MOVING AND DEFORMING FINITE-ELEMENT SIMULATION OF TWO-DIMENSIONAL STEFAN PROBLEMS

NICHOLAS ZABARAS AND YIMIN RUAN
Department of Mechanical Engineering, University of Minnesota, 111 Church Street S.E., Minneapolis, MN 55455, U.S.A.

SUMMARY
A new moving/deforming FEM analysis of two-dimensional phase change problems is presented. The region occupied by the solid phase is divided into a non-moving element region and a moving region consisting of the finite elements next to the freezing interface. A transfinite mapping method is used to generate new finite-element meshes and motion in the liquid phase during simulations. Several two-dimensional Stefan problems are analysed and discussed in relation to analytical solutions and other numerical techniques.

INTRODUCTION
Numerous engineering problems require a solution to the heat conduction equation in the presence of a phase transformation. Such applications are usually referred to as Stefan problems. A liquid at initial temperature $T_{\text{in}}(x, y)$ is assumed to occupy a region $\Omega_0$ with boundary $\partial \Omega_0$. At time $t > 0$, the boundary $\partial \Omega_0$ is cooled down to a temperature lower than the melting temperature. Solidification starts around boundary $\partial \Omega_0$ and proceeds inwards. Let us denote the isothermal interface at time $t$ as $\partial \Omega_1(t)$. The governing differential equations in the absence of heat sources are given as:

$$\rho_c S \frac{\partial T_S(x, y, t)}{\partial t} = \nabla \cdot (K_S \nabla T_S(x, y, t)), \quad (x, y) \in \Omega_S(t)$$

$$\rho_l L \frac{\partial T_L(x, y, t)}{\partial t} = \nabla \cdot (K_L \nabla T_L(x, y, t)), \quad (x, y) \in \Omega_L(t)$$

where $\Omega_S$ and $\Omega_L$ indicate the solid and liquid regions, respectively ($\Omega_S \cup \Omega_L = \Omega_0$), $T_S(x, y, t)$ is the temperature at the point $(x, y) \in \Omega_S(t)$ at time $t$, $T_L(x, y, t)$ is the temperature at the point $(x, y) \in \Omega_L(t)$ at time $t$, and $\rho$, $c$ and $K$ are the density, specific heat and conductivity, respectively. Subscripts S and L are used to indicate the solid and liquid phases, respectively.

The freezing interface conditions include the isothermal condition

$$T(x, y, t) = T_m \quad (x, y) \in \partial \Omega_1(t)$$

where $T_m$ is the melting temperature, and the energy balance equation (Stefan condition)

$$K_S \frac{\partial T_S(x, y, t)}{\partial n} - K_L \frac{\partial T_L(x, y, t)}{\partial n} = \rho L V \cdot n$$

0748-8025/90/070495-12$06.00 © 1990 by John Wiley & Sons, Ltd.
where \( \mathbf{n} \) is a unit normal vector on the interface boundary \( \partial Q_1(t) \) at a point \( (x, y) \in \partial Q_1(t) \) pointing away from the solid region, \( \mathbf{V} \) is the velocity vector at the same point on the interface, and \( L \) denotes the latent heat of fusion. It is important to emphasize that equation (4) defines the normal velocity of a point on the solid/liquid interface.

To complete the definition of the problem one should prescribe the boundary temperature \( T_0(x, y, t) \) on \( \partial Q_0 \) and flux \( q_0(x, y, t) \) on \( \partial Q_2 \), where \( \partial Q_0 \cup \partial Q_2 = \partial Q_0 \). In summary, our objective is to find the temperature history on \( \Omega_S \cup \Omega_L = \Omega \) and the solid/liquid interface velocity and location history.

**NUMERICAL IMPLEMENTATION**

**Deforming finite-element formulation of Stefan problems**

The deforming finite-element formulation used in this work mainly follows that of Lynch and O’Neill\(^2\) and Lynch,\(^3\) and is summarized as follows.

Let us divide the region \( \Omega_S \cup \Omega_L = \Omega \) into \( E_S + E_L = E \) finite elements. Denote \( M_S, M_L, \) and \( M_I \) the number of nodes in the solid phase, liquid phase, and on the moving interface, respectively. The \( E_S \) elements of the solid phase are assumed to share the same nodes on the solid/liquid interface boundary \( \partial Q_1(t) \) with the \( E_L \) elements of the liquid phase. Then the total number of nodes \( M \) in \( \Omega \) is given as \( M = M_S + M_L - M_I \).

The assumption that the finite-element nodes are moving with time implies that the shape functions explicitly depend on time through the nodal co-ordinates. For example, for each finite element \( e \) one can write

\[
T(x, y, t) = T^e(t) \Phi^e(x, y, t) \quad \text{(sum on } i), \quad i = 1, 2, \ldots, M^e
\]  

(5)

where \( T^e(t) \) are the nodal temperatures, \( \Phi^e(x, y, t) \) are the element shape functions, and \( (x, y) \) is a spatial point on the element \( e \). By assuming an isoparametric transformation of the space \( (x, y) \) to the time-independent space \( (\xi, \eta) \), with \( -1 \leq \xi, \eta \leq +1 \), one can write:

\[
x = X^e(t) \Phi^e(\xi, \eta), \quad y = Y^e(t) \Phi^e(\xi, \eta) \quad \text{(sum on } i), \quad i = 1, 2, \ldots, M^e
\]  

(6)

where \( (X^e, Y^e) \) are the nodal co-ordinates of element \( e \) and \( \Phi^e(\xi, \eta) \) are the element shape functions on master co-ordinates \((\xi, \eta)\).

The velocity components \( V_x(x, y, t) \) and \( V_y(x, y, t) \) at a point can be expressed with the isoparametric shape functions as

\[
V_x(x, y, t) = \dot{X}^e(t) \Phi^e(\xi, \eta), \quad V_y(x, y, t) = \dot{Y}^e(t) \Phi^e(\xi, \eta)
\]  

(7)

where \( (\dot{X}^e, \dot{Y}^e) \) are the \( x \) and \( y \) components of the \( i \)th nodal velocity of the element \( e \), and summation on \( i \) is implied \( (i = 1, 2, \ldots, M^e) \). Here the superimposed dot is used to denote time derivative.

After performing a Galerkin type of weak formulation of equations (1) and (2) and using equations (5) and (6) one can obtain the following assembled system of equations:

\[
C_{ij} \frac{dT^j}{dt} + (B_{ij} + K_{ij})T^j = F^i, \quad I, J = 1, 2, \ldots, M \quad \text{(sum on } J)
\]  

(8)

where \( I \) and \( J \) are the global nodal number corresponding to element nodes \( i \) and \( j \), respectively, \( (i, j = 1, 2, \ldots, M^e) \). The matrices \( K \) and \( C \) are defined as with a conventional FEM analysis, while

\[
B_{ij} = \sum_{e=1}^{E} B^e_{ij} = \sum_{e=1}^{E} \int_{\Omega_e} \rho c \Phi^e(x, y, t) \nabla \Phi^e(x, y, t) \cdot \mathbf{V}(x, y) \, d\Omega
\]  

(9)
For a node $I$ which coincides with the local node $k$ ($k = 1, 2, \ldots, M^f_0$) of the boundary segment $\partial \Omega^f_{02}$ of $\partial \Omega_{02}$, one can write
\begin{equation}
F_I = \sum_{c=1}^{E_2} F^c_k = \sum_{c=1}^{E_2} \int_{\partial \Omega^c_{02}} K \nabla T \cdot n \Psi^c_k \, d\Gamma
\end{equation}
(10)
while, when the node $I$ coincides with the local node $k$ ($k = 1, 2, \ldots, M^f_0$) of the boundary segment $\partial \Omega^f_i$ of $\partial \Omega_i$, $F_I$ takes the following form:
\begin{equation}
F_I = \sum_{b=1}^{E_I} F^b_k = \sum_{b=1}^{E_I} \int_{\partial \Omega^b_i} \left[ K_S \nabla T_S - K_L \nabla T_L \right] \cdot n \Psi^b_k \, d\Gamma
\end{equation}
(11)
Here $\partial \Omega^f_i$ ($b = 1, 2, \ldots, E_I$) and $\partial \Omega^c_{02}$ ($c = 1, 2, \ldots, E_0$) denote the boundary segments on the solid/liquid interface $\partial \Omega_i$ and the fixed boundary $\partial \Omega_{02}$ where flux is prescribed, respectively. Also, $M^f_0$ and $M^c_0$ denote the number of nodes on the segments $\partial \Omega^f_i$ and $\partial \Omega^c_{02}$, while $\Psi^b_k$ refers to the one-dimensional shape functions defined on each boundary segment. Finally, $E_I$ and $E_0$ denote the number of boundary segments in $\partial \Omega_i$ and $\partial \Omega_{02}$, respectively. Note that equation (11) is due to the heat flux discontinuity on the interface $\partial \Omega_i$ and the matrix $B$ is due to the motion of the finite-element nodes. The calculation of the matrices and vectors involved in equation (8) can be feasible only when the nodal velocities and front position are known at each time step. Obviously, to perform the time integration in equation (8), an implicit iterative scheme is required. Denoting the nodal temperature vector at time $t_n = t_{n-1} + \Delta t$ with $T^n, n = 1, 2, \ldots$, where $\Delta t$ is a time step, a stable integration scheme is
\begin{equation}
\left[ \frac{C_{n-1} + \gamma}{\Delta t} \right] T^n = F_{n-1} + \left[ \frac{C_{n-1} + \gamma}{\Delta t} - (1 - \gamma)(B_{n-1} + K_{n-1}) \right] T^{n-1}
\end{equation}
(12)
where the subscript $(n - 1 + \gamma)$ indicates the reference time for the calculation of the material properties and interface position and velocity which are required to calculate $F$, $B$, $K$, and $C$. For example, to determine $B_{n-1 + \gamma}$ one calculates the temperature-dependent material properties at time $\tilde{t}$ and then performs the integration indicated in equation (9) based on the interface location and velocity at the same time $\tilde{t}$, where
\begin{equation}
\tilde{t} = (1 - \gamma)t_{n-1} + \gamma t_n
\end{equation}
(13)
with $0.5 \leq \gamma < 1$. The location $\tilde{P}$ and velocity $\tilde{V}$ of a nodal point at the freezing front at time $\tilde{t}$ are given as
\begin{equation}
\tilde{P} = (1 - \gamma)P_{n-1} + \gamma P_n
\end{equation}
(14)
\begin{equation}
\tilde{V} = (1 - \gamma)V_{n-1} + \gamma V_n
\end{equation}
(15)
where
\begin{equation}
P_n = \tilde{V} \Delta t + P_{n-1}
\end{equation}
(16)
and the velocity $\tilde{V}$ is calculated iteratively.

Suppose that estimates of the interface velocity and material properties at time $\tilde{t}$ are known, one can then calculate $T^n$ from equation (12) by applying the boundary conditions on $\partial \Omega_0$ and the temperature boundary condition on $\partial \Omega_i(t)$ (equation (3)). This last boundary condition is enforced through a penalty method.
Calculation of the freezing interface motion

Consider that the temperature distribution and front location/velocity at time \( t_n \) are required, and that the solution is known at time \( t \leq t_{n-1} \). Using estimates of the material properties, interface location and velocity at time \( t = (1 - \gamma)t_{n-1} + \gamma t_n \), the mesh motion can be defined, and the system of equations (12) can be solved for \( T^n \). One can then return to the original equations (12) and calculate (exactly) the nodal force vector \( \mathbf{F} \) for each freezing interface node. These terms can be used to update the interface velocity at time \( \bar{t} \). Indeed, following the definition of \( \mathbf{F} \) for nodes on \( \partial \Omega_1(\bar{t}) \) (equation (11)), using the Stefan condition (equation (4)), and in each interface segment \( b \) approximating the nodal interface velocity as \( \mathbf{V} = \mathbf{V}^b_h \Psi^b_h \) (sum on \( h = 1, 2, \ldots, M^b \)), one can write

\[
\sum_{b=1}^{F_b} \int_{\partial \Omega^b} \rho L \mathbf{N}^b \Psi^b \mathbf{V}^b_h \, d\Gamma \cdot \mathbf{V}^b_h = f_j, \quad j, h = 1, 2, \ldots, M^b \quad \text{(sum on } h) \tag{17}
\]

where \( f_j \) is defined as the component of \( \mathbf{F} \) (equation (11)) which corresponds to the interface node \( j \) (\( J = 1, 2, \ldots, M_1 \)), where \( J \) is the global number of the local node \( j \) of the interface segment \( b \). Finally, \( \mathbf{N}^b \) defines the normal to the boundary segment \( b \) on the solid/liquid interface and points away from the solid region. All quantities in equation (17) are calculated at time \( \bar{t} \).

The above scheme was shown by Lynch and Sullivan\(^4\) to preserve energy, but no details of its numerical implementation and accuracy in solving classical Stefan problems in two directions were given. Basically you can pick up any direction you like for the interface node motion. A proper implementation of the assembled version of equation (17) will tell how much to move each interface node in the direction such that the normal interface motion to each freezing interface segment is correct. The tangential motion does not enter the physics of the problem. As it will be discussed later, in forming equations (17) care has to be taken that undesirable spreading or bunching of nodes and tangling of the interface do not occur. After such possibilities have been taken care of, one can select the nodal tangential interface velocities as zero and, together with the assembled version of equation (17), calculate the \( x \) and \( y \) nodal interface velocity components. More details on the calculation of these velocity components are given in the Appendix. Note that no ambiguity exists in the definition of \( \mathbf{N}^b \), but a length weighted nodal interface tangent has to be assumed using the current estimate of the interface location at time \( \bar{t} \).

Finite-element nodal motion in the solid phase

The region \( \Omega_S(t) \) is divided into two subregions. In one of the subregions the nodes are always stationary, and in the other one the nodes lying on the solid/liquid interface are moving with respect to time (Figure 1). The moving region includes only the elements of the solid phase next to the interface \( \partial \Omega_1(t) \). These elements are allowed to change their sizes to adapt to the movement of the interface front. When the sizes of these elements in the deforming region become larger than a prescribed size, a new moving region is generated and the previous moving region becomes part of the fixed region at the same time. The new region created consists of elements of zero area. The new set of elements will not introduce singularities since during the next time step and solid/liquid interface will move and finite sized elements will be created. These elements will continue deforming over the coming time steps until a necessity exists for the introduction of new elements. Obviously, the matrix \( \mathbf{B} \) is non-zero only on the moving region of \( \Omega_S(t) \). This simplified mesh generation/motion scheme is easy to implement,
Figure 1. Element configurations before and after new elements are introduced (the shaded area denotes a fixed region and the unshaded area a deforming region)

Figure 2. Transfinite mapping from the unit square region $U$ to the physical region $\Omega$, where $F(\xi, \eta)$ is defined as $F(\xi, \eta) = \{F_x(\xi, \eta), F_y(\xi, \eta)\}$

requires less computational time than the method adopted by Albert and O’Neill, and it will be shown to be very accurate.

Finite-element nodal motion in the liquid phase

Difficulties appeared in an attempt to implement in the liquid phase a mesh generation/node motion scheme similar to that used in the solid phase. In the liquid phase the transfinite mapping method is used with a bilinear projector. More on the transfinite mapping may be found in Reference 6, and an implementation of it to phase change problems was discussed by Albert and O’Neill. The region $\Omega_L$ is subdivided into simple subregions, each of them consisting of four sides. In each subregion $\Omega$, a mesh can be generated by mapping a mesh template of the unit square $U$ in the parametric space onto the subregion (Figure 2). This bilinear projector is given as follows:

$$
\begin{align*}
(x, y) &= (1 - \xi) \left( \frac{F_x(0, 0)}{F_y(0, 0)} \right) + \xi \left( \frac{F_x(1, 1)}{F_y(1, 1)} \right) + (1 - \eta) \left( \frac{F_x(\xi, 0)}{F_y(\xi, 0)} \right) + \eta \left( \frac{F_x(\xi, 1)}{F_y(\xi, 1)} \right) \\
&\quad - (1 - \xi)(1 - \eta) \left( \frac{F_x(0, 0)}{F_y(0, 0)} \right) - (1 - \xi)\eta \left( \frac{F_x(0, 1)}{F_y(0, 1)} \right) - \xi(1 - \eta) \left( \frac{F_x(1, 0)}{F_y(1, 0)} \right) - \xi\eta \left( \frac{F_x(1, 1)}{F_y(1, 1)} \right)
\end{align*}
$$

(18)

where $(\xi, \eta) \in U$ is transformed to the point $(x, y) \in \Omega$, and $F_x(\xi, \eta)$ and $F_y(\xi, \eta)$ map the boundary of the unit region $U$ onto the boundary $\partial \Omega$ of the physical region $\Omega$. Therefore, in order to use the transfinite mapping, one must specify the functions $F_x(\xi, \eta)$ and $F_y(\xi, \eta)$ that
are the $x$ and $y$ co-ordinates of the boundary $\partial \Omega$. In this paper, the discrete values of $F_x$ and $F_y$ are specified as the points $(x, y)$ of the boundary $\partial \Omega$ which are calculated during the simulation process.

In the solidification problem, let us consider that at time $t_{n-1}$ the location and velocity of the solid/liquid interface are known. In order to continue to the next time step, a new finite-element mesh is generated in each subregion of $\Omega_L$, with the following algorithmic steps:

(a) If the nodal points on the freezing boundary are distributed properly at time $t_{n-1}$, they may be used directly without any modification. If not, a linear interpolation scheme is used to rearrange the location and calculate the velocity of these new boundary nodal points. The discrete points on the boundary of each subregion are then used to determine the functions $F_x$ and $F_y$.

(b) Use the transfinite mapping [equation (18)] to provide new nodal locations, which are denoted as $(x, y)_{\text{new}}$. These new nodal locations are considered as the nodal locations at time $t_n$.

(c) In general, the new mesh will not coincide with the previous one and the new nodal locations have changed from $(x, y)_{\text{old}}$ to $(x, y)_{\text{new}}$. Therefore, an imaginary nodal motion can be considered that continuously moved the old mesh to the new one. These nodal velocities can be approximated as

$$V_{\text{nodal}} = \frac{(x, y)_{\text{new}} - (x, y)_{\text{old}}}{\Delta t} \quad (19)$$

Notice that since the interface velocity and location will be calculated through an iterative process, it is not necessary to repeat the above steps for each iteration at each time step. Rather, for each iteration at $t_n$, the interface position and velocity are recalculated while the rest of the nodes remain moving as predicted by the remeshing process at time $t_{n-1}$.

**Summary of the algorithm**

For each time step $t_n = t_{n-1} + \Delta t$, $n = 1, 2, \ldots (t_0 = 0)$:

A. Examine if the mesh at time $t = t_{n-1}$ can be used at time $t = t_n$:

1. If yes, the interface nodes are considered moving with their predicted velocity, while the non-interface nodes remain stationary.
2. If no, mesh remeshing is undertaken with the new nodal velocities calculated as explained before.

B. For each iteration $i = 1, 2, \ldots$, maximum number of iterations:

1. Calculate all matrices in equation (12) based on approximate information at time $\tilde{t}$, then apply boundary conditions on $\partial \Omega_0$ and $\partial \Omega_L$ and solve this system of equations to obtain the temperature field at time $t_n$.
2. Calculate a new interface velocity $\tilde{V}$ (at time $\tilde{t}$) using equation (22).
3. Update the location of the interface nodal points using equations (14) and (16).
4. Check the relative velocity error as

$$\text{error} = \frac{\| \tilde{V}_i - \| \tilde{V}_{i-1} \|}{\| \tilde{V}_{i-1} \|} \quad (20)$$
where $\| \cdot \|$ is the Euclidean norm of a vector and $\tilde{V}_{i-1}$ denotes the calculated vector of the nodal velocities on $\partial\Omega(t)$, at time $i = (1 - \gamma)t_n + \gamma t_{n+1}$, and at the $i$th iteration.

If error < tolerance, go to step C.
If error > tolerance, continue iteration (return to B).

End of iteration

C. Calculate the front velocity at time $t_n$ using equation (15).

Automatically adjust the time step by comparing the number of iterations $I_n$ at $t = t_n$ with the number $I_{n-1}$ at $t = t_{n-1}$.

If $I_n > I_{n-1}$, decrease time step as $\Delta t = 0.85 \Delta t$.
If $I_n \leq I_{n-1}$, increase time step as $\Delta t = 1.15 \Delta t$.

If the time limit (complete solidification) has not been reached, continue time stepping.

End of time stepping.

NUMERICAL RESULTS

In the following simulations, four-noded bilinear elements have been used for the analysis of two-dimensional one- and two-phase Stefan problems. The integration parameter $\gamma$ used in equation (12) is $0.5$.

The first example is the solidification of an infinite corner region with $K_s = K_L = 1 \text{ W/m}^\circ \text{C}$, $c_s = c_L = 1 \text{ J/kg}^\circ \text{C}$, $L = 0.25 \text{ J/kg}$ and $\rho = 1 \text{ kg/m}^3$. The melting and initial temperatures are $T_m = 0^\circ \text{C}$ and $T_i = 0.3^\circ \text{C}$, respectively. A constant temperature boundary condition $T_0 = -1^\circ \text{C}$ is applied to the surface of the region. The analytical solution is given by Rathjen and Jiji, and the non-dimensionalized interface position is given by

$$y^* = \left[ \frac{\lambda^u}{C} + \frac{C}{x^u - \lambda^u} \right]^{1/u}$$

(21)

where $C = 0.159$, $u = 5.02$, $\lambda = 0.70766$, $y^* = y|x^u(4at)$ and $x^* = x|x^u(4at)$ ($a$ is the thermal diffusivity and $a = 1 \text{ m}^2/\text{s}$ in this example). Figure 3 shows that the non-dimensionalized front

![Figure 3. Non-dimensionalized front position for solidification in an infinite corner region](image-url)
position obtained using the present methodology compares very well with the analytical solution, while Figure 4 gives a comparison of the two temperature solutions against \( x^* \) along the diagonal surface.

The second example is solidification of a \( 2 \times 2 \) (m\(^2\)) square prism with \( K_S = K_L = 1 \) W/m °C, \( c_S = c_L = 1 \) J/kg °C, \( L = 1.5613 \) J/kg and \( \rho = 1 \) kg/m\(^3\). The initial temperature is equal to the melting temperature, i.e. \( T_{in} = T_m = 0^\circ \) C. The temperature boundary condition applied to the surface of the prism is \( T_0 = -1^\circ \) C. Owing to the symmetry conditions of the problem, only one-quarter of the prism is considered. Since no analytical solution is available for solidification in a finite corner region, the results of the present algorithm are compared with the front tracking solution reported by Rao and Sastri\(^8\) and the enthalpy solution of Crowley.\(^9\)

The interface location along the diagonal and adiabatic surfaces of the prism are shown in Figure 5. The analytical solution for a half-infinite slab \( (s = 1.03376/t) \) is also shown in

Figure 4. Non-dimensionalized temperature \( T^* \) along the diagonal surface against \( x^* \) for solidification in an infinite corner region, where \( T^* = (T - T_m) / (T_m - T_0) \)

Figure 5. Front position along the diagonal and adiabatic surfaces for solidification of a square region (one-phase problem)
Figure 6. Front position along the adiabatic surface for solidification of a square region (two-phase problem)

Figure 7. Temperature history at point (0.4, 0.2) for solidification of a square region (two-phase problem)

Figure 8. Interface motion for solidification of a square region (two-phase problem)
Figure 5. It can be seen that the front position at the adiabatic surface coincides with the early time solution of the one-dimensional case.

The third example is solidification of a square prism of \(1 \times 1 \text{ m}^2\) and was also studied by Voller and Cross\(^{10}\) using the enthalpy method. The thermal properties are \(K_S = K_L = 2 \text{ W/m} \cdot \text{ C}\), \(c_S = c_L = 2.5 \text{ MJ/kg} \cdot \text{ C}\), \(L = 100 \text{ MJ/kg}\) and \(\rho = 1 \text{ kg/m}^3\). The melting and initial temperatures are \(T_0 = 0^\circ \text{ C}\) and \(T_{in} = 2^\circ \text{ C}\), respectively. A constant temperature boundary condition \(T_0 = -10^\circ \text{ C}\) is applied to the surface of the region. The results are plotted in Figures 6–8. Figure 6 gives the interface position against time on the adiabatic surface \(x = 0.5 \text{ m}\) (or \(y = 0.5 \text{ m}\)). The temperature history at the point \(x = 0.4 \text{ m}\) and \(y = 0.2 \text{ m}\) is plotted in Figure 7. It can be seen from these Figures that the present method produces solutions which are in excellent agreement with those obtained using the enthalpy method. Also the interface position at the adiabatic surface agrees well with the early time solution of the one-dimensional case \((s = 0.0346, t \text{ in hours})\). Finally, Figure 8 shows the progression of the front, which gives a general view on the interface motion and the location of the front nodal points (the symbols in the Figure indicate the front nodal points at the corresponding time).

CONCLUSIONS

A moving finite-element analysis of two-dimensional Stefan problems has been presented. An energy-preserving model of the interface motion is given, together with a new efficient mesh generation/node motion for the solid phase. The transfinite mapping technique is used for mesh generation in the liquid phase. The method is particularly tractable for solidification problems with initial temperature equal to the melting point. The solutions obtained with the present method were shown to be in excellent agreement with analytical solutions and other numerical schemes including front-fixing and fixed mesh techniques.

ACKNOWLEDGEMENTS

This work has been supported by NSF grant CBT-8802069 to the University of Minnesota. The computing has been supported by the Minnesota Supercomputer Institute and the Cornell National Supercomputer Facility, a resource of the Center for Theory and Simulation in Science and Engineering, which receives major funding from the National Science Foundation and IBM Corporation, with additional support from New York State and members of the Corporate Research Institute.

APPENDIX. CALCULATION OF INTERFACE NODAL VELOCITY COMPONENTS

The interface nodal velocity components at time \(t\) are calculated by solving the following system of equations:

\[
[A] [\alpha] = [H]
\]

(22)

where \([\alpha]\) denotes the interface nodal velocity components and the matrix \([A]\) is of order \((2M_i) \times (2M_i)\). \([A]\) and \([H]\) are calculated from assembling \(A^b\) and \(H^b\) defined on each interface segment \(b\) \((b = 1, 2, ..., E_I)\). For example, let us consider one linear interface segment
\[ b \text{ with nodes } 1 \text{ and } 2. \text{ Then, the matrix } A^b \text{ and the vectors } H^b \text{ and } \alpha^b \text{ can be obtained as:} \]

\[
A^b = \rho L\begin{pmatrix}
\int_{\Gamma} \Psi_1 \Psi_1 N_{y}^b \, d\Gamma & \int_{\Gamma} \Psi_2 \Psi_2 N_{y}^b \, d\Gamma & \int_{\Gamma} \Psi_3 \Psi_3 N_{y}^b \, d\Gamma & \int_{\Gamma} \Psi_4 \Psi_4 N_{y}^b \, d\Gamma \\
\int_{\Gamma} \Psi_1 \Psi_1 N_{x}^b \, d\Gamma & \int_{\Gamma} \Psi_2 \Psi_2 N_{x}^b \, d\Gamma & \int_{\Gamma} \Psi_3 \Psi_3 N_{x}^b \, d\Gamma & \int_{\Gamma} \Psi_4 \Psi_4 N_{x}^b \, d\Gamma \\
0 & 0 & t_{2x} & t_{2y} \\
0 & 0 & 0 & 0
\end{pmatrix}
\]

\[ (23) \]

\[
H^b = \begin{pmatrix}
\int_{\Gamma} [K_S \nabla T_S - K_L \nabla T_L] \cdot N_i \Psi_i^b \, d\Gamma \\
0 \\
\int_{\Gamma} [K_S \nabla T_S - K_L \nabla T_L] \cdot N_i \Psi_i^b \, d\Gamma \\
0
\end{pmatrix}
\]

\[ (24) \]

\[
\{\alpha^b\}^T = \{V_{ix}^b, V_{iy}^b, V_{ux}^b, V_{uy}^b\}
\]

\[ (25) \]

Here, \((N_x^b, N_y^b)\) are the directional cosines of the unit normal \(N^b\) to the segment \(b\), \((t_{ix}, t_{iy})\) denote the directional cosines of the unit tangent vector to \(\partial\Omega_i\) at the node \(i\), \(\Psi_i^b\) are the one-dimensional shape functions defined over the segment \(b\), and \((V_{ix}^b, V_{iy}^b)\) are the \(x\) and \(y\) components of the \(i\)th nodal velocity \((i = 1, 2)\).

The heat flux jump at the interface is written here explicitly for easy understanding of equation (24). The elements of the assembled vector \(H\) are calculated for each freezing front node using the compatibility equation obtained from the system of equations (12). Note that all variables involved in the above equations must be calculated at time \(t\).

REFERENCES
