A MOVING FINITE ELEMENT SOLUTION OF THE ONE-DIMENSIONAL CRANK-GUPTA PROBLEM

R.O. MOODY

NUMERICAL ANALYSIS REPORT 17/85

1. INTRODUCTION

The Crank-Gupta problem, [1],[2] (so named after its pioneers), is concerned with the absorption and diffusion of oxygen in tumour tissue. In the first stage, oxygen is permitted to diffuse in the tissue under the constraint of a fixed surface concentration until a steady state is attained. Crank and Gupta examined the second stage, that in which the boundary is sealed and the oxygen content in the tissue is reduced to zero by being externally extracted; this process causes the inward motion of the oxygen boundary, thus presenting a moving boundary problem. We follow Crank and Gupta in treating a one-dimensional model of the second stage of this problem.

The one-dimensional parabolic partial differential equation governing the concentration of oxygen, $u(x,t)$, is

$$u_t = u_{xx} - 1 \quad . \tag{1.1}$$

There is a stationary boundary at $x = 0$ which permits no oxygen leakage and therefore imposes the Neumann condition

$$u_x(0,t) = 0 \quad , \qquad t \geq 0 \quad . \tag{1.2}$$

The other boundary, which is also impermeable, may move, but maintains zero concentration: the condition is therefore

$$u_x(x_0,t) = 0 = u(x_0,t), \qquad t \geq 0 \quad , \tag{1.3}$$

where $x_0 = x_0(t)$ is the position of this boundary at time t, and must be computed as part of the solution.

The steady state solution of the first stage of the problem with no absorption supplies the initial condition

$$u(x,0) = \tfrac{1}{2}(1-x)^2, \quad 0 \leq x \leq 1 \quad , \tag{1.4}$$

with the boundary position

$$x_0(0) = 1. \tag{1.5}$$

Over the last decade numerical results for this problem have been obtained using a variety of methods, including the original finite difference approach incorporating a short-time analytical solution via Laplace transforms by Crank and Gupta (see [1] and [2]). Hansen and Hougaard [3] derived an integral equation for x_0 and an integral formula for u (as a function of x_0). They then solved the integral equation asymptotically for $t \to 0$ and numerically for all t, computing u by the integral formula. Miller, Morton and Baines [4] used an iterative finite element technique to obtain an adaptive mesh solution. The results of these different approaches were all in good agreement with each other.

The method of moving finite elements (MFE) of Miller [5],[6] for this particular problem is attractive for two main reasons: firstly, for the obvious property of being able to use a moving node in order to locate the position of the moving boundary, and secondly because of its recent success in approximately solving diffusion problems in both one and two dimensions (see Johnson [7]).

Section 2 contains a description of the method, and in section 3 we discuss mathematical aspects of its application to the problem. Section 4 presents and analyses the numerical results obtained, from which conclusions are then drawn in section 5.

2. THE MFE METHOD

The problem is solved using the local moving finite element method [8], which arises from the standard moving finite element method [5], [6], [9]. We begin with a summary of the standard MFE method but without penalty functions.

2.1 The Standard MFE Method

For an equation of the form

$$u_t = L(u) \quad , \tag{2.1}$$

where L is a linear one-dimensional spatial differential operator, we seek a piecewise linear function v of the form

$$v = \sum_{j=1}^{N} a_j \alpha_j \quad , \tag{2.2}$$

where $a_j = a_j(t)$, $j=1(1)N$, are the nodal coefficients, $\alpha_j = \alpha_j(x,\underline{s})$, $j=1(1)N$, are linear basis functions of local compact support (as shown in Figure 1), and $\underline{s} = \underline{s}(t)$ is a time-dependent vector of nodal positions, s_j, $j=1(1)N$.

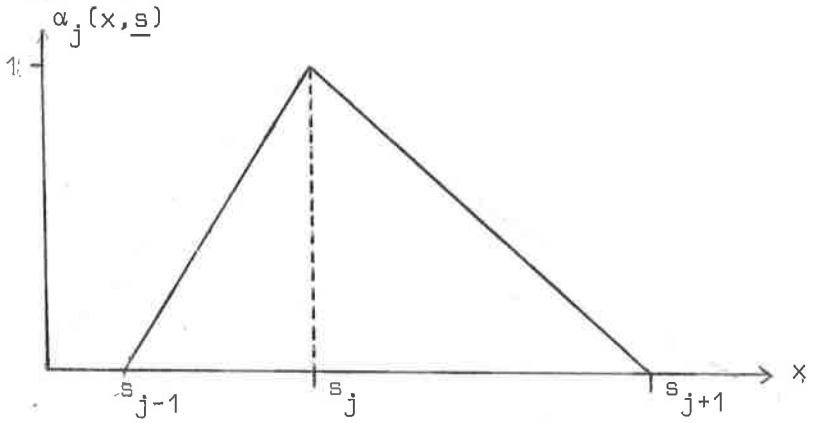

Figure 1

Differentiation with respect to time yields

$$v_t = \sum_{j=1}^{N} \{\dot{a}_j \alpha_j + \dot{s}_j \beta_j\} \quad , \tag{2.3}$$

where $\beta_j = \beta_j(t)$, $j=1(1)N$, are (in general) discontinuous piecewise linear functions (see references [5] and [10]).

$$\beta_j = -m\alpha_j, \qquad j=1(1)N \tag{2.4}$$

(where m is the local slope of v). Figure 2 displays this second type of basis function.

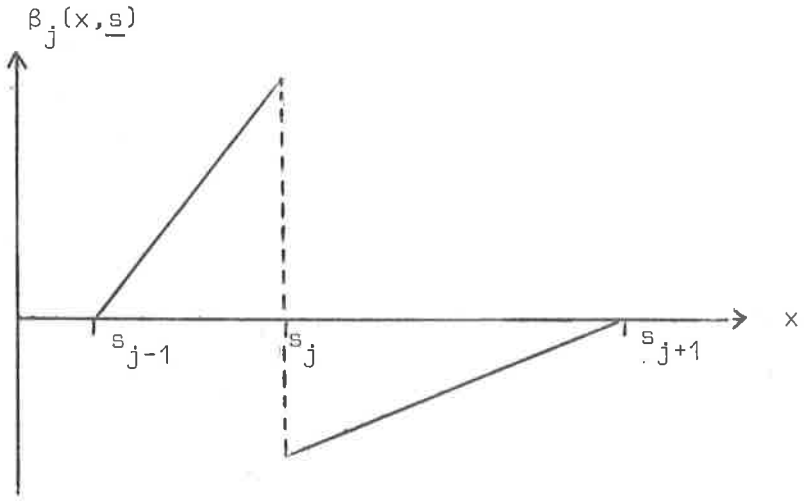

Figure 2

Minimising

$$\| v_t - L(v) \|_2 , \tag{2.5}$$

the global L_2 norm of the residual, over the variables a_j, s_j, $j=1(1)N$, leads to the extended system of Galerkin equations

$$\left.\begin{aligned} \langle \alpha_j, \; v_t - L(v) \rangle &= 0 \\ \langle \beta_j, \; v_t - L(v) \rangle &= 0 \end{aligned}\right\} \; j=1(1)N \; . \tag{2.6}$$

Substitution for v_t from (2.3) gives rise to a matrix system of ODE's which can be solved for the $\dot{a}_j$, $\dot{s}_j$, $j=1(1)N$. The nodal co-ordinates a_j, s_j, $j=1(1)N$, may then be calculated using a time-stepping scheme (see Section 3.5).

Two types of singularity which may occur (and which are also present in the local method) are described in section 2.3, together with their remedies.

2.2 The Local MFE Method

The local approach [8] expresses the v_t of (2.3), which is piecewise linear and discontinuous, in an element-wise form

$$v_t = \sum_{k=3/2}^{N-\frac{1}{2}} \{\dot{w}_{k1}\phi_{k1} + \dot{w}_{k2}\phi_{k2}\} \quad , \tag{2.7}$$

where ϕ_{k1} and ϕ_{k2}, $k = 3/2(1)N-\frac{1}{2}$, are the semi α-type element basis functions shown in figures 3(a) and (b) and $\dot{w}_{k1}$, $\dot{w}_{k2}$ are weighting velocities.

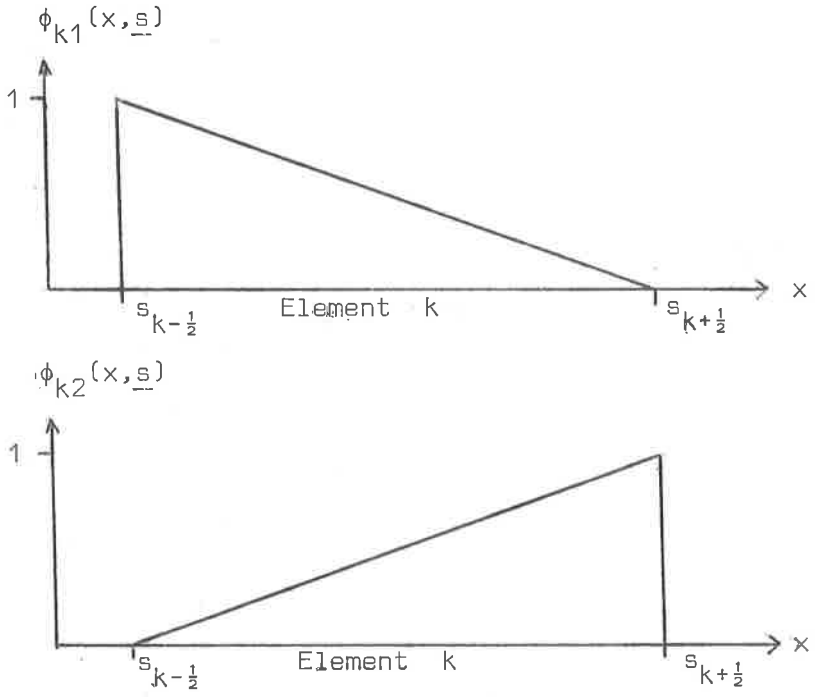

Figure 3

In the one-dimensional case the sets $\{\alpha_j,\beta_j\}_{j=1}^{N}$ and $\{\phi_{k1},\phi_{k2}\}_{k=3/2}^{N-\frac{1}{2}}$ span the same space and we may obtain expressions relating $\dot{a}_j$, $\dot{s}_j$, $j=1(1)N$, to $\dot{w}_{k1}$, $\dot{w}_{k2}$, $k = 3/2(1)N-\frac{1}{2}$ (see (2.16) below). By minimising the local element L_2-norm

$$\| v_t - L(v) \|_2 \tag{2.8}$$

over $\dot{w}_{k1}$, $\dot{w}_{k2}$, $k = 3/2(1)N-\frac{1}{2}$, we obtain the alternative elementwise Galerkin equations

$$\left.\begin{aligned} \langle \phi_{k1},\ v_t - L(v)\rangle &= 0 \\ \langle \phi_{k2},\ v_t - L(v)\rangle &= 0 \end{aligned}\right\} \quad k = {}^{3}\!/_{2}(1)N-\tfrac{1}{2} \quad . \tag{2.9}$$

Each pair of equations (2.9) yields a 2x2 matrix system of the form

$$C_k \dot{\underline{w}}_k = \underline{b}_k, \quad k = {}^{3}\!/_{2}(1)N-\tfrac{1}{2} \quad , \tag{2.10}$$

where

$$\dot{\underline{w}}_k = [\dot{w}_{k1},\ \dot{w}_{k2}]^T, \quad k = {}^{3}\!/_{2}(1)N-\tfrac{1}{2} \quad , \tag{2.11}$$

$$C_k = \begin{bmatrix} \langle \phi_{k1},\phi_{k1}\rangle & \langle \phi_{k1},\phi_{k2}\rangle \\ \langle \phi_{k2},\phi_{k1}\rangle & \langle \phi_{k2},\phi_{k2}\rangle \end{bmatrix}$$

$$= \frac{1}{6\Delta s_k} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad k = {}^{3}\!/_{2}(1)N-\tfrac{1}{2} \quad , \tag{2.12}$$

$$\Delta s_k = s_{k+1} - s_k \tag{2.13}$$

and

$$\underline{b}_k = [\langle \phi_{k1},\ L(v)\rangle\ \langle \phi_{k2}, L(v)\rangle]^T, \quad k = {}^{3}\!/_{2}(1)N-\tfrac{1}{2} \quad . \tag{2.14}$$

For each node the relationships

$$\left.\begin{aligned} \alpha_j &= \phi_{j-\frac{1}{2},2} + \phi_{j+\frac{1}{2},1} \\ \beta_j &= -m_{j-\frac{1}{2}}\phi_{j-\frac{1}{2},2} - m_{j+\frac{1}{2}}\phi_{j+\frac{1}{2},1} \end{aligned}\right\} \quad j=1(1)N \quad , \tag{2.15}$$

where node j borders elements $j-\frac{1}{2}$ and $j+\frac{1}{2}$, give rise to the simultaneous equations

$$\left.\begin{aligned} \dot{a}_j - m_{j-\frac{1}{2}}\dot{s}_j &= \dot{w}_{j-\frac{1}{2},2} \\ \dot{a}_j - m_{j+\frac{1}{2}}\dot{s}_j &= \dot{w}_{j+\frac{1}{2},1} \end{aligned}\right\} \quad j=1(1)N \quad . \tag{2.16}$$

These produce another set of 2x2 matrix systems of the form

$$M_j \dot{\underline{y}}_j = \dot{\underline{w}}_j \quad , \quad j=1(1)N \quad , \tag{2.17}$$

where

$$M_j = \begin{bmatrix} 1 & -m_{j-\frac{1}{2}} \\ 1 & -m_{j+\frac{1}{2}} \end{bmatrix} \quad , \quad j=1(1)N \, , \tag{2.18}$$

and

$$\dot{\underline{y}}_j = [\dot{a}_j, \dot{s}_j]^T, \quad j=1(1)N \quad . \tag{2.19}$$

As in the standard method, we may solve for the $\dot{a}_j, \dot{s}_j$, $j=1(1)N$, and obtain a_j, s_j, $j=1(1)N$, by a time-stepping integration method. The difference is that the local method only requires the inversion of 2x2 matrices.

2.3 Singularities

From equations (2.12) and (2.18) we see that the MFE method fails if either

$$\Delta s_j = 0 \tag{2.20}$$

or

$$m_{j-\frac{1}{2}} = m_{j+\frac{1}{2}} \tag{2.21}$$

for one or more values of j. Node overtaking, (2.20), may be prevented by limiting the step used in the time integration, and is therefore discussed later in the time-stepping section, 3.5.

In the case of the singularity due to (2.21), which is referred to as "parallelism" or "collinearity", there is no unique solution for the unknowns $\dot{a}_j$, $\dot{s}_j$. This is remedied by fixing the positions of such offending nodes, solving for their heights on a patch which includes

their neighbours and then adding a multiple of the null space, supplying them with a suitable weighted velocity (see reference [9]). For this, it is necessary to return to the α and β basis functions (see [11]). In the case of parallelism at a single node, j, we replace the β_j equation by

$$\dot{s}_j = 0 \quad , \tag{2.22}$$

from which we obtain

$$\frac{1}{6}\Delta s_{j-\frac{1}{2}}\dot{w}_{j-\frac{1}{2},1} + \frac{1}{3}\Delta s_{j-\frac{1}{2}}\dot{w}_{j-\frac{1}{2},2} + \frac{1}{3}\Delta s_{j+\frac{1}{2}}\dot{w}_{j+\frac{1}{2},1} + \frac{1}{6}\Delta s_{j+\frac{1}{2}}\dot{w}_{j+\frac{1}{2},2}$$

$$= b_{j-\frac{1}{2},2} + b_{j+\frac{1}{2},1} \quad . \tag{2.23}$$

(2.22) yields

$$\dot{w}_{j-\frac{1}{2},2} = \dot{a}_j = \dot{w}_{j+\frac{1}{2},1} \quad , \tag{2.24}$$

from which we may obtain (via (2.23)),

$$\dot{a}_j = \frac{b_{j-\frac{1}{2},2} + b_{j+\frac{1}{2},1} - \frac{1}{6}(\Delta s_{j-\frac{1}{2}}\dot{w}_{j-\frac{1}{2},1} + \Delta s_{j+\frac{1}{2}}\dot{w}_{j+\frac{1}{2},2})}{\frac{1}{3}(\Delta s_{j-\frac{1}{2}} + \Delta s_{j+\frac{1}{2}})} \quad . \tag{2.25}$$

The null space vector for $[\dot{a}_j, \dot{s}_j]^T$ is

$$[m,1]^T \tag{2.26}$$

(where $m_{j-\frac{1}{2}} = m = m_{j+\frac{1}{2}}$), a multiple of which may be added in order to ensure that this node moves along between its direct neighbours.

When more than one consecutive parallel node is present, we have to solve a tri-diagonal system of linear equations for the unknown amplitudes. Parallelism, however, very seldom occurs in problems with non-parallel initial data.

3. APPLICATION OF MFE TO THE PROBLEM

In solving the Crank-Gupta moving boundary problem, five important factors had to be considered: evaluation of the inner products, initial node placement, initial data representation, treatment of the moving boundary node and automatic time-stepping. These areas are discussed in the ensuing sections.

3.1 Determination of the Local Inner Products

The form of $L(u)$ in the Crank-Gupta problem is

$$L(u) = u_{xx} - 1 \tag{3.1}$$

(see (1.1)), which contains second order spatial derivatives. The approximating function, v, however, is only piecewise linear and so v_{xx} has the character of a delta function. The local right-hand side vectors, $\underline{b}_k$, $k = \frac{3}{2}(1)N-\frac{1}{2}$, (see (2.14)), must therefore be determined in some suitable manner. The use of integration by parts and an average element slope at the nodes is a possible approach (see references [10] and [12]).

From (2.14), we obtain for $\underline{b}_k = [b_{k1}, b_{k2}]^T$, $k = \frac{5}{2}(1)N-\frac{3}{2}$, (in a non-boundary element)

$$b_{k1} = \langle \phi_{k1}, L(v) \rangle = \langle \phi_{k1}, v_{xx} - 1 \rangle \tag{3.2}$$

$$= \int_{s_{k1}}^{s_{k2}} \phi_{k1} v_{xx} dx - \int_{s_{k1}}^{s_{k2}} \phi_{k1} dx$$

$$= [\phi_{k1} v_x]_{s_{k1}}^{s_{k2}} - \int_{s_{k1}}^{s_{k2}} \phi_{k1,x} v_x dx - \int_{s_{k1}}^{s_{k2}} \phi_{k1} dx$$

$$= - v_x(s_{k1}) - \int_{s_{k1}}^{s_{k2}} \left(- \frac{1}{\Delta s_k} \right) m_k dx - \int_{s_{k1}}^{s_{k2}} \phi_{k1} dx$$

$$= -\tfrac{1}{2}(m_{k-1} + m_k) + m_k - \tfrac{1}{2}\Delta s_k$$

$$= \tfrac{1}{2}(m_k - m_{k-1} - \Delta s_k), \quad k = \frac{5}{2}(1)N-\tfrac{3}{2}, \tag{3.3}(a)$$

and similarly

$$b_{k2} = \tfrac{1}{2}(m_{k+1} - m_k - \Delta s_k),\ k = \tfrac{3}{2}(1)N-\tfrac{1}{2} \quad . \tag{3.3)(b}$$

For boundary nodes, the conditions $u_x = 0$ are implemented via the corresponding weak form of the differential equation, (1.1), in the first and last elements, to yield

$$\left.\begin{aligned} b_{\frac{3}{2},1} &= m_{\frac{3}{2}} - \tfrac{1}{2}\,\Delta s_{\frac{3}{2}} \\ b_{N-\frac{1}{2},2} &= -m_{N-\frac{1}{2}} - \tfrac{1}{2}\Delta s_{N-\frac{1}{2}} \end{aligned}\right\} \quad . \tag{3.4}$$

From these values we can obtain the coefficients $\dot{w}_{k1}$, $\dot{w}_{k2}$, $k = \tfrac{3}{2}(1)N-\tfrac{1}{2}$, and hence $\dot{a}_j$, $\dot{s}_j$, $j = 1(1)N$, by pairing them off nodewise (see (2.16)).

3.2 Initial Node placement

Owing to the difficulty of obtaining satisfactory solutions, it was decided not to evaluate initial nodal and amplitudal values simultaneously. Consider therefore initial nodal placements alone.

Early experimentation with nodal placement revealed that for the initial quadratic data

$$u(x,0) = \tfrac{1}{2}(1-x)^2, \quad 0 \leq x \leq 1 \quad , \tag{3.5}$$

several nodes were required close to boundary mesh points in order to resolve the curvature of the solution near these points. A cubic distribution function was found to serve well, the explicit form of which being

$$-2x^3 + 3x^2 \quad . \tag{3.6}$$

3.3 Initial Data Representation

Having decided on the initial positions of the nodes, the next task was to determine an adequate initial representation of the initial condition using piecewise linears.

The simplest approach considered was merely to sample point values of the initial function at the nodes. This, however, was to require a special treatment of the moving boundary node (see Section 3.4).

A second approach was one which was designed to ensure that the total initial oxygen concentration was equal to the true value, but the solution was found to be under-determined by one equation; this vacancy was to be supplied using the same boundary treatment as in the previous case.

A third idea, one which required no special boundary treatment, was one of fitting a linear spline to the data. Normal equations, which formed a tri-diagonal system, were constructed by minimising (over the amplitudes a_j, j=1(1)N) the difference between the initial quadratic and the piecewise linear function, v, in the L_2 norm.

The first approach was finally chosen because of its simplicity and the slightly better agreement of the final results with the results in [3].

3.4 Special Boundary Treatment

The problem of treating the moving boundary caused the most difficulty. This arose from the fact that we are trying to represent a locally quadratic function by a piecewise linear approximation.

Assuming coincidence of the boundary with the final node, the equations for the velocity of the moving boundary node and its amplitude are

$$\left.\begin{aligned} \dot{a}_N - m_{N-\frac{1}{2}}\,\dot{s}_N &= \dot{w}_{N-\frac{1}{2},2} \\ \dot{a}_N &= 0 \end{aligned}\right\} \quad . \tag{3.7}$$

However, this approach did not produce good results.

The eventual approach employed was one similar to that used by Miller, Morton and Baines [4]. Their method exploited the local least squares fit

$$y(x) = -\lambda h(x - x_0 + h) + \frac{5}{6}\lambda h^2 \tag{3.8}$$

to the asymptotic solution

$$v(x,t) = \lambda(x_0 - x)^2, \qquad \lambda \in \mathbb{R} \quad , \tag{3.9}$$

adjacent to the moving boundary. When $y = 0$ we obtain

$$x = x_0 - \frac{1}{6}h \quad . \tag{3.10}$$

From Figure 4 we see that the relationship

$$a_N = -\frac{1}{5}a_{N-1} \tag{3.11}$$

between the nodal values at s_{N-1} and s_N holds, and this was used to model the $u = 0$ boudnary condition. The approach (see Figure 5)

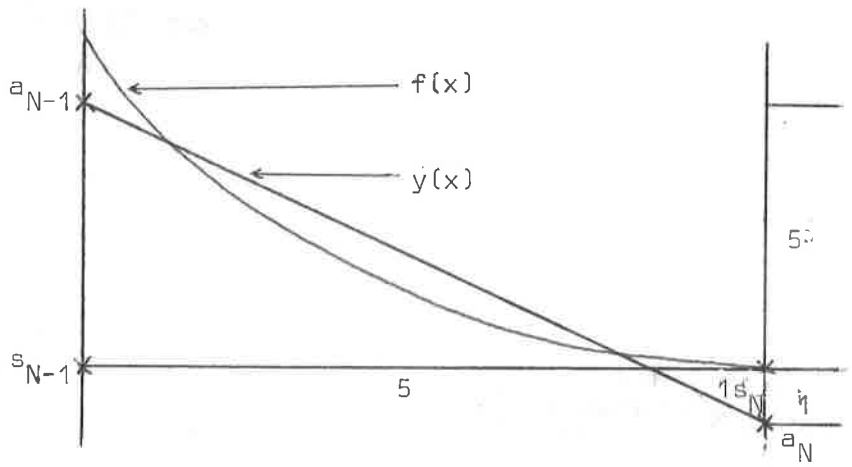

Figure 4

always locates the last node to be in such a position that the (projected) boundary with the 5:1 ratio has zero oxygen concentration,

$$u(x_0,t) = 0 , \qquad t \geq 0 \quad . \tag{3.12}$$

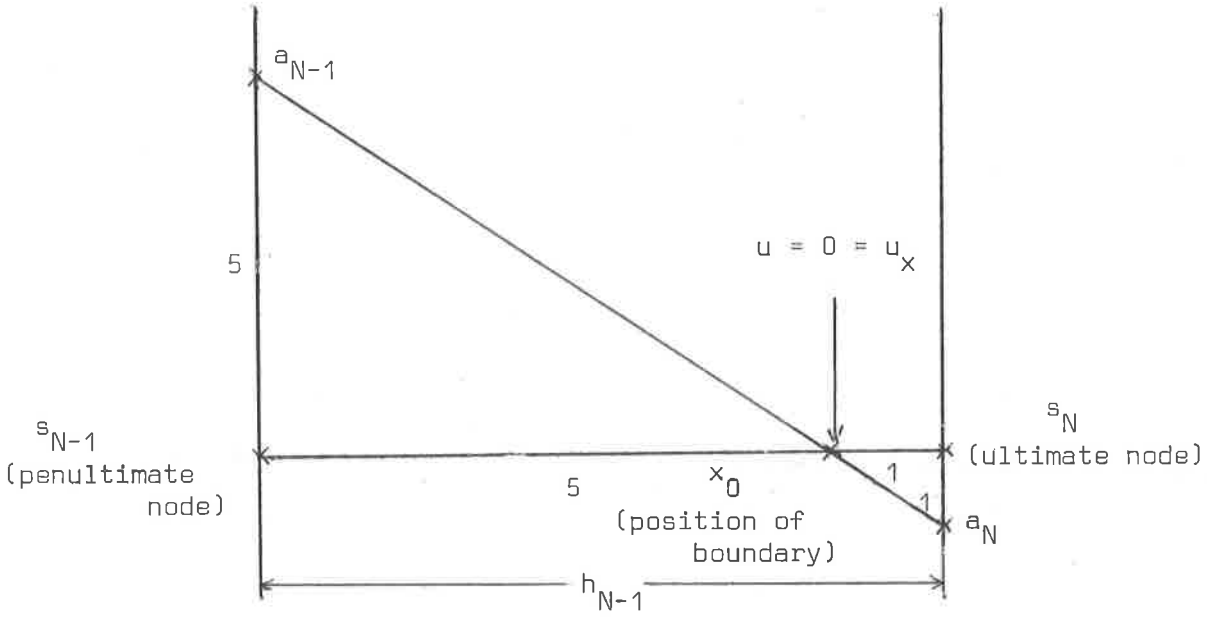

Figure 5

Several other approaches, including some variations on the above theme were attempted, in particular, one of replacing the last element slope, $m_{N-\frac{1}{2}}$, by the slope of the local line of best fit, $-\frac{1}{2}h_{N-\frac{1}{2}}$.

Other approaches included replacing the $\dot{a}_N = 0$ equation by an $\dot{a}_N \to 0$ one. For instance

$$\left.\begin{aligned} \dot{a}_N &= \theta \dot{a}_{N-1} , & 0 < \theta \leq 1 \\ \dot{a}_N &= \theta \dot{a}_{N-1} a_N / a_{N-1} , & 0 < \theta \leq 1 \end{aligned}\right\} \quad . \tag{3.13}$$

Another approach, requiring that $u \to 0$, $\dot{u} \to 0$ was derived as follows. Expanding a_N and $\dot{a}_N$ yields

$$\left.\begin{aligned} 0 = a_N(t+\Delta t) &\simeq a_N(t) + \Delta t\dot{a}_N(t) + \tfrac{1}{2}K(\Delta t)^2\ddot{a}_N(t) \\ 0 = \dot{a}_N(t+\Delta t) &\simeq \dot{a}_N(t) + \Delta t\,\ddot{a}_N(t) \end{aligned}\right\} \quad ,(3.14)$$

where K is either 0 or 1 (depending on whether consistency to order Δt or $\ddot{a}_N$ is respectively required). From (3.14)(b) we obtain, suppressing the t-dependence,

$$\Delta t \simeq -\dot{a}_N/\ddot{a}_N \quad , \tag{3.15}$$

which we then substitute into (3.14)(a), yielding

$$\dot{a}_N^2 \simeq a_N\,\ddot{a}_N/(1-\tfrac{1}{2}K) \quad , \tag{3.16}$$

Using the relationship

$$\ddot{a}_N \simeq \dot{a}_N\left(\frac{\Delta\dot{a}_N}{\Delta\dot{s}_N}\right) = \frac{\dot{a}_N(\dot{a}_N-\dot{a}_{N-1})}{(s_N-s_{N-1})} \quad , \tag{3.17}$$

(3.16) becomes

$$\dot{a}_N \simeq a_N\dot{a}_{N-1}/[a_N - (1-\tfrac{1}{2}K)(s_N-s_{N-1})] \quad , \tag{3.18}$$

and this is used to obtain an approximation to $\dot{a}_N$.

Equation (3.7)(b) was also replaced by one for $\dot{s}_N$, which is feasible since $\dot{a}_N$ occurs in (3.7)(a). An early idea was merely to differentiate the last equation in the initial placement of the mesh to arrive at one involving nodal velocities. Another exploited the locally quadratic behaviour in the last element to produce

$$\dot{s}_N = \dot{a}_{N-1}/h_{N-1} + \dot{s}_{N-1} \quad , \tag{3.19}$$

using a total derivative chain formula. The asymptotic expression in [3] was also considered.

Other approaches relied on replacing the $\dot{w}_{N-\frac{1}{2},1}$ and $\dot{w}_{N-\frac{1}{2},2}$ equations of the last element by ones obtained using the locally quadratic form of u (since these weighting velocities were of a larger order of magnitude than all others). It was, however, discovered that the further we deviated from the simplest ideas, the less accurate the results became!

3.5 Time Stepping

The time-stepping integration of the nodal and amplitudal variables a_j, s_j, $j=1(1)N$, is obtained via the first order Euler formula to yield

$$\left.\begin{aligned} a_j^{n+1} &= a_j^n + \Delta t\, \dot{a}_j^n \\ s_j^{n+1} &= s_j^n + \Delta t\, \dot{s}_j^n \end{aligned}\right\} \quad j=1(1)N\ , \tag{3.20}$$

where Δt is the time increment at the n superscripted time level, which precedes the new, $n+1$, one (see [9]).

The time increment at a particular time-step is automatically selected to ensure that neither node overtaking nor more than a specified relative change in local element slopes occurs, this amount being chosen to maintain smooth time-stepping.

Node overtaking (and hence the occurrence of a singularity due to a zero element length) is prevented by using a fixed time increment, but decreasing it if node overtaking would have occurred in a shorter time; this reduced increment is taken to be half of the smallest overtaking time of all offending nodes.

In order to restrict the relative change in element slopes by a specified percentage, p, say, we proceed as follows [13]. Approximate the change in each particular slope using a first order

expansion and then equate this to the desired percentage change in the element slope. A rearrangement then yields the appropriate time increment. Thus, let m_k^n be the slope of element k at time t, and m_k^{n+1} at a later time. Then

$$m_k^{n+1} - m_k^n \simeq \dot{m}_k^n \Delta t \quad . \tag{3.21}$$

Dropping the superscript n gives

$$\dot{m}_k \Delta t \leqq \frac{1}{100} pm_k \quad , \tag{3.22}$$

so that

$$\Delta t \leqq \frac{pm_k}{100\dot{m}_k} \quad , \tag{3.23}$$

where

$$m_k = \frac{\Delta a_k}{\Delta s_k} = \frac{a_{k+1} - a_k}{s_{k+1} - s_k} \quad . \tag{3.24}$$

The time increment is then taken as

$$\Delta t = \frac{pm_k}{100\dot{m}_k} \quad . \tag{3.25}$$

The time increment at a particular time-step is taken as the minimum of the two values described in the last two paragraphs.

4. NUMERICAL RESULTS

Early experimentation suggested the use of 20 intervals (21 nodes). In order that the results could be compared with those obtained by other workers (in particular [3]), solutions at every 0.01 seconds, up to a final time of 0.19 seconds, were output, 0.1974 seconds being the approximate time at which the oxygen concentration reduces to zero everywhere. The relative percentage change in element slopes, p, was taken to be unity.

Table 1 shows the position of the MFE moving boundary (and its deviations from the results of [3]) at 0.01 second intervals from 0 to 0.19 seconds, and in Table 2 we give the variation of the oxygen concentration with respect to time at the fixed boundary. Finally, Table 3 presents the amount of oxygen remaining at our chosen times. Graphical illustrations of the moving boundary solution (with the position of the moving boundary at each time indicated by the symbol "$\hat{\lambda}$") and the amount of oxygen remaining can be seen in Figures 6 and 7, respectively.

Time (seconds)	MFE Boundary Position	Differences ($\times 10^3$)
0.00	1.00000	0.00
0.01	0.99409	5.91
0.02	0.99409	5.91
0.03	0.99508	-
0.04	0.99539	3.79
0.05	0.99391	2.88
0.06	0.98970	2.10
0.07	0.98204	-
0.08	0.97076	0.79
0.09	0.95520	-
0.10	0.93508	-0.07
0.11	0.91015	-
0.12	0.87956	-0.40
0.13	0.84304	-
0.14	0.79977	-0.86
0.15	0.74711	-0.43
0.16	0.68360	-0.23
0.17	0.60532	-
0.18	0.50062	0.47
0.19	0.34434	1.03

TABLE 1

Time (seconds)	Concentration at Fixed Boundary (and the results of [3], where possible)	
0.00	0.50000	
0.01	0.38878	
0.02	0.34203	
0.03	0.30604	
0.04	0.27541	
0.05	0.24869	(0.24769)
0.06	0.22453	
0.07	0.20231	
0.08	0.18172	
0.09	0.16233	
0.10	0.14390	(0.14318)
0.11	0.12645	
0.12	0.10962	
0.13	0.09361	
0.14	0.07830	
0.15	0.06336	(0.06308)
0.16	0.04903	
0.17	0.03531	
0.18	0.02183	
0.19	0.00899	(0.00902)

TABLE 2

Time (seconds)	Amount of Oxygen Remaining
0.00	0.16699
0.01	0.15706
0.02	0.14706
0.03	0.13703
0.04	0.12691
0.05	0.11691
0.06	0.10694
0.07	0.09704
0.08	0.08729
0.09	0.07765
0.10	0.06815
0.11	0.05891
0.12	0.04987
0.13	0.04124
0.14	0.03306
0.15	0.02525
0.16	0.01807
0.17	0.01165
0.18	0.00604
0.19	0.00174

TABLE 3

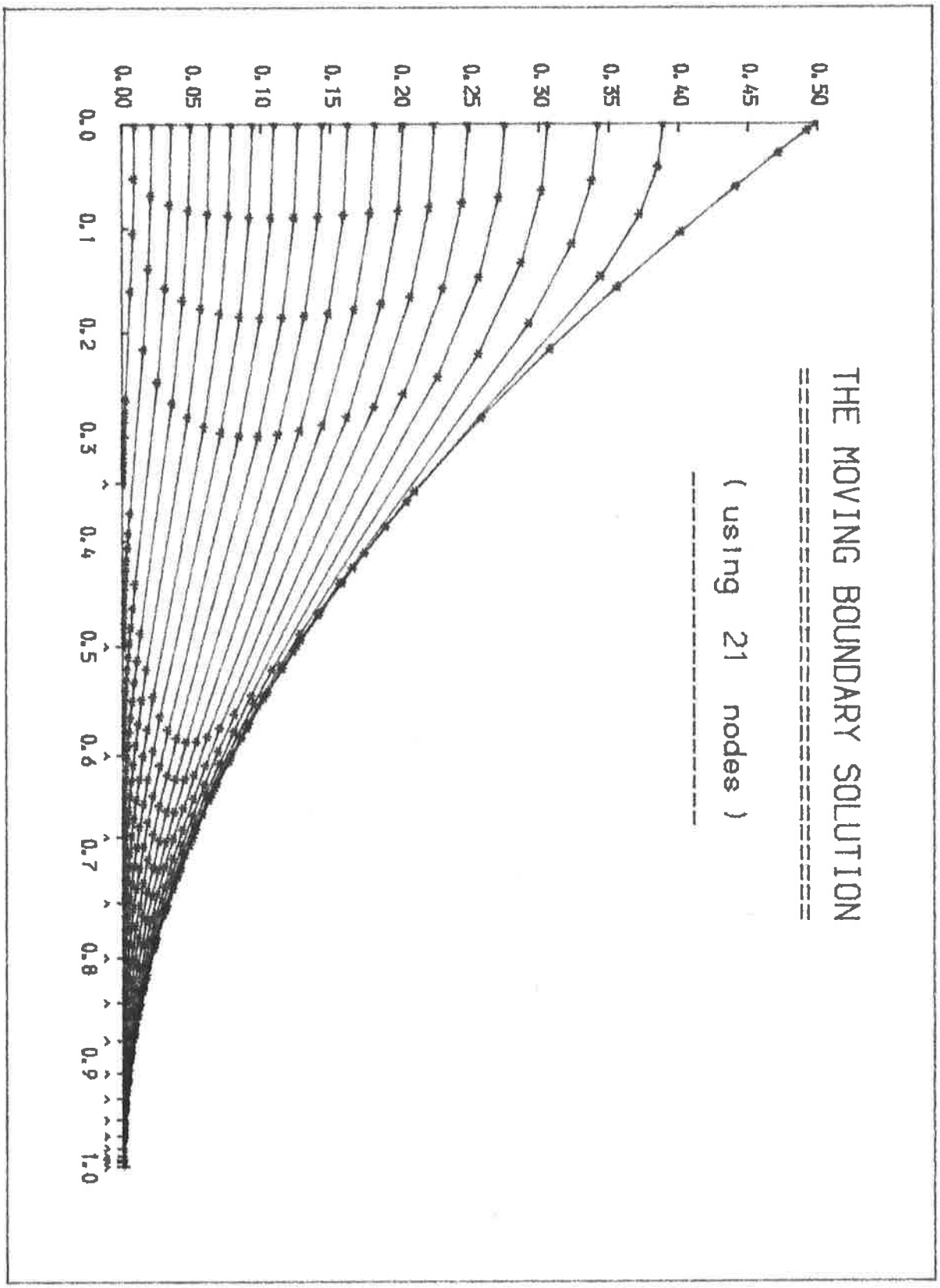
THE MOVING BOUNDARY SOLUTION
(using 21 nodes)
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0

FIGURE 6

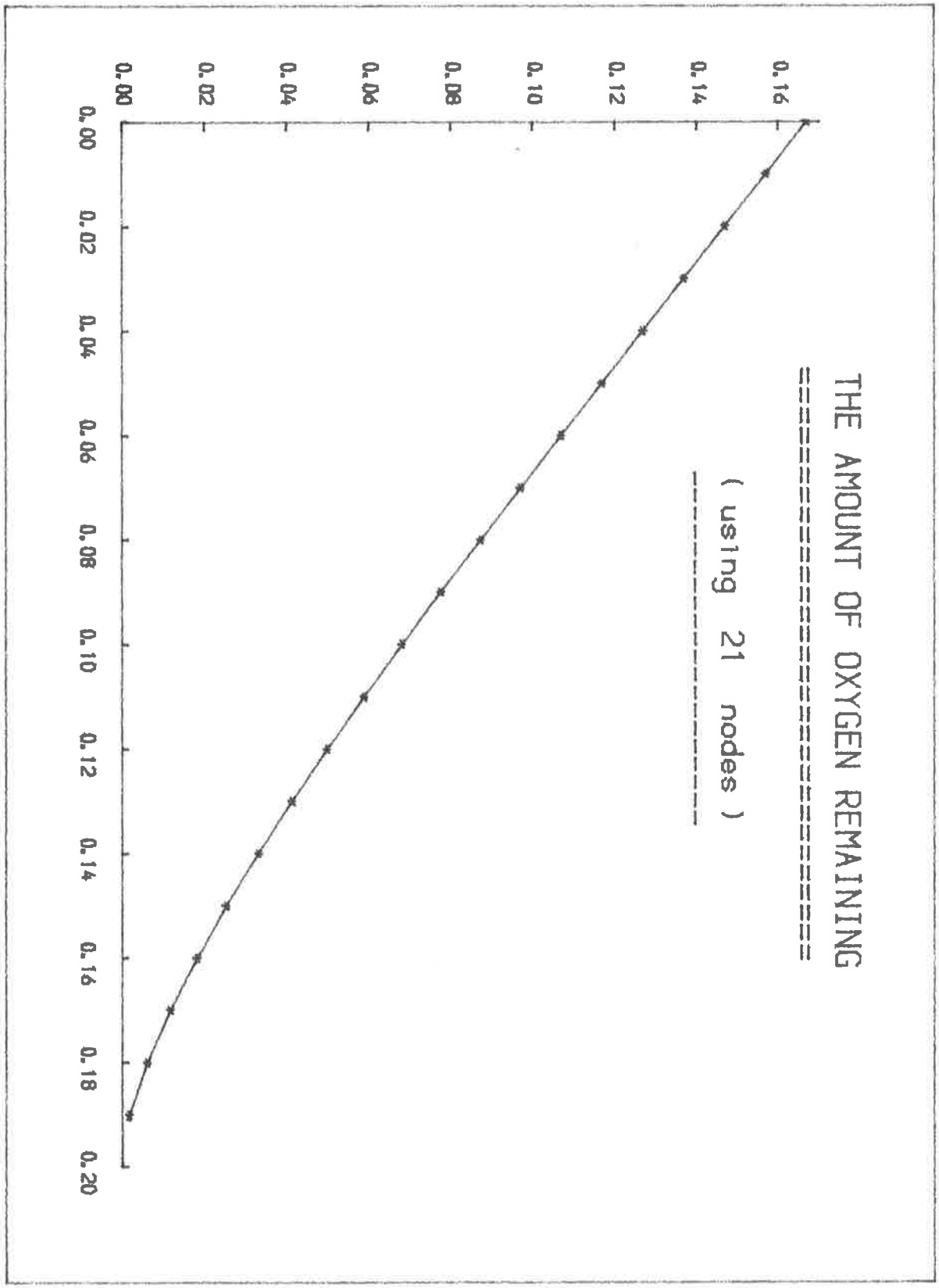
THE AMOUNT OF OXYGEN REMAINING
(using 21 nodes)
0.00
0.02
0.04
0.06
0.08
0.10
0.12
0.14
0.16
0.18
0.20

FIGURE 7

Referring to Table 1 we see that the free boundary exhibits a slight initial inward movement, then gradually retreats until about $t = 0.04$ seconds, after which the motion is towards the fixed boundary at an increasing rate. Figure 6 illustrates clearly the reaction of the nodes to this boundary motion, firstly moving to the right and then returning to the left, tracing out a concave path. The maximum time-step of 1.0×10^{-4} is greatly reduced during the early and latter stages of the absorption/diffusion process, respectively due to element slope changes and the nearness of nodes.

Table 2 demonstrates the initial rapid reduction of the oxygen concentration at the fixed boundary and the ensuing steady decrease towards zero. The first stage is due to the effects of the sink and left-hand Neumann condition and the second to the diffusive nature of the problem.

In Table 3 we show the almost uniform decay of the total oxygen content from the initial (true) value of $\frac{1}{6}$. Figure 7 exhibits the linear-like behaviour until 0.08 seconds (approximately) which is followed by a gentle curve to the final value of 0.19 seconds. Applying a linear fit to the last two points and extrapolating to the time axis yields the value of 0.1941 seconds as the time at which no oxygen remains. (A projected value of 0.1953 is obtained when matching a quadratic through the final 3 points). These values are slightly lower than the figures given in [3].

The CPU time used on a Nord 500 mini computer was approximately 38 seconds with 21 nodes, compared to 9 using 11 nodes and 234 with 41.

5. CONCLUSIONS

We have used the Moving Finite Element method to model the Crank-Gupta moving boundary problem and have succeeded in producing a relatively cheap and accurate solution.

In future work we hope to model the corresponding 2-dimensional problem. We also want to experiment with the possibility of merging nodes.

ACKNOWLEDGEMENTS

I would like to thank Dr. M.J. Baines for his supervision of this project and his helpful suggestions. The idea of the time-stepping strategy and continuous interest of Dr. C.P. Please is also acknowledged.

This work was carried out while the author was in receipt of an SERC CASE award in association with GEC.

REFERENCES

1. CRANK, J. & GUPTA, R. (1972a) J. Inst. Maths. Applics. 10, 296-304.

2. CRANK, J. & GUPTA, R. (1972b) J. Inst. Maths. Applics. 10, 19-33.

3. HANSEN, E. & HOUGAARD, P. (1974) J. Inst. Maths. Applics., 13, 385-398.

4. MILLER, J.V., MORTON, K.W., & BAINES, M.J. (1978) J. Inst. Maths. Applics. 22, 467-477.

5. MILLER, K. & MILLER, R.N. (1981) "Moving Finite Elements, Part I", SIAM J. Numer. Anal. 18, 1019-1032.

6. MILLER, K. (1981) "Moving Finite Elements, Part II", SIAM J. Numer. Anal 18, 1033-1057.

7. JOHNSON, I.W. (1985) "Moving Finite Elements for Nonlinear Diffusion Problems in One and Two Dimensions", Numerical Analysis Report 12/85, Dept. of Mathematics, University of Reading.

8. BAINES, M.J. (1985) "Local Moving Finite Elements", Numerical Analysis Report 9/85, Dept. of Mathematics, University of Reading,

9. WATHEN, A.J. & BAINES, M.J. (1983) "On the Structure of the Moving Finite-element Equations", IMA J. Numer. Anal. 5, 161-182.

10. LYNCH, D.R. (1982) "Unified Approach to Simulation on Deforming Elements with Application to Phase Change Problems", J. Comput. Phys. 47, 387-411.

11. BAINES, M.J. (1985) "On Approximate Solutions of Time-dependent Partial Differential Equations by the Moving Finite Element Method", Numerical Analysis Report 1/85, Dept. of Mathematics, University of Reading.

12. JOHNSON, I.W. (1984) "The Moving Finite Element Method for the Viscous Burgers' Equation", Numerical Analysis Report 3/84, Dept. of Mathematics, University of Reading.

13. PLEASE, C.P. (1985) Private communication.