+1}$.

(3) Solve (21) for $u_{\Delta,s}^{n+1}$ and then determine $x_* = x_{(*,s+1)}^{n+1}$ using linear interpolation for the phase change temperature.

(4) If $|x_{(*,s+1)}^{n+1} - x_{(*,s)}^{n+1}| < Tol_{grid}$ then $u_\Delta^{n+1} = u_{\Delta,s}^{n+1}$, $\mathbf{x}_\Delta^{n+1} = \mathbf{x}_{\Delta,s}^{n+1}$, and $x_*^{n+1} = x_{(*,s+1)}$. Otherwise $s = s + 1$ and goto (2).

There is rarely any need to use a very strict tolerance for the convergence of the grid points and in all the calculations presented in the following section we set $Tol_{grid} = 10^{-3}$. The simple initial extrapolation step is extremely useful to speed up convergence. By only extrapolating the estimate of x_* we of course ensure that we have a non-overlapping grid. Clearly the efficiency of this approach depends on how quickly convergence is reached. The calculations for $\mathbf{u}_{\Delta,s+1}^{n+1}$ and $\mathbf{x}_{\Delta,s+1}^{n+1}$ can be accelerated if the initial guesses for these calculations are $\mathbf{u}_{\Delta,s}^{n+1}$ and $\mathbf{x}_{\Delta,s}^{n+1}$.

5. NUMERICAL EXPERIMENTS

5.1. Example 1. The first test case we consider is a classical Stefan problem describing the freezing of water. This example has also been considered by Bonacina *et al.* [4] and has been used by Furzeland [15] to compare the performance of different numerical techniques for solving moving boundary value problems. Equation (1) is solved subject to boundary and initial conditions

$$T_1(0,t) = -20, \qquad \frac{\partial T_2}{\partial x} = 0 \quad \text{as } x \to \infty, t > 0,$$

and

$$T_1(x,0) = T_2(x,0) = 10, \qquad \text{for } x \geq 0.$$

The thermal properties are

$$k_1 = 2.22,\ k_2 = 0.556,\ C_1 = 1.762,\ C_2 = 4.226,\ \lambda_1 = 338,\ T_{m_1} = 0.$$

The similarity solution for this problem is given in Carslaw and Jaegar [7] and takes the form

$$\begin{aligned} s(t) &= 2\phi\sqrt{\kappa_1 t}, \\ T_1(x,t) &= -20\left(1 - \frac{\operatorname{erf}(x/2\sqrt{\kappa_1 t})}{\operatorname{erf}\phi}\right), \\ T_2(x,t) &= 10\left(1 - \frac{\operatorname{erfc}(x/2\sqrt{\kappa_2 t})}{\operatorname{erfc}(\phi\sqrt{\kappa_1/\kappa_2})}\right), \end{aligned}$$

where $\kappa_i = k_i/C_i$ and ϕ is the root of the equation

$$\frac{e^{-\phi^2}}{\operatorname{erf}\phi} - \frac{k_2}{k_1}\sqrt{\frac{\kappa_1}{\kappa_2}}\frac{e^{-\kappa_1\phi^2/\kappa_2}}{2\operatorname{erfc}(\phi\sqrt{\kappa_1/\kappa_2})} - \frac{\phi\lambda\sqrt{\pi}}{20C_1} = 0.$$

To avoid any difficulties with the discontinuity in the initial and boundary conditions, and to compare the results with those of Furzeland [15], the problem was solved for $0.0012 \le t \le 0.288$. Figure 6a shows the computed mesh trajectories with $\Delta t = 0.0012$, $N = 40$, $\varepsilon_1 = \varepsilon_1^+ + \varepsilon_1^- = 0.25$ and the mesh has been generated with $\mu_1 = \mu_2 = 200$. We can see that the mesh has followed smoothly the movement of the phase boundary and from

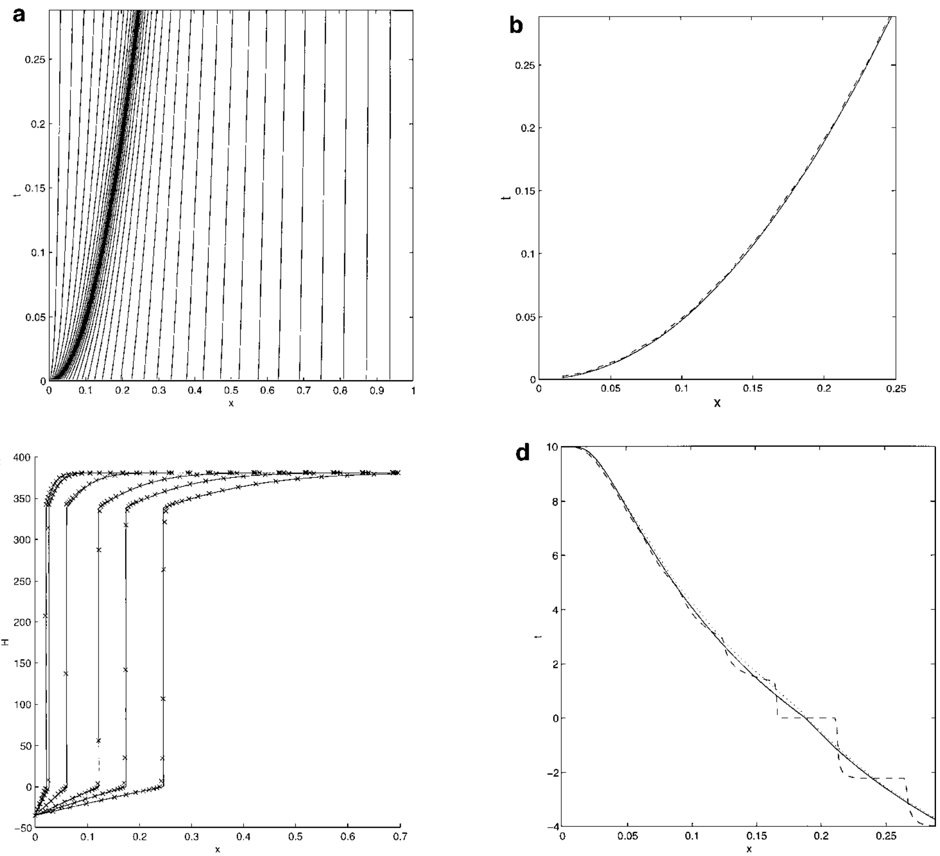

FIG. 6. Results for Example 1 with $N = 40$ and $\varepsilon = 0.25$; (a) mesh trajectories; (b) front position — exact, $\cdots$ adaptive, – – uniform; (c) enthalpy; (d) temperature at $x = 0.2$; — exact, $\cdots$ adaptive, – – uniform.

TABLE I
Results for Example 1 with N = 40 and ε_1 = 0.5

μ	Newt	SOR	$\|E_f\|_{L_\infty(\Omega)}$	$\|E_T\|_{L_2(\Omega)}$	$\|E_H\|_{L_1(\Omega)}$
50	2486	351	1.726E−3	8.879E−2	3.951E−1
100	1044	10	1.328E−3	5.352E−2	3.776E−1
200	849	0	1.408E−3	5.075E−2	3.883E−1
400	759	0	1.736E−3	5.257E−2	4.078E−1

Fig. 6b we see that the clustering of mesh points has led to a very accurate prediction of the position of the front. Included in this figure are the results obtained using a stationary uniform mesh. Equation (7) has been discretised using a central difference approximation and the resulting nonlinear equations solved for the enthalpy using a Newton iteration. This allows a liquid fraction H/λ to be calculated for the mesh cell undergoing the phase transition which can be used to improve the accuracy of the predicted front position. However, even with this modification we can see that the uniform mesh results are poorer than the moving mesh prediction. Figure 6c shows the computed enthalpy at $t = 0.0024, 0.0036, 0.018, 0.072, 0.144$, and 0.288. These again are very accurate. Finally, Fig. 6d shows the temperature history of the point $x = 0.2$. We can see the unphysical staircase behaviour using a fixed uniform mesh and that the moving mesh results are in very good agreement with the analytical solution.

Table I compares the performance of the moving mesh method as $\mu_1 = \mu_2 = \mu$ is varied with $\varepsilon_1 = 0.5$ fixed, where E_f, E_T, and E_H denote the error in the front position, temperature, and enthalpy, respectively. The CPU times have all been normalised to the $\varepsilon_1 = 2$ case and all calculations were performed using double precision arithmetic with the SOR parameter $\omega = 1.4$. We see that as long as μ is sufficiently large then the algorithm is very efficient requiring only two or three Newton steps per time step. For smaller values of μ the grid does not resolve the steep front in H and this leads to the increased number of Newton and SOR steps. Table II shows the sensitivity of the numerical results to the choice of ε_1 with $\mu = 200$ fixed. When ε_1 is large we see that the resulting nonlinear systems are relatively easy to solve but this is at the cost of reduced accuracy. As we decrease ε_1 we find that the problem becomes slightly more difficult to integrate forward but that we get a considerable improvement in accuracy. Eventually if ε_1 is taken too small then we see an increase in the overall cost of the algorithm with little improvement in accuracy. The last row of this table shows the results using a fixed mesh. The moving mesh results are clearly much better especially for the temperature. In fact to reduce the error in the temperature to that using the moving mesh requires an order of magnitude more mesh points. The two

TABLE II
Results for Example 1 with N = 40 and μ = 200

ε_1	Newt	SOR	$\|E_f\|_{L_\infty(\Omega)}$	$\|E_T\|_{L_2(\Omega)}$	$\|E_H\|_{L_1(\Omega)}$	CPU
2	742	0	4.908E−3	1.516E−1	1.058E+0	1.0
1	754	0	2.285E−3	7.948E−2	6.148E−1	1.1
0.5	849	0	1.408E−3	5.075E−2	3.883E−1	1.2
0.25	1068	10	1.558E−3	4.286E−2	2.868E−1	1.4
Uniform	497	0	6.207E−3	5.222E−1	8.756E−1	0.63

TABLE III
Comparison of Adaptive Grid Results with Those Considered by Furzeland [15]; μ = 200, ε_1 = 0.25

Method/t	2.4E−3	3.6E−3	1.8E−2	7.2E−2	1.44E−1	2.88E−1
(i)	2.240E−2	2.760E−2	6.180E−2	1.238E−1	1.750E−1	2.476E−1
(ii)	1.960E−2	2.560E−2	6.170E−2	1.236E−2	1.749E−1	2.474E−1
(iii)	2.280E−2	2.790E−2	6.190E−2	1.237E−1	1.750E−1	2.474E−1
(iv)	2.500E−2	2.500E−2	5.000E−2	1.250E−1	1.750E−1	2.490E−1
Adaptive	2.155E−2	2.670E−2	6.053E−2	1.234E−1	1.748E−1	2.478E−1
$s(t)$	2.260E−2	2.790E−2	6.190E−2	1.238E−1	1.750E−1	2.475E−1

tables do show that very accurate solutions can be obtained using the moving mesh method without the need for the grid being overly refined and for moderate values of ε_1.

Finally, Table III compares the predicted front position with the four methods considered by Furzeland [15]. Methods (i), (ii), and (iii) are based on front-tracking techniques whereas method (iv) is based on a discretisation of an unsmoothed enthalpy formulation using a stationary uniform grid. It should be noted that the results for method (i) are for $N = 80$ and the results for method (iii) are obtained by a method of lines approach using adaptive time stepping. We see clearly that the moving mesh results are a significant improvement over method (iv) and are very competitive with the three front-tracking methods.

5.2. Example 2. The second test case considered involves the simulation of the spot-welding of two large sheets of steel using a high electric current as a body heating source. The model used was proposed by Atthey [2] and this example has also been used as a test case in the numerical work of Li [18]. In non-dimensionalised form the governing equation is given by (7) with

$$\varphi(u) = \begin{cases} A_0 + uE_0, & u \le 0.6 \\ A + uE, & u > 0.6, \end{cases} \tag{33}$$

and

$$H(u) = \begin{cases} Bu, & u < 1 \\ B \le H \le B + C, & u = 1 \\ Bu + C & u > 1, \end{cases} \tag{34}$$

where $A = 1.708$, $A_0 = 0.336$, $E = 1.220$, $E_0 = 3.457$, $B = 0.780$, and $C = 0.331$. The boundary and initial conditions take the form

$$\frac{\partial u}{\partial x} = \begin{cases} 0 & \text{on } x = 0 \\ -1.949u & \text{on } x = 1, \end{cases} \tag{35}$$

and $u(x, 0) = 0$, $0 \le x \le 1$.

Before the melting temperature is reached at $x = 0$ this is just a simple heat conduction problem. Once the melting temperature is reached it remains fixed for a time $t_* = C/(A + E) = 0.11305$ while enough heat is added for the material to change phase. During this time other parts of the material reach the melting temperature due to the body heating

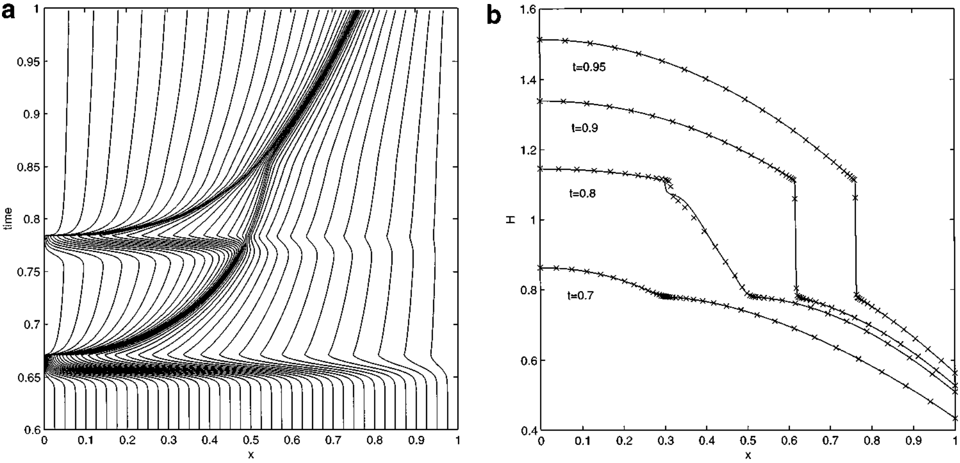

FIG. 7. Results for Example 2 with $N = 40$. (a) Mesh trajectories, (b) enthalpy; × coarse grid, — fine grid.

term thus leading to a finite mushy region. The interfaces between the solid-mush regions and the liquid-mush regions will be referred to as the solidus and liquidus interfaces, respectively. After the liquidus interface appears at $x = 0$ it rapidly moves across the domain and eventually merges with the solidus interface.

Numerically, we define the position of the solidus and liquidus interfaces as $x_s(t) = x(H(B))$ and $x_l(t) = x(H(B + C))$, where H is the smoothed enthalpy function. Using the smoothed enthalpy function the temperature at $x = 0$ increases continuously during the change of phase rather than remaining fixed. To calculate the times when the solidus and liquidus interfaces appear at $x = 0$ we use linear extrapolation in time from the solid and liquid phases, respectively.

Figure 7 shows the computed solutions and mesh trajectories with $N = 40$, $\Delta t = 0.001$, and $\varepsilon_1 = 0.001$. Before the temperature at the lefthand boundary reaches the melting temperature we can see that a stationary uniform grid is being used. To ensure that the computational grid is in the correct position to track the solidus interface, the method first detects the presence of the interface at $x = 0$ and then retakes a number of time steps such that μ_1 is increased exponentially to its final value. When a sufficient amount of heat has been added the liquidus interface appears at $x = 0$ and moves very rapidly across the domain. At this stage some of the grid points migrate smoothly from the solidus interface to resolve the liquidus interface. Thereafter, we can see that the adaptive algorithm follows accurately both interfaces which converge towards each other. The predicted interface positions are shown in Fig. 8 which move very smoothly.

No exact solution exists for this test case so Fig. 7b compares the computed enthalpy with the solutions obtained using a fine grid with $N = 320$. We see excellent agreement and note that the enthalpy at the solidus interface is continuous whereas it is almost discontinuous at the liquidus interface. An analytical argument to explain this behaviour is given in [17].

Tables IV and V show the convergence of a number of parameters describing the simulation. Here, t_0 and t_1 are the times of the appearance of the solidus and liquidus interfaces, respectively. The positions of the solidus interface at $t = t_1$ and $t = 1$ are denoted by $S(t_1)$ and $S(1)$. The prediction of $S(1)$ appears to be converging to around $x = 0.76$ which is in reasonable agreement with the experimental value of $0.6 < x < 0.7$ and is closer then the predictions of Atthey ($x \approx 0.85$) and Li ($x \approx 0.784$).

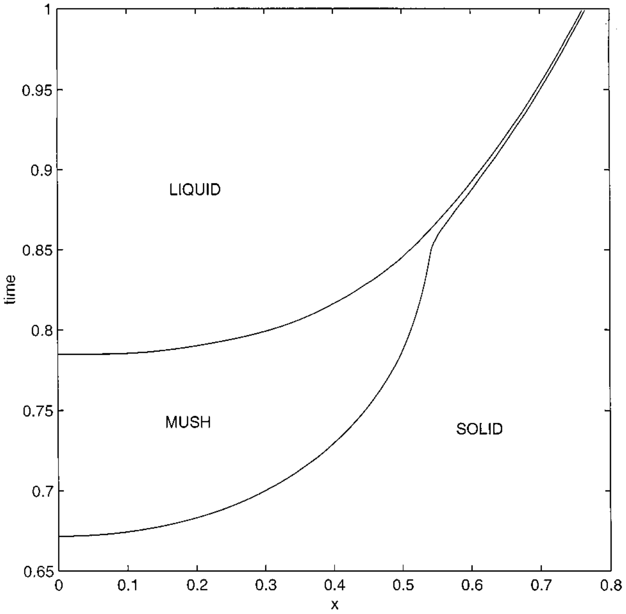

FIG. 8. Phase front propagation for Example 2.

5.3. Example 3. The final case we consider involves the appearance and disappearance of four phase fronts. This example was originally proposed by Duncan [11]. The governing equation is given by (7) with $\phi = 0$ and the temperature-enthalpy function is

$$H(u) = \begin{cases} u & \text{for } u < -0.4 \\ u + 0.2 & \text{for } -0.4 < u < 0 \\ u + 0.4 & \text{for } 0 < u. \end{cases} \tag{36}$$

The initial and boundary conditions are

$$H(x, 0) = -1 \qquad \text{for } x \in [0, 1],$$

$$H(0, t) = \begin{cases} 4t - 1 & \text{for } t \in [0, 0.5] \\ 1 & \text{for } t \in [0.5, \infty), \end{cases}$$

$$H(1, t) = \begin{cases} -1 & \text{for } t \in [0, 0.075] \\ 4t - 1.3 & \text{for } t \in [0.075, 0.575], \\ 1 & \text{for } t \in [0.575, \infty). \end{cases}$$

TABLE IV
Results for Example 2 with N = 40, μ = 200, and Δt = 1$E-3$

ε_1	t_0	t_1	$t_1 - t_0$	$S(t_1)$	$S(1)$
5.0E−3	0.67072	0.78594	0.11521	0.52347	0.75651
2.0E−3	0.67121	0.78553	0.11439	0.50807	0.75998
1.0E−3	0.67119	0.78488	0.11369	0.49832	0.76101

TABLE V
Results for Example 2 with ε_1 = 2E−3, μ = 200

N	Δt	t_0	t_1	$t_1 - t_0$	$S(t_1)$	$S(1)$
40	4.0E−3	0.66487	0.77388	0.10901	0.49949	0.79713
80	2.0E−3	0.66863	0.78340	0.11477	0.50642	0.76939
160	1.0E−3	0.67124	0.78565	0.11441	0.50741	0.76019

The solution of this problem has two phase-change boundaries moving to the right and two moving to the left. The fronts moving to the right appear at $t = 0.2$ and 0.35 and the fronts moving to the left at $t = 0.275$ and 0.425. The fronts moving to the right and left annihilate each other and hence we have the added difficulty of the disappearance of fronts during the lifetime of the simulation.

Figure 9 shows the computed mesh trajectories and enthalpy at the times $t = 0.15$, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.53, 0.56, and 0.6 using the parameters $N = 40$, $\varepsilon_1 = \varepsilon_2 = 0.001$, $\Delta t = 0.001$, and $Tol_{grid} = 0.001$. As no exact solution is available we have compared the solution with a fine grid solution with $N = 320$. We can see that the grid evolves smoothly to track the appearance and disappearance of all the interfaces. Finally, Fig. 10 shows the predicted front positions.

6. EXTENSIONS

A major benefit of an adaptive moving mesh approach will be for problems in more than one dimension. To generate an adaptive moving mesh it is useful to regard the physical domain Ω_p as the image of a computational (logical) domain Ω_c under the invertible maps

$$x = x(\xi, \eta, t), \; y = y(\xi, \eta, t) \qquad \text{and} \qquad \xi = \xi(x, y, t), \; \eta = \eta(x, y, t), \tag{37}$$

where (x, y) and (ξ, η) are the physical and computational coordinates, respectively. A mesh covering Ω_p is obtained by applying the mapping given in (37) to a partitioning of Ω_c.

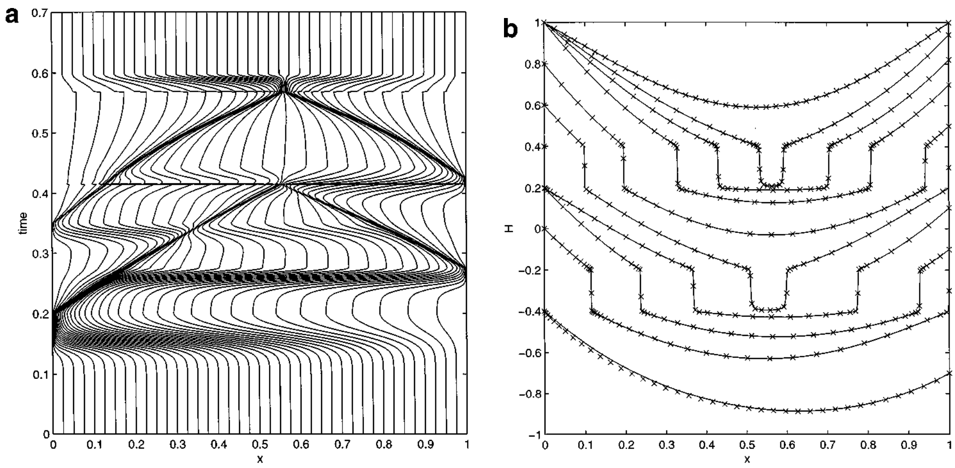

FIG. 9. Results for Example 3 with $N = 40$. (a) Mesh trajectories, (b) enthalpy; × coarse grid, — fine grid.

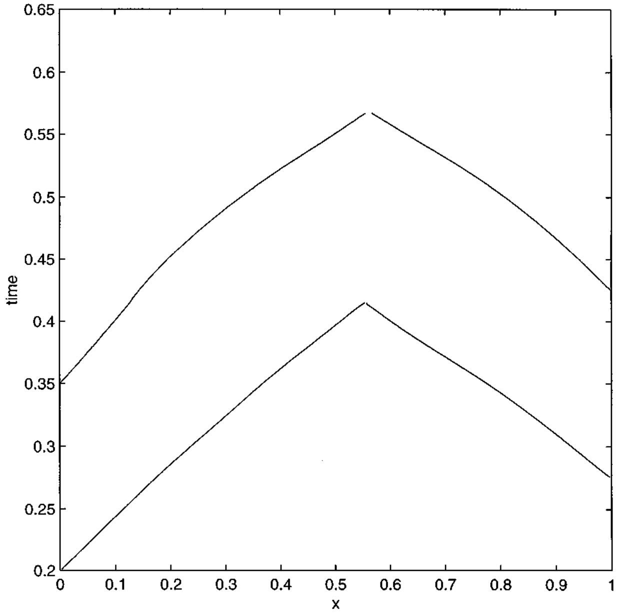

FIG. 10. Phase front propagation for Example 3 with $N = 40$.

A popular way to choose the coordinate transformation is to require that it minimizes a functional of the form

$$I(\xi, \eta) = \frac{1}{2} \int_{\Omega_p} (\nabla \xi^T G^{-1} \nabla \xi + \nabla \eta^T G^{-1} \Delta \eta) \, \mathrm{d}x \, \mathrm{d}y, \tag{38}$$

where $\nabla = (\partial/\partial x, \partial/\partial y)$ and $G(x, y)$ is a 2×2 symmetric positive definite matrix, often referred to as a monitor matrix. The idea in adaptive mesh generation is to choose G to concentrate mesh points in Ω_p where the solution of the PDE is difficult to solve. Such an approach has been used recently by Cao *et al.* [5] as the basis of an r-adaptive finite element method. To apply these ideas to two-dimensional problems work is underway to initially consider the monitor matrix

$$G = \left(1 + \frac{\mu_1}{\sqrt{\mu_2^2 (\mathbf{x} - \mathbf{x}_*)^2 + 1}}\right) I,$$

where I is the 2×2 identity matrix and $\mathbf{x}_*$ is the closest point on the phase boundary to the point $\mathbf{x}$. Preliminary results in this direction are very encouraging.

7. CONCLUSIONS

In this paper we have developed a very simple adaptive moving mesh method for phase-change problems in one dimension. The method uses a smoothed enthalpy-temperature relationship and the grid is moved to equidistribute an analytically integrable monitor function. The algorithm gives rise to smoothly clustered mesh trajectories which allow a very accurate prediction of the position of phase-change boundaries. We have used a novel semi-implicit discretisation that gives rise to systems of nonlinear algebraic equations that can be solved efficiently using Newton's method. We have also shown theoretically that a unique solution of these systems exists. Future work will include a detailed convergence analysis of the method and the extension of the moving mesh methodology to multidimensional problems.

REFERENCES

1. S. Adjerid and J. E. Flaherty, A moving finite element method with error estimation and refinement for one-dimensional partial differential equations, *SIAM J. Numer. Anal.* **23**, 778 (1986).
2. D. R. Atthey, A finite difference scheme for melting problems, *J. Inst. Math. Appl.* **13**, 353 (1974).
3. J. G. Blom, J. M. Sanz-Serna, and J. G. Verwer, On simple moving grid methods for one-dimensional evolutionary partial differential equations, *J. Comput. Phys.* **74**, 191 (1988).
4. C. Bonacina, G. Comini, A. Fasano, and M. Primicerio, Numerical solution of phase-change problems, *Int. J. Heat Mass Transfer* **16**, 1825 (1973).
5. W. Cao, W. Huang, and R. D. Russell, An r-adaptive finite element method based upon moving mesh PDEs, *J. Comput. Phys.* **149**, 221 (1999).
6. G. F . Carey and H. T. Dinh, Grading functions and mesh redistribution, *SIAM J. Numer. Anal.* **22**, 1028 (1985).
7. H. S. Carslaw and J. C. Jaeger, *Conduction of Heat in Solids* (Oxford Univ. Press, London, 1959).
8. J. Crank, *Free and Moving Boundary Value Problems* (Oxford Science Publications, 1984).
9. C. de Boor, Good approximation by splines with variables knots. II. in *Lecture Notes in Mathematics* (Springer-Verlag, New York / Berlin, 1974), Vol. 363, p. 12.
10. E. A. Dorfi and L. O'c. Drury, Simple adaptive grids for 1-D initial value problems, *J. Comput. Phys.* **69**, 175 (1987).
11. D. B. Duncan, A simple and effective self-adaptive moving mesh for enthalpy formulations of phase change problems, *IMA J. Numer. Anal.* **11**, 55 (1991).
12. P. W. Egolf and H. Manz, Theory and modeling of phase change materials with and without mushy regions, *Int. J. Heat Mass Transfer* **37**, 2917 (1994).
13. C. M. Elliott, On the finite element approximation of an elliptic variational inequality arising from an implicit time discretisation of the Stefan problem, *IMA J. Numer. Anal.* **1**, 115 (1981).
14. K. Farrell and L. Drury, An explicit, adaptive grid algorithm for one-dimensional initial value problems, *Appl. Numer. Math.* **26**, 3 (1997).
15. R. M. Furzeland, A comparitive study of numerical methods for moving boundary value problems, *J. Inst. Math. Appl.* **26**, 411 (1980).
16. W. Huang, Y. Ren, and R. D. Russell, Moving mesh PDEs based on the equidistribution principle, *SIAM J. Numer. Anal.* **31**, 709 (1994).
17. A. A. Lacey and A. B. Taylor, A mushy region in a Stefan problem, *IMA J. Appl. Math.* **30**, 303 (1983).
18. C. H. Li, A finite-element front-tracking enthalpy method for Stefan problems, *IMA J. Numer. Anal.* **3**, 87 (1983).
19. G. H. Meyer, Multidimensional Stefan problems, *SIAM J. Numer. Anal.* **10**, 522 (1973).
20. R. H. Nochetto, M. Paolini, and C. Verdi, An adaptive finite element method for two-phase Stefan problems in two space dimensions. Part II. Implementation and numerical experiments, *SIAM J. Sci. Comput.* **12**(5), 1207 (1991).
21. R. H. Nochetto, A. Schmidt, and C. Verdi, Error control for phase change problems, in *IVTAM Symposium on Variations of Domain and Free Boundary Problems, Paris* (Kluwer Academic, Dordrecht, 1997).
22. J. R. Ockendon and W. R. Hodgkins, *Moving Boundary Problems in Heat Flow and Diffusion* (Oxford Univ. Press, London, 1975).
23. N. Shamsunder, in *Moving Boundary Problems*, edited by D. G. Wilson and P. T. Boggs (Academic Press, San Diego, 1978), p. 165.
24. V. Voller and M. Cross, Accurate solutions of moving boundary problems using the enthalpy method, *Int. J. Heat Mass Transfer* **24**, 545 (1981).