JOURNAL OF COMPUTATIONAL PHYSICS 25, 163–181 (1977)

Numerical Computation of the Free Boundary for the Two-Dimensional Stefan Problem by Space-Time Finite Elements

R. BONNEROT AND P. JAMET

Commissariat à l'Énergie Atomique, Centre d'Études de Limeil,
Service de Mathématiques Appliquées, B.P. 27, 94190 Villeneuve-St-Georges, France

Received September 22, 1976

We describe a finite element method for the numerical solution of the two-dimensional Stefan problem. At each time step, the free boundary is approximated by a polygonal line whose vertices coincide with triangulation nodes. This is achieved by using space-time finite elements, which allow a change in the position of the nodes at each time step. Numerical results are given.

1. INTRODUCTION

The computation of the free boundary is the main difficulty in the numerical solution of the Stefan problem. In the one-dimensional case, there are various numerical methods which involve an explicit computation of the free boundary [8, 10, 14, 21, 23]. But, in the multidimensional case, the computation of the free boundary is much harder [9, 18]. Hence, during the last few years, several authors have developed methods which avoid computing the free boundary. These methods are based either on the numerical solution of a nonlinear partial differential equation [7, 22] according to a formulation of Oleinik [24] (see also [19]), or on the numerical solution of a variational inequality [1, 5, 13], according to a formulation of Baiocchi [2] and Duvaut [11, 12] (see also [20]). Let us also mention the alternating phase truncation method of Berger *et al.* [3] which is also related to the solution of a variational inequality.

Despite the interest of such methods, we have thought that the explicit computation of the free boundary in the multidimensional case remained an interesting problem and we have studied a method for that. This method uses space-time finite elements (see [15]), which allow one to move the nodes of the triangulation at each time step in such a way that the free boundary is always approximated by a polygonal curve whose vertices are nodes of the triangulation. This method can be applied provided the free boundary is sufficiently regular. It has already been used by the authors for two types of one-dimensional free boundary problems: the Stefan problem [4], and compressible flow [17].

Let us notice that although our method is basically different from the Isotherm Migration Method of Crank and Gupta [9], there is a certain resemblance in the

ISSN 0021-9991

approximation of the free boundary. Our method is more general since the initial function u^0 can be arbitrary whereas the I.M.M. method requires the monotonicity of u^0 with respect to one space-variable; also, it can be applied to a great variety of domains which need not even be simply connected.

In Section 2, we describe the problem that we want to solve and we derive an integral identity which is the basis of our method. In Section 3, we assume that the free boundary is known and we describe a numerical method to solve the heat equation in a given variable domain. In Section 4, we describe the computation of the free boundary. In Section 5 we give an account of numerical experiments. Section 6 contains some general remarks and comments. Finally, explicit expressions for the coefficients of the discrete equations are given in an Appendix.

2. Description of the Problem

Let D be a given domain in $\mathbb{R}^2$ and Γ_1 be a given portion of its boundary ∂D. Let $\Omega(t)$ be a bounded subdomain of D which depends on the time t and whose boundary $\partial\Omega(t)$ contains Γ_1. Let us write $\partial\Omega(t) = \Gamma_1 \cap \Gamma_2(t) \cup \mathscr{C}(t)$, where $\Gamma_2(t) \subset \partial D$, $\mathscr{C}(t) \subset D$ and $\Gamma_1 \cup \Gamma_2(t) = \varnothing$. The curve $\mathscr{C}(t)$ is the *free boundary*.

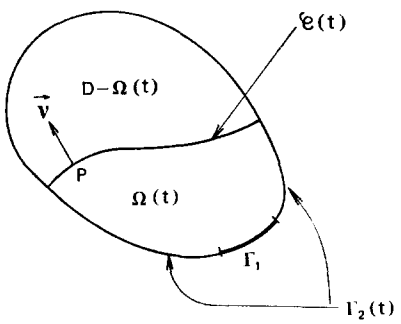

FIG. 1. The fixed domain D, the variable domain $\Omega(t)$, and the free boundary $\mathscr{C}(t)$.

We will use the following notation.

$\nu = \nu(P)$ = outward normal unit vector at a point $P \in \partial\Omega(t)$,

v_ν = outward normal speed of propagation of the free boundary $\mathscr{C}(t)$,

$\partial\varphi/\partial\nu$ = derivative along the vector ν of an arbitrary function φ,

$\Delta\varphi$ = Laplacian of the function φ.

$\mathscr{R} = \{(P, t);\ P \in \Omega(t),\ t > 0\}$,

$\Sigma_1 = \{(P, t);\ P \in \Gamma_1,\ t > 0\}$,

$\Sigma_2 = \{(P, t);\ P \in \Gamma_2(t),\ t > 0\}$,

$\mathscr{S} = \{(P, t);\ P \in \mathscr{C}(t),\ t > 0\}$.

Finally, for simplicity of notation, we will use the same notation for the domain $\Omega(t) \subset \mathbb{R}^2$ and the corresponding section of $\mathscr{R}$, i.e., the set $\{(P, t);\ P \in \Omega(t),\ t \text{ fixed}\}$. This should not lead to any confusion.

Let c be a given positive constant. Let $\Omega(0)$ be given and u^0 be a given positive function defined in $\Omega(0)$. Let g be a given positive function defined on Σ_1. Our problem is to determine the variable domain $\Omega(t)$ and a positive function u defined in $\mathscr{R}$ such that

$$(\partial u/\partial t) - \Delta u = 0 \qquad \text{in } \mathscr{R}, \tag{2.1}$$

$$u = u^0 \qquad \text{on } \Omega(0), \tag{2.2}$$

$$u = g \qquad \text{on } \Sigma_1, \tag{2.3}$$

$$(\partial u/\partial \nu) = 0 \qquad \text{on } \Sigma_2, \tag{2.4}$$

$$u = 0 \qquad \text{on } \mathscr{S}, \tag{2.5}$$

$$v_\nu = -c\,(\partial u/\partial \nu) \qquad \text{on } \mathscr{S}. \tag{2.6}$$

Equation (2.1) is the heat equation; (2.2) is the initial condition; (2.3), (2.4), and (2.5) are the boundary conditions; (2.6) determines the propagation of the free boundary.

Now, we transform the partial differential Eq. (2.1) into an integral identity which is the basis of our finite element method. Let $\mathscr{E}$ be the set of all continuous functions defined in $\mathscr{R}$, which admit bounded first-order derivatives in $\mathscr{R}$ and which vanish on $\Sigma_1 \cup \mathscr{S}$. Let θ_1 and θ_2 be two arbitrary nonnegative numbers, $0 \leqslant \theta_1 < \theta_2$, and let G be the intersection of $\mathscr{R}$ with the strip $\theta_1 < t < \theta_2$, i.e., $G = \{(P, t);\ P \in \Omega(t),\ \theta_1 < t < \theta_2\}$. Multiplying Eq. (2.1) by an arbitrary function $\varphi \in \mathscr{E}$, integrating by parts in G, and taking account of the boundary condition (2.4), we get

$$A_G(u, \varphi) = 0, \qquad \text{for all } \varphi \in \mathscr{E}, \tag{2.7}$$

where $A_G(u, \varphi)$ is the bilinear form

$$A_G(u, \varphi) = -\iiint_G u \frac{\partial \varphi}{\partial t}\, dP\, dt + \iiint_G \operatorname{grad} u \cdot \operatorname{grad} \varphi\, dP\, dt + \iint_{\Omega(\theta_2)} u\varphi\, dP - \iint_{\Omega(\theta_1)} u\varphi\, dP. \tag{2.8}$$

Conversely, any function u which satisfies the integral identity (2.7) for all θ_1 and θ_2 and which is sufficiently smooth, satisfies the partial differential Eq. (2.1) and the boundary condition (2.4). Subsequently, we will replace the two Eqs. (2.1) and (2.4) by the integral identity (2.7).

3. Numerical Solution of the Heat Equation in a Variable Domain

In this section, we assume that $\mathscr{R}$ is known and we describe a finite element method to solve the problem {(2.1)–(2.5)}. (We have excluded Eq. (2.6) since the free boundary is known.)

First, we discretize the domain $\mathscr{R}$ by means of isoparametric finite elements corresponding to a six-noded triangular prism (see [25, Section 7.12]). These elements are arranged in the following way. Let K_i^n denote an arbitrary element, where i is a space-index and n a time-index, $1 \leqslant i \leqslant I$, $n \geqslant 0$. Each element K_i^n admits two triangular bases T_i^n and T_i^{n+1} contained in the planes $t = t^n$ and $t = t^{n+1}$, respectively, with $t^{n+1} > t^n \geqslant t^0 = 0$. The two elements K_i^n dans K_i^{n-1} have in common the base T_i^n. Moreover, let $\Omega^n = \{\bigcup T_i^n;\ 1 \leqslant i \leqslant I\}$; its boundary $\partial\Omega^n$ is a polygonal line whose vertices are located on $\partial\Omega(t^n)$. Finally, let P_l^n, $1 \leqslant l \leqslant L$, denote the vertices of the triangles T_i^n, $1 \leqslant i \leqslant I$; these points are called the nodes of the triangulation and we say that two nodes P_l^n and $P_{l'}^{n'}$ are neighbors if they belong to a same edge of one of the elements K_i^n. Each node P_l^n admits a unique neighbor P_l^{n+1} in the plane $t = t^{n+1}$ and the following conditions must be satisfied. If $P_l^n \in \Sigma_1$ (resp. Σ_2 or the interior of $\mathscr{R}$), then $P_l^{n+1} \in \Sigma_1$ (resp. Σ_2 or the interior of $\mathscr{R}$); if $P_l^n \in \mathscr{S}$, then $P_l^{n+1} \in \mathscr{S} \cup \Sigma_2$. Note that, in the latter case, P_l^{n+1} does not necessarily belong to $\mathscr{S}$; this is related to the fact that the free boundary can disappear if the domain $\Omega(t)$ invades the whole domain D.

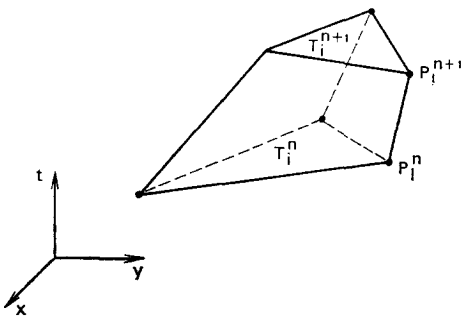

FIG. 2. A finite element K_i^n.

Now, we want to discretize the integral identity (2.7). Let $G_h^n = \{\bigcup K_i^n;\ 1 \leqslant i \leqslant I\}$ and $\mathscr{R}_h = \{\bigcup G_h^n;\ n \geqslant 0\}$, where h is a certain parameter which characterizes the discretization and which tends to zero as the mesh is uniformly refined. Let V_h be the set of all functions which are defined and continuous in $\mathscr{R}_h$ and whose restriction to each element K_i^n is linear along the edges and linear with respect to the space variables for arbitrary fixed t. For each n, we denote by $\mathscr{E}_h^n$ the set of all functions $\varphi_h \in V_h$ which satisfy $\varphi_h(P_l^n) = \varphi_h(P_l^{n+1})$ for all l and which vanish at the points $P_l^{n+1} \in \Sigma_1 \cup \mathscr{S}$. Let us choose $\theta_1 = t^n$, $\theta_2 = t^{n+1}$ and let $A^n(u, \varphi)$ be an approximation of the bilinear form $A_G(u, \varphi)$ obtained by replacing G by G_h^n and by approximating the integrals by means of numerical quadrature formulas that we will specify later on. Then, our discrete problem is the following. Find $u_h \in V_h$ such that

$$A^n(u_h, \varphi_h) = 0, \quad \text{for all } \varphi_h \in \mathscr{E}_h^n \text{ and all } n \geqslant 0, \tag{3.1}$$

$$u_h = u^0 \quad \text{at all the nodes } P_l^0, \tag{3.2}$$

$$u_h = g \quad \text{at all the nodes } P_l^n \in \Sigma_1, \tag{3.3}$$

$$u_h = 0 \quad \text{at all the nodes } P_l^n \in \mathscr{S}. \tag{3.4}$$

For each $n \geqslant 0$, (3.1) is a system of linear algebraic equations whose unknowns are the values of the function u_h at the nodes $P_l^{n+1} \in \mathscr{R} \cup \Sigma_2$. By taking φ_h equal to each of the functions $\varphi_h^{(l)} \in \mathscr{E}_h{}^n$ such that $\varphi_h^{(l)}(P_l{}^n) = 1$ and $\varphi_h^{(l)}(P_j{}^n) = 0$ for $j \neq l$, we get a system of equations with a sparse matrix; each equation involves only the unknown values of u_h at the node P_l^{n+1} and its neighbors in the plane $t = t^{n+1}$.

It remains to specify the quadrature formulas that we have chosen to approximate the bilinear form (2.8).

The space integrals are approximated by using the following formula on each triangle $T_i{}^n$

$$\iint_{T_i^n} \psi \, dP \sim \tfrac{1}{3} (\text{Meas. } T_i{}^n) \sum_l \psi(P_l{}^n) = \mathscr{J}(\psi; T_i{}^n), \tag{3.5}$$

where ψ is an arbitrary function, Meas. $T_i{}^n$ denotes the area of the triangle $T_i{}^n$ and the sum is taken for the three vertices $P_l{}^n$ of the triangle $T_i{}^n$.

The triple integrals are approximated by using the following formula on each element $K_i{}^n$

$$\iiint_{K_i^n} \psi \, dP \, dt \sim \tfrac{1}{2}(t^{n+1} - t^n)(\mathscr{J}(\psi; T_i{}^n) + \mathscr{J}(\psi; T_i^{n+1})). \tag{3.6}$$

The coefficients of Eqs. (3.1) can be computed in the same way as in the one-dimensional case [4].We will not describe the details of this computation. We refer to the Appendix for the explicit expressions of the coefficients.

Particular case. Assume that the elements $K_i{}^n$ are *right* prisms and that their bases are equal right triangles whose sides are parallel to three fixed directions (see Fig. 3). Then, it is easy to check that, our method is identical to the *Crank–Nicolson scheme* (but, variable domains cannot be treated in this way, since the position of the nodes is kept fixed).

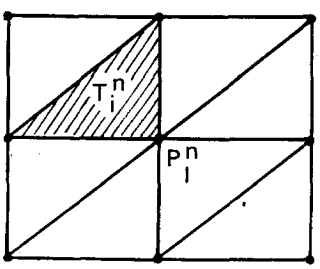

FIG. 3. Triangulation with equal right triangles.

4. COMPUTATION OF THE FREE BOUNDARY

We will now describe how we determine the free boundary at each time-step. In this section, we consider only the space position of the points and curves of $\mathscr{R}$; but, for simplicity, we use the same notation to denote a point or curve of $\mathscr{R}$ and its projection on the space $\mathbb{R}^2$.

Let $\mathscr{C}^n$ be the portion of $\partial\Omega^n$ which approximates the free boundary $\mathscr{C}(t^n)$ at the time t^n; it is composed by all the straight segments of $\partial\Omega^n$ which have at least one extremity inside $\mathscr{R}$. The extremities of $\mathscr{C}^n$, are located on the boundary of D (see Fig. 4).

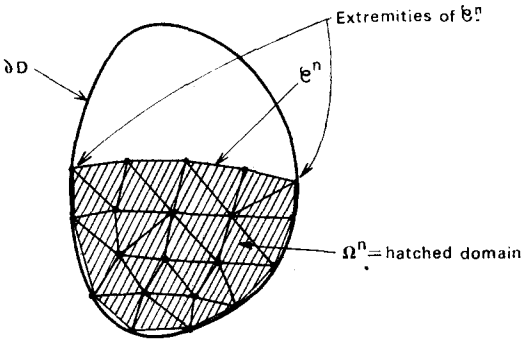

FIG. 4. The discrete domain Ω^n and free boundary $\mathscr{C}^n$.

Let P_l^n be an arbitrary vertex of $\mathscr{C}^n$. We want to determine the corresponding point P_l^{n+1} at the time $t = t^{n+1}$. To simplify the notation, we will omit the index l; thus, we will write P^n and P^{n+1} instead of P_l^n and P_l^{n+1}.

We consider two cases.

Case 1. The point P^n is not an extremity of $\mathscr{C}^n$. Then, it admits two neighbors on $\mathscr{C}^n$, say P_1^n and P_2^n. Let us remark that Eq. (2.6) involves only the normal speed of propagation. Therefore, we can move the point P^n in an arbitrary direction. Let δ be the straight line along which we want to move the point P^n in order to get the point P^{n+1}. We impose the two conditions: the two segments $P^nP_1^n$ and $P^nP_2^n$ must be located on opposite sides of δ and the angle of each of them with δ must not be "too small"; for example, we could choose δ to be the bissectrix of the angle determined by these two segments. In the same way, we denote by δ_1 and δ_2 the two straight lines along which we want to move the points P_1^n and P_2^n in order to get P_1^{n+1} and P_2^{n+1}.

Let ν_j be the unit vector which is perpendicular to the segment $P^nP_j^n$, for $j = 1$ and 2, and oriented towards the outside of Ω^n. Let $\tau_j = -c(t^{n+1} - t^n)(\partial u_h/\partial\nu)_j\,\nu_j$, where $(\partial u_h/\partial\nu)_j$ denotes the constant value of the outward normal derivative of u_h on the segment $P^nP_j^n$. Let S_j^n be the straight line through the points P^n and P_j^n, and let S_j' be its image by the τ_j translation. We consider the following points: $P_j' = S_j' \cap \delta$, $P_{jj}' = S_j' \cap \delta_j$, Q_j = midpoint of the segment $P_j'P_{jj}'$ for $j = 1$ and 2, $P_3' = \delta \cap [Q_1Q_2]$ where $[Q_1Q_2]$ denotes the straight segment joining Q_1 and Q_2. The three points P_1', P_2', and P_3' are located on the straight line δ; we denote by P' the point among these three which is located between the two others or possibly coincides with one of the others. (See Figs. 5 and 5′ which show two different cases.)

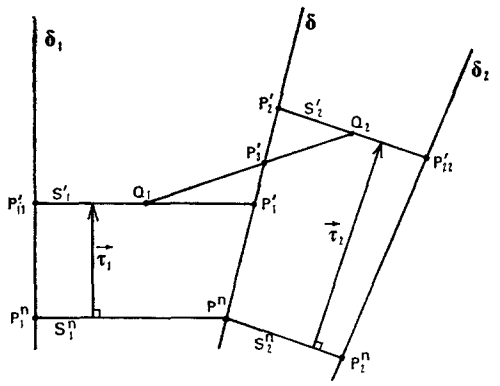

FIG. 5. Displacement of the free boundary. In this case, we have $P' = P_3'$.

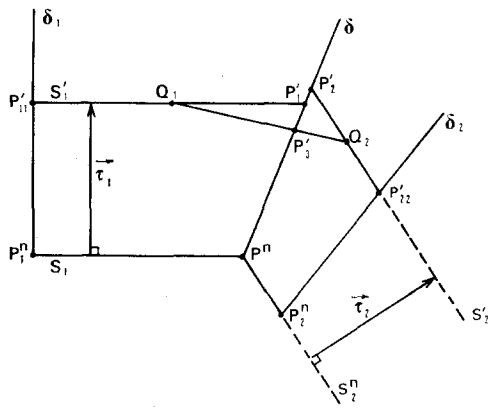

FIG. 5'. Displacement of the free boundary. In this case, we have $P' = P_1'$.

Let (P^nP') denote the open straight segment joining the points P^n and P', and $[P^nP']$ denote the corresponding closed segment. Then, we determine the point P^{n+1} as follows.

If $(P^nP') \subset D$, we take $P^{n+1} = P'$.

If (P^nP') is not contained in D, we take P^{n+1} such that $P^{n+1} \in [P^nP'] \cap \partial D$ and $(P^nP^{n+1}) \subset D$.

Remark. A simpler method to determine the point P^{n+1} is to take it at the midpoint of the segment $P_1'P_2'$. However, numerical experiments have exhibited oscillations of the free boundary; although these oscillations were small, we have modified this method in order to cancel them. By taking $P^{n+1} = P_3'$, we have introduced a dissipative effect in the computation of the free boundary; but, in order that the scheme remain consistent for arbitrarily small time steps, we have added the condition that the point P^{n+1} should remain between the points P_1' and P_2' which led us to the method described above.

Case 2. The point P^n is an extremity of $\mathscr{C}^n$. Then P^n is located on ∂D and admits one neighbor $P_1^n \in D \cap \mathscr{C}^n$. The point P^{n+1} must also be located on ∂D. To determine this point, we use one of the two following methods.

Method 1. We take $P^{n+1} = \partial D \cap S_1'$, where S_1' is the straight line previously defined.

Method 2. We do not move the point P^n; we take $P^{n+1} = P^n$. The point P^n will cease to be an extremity of $\mathscr{C}^n$ when its neighbor P_1^n reaches the boundary ∂D. We apply this method when the neighbor point P_1^n is moved towards the boundary. We refer to the next section for practical examples.

5. Numerical Experiments

We will present numerical results for two different problems.

(a) *Problem* 1

Let $\{x, y\}$ be a set of Cartesian orthogonal coordinates in $\mathbb{R}^2$ and

$$D = \{(x, y);\ 0 < x < 1,\ 0 < y < 4\}, \qquad \Gamma_1 = \{(x, 0);\ 0 \leqslant x \leqslant 1\},$$
$$\Omega(0) = \{(x, y);\ 0 < x < 1,\ 0 < y < 2 + \cos \pi x\},$$
$$u^0(x, y) = 1 - (y/(2 + \cos \pi x)), \qquad g = 1,\ c = 1.$$

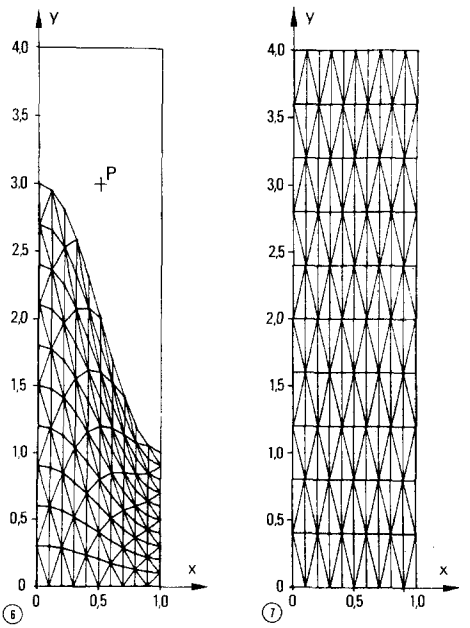

FIG. 6. Problem 1. The domain D, the domain $\Omega(0)$, and the initial triangulation. The point P is the one which corresponds to Fig. 9.

FIG. 7. The triangulation at final time $t = 15$. We have $\Omega(t) = D$; the free boundary has vanished.

We choose a triangulation whose nodes are located on fixed straight lines which are parallel to the y-axis and equidistant; on each of these lines the nodes are equally spaced. Figure 6 represents the triangulation at time $t = 0$; we have taken ten intervals in the direction of each of the coordinates x and y, which yields 200 triangles.

At each time-step, the nodes are moved in the y-direction. The displacement of those which are on the free boundary is computed by the method described in Section 4. For the extremities of the free boundary, we use method 1 of Section 4 if they are located on either of the two sides of the rectangle D which are parallel to the y-axis; we use method 2 if they are located on the side Γ_1' opposite to Γ_1 (which happens when the free boundary reaches Γ_1'). The interior nodes are moved in such a way that they remain equally spaced on each node line parallel to the y-axis.

We have performed the computations until time $t = 15$ with time-step $\Delta t = 0.1$. Figure 7 represents the final triangulation: the domain $\Omega(t)$ has invaded the whole domain D and the free boundary has vanished. Figure 8 shows the propagation of

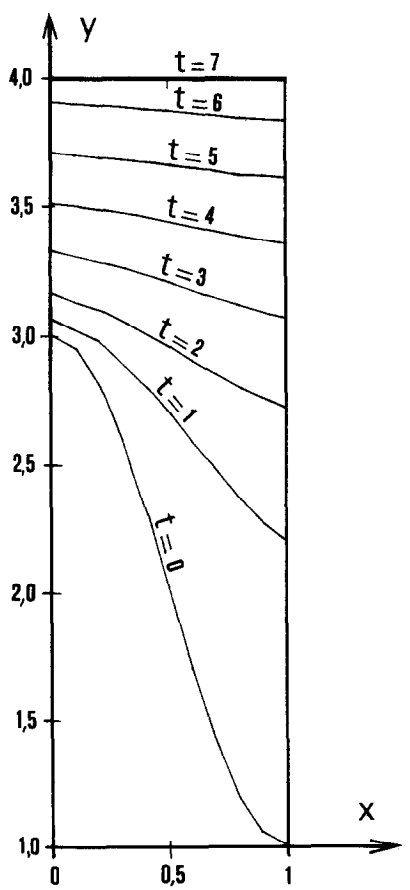

FIG. 8. Problem 1. Propagation of the free boundary.

the free boundary. Each curve represents the free boundary for a fixed value of t. Figure 9 represents the temperature $u(P, t)$ at the fixed point $P = (0.5, 3)$ as a function of t. On the corresponding curve we can distinguish the four following periods.

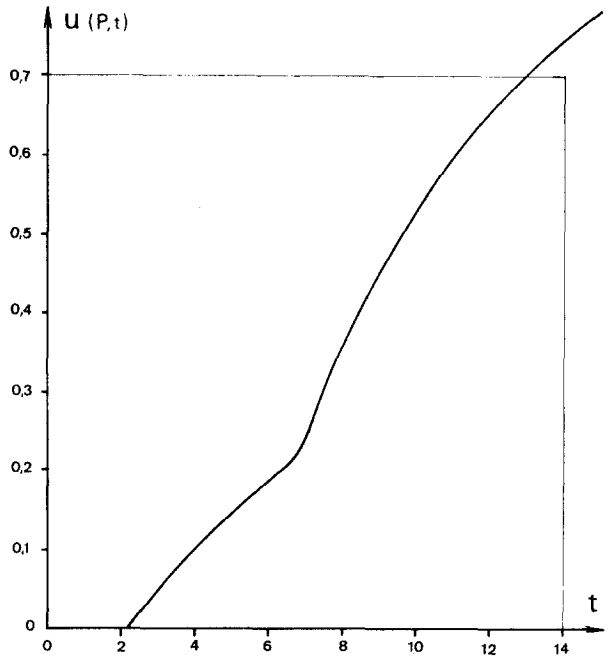

FIG. 9. Problem 1. Variation of the temperature u as a function of time t at the fixed point P. $x = 0.5$, $y = 3$.

Period 1. $0 < t < 2.1$. The free boundary has not yet reached the point P. We have $u(P, t) = 0$.

Period 2. $2.1 < t < 6.6$. The free foundary has passed the point P, but has not yet reached the side Γ_1'. The temperature increases.

Period 3. $6.6 < t < 6.9$. The free boundary has reached the side Γ_1', but the corresponding domain $\Omega(t)$ has not yet invaded the whole domain D. The increase of temperature speeds up; the curve admits an upward curvature.

Period 4. $t > 6.9$. The domain $\Omega(t)$ has invaded the whole domain D; the free boundary has disappeared. The curve admits a downward curvature. The temperature keeps on increasing towards the asymptotic value $u = 1$.

Finally, Fig. 10 represents the temperature along the midline $x = 0.5$ for fixed values of t.

We have not worked on minimizing the computation time. At each time-step, the system of linear algebraic Eq. (3.1) has been solved by overrelaxation with the coefficient $\omega = 1.6$; the iterations are stopped when the relative difference between two

consecutive iterated values of u goes below 0.00001 at all nodes, which yields 30 iterations on the average. In this way, the computation takes 15 seconds on a computer CDC 7600.

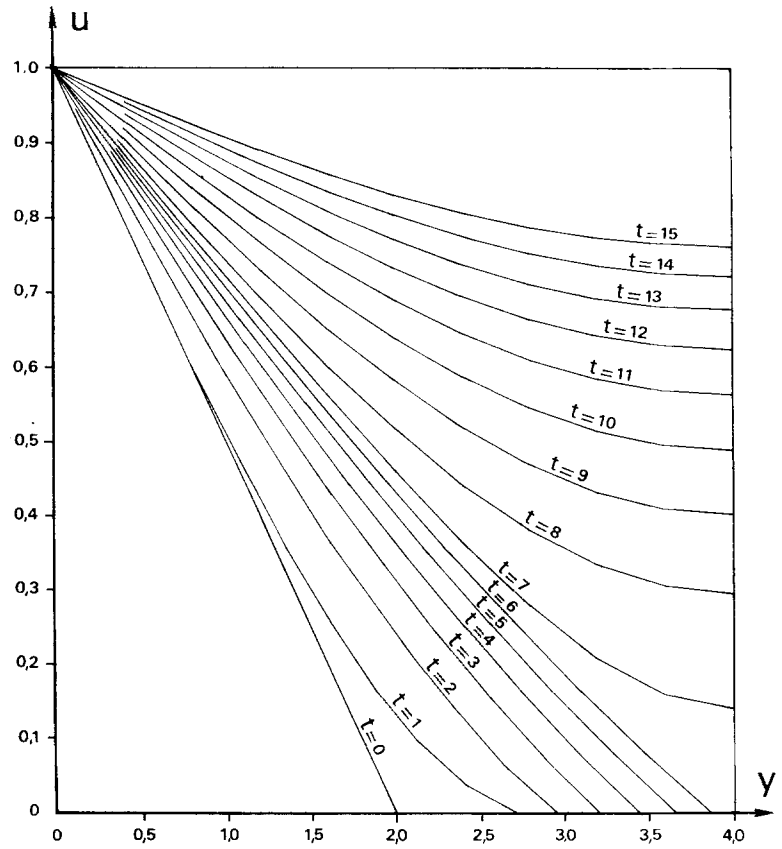

FIG. 10. Problem 1. Variation of the temperature u along the midline $x = 0.5$ for fixed values of t.

Stability. In order to study the stability of the method, we have made computations with various values of the time-step. We have also experimented with various values of the constant c of Eq. (2.6) to check if the stability is related to the speed of propagation of the free boundary.

These experiments have shown no instability. However, for each Δx and c fixed, oscillations appear as the time-step is increased. This phenomenon is exhibited on Figs. 11 and 12. They represent the value of u_h at the mesh point P_1^n located next to the free boundary on the line $x = 0$ as a function of the time-index n, for $\Delta x = 1/16$, $c = 1$, and for two different values of Δt. On Fig. 11, we have $\Delta t = \Delta x/4$; the curve is regular. On Fig. 12, we have $\Delta t = 4\Delta x$; the curve oscillates and some values of u_h are negative; but the oscillations are damped and the method is stable.

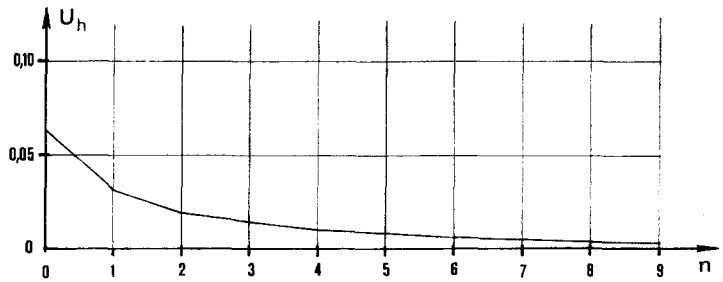

FIG. 11. Problem 1. The value of u_h at a mesh-point near the free boundary as a function of the time-index n for $c = 1$, $\Delta x = 1/16$, and $\Delta t = \Delta x/4$.

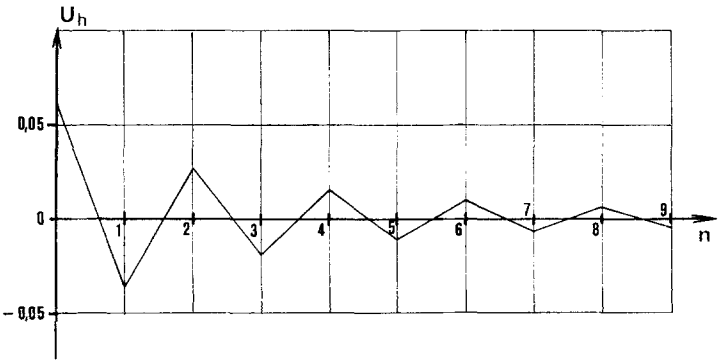

FIG. 12. Same as Fig. 11 with $\Delta t = 4\Delta x$. Oscillations appear, but the method remains stable.

TABLE I

Occurrence of Oscillations and Negative Values of u_h ;
$\lambda = \Delta t/\Delta x$, $\Delta x = \frac{1}{16}$

	λ			
c	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1
1	+	+	+	−
2	+	+	−	−
4	+	−	−	−

Table I shows that the value of Δt above which oscillations occur depends on the speed of propagation of the free boundary; we have indicated for each value of c and Δt whether u_h admits negative values or remains positive at all mesh points; we have noted − in the first case and + in the second case.

Convergence and order of accuracy. In order to test the accuracy of the method, we have performed a series of computations with $\Delta t = \Delta x = h = 2^{-p}$ for $p = 2$,

3, 4, 5, 6. Let $y(x, t)$ be the value of the ordinate y on the free boundary as a function of x and t and let y_h denote the corresponding approximation. The values of y_h for $x = 0, 0.5, 1$ and for $t = 2$ are given on Table II with the values of the differences $\delta y_h = y_h - y_{2h}$ and of the ratios $Ry_h = \delta y_h / \delta y_{2h}$. Due to the greater complexity of the two-dimensional computations, it has not been possible to decrease h as much as in the one-dimensional case [4]. However the table shows that δy_h decreases as h decreases and Ry_h is of the order of 2 for the smallest value of h. Hence, it appears that the method is accurate of order 1.

TABLE II

Convergence and Accuracy of the Method

	$x = 0$			$x = 0.5$			$x = 1$		
$1/h$	y_h	δy_h	Ry_h	y_h	δy_h	Ry_h	y_h	δy_h	Ry_h
4	3.353			3.259			2.949		
8	3.181	0.172		2.978	0.280		2.712	0.236	
16	3.122	0.058	2.9	2.902	0.076	3.7	2.679	0.033	7.1
32	3.085	0.037	1.6	2.843	0.058	1.3	2.635	0.043	0.7
64	3.068	0.017	2.1	2.810	0.033	1.7	2.610	0.025	1.6

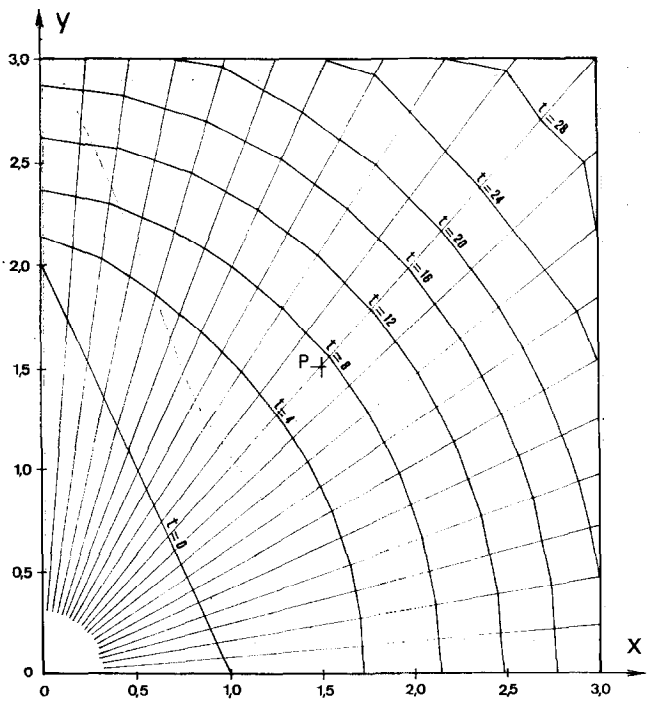

FIG. 13. Problem 2. Propagation of the free boundary. The point P is the one which corresponds to Fig. 14.

(b) *Problem* 2

Let $D = \{(x, y);\ 0 < x < 3,\ 0 < y < 3\}$, $\Gamma_1 =$ the point $(0, 0)$, $\Omega(0) = \{(x, y);\ x > 0,\ y > 0,\ 2x + y < 2\}$, $u^0(x, y) = 1 - x - y/2$, $g \equiv 1$, $c = 1$.

We choose the triangulation nodes equally spaced on fixed radial straight lines through the point $0 = (0, 0)$. We have divided the right angle (Ox, Oy) into 20 equal angles and each radius into 20 equal subintervals, which yields 800 triangles. At each time-step the nodes are moved along the corresponding radii. For the extremities of the free boundary, we use method 1 or method 2 of Section 4 according to whether they are located on the sides of the square D which coincide with the coordinate axes or on the two opposite sides. We have performed the computation until $t = 40$ with the time-step $\Delta t = 0.2$.

Figure 13 shows the propagation of the free boundary. Let us notice that the representation of the free boundary is not good when t approaches the time of its disappearance; in fact, it is near the point (1, 1), where the disappearance of the free boundary occurs, that the triangulation is the coarser; by refining the triangulation in this region we could get a better representation of the free boundary near the end of its existence.

Figure 14 represents the temperature at the center P of the square D, as a function

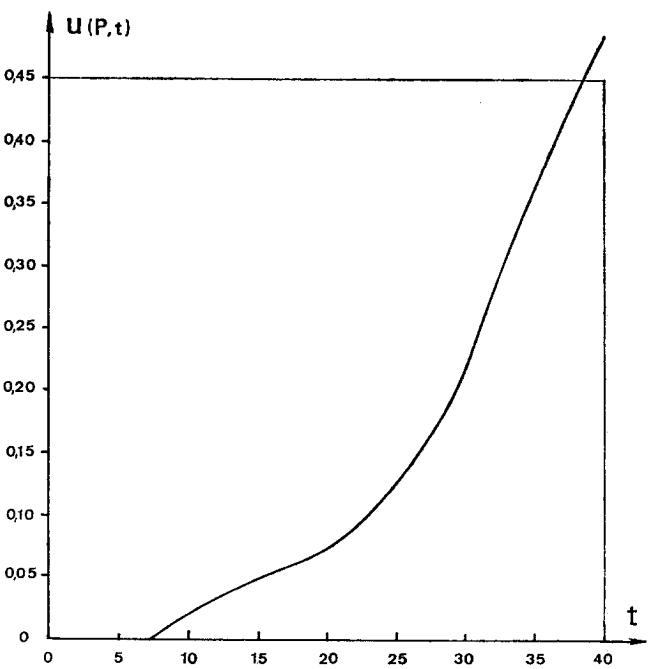

FIG. 14. Problem 2. Variation of the temperature u as a function of time t at the fixed point P. $x = 1.5, y = 1.5$.

of t; on the curve we distinguish four periods which are similar to those described in the previous problem. Finally, Fig. 15 represents the temperature along the diagonal line $x = y$ as a function of the distance from the point 0 for fixed values of t.

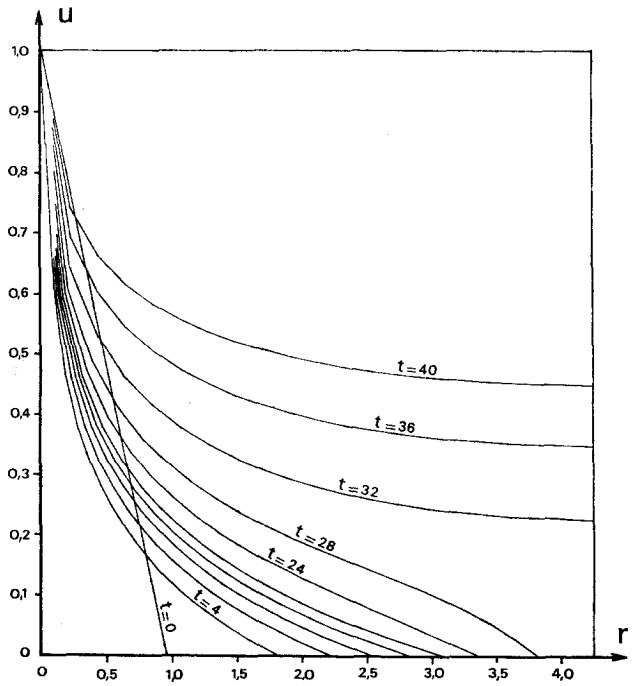

FIG. 15. Problem 2. Variation of the temperature u along the diagonal $x = y$ as a function of the distance r from the point 0. The curves correspond to values of t which are multiples of 4.

6. GENERAL REMARKS

6.1. *Time Discretization*

To the authors' knowledge, all other finite element methods for time-dependent problems use spatial finite elements and lead to a system of ordinary differential equations with respect to the time t; then, the numerical solution of these differential equations requires a discretization in time. In the present method, both discretizations (in space and in time) are made simultaneously.

In the case of finite elements whose position is kept fixed, our approach provides a new way of deriving some well-known methods for solving partial differential equations. We have already mentioned that the discrete Eqs. (3.1) contain the *Crank–Nicolson scheme* as a particular case. Other schemes corresponding to other time discretizations can be derived by changing either the space of trial functions $\mathscr{E}_h^n$

or the quadrature formula (3.6). For example, one obtains the classical *first-order explicit scheme* for the heat equation in *either* of the two following ways.

(i) Replace the space $\mathscr{E}_h^n$ defined in Section 3, by the space of all functions $\varphi_h \in V_h$ such that $\varphi_h(P_l^{n+1}) = 0$ at all nodes P_l^{n+1}, $1 \leqslant l \leqslant L$, and $\varphi_h(P_l^n) = 0$ if $P_l^{n+1} \in \sum_1 \cup \varphi$.

(ii) Replace the quadrature formula (3.6) by

$$\iiint_{K_i^n} \psi \, dP \, dt \sim (t^{n+1} - t^n) \, \mathscr{J}(\psi; T_i^n).$$

(Note that this formula which can be applied to right prisms is not good if the two bases T_i^n and T_i^{n+1} do not have equal areas.)

Similarly, one obtains the classical *first-order implicit scheme* in either of the two following ways.

(i) Replace the space $\mathscr{E}_h^n$ by the space of all functions $\varphi_h \in V_h$ such that $\mathscr{S}_h(P_l^n) = 0$ at all nodes P_l^n, $1 \leqslant l \leqslant L$, and $\varphi_h(P_l^{n+1}) = 0$ if $P_l^{n+1} \in \sum_1 \cup \mathscr{S}$.

(ii) Replace the quadrature formula (3.6) by

$$\iiint_{K_i^n} \psi \, dP \, dt \sim (t^{n+1} - t^n) \, \mathscr{J}(\psi; T_i^{n+1}).$$

We have not retained the explicit scheme because of the well-known stability condition which is too stringent (it is necessary to take the time-steps of the order of d^2, where d is the space mesh-size). On the contrary, the numerical experiments described, in Section 5 show that, our method remains stable for large values of the time-step like the Crank–Nicolson scheme.

More accurate time discretizations can be obtained by choosing more complicated trial functions and a more accurate quadrature formula.[1]

But the essential interest of space-time finite elements lies in their application to time-dependent meshes. In particular, they provide a practical way of extending to variable meshes certain usual finite difference or finite element methods.

6.2. *Accuracy*

The numerical experiments of Section 5 show that the method is of order 1.

It would be useless to improve the accuracy of the time discretization without improving also the accuracy of the space discretization; this would require the use of curved finite elements near the free boundary; the normal derivative of u_h would not be piecewise constant anymore along the free boundary and would also require an improved approximation; the method for computing the displacement of the free

[1] A mathematical theory of the approximation of parabolic problems in a time-dependent *given* domain by means of space-time finite elements will be given in a forthcoming paper with stability results and error estimates of arbitrary order [26].

boundary should also be changed and improved. It seems that the improvement of the global order of accuracy of our method in the two-dimensional case cannot be obtained without many complications. Let us recall that the situation is different in the one-dimensional case. In [4] a simple method of the second order of accuracy is described and tested.

6.3. *Displacement of the Interior Nodes*

The displacement of the interior nodes at each time step can be chosen arbitrarily provided the corresponding prisms K_i^n are "well conditioned" (see [6]). This condition implies that the angle of any two edges which have a common vertex must not be too small. A weaker condition can be derived from [16]. Each triangle T_i^n may admit one small angle, but no angle should approach π. This weaker condition is satisfied by the triangles used in problem 2 of Section 5; in particular, the triangles, which admit the origin as a vertex, have a small angle at the origin, especially if the mesh is refined, but none of the two other angles approaches π.

One possibility is to keep the interior nodes fixed for a certain number of time steps and then update the whole triangulation. The advantage of this procedure would be to avoid computing new coefficients for the discrete Eqs. (3.1) at each time-step. However, since the domain is expanding, the triangles which have vertices on the free boundary would become much larger than the interior triangles; this would be specially bad for the approximation of the normal derivative $\partial u/\partial \nu$.

An intermediate procedure is to fix only the nodes which are "far enough" from the free boundary and share the effect of the domain expansion between several layers of triangles along the free boundary. The complete rearrangement of all the nodes can be made periodically after a fixed number of steps. This procedure, which was suggested by one of the referees, would save computer time.

6.4. *Application to More Complex Problems*

We have considered only the simplest case of the Stefan problem. The method can also be applied to the *two-phase Stefan problem* in which the temperature u has to be determined in both phases on each side of the free boundary.

APPENDIX: Explicit Expressions for the Coefficients of Discrete Eqs. (3.1)

Let $P_{l_0}^{n+1}$ be an arbitrary node which is located in the interior of $\mathscr{R}$ or on Σ_2. There corresponds to this node the equation obtained by taking $\varphi_h = \varphi_h^{(l_0)}$ in (3.1), with $\varphi_h^{(l)}$ defined in Section 3.

This equation is of the form $L(u_h) = 0$, where $L(u_h)$ is the sum of linear forms $L_T(u_h)$ relative to each of the triangles $T = T_i^n$ which admit $P_{l_0}^n$ as a vertex. Each form $L_T(u_h)$ is obtained by restricting the integrals of (2.8) to the prism K_i^n which corresponds to the triangle T_i^n and by performing numerical integration by means of the quadrature formulas (3.5), (3.6). We will give the explicit expression of the

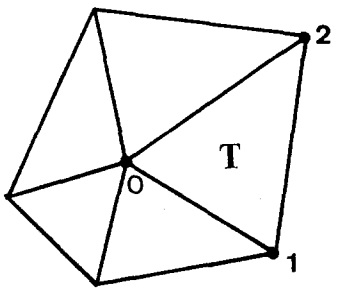

FIG. 16. The triangle T with vertices P_{l_s}; only the index s is indicated.

linear form $L_T(u_h)$ for an arbitrary triangle T with vertices $P^n_{l_0}$, $P^n_{l_1}$, $P^n_{l_2}$, ordered counterclockwise. Let $u_s{}^n = u_h(P^n_{l_s})$ for $s = 0, 1, 2$; then

$$L_T u_h = \sum_{s=0}^{2} (a_s{}^n u_s{}^n + a_s^{n+1} u_s^{n+1}),$$

where the expressions of $a_s{}^n$ and a_s^{n+1} are given below. All the coefficients and all the variables which appear in these expressions admit a superscript n or $n+1$. For simplicity, we omit this superscript when it can take either of these two values; thus, in each expression, all the missing superscripts must be replaced by $n + \tau$, with $\tau = 0$ or 1. On the other hand, we denote by x_s and y_s the coordinates of the point P_{l_s} with respect to an arbitrary system of orthogonal coordinate axes. With these notations, we have

$$a_s = \frac{\alpha\beta_s - \beta\alpha_s}{12\alpha} + \frac{(t^{n+1} - t^n)}{4\alpha}\gamma_s + \frac{\alpha}{6}\epsilon_s,$$

for $s = 0$, 1 and 2, with

$$\alpha = (x_1 - x_0)(y_2 - y_0) - (y_1 - y_0)(x_2 - x_0),$$

$$\alpha_s = (x_s - x_0)(y_2 - y_1) - (y_s - y_0)(x_2 - x_1),$$

$$\begin{aligned}\beta = {} & (x_1 - x_0)[(y_2 - y_0)^{n+1} - (y_2 - y_0)^n] - (y_1 - y_0) \\ & \times [(x_2 - x_0)^{n+1} - (x_2 - x_0)^n] \\ & - (x_2 - x_0)[(y_1 - y_0)^{n+1} - (y_1 - y_0)^n] + (y_2 - y_0) \\ & \times [(x_1 - x_0)^{n+1} - (x_1 - x_0)^n],\end{aligned}$$

$$\begin{aligned}\beta_s = {} & (x_s - x_0)[(y_2 - y_1)^{n+1} - (y_2 - y_1)^n] - (y_s - y_0) \\ & \times [(x_2 - x_1)^{n+1} - (x_2 - x_1)^n] \\ & + (x_2 - x_1)(y_0^{n+1} - y_0{}^n) - (y_2 - y_1)(x_0^{n+1} - x_0{}^n),\end{aligned}$$

$$\gamma_s = (y_2 - y_1)\gamma_s' + (x_2 - x_1)\gamma_s'',$$

$$\gamma_0' = y_2 - y_1, \qquad \gamma_1' = y_0 - y_2, \qquad \gamma_2' = y_1 - y_0,$$

$$\gamma_0'' = x_2 - x_1, \qquad \gamma_1'' = x_0 - x_2, \qquad \gamma_2'' = x_1 - x_0,$$

$$\epsilon_0{}^n = -1, \qquad \epsilon_0^{n+1} = 1, \qquad \epsilon_1 = \epsilon_2 = 0.$$

Let us notice that the foregoing expressions are antisymmetric with respect to the indices 1 and 2; if these two indices are interchanged the coefficients a_0, a_1, and a_2 are changed into $-a_0$, $-a_2$, and $-a_1$, respectively.

REFERENCES

1. J. AGUIRE-PUENTE AND M. FREMOND, Frost propagation in wet porous media, *in* "Joint IUTAM/IMU Symposium on Applications of Methods of Functional Analysis to Problems of Mechanics, Marseille 1975," Springer–Verlag, New York/London, 1976.
2. C. BAIOCCHI, *C. R. Acad. Sci. Paris* **273** (1971), 1215–1217.
3. A. E. BERGER, M. CIMENT, AND J. C. W. ROGERS, Numerical solution of a Stefan problem by a technique of alternating phase truncation *in* "Séminaire IRIA, Analyse et contrôle des systèmes, 1975."
4. R. BONNEROT AND P. JAMET, *Internat. J. Num. Met. Engrg.* **8** (1974), 811–820.
5. J. F. BOURGAT, Analyse numérique du problème de Stefan à deux phases, rapport IRIA, to appear.
6. P. G. CIARLET AND P. A. RAVIART, *Comp. Meth. Appl. Engrg.* **1** (1972), 217–249.
7. J. F. CIAVALDINI, *SIAM J. Numer. Anal.* **12** (1975), 464–487.
8. J. CRANK AND R. S. GUPTA, *J. Inst. Math. Appl.* **10** (1972), 296–304.
9. J. CRANK AND R. S. GUPTA, *Int. J. Heat Mass Transfer* **18** (1975), 1101–1107.
10. J. DOUGLAS AND T. M. GALLIE, *Duke Math. J.* **22** (1955), 557–571.
11. G. DUVAUT, *C. R. Acad. Sci. Paris* **276** (1973), 1461–1463.
12. G. DUVAUT, "Problèmes à frontière libre en théorie des milieux continus, rapport IRIA," No. 185 (1976).
13. M. FREMOND, Variational formulation of the Stefan problem, coupled Stefan problem, frost propagation in porous media, *in* "International Conference on Computational Methods in Nonlinear Mechanics, Austin, Texas, 1974."
14. J. GRAVELEAU, rapport du Com. Energ. Atomique, Centre d'Études de Limeil, 1972.
15. P. JAMET, Éléments finis espace-temps pour la résolution numérique de problèmes de frontières libres, *in* "Séminaire de mathématiques supérieures Montréal 1975," Presses de l'Université de Montréal.
16. P. JAMET, *R.A.I.R.O. Anal. Numer.* **10**(3) (1976), 43–61.
17. P. JAMET AND R. BONNEROT, *J. Computational Phys.* **18** (1975), 21–45.
18. A. LAZARIDIS, *Int. J. Heat Mass Transfer* **13** (1970), 1459–1477.
19. J. L. LIONS, "Quelques méthodes de résolution des problèmes aux limites non linéaires," Dunod, Paris, 1969.
20. J. L. LIONS, Introduction to some aspects of free surface problems, *in* "Numerical Solution of Partial Differential Equations-III," SYNSPADE (B. Hubbard, Ed.), Academic Press, New York, 1975.
21. G. H. MEYER, *Numer. Math.* **16** (1970), 248–267.
22. G. H. MEYER, *SIAM J. Numer. Anal.* **10** (1973), 522–538.
23. T. NOGI, *Publ. RIMS*, Kyoto Univ. **9** (1974), 543–575.
24. O. OLEINIK, On Stefan-type free boundary problems for parabolic equations, *Sem. Ist. Naz. Alta Mat.* (1962–1963), 388–403.
25. O. C. ZIENKIEWICZ, "The Finite Element Method in Engineering Science," McGraw–Hill, London, 1971.
26. P. JAMET, "Galerkin-type approximations which are discontinuous in time for parabolic equations in a variable domain," Report No. 1771, Mathematics Research Center, University of Wisconsin, July 1977.
27. M. MORI, "A finite element method for solving moving boundary problems," IFIP working conference on modelling of environmental systems, Tokyo, pp. 167–171, 1976.