ON APPROXIMATE SOLUTIONS OF TIME-DEPENDENT PARTIAL DIFFERENTIAL EQUATIONS BY THE MOVING FINITE ELEMENT METHOD

M.J. BAINES

NUMERICAL ANALYSIS REPORT 1/85

The work reported here forms part of the research programme
of the Oxford/Reading Institute for Computational Fluid Dynamics

CONTENTS

1. INTRODUCTION

Recent work on the Moving Finite Element (MFE) method at Reading (Wathen (1982), (1984), Wathen & Baines (1983), Wathen, Baines & Morton (1984), Johnson (1984)), has shown that, contrary to earlier impressions, the method is practical and effective, particularly for hyperbolic problems, and possesses some unexpected properties. The original idea of Miller (Miller & Miller (1980), Miller (1981)) has been developed by the above workers without recourse to penalty functions and good results have been obtained for scalar hyperbolic conservation laws with shocks and for scalar convection diffusion problems. Here we review and discuss further the application of the MFE method to evolutionary partial differential equations of the form

$$u_t = L(u) \quad , \tag{1.1}$$

where $L(u)$ is an operator involving only spatial derivatives.

A summary of the MFE method is as follows. The object function u is represented by a piecewise linear continuous spline v which can be written as a linear sum of time dependent linear basis functions α , ones which take the value 1 at a (moving) node and zero at all other nodes. The time derivative of this function v_t is a linear sum of the α functions and related $\underline{\beta}$ functions where $\underline{\beta} = -(\underline{\nabla}v)\alpha$ (see Section 2 for more details). The coefficients of α and $\underline{\beta}$ in v_t are the time derivatives of the nodal heights a and nodal positions $\underline{s}$ denoted by $\dot{a}$ and $\dot{\underline{s}}$ respectively.

The L_2 residual

$$||v_t - L(v)||_2 \tag{1.2}$$

is then minimised over the $\dot{a}$ and $\dot{\underline{s}}$, giving the MFE equations

$$A(\underline{y})\dot{\underline{y}} = \underline{g}(\underline{y}) \quad , \tag{1.3}$$

where $\underline{y}$ is a vector of a's and $\underline{s}$'s, $A(\underline{y})$ is a symmetric positive definite matrix with block elements of the form

$$\begin{bmatrix} <\alpha,\alpha> & <\alpha,\underline{\beta}> \\ <\underline{\beta}^T,\alpha> & <\underline{\beta}^T,\underline{\beta}> \end{bmatrix} \tag{1.4}$$

and

$$\underline{g}(\underline{y}) = \begin{bmatrix} <\alpha,L(v)> \\ <\underline{\beta},L(v)> \end{bmatrix} . \tag{1.5}$$

The MFE equations (1.3) are a set of ordinary differential equations for $\underline{y}$ and hence a and $\underline{s}$. The variation of the function v (and the mesh) with time is obtained by numerical integration of these equations.

If D is the matrix consisting only of the diagonal blocks of A, it has been shown that $D^{-1}A$ is exceptionally well conditioned. Moreover A may be decomposed into the form

$$A = N^TQ^TCQN \tag{1.6}$$

where C is symmetric positive definite block diagonal, Q is a permutation matrix (unnecessary in the one dimensional case) and N is rectangular block diagonal.

The method was originally used to solve parabolic equations with the use of additional penalty functions to deal with possible singularities of $A(\underline{y})$. It has also been used without penalty functions to solve hyperbolic problems with shocks and diffusion problems with steep fronts.

The pattern of the report is as follows. In Section 2 we discuss in detail the form of basis functions and their time derivatives. Then in Sections 3-5 we discuss exact and approximate solutions of scalar partial differential equations of the form (1.1) for certain special $L(u)$, concentrating first on general results and then specialising to one dimension. In Section 6 time stepping considerations are discussed including shock modelling. Section 7 is concerned with boundary conditions while Sections 8 and 9 cover the extensions of the method to systems and to multi-dimensions, respectively.

Finally, in Section 10, some pointers as to the likely success and limitations of the model are presented.

I should like to acknowledge useful discussions with several people at Reading University, particularly Andy Wathen.

2. BASIS FUNCTIONS

The essential additional feature of the MFE representation is the inclusion of mesh variation with time in the usual Galerkin finite element approach.

We shall write the finite element approximation in the form

$$v = \sum_j a_j \alpha_j \tag{2.1}$$

where a_j is the coefficient of the basis function α_j. In the case of fixed finite elements (FFE) α_j is a spatial function depending (in a passive way) on the position of fixed nodes while a_j may depend on time t. This dependence is expressed by writing

$$v = \sum_j a_j(t)\alpha_j(\underline{r},\underline{s}) \tag{2.2}$$

where the position vector $\underline{r}$ gives the spatial variation and the vector $\underline{s}$ gives the (passive) dependence on the nodal co-ordinates s_j.

The extension to moving finite elements is effected by allowing $\underline{s}$ to depend on t, viz.

$$v = \sum_j a_j(t)\alpha_j(\underline{r},\underline{s}(t)) \quad . \tag{2.3}$$

In order to study the solution of evolutionary differential equations it is necessary to differentiate (2.3) with respect to time. Since t appears twice in (2.3) we obtain

$$\frac{\partial v}{\partial t} = \sum_j \ [\dot{a}_j(t)\alpha_j(\underline{r},\underline{s}(t)) + a_j(t)\dot{\alpha}_j(\underline{r},\underline{s}(t))], \tag{2.4}$$

where the dot denotes differentiation with respect to time. Now, by the chain rule,

$$\dot{\alpha}_i(\underline{r},\underline{s}(t)) = \sum_j \dot{s}_j(t)\frac{\partial \alpha_i}{\partial s_j}(r,s) = \underline{s}(t) \cdot \underline{\nabla}_{\underline{s}_j}\alpha_i(\underline{r},\underline{s}) \tag{2.5}$$

so that (2.4) becomes (dropping dependent variables)

$$\frac{\partial v}{\partial t} = \sum_j \dot{a}_j \alpha_j + \sum_i a_i \sum_j \dot{s}_j \frac{\partial \alpha_i}{\partial s_j} \quad . \tag{2.6}$$

By interchanging the order of summation and writing

$$\underline{\beta}_j = \sum_i a_i \underline{\nabla}_{s_j} \alpha_i \tag{2.7}$$

we obtain

$$\frac{\partial v}{\partial t} = \sum_j [\dot{a}_j \alpha_j + \underline{\dot{s}}_j \cdot \underline{\beta}_j] \quad , \tag{2.8}$$

where $\underline{\beta}_j$ is a second type basis function dependent not only on $\underline{r}$ and $\underline{s}$ but also on $\underline{a}$, the vector of nodal coefficients. From (2.7) and (2.3) we can write

$$\underline{\beta}_j = \underline{\nabla}_{s_j} v \quad , \quad \beta_j = \frac{\partial v}{\partial s_j} \quad . \tag{2.9}$$

We illustrate the form of $\underline{\beta}_j$ in the case of piecewise linear basis functions α_j which take the value 1 at node j and zero at all other nodes. In any element adjacent to node j (with a co-ordinate s_j) we can write the linear function v as

$$v = a_k + m_{jk} (\sigma_j - s_k) \tag{2.10}$$

where σ_j is a co-ordinate in the s_j direction and a_k is the value of v at an arbitrary point s_k in the element. The slope m_{jk} is given by

$$m_{jk} = \frac{\partial v}{\partial \sigma_j} = \frac{a_j - a_k}{s_j - s_k} \tag{2.11}$$

where a_j is the coefficient associated with the basis function.

Then

$$\alpha_j = \frac{\sigma_j - s_k}{s_j - s_k} \tag{2.12}$$

taking the value 1 at $\sigma_j = s_j$, and

$$\underline{\beta}_j = \underline{\nabla}_{s_j} v = \underline{\nabla}_{s_j} \left[a_k + \left(\frac{a_j - a_k}{s_j - s_k} \right) (\sigma_j - s_k) \right]$$

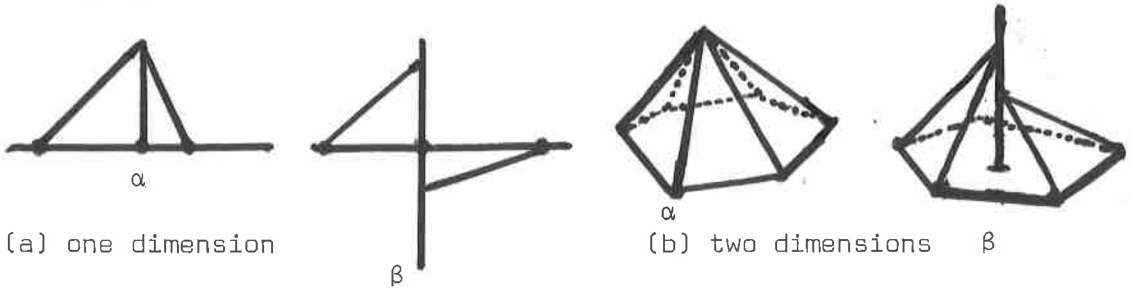
$$= - \frac{(a_j - a_k)}{(s_j - s_k)^2} (\sigma_j - s_k) \hat{\underline{s}}_j \qquad (2.13)$$

where $\hat{\underline{s}}_j$ is a unit vector in the direction of increasing σ_j. Finally, using (2.11) and (2.12), we obtain

$$\underline{\beta}_j = - m_{jk} \alpha_j \hat{\underline{s}}_j = - \left(\underline{\nabla}_{\sigma_j} v \right) \alpha_j \qquad (2.14)$$

in that element. [See also Lynch (1981)].

Thus in this case the second type basis function $\underline{\beta}_j$ has components which are multiples of α_j and have the same support as α_j. The result is true for linear basis functions in any number of dimensions. Diagrams illustrating α's and β's in one and two dimensions are shown in Fig. 2.1.



(a) one dimension

(b) two dimensions

FIG. 2.1 : Basis functions α and β

Consider now a single element k. For each node i which is a vertex of the element, let ϕ_{ki} be the linear function which takes the value 1 at node i and zero at the other vertices. Then the approximation

$$v = \sum_k \sum_i a_{ki} \phi_{ki} \qquad (2.15)$$

is linear in each element but discontinuous from element to element in general, unless the a_{ki}'s are constrained. Then, using (2.8),

$$v_t = \sum_k \sum_i [\dot{a}_{ki} \phi_{ki} + \dot{\underline{s}}_{ki} \cdot \underline{\psi}_{ki}] \qquad (2.16)$$

where, from (2.14),

$$\underline{\psi}_{ki} = - m_{ki}\phi_{ki}\,\hat{\underline{s}}_i = - \left(\underline{\nabla}_{s_{ki}} v\right) \phi_{ki} \qquad (2.17)$$

since ϕ_{ki} is linear in the element k and takes the value 1 at node i. Thus

$$v_t = \sum_k \sum_i [\dot{a}_{ki} - \dot{\underline{s}}_{ki} \cdot \underline{\nabla}_{s_{ki}} v]\, \phi_{ki} \qquad (2.18)$$

$$= \sum_k \sum_i w_{ki}\phi_{ki} \quad , \qquad (2.19)$$

where * $$w_{ki} = \dot{a}_{ki} - \dot{\underline{s}}_{ki} \cdot \underline{\nabla}_{s_{ki}} v \quad . \qquad (2.20)$$

In this case v_t , as given by (2.19), lies in the same space as v.

If a_{ki} is constrained in (2.15) so that v is continuous from element to element, as is usual, then v can be written

$$v = \sum_j a_j\alpha_j \qquad (2.21)$$

where α_j takes the value 1 at node j and zero at surrounding nodes. This is the usual nodewise representation which results in a continuous function v.

Since the basis function α_j is made up of several element basis functions ϕ_{ki}, each of which is linear and takes the value 1 at node j, we have from (2.8) and (2.14)

$$v_t = \sum_j [\dot{a}\alpha_j + \dot{\underline{s}}_j \cdot \underline{\beta}_j] \qquad (2.22)$$

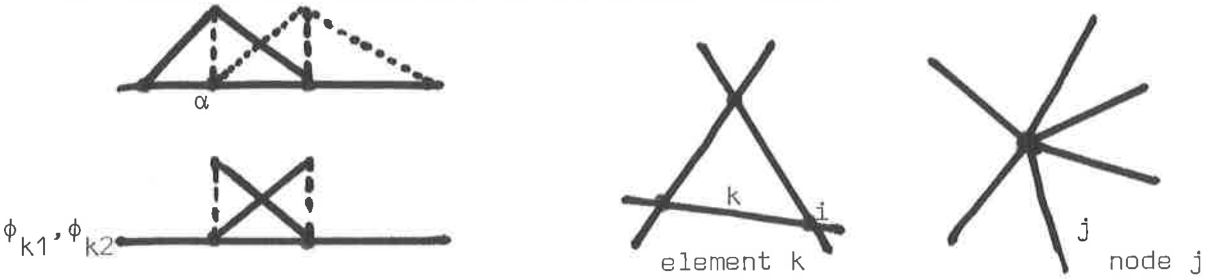
where

$$\underline{\beta}_j = - (\underline{\nabla}_{\underline{\sigma}_j} v)\alpha_j \quad , \quad \beta_j = - \frac{\partial v}{\partial \sigma_j} \alpha_j \quad . \qquad (2.23)$$

Now $\underline{\beta}_j$ is a different multiple of α_j in each element adjacent to node j, since the gradient of v varies from element to element. Thus, although the function v is continuous, its time derivative v_t is discontinuous and lies in the space $S_{\alpha\beta}$ spanned by the basis functions α_j and $\underline{\beta}_j$. Note that in two or more dimensions the space $S_{\alpha\beta}$ is smaller than the space

* N.B. w_{ki} is called $\dot{w}_{ki}$ in Wathen & Baines (1985).

S_ϕ spanned by the ϕ_{ki} although contained in it: this is because in those cases each element has fewer nodes than a node has surrounding elements (see Fig. 2.2(b)). With this proviso the form (2.18)-(2.19) can be used for v_t. These results hold generally in any number of dimensions.



(a) one dimensional basis functions (b) two dimensions (plan view)

FIG. 2.2 : (a) Nodewise and elementwise basis functions in one dimension
(b) Elementwise and nodewise numbering in two dimensions (plan view)

Specialising now to one dimension, we note here that $S_{\alpha\beta}$, the space spanned by the α_j and β_j (there is only one component of $\underline{\beta}_j$) is of the same dimension as S_ϕ, the space spanned by the ϕ_{ki} (see Fig. 2.2(a)). This is special to one dimension and ϕ_{k1}, ϕ_{k2} or α_j, β_j are equivalent alternative basis functions for v_t. It follows that, for a continuous function v

$$v = \sum_j a_j \alpha_j = \sum_k [a_{k1}\phi_{k1} + a_{k2}\phi_{k2}] \quad , \tag{2.24}$$

say, the time derivative v_t can be written, using $\beta_j = -m\alpha_j$ from (2.23) (m is v_x), in the form

$$v_t = \sum_j (\dot{\alpha}_j \alpha + \dot{s}_j \underline{\beta}_j) = \sum_k [\dot{a}_{k1}\phi_{k1} + \dot{a}_{k2}\phi_{k2} - m_k\dot{s}_{k1}\phi_{k1} - m_k\dot{s}_{k2}\phi_{k2}]$$

$$= \sum_k [w_{k1}\phi_{k1} + w_{k2}\phi_{k2}] \tag{2.25}$$

where

$$\left.\begin{aligned} w_{k1} &= \dot{a}_{k1} - m_k\dot{s}_{k1} \\ w_{k2} &= \dot{a}_{k2} - m_k\dot{s}_{k2} \end{aligned}\right\} \tag{2.26}$$

(m_k is m in the k'th element).

A way of increasing the degree of approximation of v while preserving continuity is to add to (2.24), in a hierarchical manner, higher order terms with various continuity properties. For example the basis function

$$\Phi_k = (s_{k2} - x)(x - s_{k1}) \tag{2.27}$$

may be added to give

$$v = \sum a_{k1}\phi_{k1} + a_{k2}\phi_{k2} + \sum_k c_k(s_{k2} - x)(x - s_{k1}) \tag{2.28}$$

which is piecewise quadratic with simple continuity at the nodes. Differentiation of (2.28) with respect to time yields

$$v_t = \sum_k [w_{k1}\phi_{k1} + w_{k2}\phi_{k2} + \dot{c}_k\Phi_k + c_k\{\dot{s}_{k2}\Delta_k s\ \phi_{k2} - \dot{s}_{k1}\Delta_k s\ \phi_{k1}\}] \tag{2.29}$$

where

$$\Delta_k s = s_{k2} - s_{k1} \quad . \tag{2.30}$$

That is

$$v_t = \sum_k [\{w_{k1} - \Delta_k s\ c_k\dot{s}_{k1}\}\phi_{k1} + \{w_{k2} + \Delta_k s\ c_k\dot{s}_{k2}\}\phi_{k2} + \dot{c}_k\Phi_k\ ] \tag{2.31}$$

which lies in the same space as v, spanned by the ϕ_{k1}, ϕ_{k2} and Φ_k.

In the above discussion on piecewise linear basis functions the gradients are piecewise constant and therefore have low accuracy. We pass on now to consider the right hand side of (1.1) and special forms of $L(v)$ which lie in the spaces already constructed.

3. "EXACT" L(u)

Having considered the form of v_t in some detail in Section 2, we now investigate the form of $L(v)$, where L is an operator which contains v and its space derivatives but does not contain time derivatives of v.

Interest centres on forms of $L(v)$ which lie in the same space as v_t, since there is then the capability of matching $L(v)$ and v_t in the partial differential equation (1.1) with no error. We call such a correspondence an "exact" match.

If v is of the form (2.21) v_t lies in a subspace $S_{\alpha\beta}$ of the space S_ϕ spanned by the ϕ_{ki}. The space S_ϕ consists of all functions which are linear in an element but not necessarily continuous across elements. We note that any linear function $\ell(v)$ of v lies in S_ϕ although it is also continuous across elements.

Moreover $\underline{\nabla}v$ also lies in S_ϕ since it is piecewise constant and discontinuities across elements are allowed. Any continuous function $f(\underline{\nabla}v)$ of $\underline{\nabla}v$ will also have the same property.

Taking these two observations together we note that the function

$$L(v) = f(\underline{\nabla}v)\ell(v) \tag{3.1}$$

of v lies in S_ϕ. For an exact match, however, it must lie in the subspace $S_{\alpha\beta}$ and this is not generally true in more than one dimension.

In one dimension, however, an exact match is possible between v_t and $L(v)$. As a result expressions for $\dot{a}_j$ and $\dot{s}_j$ may be obtained giving ordinary differential equations for a_j and s_j. If these equations can be integrated exactly in time, exact solutions of the partial differential equation

$$u_t = f(u_x)\ell(u) \tag{3.2}$$

may be found for piecewise linear data. As we shall see this can be done for the equation

$$u_t = - uu_x \quad . \tag{3.3}$$

To illustrate how far this method can be taken, consider the equation

$$u_t = u\,f(u_x) + g(u_x) \tag{3.4}$$

With v given by (2.24) and v_t by (2.25) we have

$$\sum_k [w_{k1}\phi_{k1} + w_{k2}\phi_{k2}] = \sum_k [\{a_{k1}f(m_k) + g(m_k)\}\phi_{k1} + \{a_{k2}f(m_k) + g(m_k)\}\phi_{k2}] \tag{3.5}$$

from which, by exact matching,

$$\left.\begin{aligned} w_{k1} &= \dot{a}_{k1} - m_k\dot{s}_{k1} = a_{k1}f(m_k) + g(m_k) \\ w_{k2} &= \dot{a}_{k2} - m_k\dot{s}_{k2} = a_{k2}f(m_k) + g(m_k) \end{aligned}\right\} \tag{3.6}$$

using (2.26). Dropping the suffix k we have

$$\left.\begin{aligned} \dot{a}_1 - m\dot{s}_1 &= a_1f(m) + g(m) \\ \dot{a}_2 - m\dot{s}_2 &= a_2f(m) + g(m) \quad . \end{aligned}\right\} \tag{3.7}$$

Now

$$m = \frac{a_2 - a_1}{s_2 - s_1} \tag{3.8}$$

and, subtracting the two equations (3.7), we obtain

$$\dot{a}_2 - \dot{a}_1 - m(\dot{s}_2 - \dot{s}_1) = (a_2 - a_1)f(m) \tag{3.9}$$

which, on division by $s_2 - s_1$ gives

$$\frac{dm}{dt} = mf(m) \quad . \tag{3.10}$$

Exact integration is possible in a number of special cases of f. In particular if

$$f(m) = m^p \tag{3.11}$$

where $p \neq 0$, we obtain

$$\int \frac{dm}{m^{p+1}} = \int dt \tag{3.12}$$

which gives

$$m^p = \frac{1}{p(c - t)} \qquad (c \quad \text{a constant}) \tag{3.13}$$

while if $f(m)$ is constant m is an exponential function of t, and if $f(m) \equiv 0$ m is constant for all time.

Returning to the set of equations (3.6) for all k and taking the two equations either side of node j, we have

$$\left.\begin{aligned} \dot{a}_j - m_L \dot{s}_j &= a_j f(m_L) + g(m_L) \\ \dot{a}_j - m_R \dot{s}_j &= a_j f(m_R) + g(m_R) \end{aligned}\right\} \tag{3.14}$$

where L, R stand for the left and right elements adjacent to the node j. Eliminating $\dot{s}_j$ from the equations (3.14) and excluding the case $m_L = m_R$, we have

$$\left(\frac{1}{m_L} - \frac{1}{m_R}\right) \dot{a} = \left[\frac{f(m_L)}{m_L} - \frac{f(m_R)}{m_R}\right] a + \left[\frac{g(m_L)}{m_L} - \frac{g(m_R)}{m_R}\right] \tag{3.15}$$

dropping the suffix j.

If $f(m) = m^p$ where $p \neq 0$, m is given by (3.13) and

$$\dot{a} = \left(\frac{m_L^{p-1} - m_R^{p-1}}{m_L^{-1} - m_R^{-1}}\right) a + \left(\frac{m_L^{-1} g(m_L) - m_R^{-1} g(m_R)}{m_L^{-1} - m_R^{-1}}\right) . \tag{3.16}$$

Now

$$\begin{aligned} \frac{m_L^{p-1} - m_R^{p-1}}{m_L^{-1} - m_R^{-1}} &= \frac{1}{p}\left[\frac{(c_L - t)^{\frac{1}{p} - 1} - (c_R - t)^{\frac{1}{p} - 1}}{(c_L - t)^{\frac{1}{p}} - (c_R - t)^{\frac{1}{p}}}\right] \\ &= \frac{d}{dt} \log \left|(c_L - t)^{\frac{1}{p}} - (c_R - t)^{\frac{1}{p}}\right| \\ &= \frac{d}{dt} \log \left\{ p^{\frac{-1}{p}} \left|m_L^{-1} - m_R^{-1}\right|\right\} \end{aligned} \tag{3.17}$$

so that (3.16) can be written

$$\frac{d}{dt}\left(\frac{a}{|m_L^{-1} - m_R^{-1}|}\right) = \left(\frac{m_L^{-1}g(m_L) - m_R^{-1}g(m_R)}{\left(m_L^{-1} - m_R^{-1}\right)^2}\right) \tag{3.18}$$

Again, exact integration is possible in a number of special cases of g, notably $g(m) = m$ and $g(m) = \text{const.}$

If $f = 0$, m is constant, and integration of (3.15) gives

$$a = \left(\frac{m_L^{-1}g(m_L) - m_R^{-1}g(m_R)}{m_L^{-1} - m_R^{-1}}\right) t + a_0 \tag{3.19}$$

which, in the special case $g(m) = m$, becomes

$$a = a_0 \quad . \tag{3.20}$$

For other f and g numerical integration is possible.

A different linear combination of equations (3.14) gives

$$-(m_L - m_R)\dot{s} = \{f(m_L) - f(m_R)\}a + g(m_L) - g(m_R) \tag{3.21}$$

which leads to s as a function of t. However, it may be better to use the relationship

$$s_{k2} - s_0 = \sum_{\ell=1}^{k} (s_{\ell 2} - s_{\ell 1}) = \sum_{1}^{k} (a_{\ell 2} - a_{\ell 1})m_\ell^{-1} \tag{3.22}$$

if the a's and m's are already known.

Using the above method exact solutions are obtained for the particular partial differential equations

$$\left.\begin{aligned} u_t + uu_x^p &= 0 \qquad &\text{(a)}\\ u_t + u_x^q &= 0 \qquad &\text{(3.23)(b)}\\ u_t + uu_x^p &= 1 \qquad &\text{(c)} \end{aligned}\right\}$$

with piecewise linear data. In the case (3.23)(a) with $p = 1$ the method is precisely the characteristic method (Wathen (1984)).

Turning now to v's of the form (2.28) we note that here v_t lies in the same space as v (in one dimension). Since in any element v is quadratic any linear function $\ell(v)$ of v lies in the same space. Also, since v_x is linear, any quadratic function $q(v_x)$ lies in the space. Moreover v_{xx} is constant so any continuous function $f(v_{xx})$ lies in the space. Taken together, the function

$$L(v) = f(v_{xx})(q(v_x) \text{ or } \ell(v)) \tag{3.24}$$

of v lies in the space spanned by the ϕ_{k1}, ϕ_{k2} and Φ_k. An exact match is then obtainable between v_t and $L(v)$, so that expressions for $\dot{a}_j$ $\dot{s}_j$ and $\dot{c}_k$ can be found and ordinary differential equations solved for a_j, s_j and c_j. Included in this class of partial differential equations is

$$u_t = u_{xx} \tag{3.25}$$

and

$$u_t = (uu_x)_x \tag{3.26}$$

With v given by (2.28) and v_t by (2.31), equation (3.25) becomes

$$\sum_k [(w_{k1} - \Delta_k s\, c_k \dot{s}_{k1})\phi_{k1} + (w_{k2} + \Delta_k s_k c_k \dot{s}_{k2}) + \dot{c}_k \Phi_k] = - \sum_k 2c_k(\phi_{k1} + \phi_{k2}) \tag{3.27}$$

so that, dropping the suffix k,

$$\left.\begin{aligned} w_1 - \Delta s\ c\ \dot{s}_1 &= -2c \\ w_2 + \Delta s\ c\ \dot{s}_2 &= -2c \\ \dot{c} &= 0 \end{aligned}\right\} \tag{3.28}$$

from which the local curvature c of the approximation is constant, c_0 say, and

$$\left.\begin{aligned} w_1 - \Delta s\ c_0\ \dot{s}_1 &= -2c_0 \\ w_2 + \Delta s\ c_0\ \dot{s}_2 &= -2c_0 \end{aligned}\right\} \tag{3.29}$$

Using (2.26) we obtain, for each element,

$$\left.\begin{aligned} \dot{a}_1 - (m - c_0\Delta s)\dot{s}_1 &= -2c_0 \\ \dot{a}_2 - (m + c_0\Delta s)\dot{s}_2 &= -2c_0 \end{aligned}\right\} \tag{3.30}$$

so that, subtracting,

$$\dot{a}_2 - \dot{a}_1 - m(\dot{s}_2 - \dot{s}_1) - c_0 \Delta s(\dot{s}_2 + \dot{s}_1) = 0 \qquad (3.31)$$

Dividing by Δs and using (3.8) gives

$$\frac{dm}{dt} = c_0(\dot{s}_2 + \dot{s}_1) \qquad (3.32)$$

leading to

$$m = c_0(s_2 + s_1) + m_0 \qquad (3.33)$$

where m_0 is a constant. Then (3.30) becomes

$$\left.\begin{aligned} \dot{a}_1 - (2\,c_0 s_1 + m_0)\dot{s}_1 &= -\,2c_0 \\ \dot{a}_2 - (2\,c_0 s_2 + m_0)\dot{s}_2 &= -\,2c_0 \end{aligned}\right\} \qquad (3.34)$$

which integrates to give the relation

$$a - \left(\frac{cs_1^2}{2} + m_0 s_1\right) = -\,2ct + d \qquad (3.35)$$

in each element.

Consider now the node j and the two equations from the set (3.30) relevant to that node. Denoting elements to the left and right of node j by L and R as before, we have

$$\left.\begin{aligned} a - (c_{0L}\frac{s^2}{2} + m_{0L}s) &= -\,2\,c_{0L}t + d_{2L} \\ a - (c_{0R}\frac{s^2}{2} + m_{0R}s) &= -\,2\,c_{0R}t + d_{1R} \end{aligned}\right\} \qquad (3.36)$$

Subtraction gives

$$(c_{0R} - c_{0L})\frac{s^2}{2} + (m_{0R} - m_{0L})s + 2(c_{0R} - c_{0L})t - (d_{2L} - d_{1R}) = 0 \qquad (3.37)$$

the solution of which provides s as a function of t. Then (3.36) provides a as a function of t.

For the right hand side $(uu_x)_x$ of (3.26) we consider $uu_x = uu_{xx} + u_x^2$ which in the k'th element has the form

$$\begin{aligned}- 2c_k a_{k1} \phi_{k1} - 2c_k a_{k2} \phi_{k2} - 2c_k^2 \Phi_k - 4c_k^2 \Phi_k + (m_k + c_k \Delta s)^2 \phi_{k1} \\ - (m_k - c_k \Delta s)^2 \phi_{k2}\end{aligned} \tag{3.38}$$

so that, as in (3.28)

$$\left.\begin{aligned} w_1 - c\Delta s\, \dot{s}_1 &= - 2ca_1 + (m + c\Delta s)^2 \\ w_2 + c\Delta s\, \dot{s}_2 &= - 2ca_2 - (m - c\Delta s)^2 \\ \dot{c} &= - 6c^2 \end{aligned}\right\} \tag{3.39}$$

The last of these equations provides

$$c = \frac{1}{(6t + c_0^{-1})} \quad , \tag{3.40}$$

which shows that the local curvature c in an element decreases with time, while the first two equations, in terms of $\dot{a}$ and $\dot{s}$ become

$$\left.\begin{aligned} \dot{a}_1 - (m + c\Delta s)\dot{s}_1 &= - 2ca_1 + (m + c\Delta s)^2 \\ \dot{a}_2 - (m - c\Delta s)\dot{s}_2 &= - 2ca_2 - (m - c\Delta s)^2 \quad . \end{aligned}\right\} \tag{3.41}$$

At a node j

$$\left.\begin{aligned} \dot{a} - (m_L - c_L \Delta_L s)\dot{s} &= - 2c_L a - (m_L - c_L \Delta_S s)^2 \\ \dot{a} - (m_R - c_R \Delta_R s)\dot{s} &= - 2c_R a + (m_R - c_R \Delta_R s)^2 \end{aligned}\right\} \tag{3.42}$$

which gives $\dot{a}$ and $\dot{s}$ in terms of a, s and time t.

In higher dimensions where the function $L(v)$ may lie in S_ϕ but does not necessarily lie in the appropriate subspace $S_{\alpha\beta}$ it is not possible to get a direct match between v_t and $L(v)$. But we can minimise the difference $v_t - L(v)$ in various ways, one way being to project $L(v)$ into the space $S_{\alpha\beta}$ containing v_t. This will lead to a set of normal equations for the $\dot{a}$ and $\dot{s}$.

We follow up this method in the next section.

4. MINIMISATION OF THE RESIDUAL

For general forms of v and $L(v)$ there will be no match between the two because they lie in different spaces. In particular this is true in one dimension when $L(v)$ is not one of the forms discussed in Section 3. For example, if v is piecewise linear and $L(v) \equiv v^2 v_x$ no match is possible, and again if v is piecewise quadratic and $L(v) \equiv vv_x$ there is no match. An important case is when v is piecewise linear and $L(v) \equiv v_{xx}$ as in the diffusion equation. The treatment of $L(v)$ in that case will be discussed in detail in Section 5.

The approach adopted here is a little different from the standard approach described briefly in Section 1. We choose an approximate form v for the object function u and calculate $L(v)$ as usual. But we then seek a locally linear best fit to $L(v)$ within an element and match it directly or indirectly to v_t. In one dimension the method is equivalent to that described in Section 1.

We choose the standard linear approximation (2.21) for v and v_t is then of the form (2.22). But we shall prefer to use the form (2.19) with the proviso that the point in the ϕ_{ki} space so obtained may not be in the space spanned by the α_j and $\underline{\beta}_j$. This proviso is not needed in one dimension, however.

With the linear form (2.21) for v we calculate $L(v)$ which will generally be smooth within an element but not lie in the space S_ϕ of piecewise linear functions spanned by the ϕ_{ki}. We therefore project $L(v)$ into the space S_ϕ by obtaining some linear fit to $L(v)$. This can be done in any convenient norm, although the L_2 norm has some useful advantages.

Denoting the result of this projection by $PL(v)$ we now try to match v_t with $PL(v)$. In one dimension this can be done directly and we shall concentrate on this case now. The extension to higher dimensions (in which a second projection is proposed) will be dealt with in Section 9.

In one dimension, then, within an element k we have

$$PL(v) = c_{k1}\phi_{k1} + c_{k2}\phi_{k2} \tag{4.1}$$

whereas v_t takes the form (2.25). Thus, using (2.26), we have a match between v_t and $PL(v)$ if

$$\begin{aligned} w_{k1} &= \dot{a}_{k1} - m_k \dot{s}_{k1} = c_{k1} \\ w_{k2} &= \dot{a}_{k2} - m_k \dot{s}_{k2} = c_{k2} \end{aligned} \tag{4.2}$$

which, as with (3.10), can be used to give

$$\frac{dm}{dt}k = \frac{c_{k2} - c_{k1}}{s_{k2} - s_{k1}} \quad , \tag{4.3}$$

or, as with (3.14), can be regrouped to give

$$\left.\begin{aligned} \dot{a}_j - m_{jL}\dot{s}_j &= c_{jL} \\ \dot{a}_j - m_{jR}\dot{s}_j &= c_{jR} \quad . \end{aligned}\right\} \tag{4.4}$$

Provided that $m_{jL} \neq m_{jR}$ we can solve equations (4.4) for

$$\dot{a}_j = \frac{m_{jR}c_{jL} - m_{jL}c_{jR}}{m_{jR} - m_{jL}} \tag{4.5}$$

$$\dot{s}_j = \frac{c_{jL} - c_{jR}}{m_{jR} - m_{jL}} \tag{4.6}$$

having obtained m from (4.3) or directly.

If $m_L = m_R = m$, say, the pair of equations (4.4) is singular with null space vector $[m \;\; 1]^T$. We may still solve the equations by taking a particular solution, say $\dot{a}_j = \dot{a}$, $\dot{s}_j = 0$, and adding an arbitrary multiple of the null vector chosen to satisfy some external criterion (see Section 10).

Integration of (4.3), (4.5) or (4.6) will depend on c_{jL}, c_{jR} and generally must proceed numerically. Note that the MFE method up to here is semi-discrete (leading to ordinary differential equations), and that any numerical time integration making it fully discrete is simply tacked on to the MFE equations: a fully discrete MFE approach does not appear tractable because of the complexity of the $\underline{\beta}$ basis functions.

An interesting feature of the MFE method, as described above is its local rather than global nature. The calculation of $L(v)$, projection into S_ϕ and the matching with v_t are all done within an element. The w_{k1}, w_{k2} are evaluated within an element and the $\dot{a}_j$, $\dot{s}_j$ are evaluated from information spread only over elements adjacent to node j. The method therefore appears particularly suitable to equations of hyperbolic type with local characteristics. Of course a procedure for the formation of shocks is needed for hyperbolic equations but this has already been done by Wathen and Baines (1985)(see Section 6). By the same argument it is less appropriate to diffusion problems where global properties, such as a maximum principle apply.

If the projection step is chosen to be an L_2 projection the standard MFE method is obtained. For then the c_{k1}, c_{k2} in the projection (4.1) (which are equal to the w_{k1}, w_{k2} (see (4.2))) are given by

$$\begin{bmatrix} <\phi_{k1}, \phi_{k1}> & <\phi_{k1}, \phi_{k2}> \\ <\phi_{k2}, \phi_{k1}> & <\phi_{k2}, \phi_{k2}> \end{bmatrix} \begin{bmatrix} c_{k1} \\ c_{k2} \end{bmatrix} = \begin{bmatrix} <\phi_{k1}, L(v)> \\ <\phi_{k2}, L(v)> \end{bmatrix} , \qquad (4.7)$$

which also comes from minimising the L_2 norm $\| v_t - L(v) \|_2$ over w_{k1}, w_{k2}. Writing the matrix of (4.7) as C_k and $[c_{k1}, c_{k2}]^T = (w_{k1}, w_{k2})^T = \underline{w}_k$, this is

$$C_k \underline{w}_k = \underline{b}_k , \qquad (4.8)$$

where $\underline{b}_k = [b_{k1}, b_{k2}]^T$ and

$$b_{k1} = <\phi_{k1}, L(v)> \qquad b_{k2} = <\phi_{k2}, L(v)> . \qquad (4.9)$$

To relate $\underline{w}_k$ to the $\dot{a}_j$, $\dot{s}_j$ we use (4.4) which in matrix form is

$$M_j \dot{\underline{y}}_j = \underline{w}_j \qquad (4.10)$$

where

$$M_j = \begin{bmatrix} 1 & -m_{jL} \\ 1 & -m_{jR} \end{bmatrix} , \qquad (4.11)$$

$$\dot{\underline{y}}_j = [a_j, s_j]^T \qquad (4.12)$$

and $\underline{w}_j$ consists of the two adjacent w's associated with node j.

Finally, denoting by C the diagonal block matrix with blocks C_k and by M the diagonal block matrix with blocks M_j we have apart from end effects

$$C\underline{w} = \underline{b} \tag{4.13}$$

$$M\dot{\underline{y}} = \underline{w} \tag{4.14}$$

where $\underline{w}^T = \{w_j\}^T$, $\underline{b}^T = \{b_k\}^T$ and $\dot{\underline{y}}^T = \dot{a}_j, \dot{s}_j$. Note that the blocks of C and of M are staggered with respect to each other.

Now the $\underline{b}$ vector can be written in terms of the $\underline{g}$ vector of (1.5) by making use of the relations

$$\left.\begin{aligned} \alpha_j &= \phi_{jL} + \phi_{jR} \\ \beta_j &= - m_{jL}\phi_{jL} - m_{jR}\phi_{jR} \end{aligned}\right. \tag{4.15}$$

(c.f. (2.24), (2.25)), which give

$$\begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix} = M^T \begin{bmatrix} \phi_{jL} \\ \phi_{jR} \end{bmatrix} . \tag{4.16}$$

Then, from (4.9) and (1.5) we have

$$M^T\underline{b} = \underline{g} \qquad . \tag{4.17}$$

Combining (4.13), (4.14) and (4.17) leads to

$$M^TCM\dot{\underline{y}} = \underline{g} \tag{4.18}$$

as for the standard MFE method (c.f. Wathen & Baines (1985)). These are the equations obtained by minimising the L_2 norm $\|v_t - L(v)\|_2$ over $\dot{a}$ and $\dot{s}$ when v_t is given by (2.22)(in one dimension). The normal equations are

$$\left.\begin{aligned} \langle\alpha_j, v_t - L(v)\rangle &= 0 \\ \langle\beta_j, v_t - L(v)\rangle &= 0 \end{aligned}\right\} \tag{4.19}$$

which leads to

$$A(\underline{y})\dot{\underline{y}} = \underline{g}(\underline{y}) \quad , \tag{4.20}$$

where $A(\underline{y})$ is as in Section 1.

A consequence of using the L_2 norm is that, summing the first equation of (4.19) over all j (or the equivalent ϕ_{ki} equations over all k,i) gives

$$\int_{s_0}^{s_{N+1}} \{v_t - L(v)\}dx = 0 \tag{4.21}$$

and, if the, s_0, s_{N+1} are fixed, this becomes

$$\frac{d}{dt}\left[\int_{s_0}^{s_{N+1}} vdx\right] = \int_{s_0}^{s_{N+1}} L(v)\ dx. \tag{4.22}$$

Thus, apart from boundary terms, the $\int_{s_0}^{s_{N+1}} vdx$ is conserved with time, a useful property to build into the modelling of conservation laws.

Moreover, if the first equation of (4.19) is multiplied by a_j and then summed over j, we have

$$\int_{s_0}^{s_{N+1}} \{vv_t - vL(v)\}dx = 0 \tag{4.23}$$

and again, if s_0, s_{N+1} are fixed, this gives

$$\frac{d}{dt}\int_{s_0}^{s_{N+1}} v^2\ dx = 2\int_{s_0}^{s_{N+1}} vL(v)dx \quad . \tag{4.24}$$

If L is such that $\langle v, L(v)\rangle = 0$, then $\int_{s_0}^{s_{N+1}} v^2 dx$ is constant in the same way as $\int_{s_0}^{s_{N+1}} u^2 dx$, another useful modelling property.

Because we have both equations (4.19) satisfied, (4.21) and (4.23) hold locally, i.e. with s_0, s_{N+1} replaced by s_j, s_{j+1}. But (4.22) and (4.24) do not follow because the local boundaries are moving and "leak" the conserved quantities.

The L_2 norm is essential for forms of the operator L involving u_{xx} (see Section 5).

Non-standard forms of the MFE method are possible, by means of projections using norms other than L_2, although for the reasons above the L_2 norm appears preferable.

One result that comes out of using the L_2 norm is on the behaviour of the slope m_k that appears in equation (4.3), which may be written

$$\frac{dm_k}{dt} = \frac{1}{\Delta_k s}\,[-1\ \ 1]\,\underline{w}_k = \frac{1}{\Delta_k s}\,[-1\ \ 1]\,C_k^{-1}\,\underline{b}_k \tag{4.25}$$

$$= \frac{6}{(\Delta_k s)^2}\,[-1\ \ 1]\begin{bmatrix} <\phi_{k1}, L(v)> \\ <\phi_{k2}, L(v)> \end{bmatrix}$$

$$= \frac{6}{(\Delta_k s)^2}\,<\psi_k, L(v)> \;=\; \frac{6}{(\Delta_k s)^2}\int_{s_{k1}}^{s_{k2}} \psi_k L(v)\,dx \tag{4.26}$$

where $\psi_k = -\phi_{k1} + \phi_{k2}$. [Note that (-1,1) is an eigenvector of C]. (4.27)

Hence, if $L(v) = -f_x(v)$

$$\frac{dm_k}{dt} = \frac{-6}{(\Delta_k s)^2}\int_{s_{k1}}^{s_{k2}} \psi_k f_x(v)\,dx \quad . \tag{4.28}$$

$$= \frac{-6}{(\Delta_k s)^2}\left[[f_{k2} + f_{k1} - \frac{2}{\Delta_k s}\int_{s_{k1}}^{s_{k2}} f(v)\,dx]\right]$$

$$= \frac{-12}{(\frac{1}{2}\Delta_k s)^2}\,\{\bar{f}_k - \hat{f}_k\} \tag{4.29}$$

where $\bar{f}_k = \frac{1}{2}(f_{k1} + f_{k2})$ and $\hat{f}_k = \frac{1}{\Delta_k s}\int_{s_{k1}}^{s_{k2}} f(v)\,dx$. (4.30)

It follows from (4.29) that if f is convex upwards the slope m_k decreases and if f is convex downwards the slope m_k increases. If f is linear $\frac{dm_k}{dt}$ is zero and if f is quadratic $\frac{dm_k}{dt} \propto m^2$, as before.

The separation of the stages of the method into a local projection in the space S_ϕ together with a further projection also clarifies the role of matrix singularities, a feature which early workers went to great lengths to avoid.

Thus, singularity of C in (4.13) depends on singularity of C_k in (4.8). From the fact that C_k is symmetric and positive semi-definite and a multiple of $\Delta_k s$, singularity can occur only if $\Delta_k s = 0$. The occurrence of this type of singularity will be considered in detail in Section 6.

Singularity of M in (4.14), which arises through singularity of M_j in (4.11), occurs only when $m_{jL} = m_{jR}$, which corresponds to collinearity of nodes or parallelism. In this case the spaces S_ϕ and $S_{\alpha\beta}$ spanned by the ϕ_{ki} and α_j, β_j are not equivalent. The projection $PL(v)$ of $L(v)$ into S_ϕ cannot be mapped uniquely into a point of $S_{\alpha\beta}$. Roughly speaking, the solution has a "loose" point whose position is undefined.

To get round this latter difficulty we can solve (4.10) for a particular solution $\dot{\underline{y}}_j^*$ in the space $S_{\alpha\beta}$ (with $\dot{s}_j = 0$, say) and add an arbitrary multiple of the null space (see Wathen & Baines (1985)). The arbitrariness can then be eliminated by imposing some external criterion, such as that the loose point should move at an average speed or should be located after a time step at an averaged point.

Note that in the case of parallelism there is less information contained in the vector $\underline{g}$ of (1.5) than there is in the vector $\underline{b}$ of (4.13). Thus if $\underline{g}$ is calculated, as in the standard MFE method, and (4.18) solved for $\dot{\underline{y}}$ then in the case of parallelism the effort in obtaining a particular solution $\dot{\underline{y}}^*$ is greater because of the lack of eigenvalue clustering of the elementary spectrum of A^*. If, however, we work with $\underline{b}$ and solve (4.13) for $\underline{w}$ (which we can always do provided that $\Delta_k s \neq 0$) the difficulty of obtaining $\dot{\underline{y}}_j^*$ reduces to that of finding $\underline{y}_j^*$ from (4.10), a relatively trivial matter.

The particular solution obtained by setting $\dot{s}_j = 0$ is equivalent to applying the method with one fixed node. Applying this constraint links the elements to the left and right of the node j and the method loses its local nature. This may be understood in terms of adjacent elements temporarily acting as a single element.

If several, or perhaps most, of the nodes are parallel (collinear) and are temporarily fixed to obtain a particular solution, the constraints become more widespread and the local nature of the method is lost, functions in one element affecting the behaviour of distant elements. The fixed finite element method has this character: indeed it is the limit of the MFE method when all the $\dot{s}_j = 0$.

Although boundary conditions will be discussed in detail in Section 7 we note here that the imposition of a fixed boundary also gives a constraint on the method. As we shall see, some condition must be imposed at boundaries to prevent non-uniqueness but the constraining effects do not affect the local nature of the method.

Other kinds of constraints will be considered in Sections 6, 7 and 8 which are concerned respectively with time stepping, boundary conditions and systems of equations. Next, however, we consider problems of diffusion type involving second derivatives of u in space when the approximating function is piecewise linear. This causes special problems in the evaluation of $L(v)$ and of inner products containing this term. It is these that we consider in the next section.

5. DIFFUSION OPERATORS

If we attempt to carry through the MFE method with a diffusion equation of the form

$$u_t = L(u) \equiv \underline{\nabla} \cdot (D(u)\underline{\nabla}u) \qquad (5.1)$$

with a piecewise linear approximation (2.21) for v, difficulties arise with the second derivative of v which does not exist except in a distributional sense.

There are two ways forward here. One, due to Miller (see also Mueller (1983)) is to use the L_2 form of the residual and evaluate the troublesome inner products using integration by parts, as in the usual fixed finite element method. Special justification of this procedure is required when using inner products with the $\underline{\beta}$ function; this is done by Miller on the basis of mollification of the basis functions. Using the $S_{\alpha\beta}$ space the inner products (1.5) for $\underline{g}$, the right hand side of (1.3), are

$$\left.\begin{array}{ll} \langle\alpha,\ \underline{\nabla} \cdot (D(v)\underline{\nabla}v)\rangle & \\ \text{and} \quad \langle\beta_j,\ \underline{\nabla} \cdot (D(v)\underline{\nabla}v)\rangle\ , & \underline{\beta} = \{\beta_j\}. \end{array}\right\} \qquad (5.2)$$

Following Mueller, we use Green's identities and the vanishing of α (except for boundary nodes) over the border of its support to obtain

$$\langle\alpha,\ \underline{\nabla} \cdot (D(v)\underline{\nabla}v)\rangle = -\langle\underline{\nabla}\alpha,\ D(v)\underline{\nabla}v\rangle \qquad (5.3)$$

and (c.f. (2.23)

$$\langle\beta_j,\ \underline{\nabla} \cdot (D(v)\underline{\nabla}v)\rangle = -\langle\alpha\frac{\partial v}{\partial\sigma_j},\ \underline{\nabla} \cdot (D(v)\underline{\nabla}v)\rangle \qquad (5.4)$$

$$= -\int \underline{\nabla}\cdot\left\{\alpha\frac{\partial v}{\partial\sigma_j}\ D(v)\underline{\nabla}v\right\}d\tau + \langle\underline{\nabla}\left\{\alpha\frac{\partial v}{\partial\sigma_j}\right\}, D(v)\underline{\nabla}v\rangle \qquad (5.5)$$

where the first term of (5.5) vanishes for interior nodes.

Thus

$$\langle\beta_j, \underline{\nabla}\cdot(D(v)\underline{\nabla}v)\rangle = \langle\alpha\underline{\nabla}\left(\frac{\partial v}{\partial\sigma_j}\right) + \frac{\partial v}{\partial\sigma_j}\underline{\nabla}\alpha, D(v)\underline{\nabla}v\rangle$$
$$= \langle\alpha\underline{\nabla}\left(\frac{\partial v}{\partial\sigma_j}\right), D(v)\underline{\nabla}v\rangle + \langle\left(\frac{\partial v}{\partial\sigma_j}\right)\underline{\nabla}\alpha, D(v)\underline{\nabla}v\rangle \quad . \tag{5.6}$$

The second term contains only first derivatives of v while the first is equal to

$$\int\alpha D(v)\frac{\partial}{\partial\sigma_j}\left\{\tfrac{1}{2}(\underline{\nabla}v)^2\right\}d\tau = \int\frac{\partial}{\partial\sigma_j}\left\{\frac{\alpha}{2}D(v)(\underline{\nabla}v)^2\right\}d\tau - \int\tfrac{1}{2}(\underline{\nabla}v)^2\frac{\partial}{\partial\sigma_j}\{\alpha D(v)\}d\tau \tag{5.7}$$

where, again, the first term vanishes for interior nodes. Thus we are left with

$$\langle\beta_j, \underline{\nabla}\cdot(D(v)\underline{\nabla}v)\rangle = \langle\frac{\partial v}{\partial\sigma_j}\underline{\nabla}\alpha, D(v)\underline{\nabla}v\rangle - \tfrac{1}{2}\int(\underline{\nabla}v)^2\frac{\partial}{\partial\sigma_j}\{\alpha D(v)\}d\tau \tag{5.8}$$

which contains only first derivatives of v. The result is conveniently written

$$\langle\underline{\beta}_j, \nabla\cdot(D(v)\underline{\nabla}v)\rangle = \int\left[D(v)\,\underline{\nabla}_{\sigma_j}v\,(\underline{\nabla}\alpha\cdot\underline{\nabla}v) - \tfrac{1}{2}(\underline{\nabla}v)^2\underline{\nabla}_{\sigma_j}\{\alpha D(v)\}\right]d\tau \quad . \tag{5.9}$$

For boundary nodes, extra terms are required whose evaluation depends on the application of boundary conditions (see Section 7).

The forms (5.3) and (5.9) contain only first derivatives of v and evaluation of $\underline{g}$ can proceed in the usual way. Some choice needs to be made of the values of $\underline{\nabla}v$ at the element boundaries and this must be done in a manner to be determined: Miller chooses, in one dimension, simple averages of the left and right slopes (see below).

Another approach, suggested by Morton (1982), is to replace the second derivative of v where it occurs in (5.2) by the second derivative of an appropriate "recovered" function W. The recovered function is chosen to be smoother than v, so that it may be twice continuously differentiated, but is close to v in some norm. This mapping is a sort of anti-projection guided by external information and is usually far from unique. We illustrate the possibilities in one dimension where several candidates exist. One possible choice is the Hermite cubic W_H within each element which is chosen to match the function v at the end points of the element as well as matching the average slope of v at the end points. The second derivative of W_H is linear in each element and piecewise linear over the whole range, being discontinuous from element to element. Hence $\frac{\partial^2 W_H}{\partial x^2} \in S_\phi$ although $\frac{\partial}{\partial x}\left\{D(v)\frac{\partial W_H}{\partial x}\right\} \notin S_\phi$ in general.

We follow through the consequences in the case $D(v) \equiv 1$. In that case the right hand side of (5.1), which now takes the form

$$u_t = L(u) \equiv \frac{\partial^2 W_H}{\partial x^2} , \tag{5.10}$$

lies in the space S_ϕ and there is an exact match within each element between the left and right hand sides of this equation. Denoting by M_1 and M_2 the average slopes chosen to match the Hermite cubic at s_1 and s_2 respectively (see Fig. 5.1), we have

$$\left.\begin{aligned} \frac{\partial^2 W_H}{\partial x^2} &= \frac{6}{(\Delta s)^2}(M_1 + M_2 - 2m)x + \gamma , \\ \text{where} \quad \gamma &= -\frac{6m}{\Delta s} - \frac{2s_2}{(\Delta s)^2}(M_2 + 2M_1) - \frac{2s_1}{(\Delta s)^2}(M_1 + 2M_2) , \end{aligned}\right\} \tag{5.11}$$

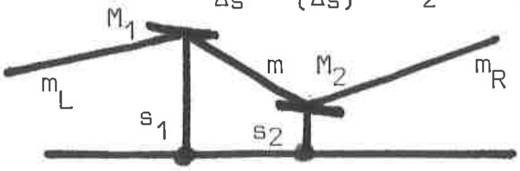


FIG. 5.1 : Averaged slopes at nodes

which leads to the match (c.f. (3.6))

$$\left.\begin{aligned} w_1 &= \dot{a}_1 - m\dot{s}_1 = \frac{6}{(\Delta s)^2}(M_1 + M_2 - 2m)s_1 + \gamma \\ w_2 &= \dot{a}_2 - m\dot{s}_2 = \frac{6}{(\Delta s)^2}(M_1 + M_2 - 2m)s_2 + \gamma . \end{aligned}\right\} \tag{5.12}$$

Subsequent subtraction gives, as in (3.8) et seq.,

$$\frac{dm}{dt} = \frac{6}{(\Delta s)^2}(M_1 + M_2 - 2m) , \tag{5.13}$$

an interesting result since, if the straightforward choice

$$\left.\begin{aligned} M_1 &= \tfrac{1}{2}(m + m_L) \\ M_2 &= \tfrac{1}{2}(m + m_R) \end{aligned}\right\} \tag{5.14}$$

is made (see Fig. 5.1), (5.13) gives

$$\frac{dm}{dt} = \frac{3}{(\Delta s)^2}(m_L + m_R - 2m) \tag{5.15}$$

(a finite difference discretisation of the equation

$$m_t = 3m_{xx} \tag{5.16}$$

on a smoothed out mesh). It is easily shown that (5.15) possesses a maximum principle, so that m does not rise above initial and boundary values (at least in the semi-discrete solution).

Indeed, as long as the averaging in (5.14) leads to a convex function in (5.15) the maximum principle applies (c.f. (4.29)). In particular, if we choose

$$M_1 = \frac{1}{6}(m_L + 5m), \; M_2 = \frac{1}{6}(m_R + 5m) \tag{5.17}$$

- an unsymmetric choice - the 3 in (5.1) disappears and the equation for m is precisely that for u_x (c.f. (3.25)).

$$m_t = m_{xx} \quad , \tag{5.18}$$

otherwise obtained from (3.25) by differentiation. This form of recovery appears to give some consistency with the differential equation for m.

We can be more sophisticated by employing the recovery

$$\begin{aligned} M_1 &= \theta(m_L - m) + m \\ M_2 &= \phi(m_R - m) + m \end{aligned} \tag{5.19}$$

where θ,ϕ are chosen so as to give consistency with the differential equation (5.17) differenced on an irregular grid with m taken to be at the mid-pts. of elements. Then, denoting by $\bar{\sigma}$ the co-ordinate of a mid-pt. of an element, the appropriate θ, ϕ are

$$\left.\begin{aligned} \theta &= \frac{1}{3}\,\frac{(\Delta s)^2}{(\sigma-\bar{\sigma}_L)(\sigma_R-\sigma_L)} \\[2em] \phi &= \frac{1}{3}\,\frac{(\Delta s)^2}{(\sigma_R-\bar{\sigma})(\sigma_R-\sigma_L)} \quad . \end{aligned}\right\} \tag{5.20}$$

The choice (5.14) is used by Miller in the evaluation of his inner products (see above). It has been shown by Johnson (1984) that in the above case the Miller mollification method and the Hermite cubic recovery method lead to the same MFE equations. This result shows that the recovery method is as powerful as the direct method and more flexible.

An obvious alternative recovery is to seek a quadratic function $Q(x)$ which matches the derivative v_x in a suitable way. By matching Q with m at the mid point of an element and with M_1, M_2 at the end points a different w_{xx} equal to Q_x, is obtained. The corresponding forms of (5.12) and (5.13) are

$$\left.\begin{aligned} w_1 &= \dot{a}_1 - m\dot{s}_1 = \frac{4}{(\Delta s)^2}(M_1 + M_2 - 2m)s_2 + \gamma^1 \\ w_2 &= \dot{a}_2 - m\dot{s}_2 = \frac{4}{(\Delta s)^2}(M_1 + M_2 - 2m)s_1 + \gamma^1 \end{aligned}\right\} \tag{5.21}$$

and (c.f. (5.13))

$$\frac{dm}{dt} = \frac{4}{(\Delta s)^2}(M_1 + M_2 - 2m) \tag{5.22}$$

showing that the slope decreases more slowly for this recovery. The nodes will not move in the same way in the two recoveries so there is no incompatibility.

It can be shown using the above arguments that a recovery of any lower order gives an unchanging slope which may not be sufficient for a good representation of the solution,

A different choice of cubic recovery is afforded by the cubic spline function W_s, which matches the function v and its first and second derivatives at the interior nodes together with special conditions at the end points. The second derivative of W_s is again piecewise linear but this time is continuous at the nodes. It belongs to the space S_ϕ, in fact to the subset S_α of continuous functions in this space. It follows from matching this second derivative to v_t in the equation

$$v_t = \sum (\dot{a}_j\alpha_j + \dot{s}_j\beta_j) = \frac{\partial^2 W_s}{\partial x^2} \tag{5.23}$$

that, since there are no discontinuities in $\dfrac{\partial^2 W_s}{\partial x^2}$,

$$\dot{s}_j = 0 \qquad \forall j \quad . \tag{5.24}$$

So there is no nodal movement and the method is a fixed finite element method. This type of recovery indicates that the motion of the nodes is very sensitive to the precise form of the recovery.

One advantage of the spline recovery is that it links together information from all parts of the region when evaluating the spatial second derivatives. It therefore gives , as with implicit finite difference methods, a means of linking boundary data to interior data to support global properties like the maximum principle. Equation (5.15), in common with explicit finite difference methods, fails to give this strong connection. Thus both the Miller method and the local cubic recovery are susceptible to instabilities. A mitigating feature is that the diffusion coefficient $\frac{3}{(\Delta s)^2}$ in (5.15) becomes very high when the nodes run closely together, giving high diffusion. Thus the method does not appear to need artificial diffusion.

Returning to the general case (5.1) in one dimension

$$u_t = (D(u)u_x)_x \tag{5.25}$$

there will be no match between the two sides of the equation for general D(v) (except perhaps for rather special recoveries), and the minimisation of the residual (or the best fit) in an appropriate norm will be needed. The L_2 norm is necessary for the Miller-Mueller approach described at the beginning of this section although with recovery this is not an absolute requirement. However, the conservation properties which follow when the L_2 norm is used make this norm a natural choice in diffusion problems (see (4.21) et. seq.).

In calculating the L_2 projection, however, note that it is possible to project into the space S_ϕ rather than $S_{\alpha\beta}$ which gives more information to the solution mechanism in higher space dimensions, and does so even in one dimension in the case of parallelism.

Alternatively we can use a recovery approach with m replaced everywhere by $D(v)m$. This may be the way to deal with the non-linear case.

We end this section with a study of the model convection-diffusion Burgers' equation

$$u_t = - uu_x + \varepsilon u_{xx} \qquad (5.26)$$

which has been studied by Herbst (1983) and Johnson (1984). Herbst showed that for a solution with a steep front with a shock structure and Neumann boundary conditions at s_0, s_{N+1}

$$\varepsilon \int_{s_0}^{s_{N+1}} u_x^2 \, dx = \frac{1}{12} (u_{N+1} - u_0)^3 \qquad (5.27)$$

holds both for the exact solution and for the finite element solution, in which case it takes the form

$$\varepsilon \sum_k m_k^2 = \frac{1}{12} (v_{N+1} - v_0)^3 \quad . \qquad (5.28)$$

He pointed out that, since the right hand side of (5.37) is fixed, small ε's go with large $\sum m_k^2$ and, if the mesh is equidistant, a sufficiently small ε will require such a large $\sum m_k^2$ that the only way of providing it is for the solution to exhibit oscillations (see Fig. 5.2(a)). If the mesh is

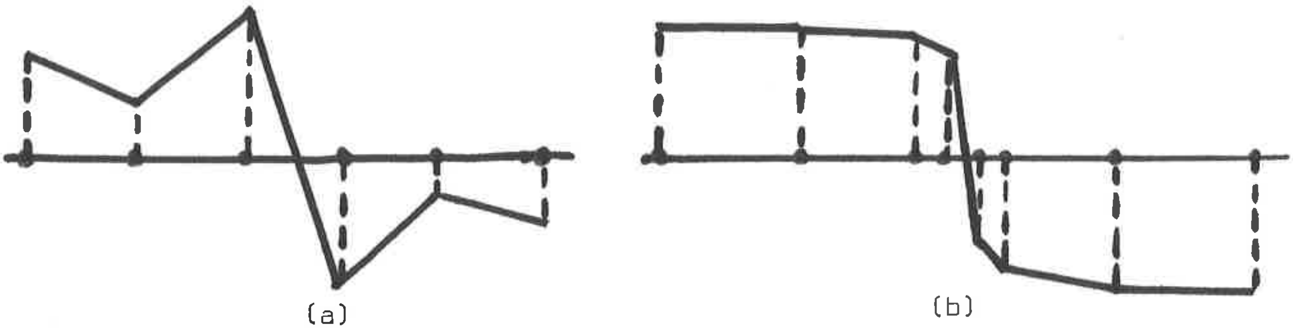


FIG. 5.2 : Representation of a front by linear finite elements for (a) equidistant points, (b) unequally spaced points.

allowed to be irregular, or to move, however, there is no need for such oscillations (Fig. 5.2(b)).

Using the recovery described in (5.11) et. seq. and including the term $-uu_x$ of (5.25) in the manner of (3.4) leads to the equations

$$\left.\begin{aligned} \dot{a}_1 - m\dot{s}_1 + a_1 m &= \frac{6\varepsilon}{(\Delta s)^2}(M_1 + M_2 - 2m)s_1 + c \\ \dot{a}_2 - m\dot{s}_2 + a_2 m &= \frac{6\varepsilon}{(\Delta s)^2}(M_1 + M_2 - 2m)s_2 + c \end{aligned}\right\} \qquad (5.29)$$

(c.f. (3.7) and (5.12)). Subtraction and division by Δs gives

$$\frac{dm}{dt} + m^2 = \frac{6\varepsilon}{(\Delta s)^2}(M_1 + M_2 - 2m). \qquad (5.30)$$

How are M_1, M_2 to be chosen in this case? If we take the differential equation for m as a guide, we must differentiate (5.25) with respect to x giving

$$m_t + um_x + m^2 = \varepsilon m_{xx} \quad . \qquad (5.31)$$

If we regard the $\frac{d}{dt}$ operator in (5.30) as a moving operator, equivalent to $\frac{D}{Dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x}$, (since (5.30) arises from a moving grid differentiation) consistency with the differential equation on a smoothed out grid is obtained as in the pure diffusion case (5.17). Similarly (5.19) and (5.20) give the appropriate values of M_1 and M_2 in the more accurate irregular grid differencing case.

When $\varepsilon = 0$ (5.30) has the solution

$$m = \frac{1}{t - c} \qquad (5.32)$$

corresponding to pure convection. If m starts negative at $t = 0$ (c positive) m then approaches $\dot{\infty}$ through negative values. This is correct for this data since we expect a shock to form. For the convection-diffusion equation (5.26), however, this is not the expected behaviour. We do not expect a shock and do not expect $m \to -\infty$. Rather we expect a dynamic steady state to be reached where the solution of

$$uu_x = \varepsilon u_{xx} \tag{5.33}$$

with its steep front is convected with a steady speed. Hence the recovery of u_{xx} should eventually be sufficient to match the left-hand side of (5.33) which will be large for large u_x. This is unlikely to be achieved by a recovery based on polynomial fitting, however, since near steep fronts polynomial approximation is notoriously suspect. A recovery with a built-in limiter might be the answer.

Alternatively, if we choose M_1, M_2 to be given by (5.14) when m is small, but choose them to satisfy

$$\frac{6\varepsilon}{(\Delta s)^2}(M_1 + M_2 - 2m) = m^2 \tag{5.34}$$

when m is large we obtain the right character of the recovery. This can be achieved with the recovery (5.19) taking θ, ϕ to be

$$\theta = \min\left\{ \tfrac{1}{2}, \frac{(\Delta a)^2}{12\varepsilon(m_L - m)}\right\}, \quad \phi = \min\left\{ \tfrac{1}{2}, \frac{(\Delta a)^2}{12\varepsilon(m_R - m)}\right\} \tag{5.35}$$

since these values will reduce the rate of change of m to zero preventing the shock forming. For very small m ,however,(5.35) will give almost constant m and no diffusion at all in (5.30), allowing the nodes to move into the front. As the front gets steeper the rate of change of m is reduced to zero preventing the formation of a shock.

In the latter part of this section we have used approximate consistency with the differential equation to determine the form of recovery. The aim is to build in known features of the problem when the recovery is done rather than depend on straight averaging which might be expected to work only for smooth solutions.

Care has to be exercised in using exotic recoveries, however, to make sure that each M is symmetric from both sides, e.g. (5.14) but not (5.17), since otherwise conservation properties are lost. A better form of (5.14), within the family (5.19), is

$$M_1 = \frac{(\Delta s)^{-1} + (\Delta_L s)^{-1} m_L}{(\Delta s)^{-1} + (\Delta_L s)^{-1}}\,, \quad M_2 = \frac{(\Delta s)^{-1} m + (\Delta_R s)^{-1} m_R}{(\Delta s)^{-1} + (\Delta_R s)^{-1}}\,, \tag{5.36}$$

while a symmetric version of the recovery corresponding to (5.35) is

$$\left.\begin{aligned} M_1 &= \frac{m + m_L}{1 + \left\{1 + \dfrac{((\Delta s)^2 m^2 + (\Delta_L s)^2 m_L^2)}{12\varepsilon(m+m_L)}\right\}^{-1}} \\[2em] M_2 &= \frac{m + m_R}{1 + \left\{1 + \dfrac{((\Delta s)^2 m^2 + (\Delta_R s)^2 m_R^2)}{12\varepsilon(m+m_R)}\right\}^{-1}} \end{aligned}\right\} \tag{5.37}$$

Generalisations of these ideas have been made to two dimensions.

We move on now to time stepping considerations and possible strategies for coping with overtaking nodes, a source of singularity of the method.

6. TIME STEPPING

As remarked upon in a previous section the MFE method, in its standard form or any of the variants mentioned here, is essentially a semi-discrete method which transforms the original set of time dependent partial differential equations into a set of ordinary differential equations (the MFE equations).

If we can solve the MFE equations exactly (as in some parts of Section 3 above) then the quality of the projection of the function $L(v)$ into S_ϕ or $S_{\alpha\beta}$ is maintained for all times. If however we solve the MFE equations by a numerical time stepping procedure the projection will be degraded by the approximation, the more so for low order integration schemes or large time steps.

The division of the method into two steps, namely, the projection of $L(v)$ into a "local" subspace at each instant of time, together with a time integration of the consequent MFE equations, is practically convenient rather than natural in any sense. The more satisfactory approach of using moving finite elements applied fully in both space and time, however, does not seem to be tractable.

In the original MFE method of Miller and his associates the time stepping was carried out using a stiff ODE solver, suitably modified. The necessity for this type of integrator came from the use of penalty functions which were used to prevent the occurrence of the two kinds of singularity ($|C| = 0$ and $|M| = 0$) mentioned in Section 4. We have already discussed how singularities arising from $|M| = 0$ can be avoided and in this section, in conjunction with the problem of time stepping, we consider how the $|C| = 0$ singularity may be treated. In this way we avoid the necessity of penalty functions and therefore of a stiff solver on these grounds.

Before considering the general problem of what scheme to use, time step restrictions etc., we look back to Section 3 where the MFE equations for the one-dimensional problem

$$u_t = -\lambda u^q u_x \qquad (q = 0,1) \qquad (6.1)$$

(λ constant) were found to be

$$\dot{a}_j = 0, \qquad \dot{s}_j = -\lambda a_j^q \qquad (6.2)$$

((3.16) and (3.21) with appropriate f and g). These are cases where an exact solution is possible but, more importantly, the nodes move along the characteristics for the problem. The characteristics here are straight lines and u is constant along them, so that an exact solution to the characteristic problem is easy, like the MFE solution. More generally there will be no such close correspondence but Morton (1982) has shown that the semi-discrete MFE equations correspond to transporting the best L_2 fit to the exact solution. This indicates that nodal velocities with components $(\dot{s}_j, \dot{a}_j)$ are tangential to characteristics.

Approximate time integration will degrade this property but if the time step is not too large we may expect the MFE solution to approximately follow characteristic paths. One consequence is that the points where characteristics cross and form shocks will be indicated by the crossing of nodal paths, i.e. node overtaking. This is consistent with equation (4.29) which shows that m increases for convex f and ~~decreases~~ for concave f. This is also the situation corresponding to one of the sources of singularity discussed before, namely, that of $|C| = 0$.

If however shocks are not expected, as in the diffusion problems of Section 5, node overtaking is a form of instability (effectively oscillations of the solution in the x (or s) direction). The solution breaks down if this happens. Special time restrictions are required in this case.
We discuss hyperbolic and diffusion equations in turn.

For hyperbolic equations there appears to be no case for using implicit methods, since there is a qualitative rationale for node overtaking. Using an explicit method, then, we admit time steps which lead to node overtaking, recognising such an occurrence (in one dimension) as indicative of the formation of a shock. If, after a time step, node k1 catches up with node k2, the nodes having amplitudes a_{k1} and a_{k2} respectively (see Fig. 6), then $\Delta_k s = 0$ and the matrix C_k and the vector $\underline{b}_k$ in (4.8) are zero, which makes $\underline{w}_k$ indeterminate. Thus, from (4.10), the two equations

$$\begin{aligned} \dot{a}_j - m_{jL}\dot{s}_j &= w_{jL} \\ \dot{a}_j - m_{jR}\dot{s}_j &= w_{jR} \end{aligned} \qquad (6.3)$$

for the node j (the coincidence of the nodes k1 and k2 with common co-ordinate $s_j = s_{k1} = s_{k2}$) are lost. The situation corresponds to singularity of C and hence of A (see (4.8) and (4.18) where $A = M^T CM$: c.f. also (1.6)) in the discussion following (4.31).

We seek two replacement equations for (6.3) on the basis that the new configuration (see Fig. 6.1(b)) moves as a shock, i.e. for the equation $u_t + f_x = 0$ there is a common shock speed (from the jump relation)

$$\dot{s}_{k1} = \dot{s}_{k2} = \frac{f(a_{k2}) - f(a_{k1})}{a_{k2} - a_{k1}} \quad . \qquad (6.4)$$

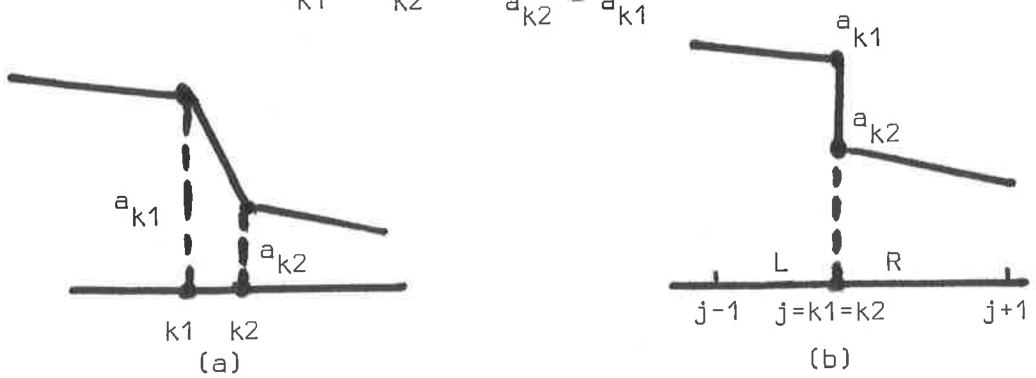


FIG. 6.1 : Formation of a shock

Replacing (6.3) by (6.4), we find that since $\dot{s}_{k1}$ is known only one equation is now needed in the element L (see Fig. 6.1(b)) to determine the w's in that element, and this comes from the other end, namely,

$$\dot{a}_{j-1} - m_{j-1,L}\dot{s}_{j-1} = w_{j-1,L} \quad . \qquad (6.5)$$

Similarly for the element R.

Thus with (6.4) and the set of equations (4.10) (without (6.3)) we can solve generally for $\dot{a}_j$ and $\dot{s}_j$. The status of (6.4) is that of an internal boundary condition.

In higher dimensions the situation is more complex and this is described in Section 9.

An exceptional result occurs when $L(u) = g(u_x)$. In this case m_k = constant for all times (c.f. (3.4), (3.10) or (4.29)) and nodes which overtake will not cause jumps, even if these are expected. The only way in which nodes can approach one another (with m_k finite) is by merging, i.e. $a_{k2} - a_{k1} \to 0$ as $s_{k2} - s_{k1} \to 0$. Consequently resolution is lost and shocks which should appear do not do so.

In this case the piecewise linear nature of the approximation is too crude and a case may be made for recovery as in Section 5, for the first time in hyperbolic equations. The piecewise constant nature of $g(v_x)$ does not represent the phenomena we are trying to describe in a sufficiently accurate way and a procedure consistent with allowing $g(v_x)$ to roam the whole of the $S_{\alpha\beta}$ space is to perform quadratic recovery on v_x before substitution into $\underline{b}$ or $\underline{g}$.

For equations of diffusion type the overtaking of nodes will generally be indicative of excessively large time steps. Unless the nodal amplitudes a_{k1} and a_{k2} are equal when overtaking takes place (merging) an unacceptable feature will be created which may make the solution go unstable or substantially lose accuracy (see below). For this reason the time step has to be controlled. Using the one-dimensional linear heat equation as a guide, the equation for the slope m is given by (5.13) which, for an appropriate recovery, is (5.18), namely

$$\frac{dm}{dt} = \frac{1}{(\Delta s)^2} (m_L + m_R - 2m) \quad . \tag{6.6}$$

Simple explicit forward time stepping yields

$$m^{n+1} = m^n + \frac{\Delta t}{(\Delta s)^2} (m_L^n + m_R^n - 2m^n) \tag{6.7}$$

$$= \left\{1 - \frac{2\Delta t}{(\Delta s)^2}\right\} m^n + \frac{\Delta t}{(\Delta s)^2} (m_L^n + m_R^n) \tag{6.8}$$

which is such that m will decrease with time if

$$\Delta t < \tfrac{1}{2}(\Delta s)^2 \quad . \tag{6.9}$$

As nodes close up Δs becomes small and the time restriction is severe, just as for explicit finite difference or fixed finite element methods. Because the implicit form of (6.6) satisfies a maximum principle, an implicit approach to (6.6) will lead to decreasing m. Overtaking then cannot take place unless nodes actually merge together (since m remains finite).

One way of carrying out an implicit time stepping in the standard MFE formulation is to solve

$$A(\underline{y}^{n+1})(\underline{y}^{n+1} - \underline{y}^n) = \Delta t \, \underline{g}(\underline{y}^{n+1}) \tag{6.10}$$

by writing $\underline{z} = \underline{y}^{n+1}$, setting up an iteration on $\underline{z}$ of the form

$$A(\underline{z}^{(k)})(\underline{z}^{(k+1)} - \underline{y}^n) = \Delta t g(\underline{z}^{(k)}) \tag{6.11}$$

and solving this for $\underline{z}^{(k+1)}$ using the conjugate gradient method (preconditioning by D^{-1}) as discussed fully in Wathen & Baines (1985). The exceptional eigenstructure of A gives very rapid convergence of the method. Larger time steps will be possible and the maximum principle operates.

For the more general equation

$$u_t = (D(u)u_x)_x \quad , \tag{6.12}$$

where no maximum principle applies in general, we can nevertheless envisage a split version of the problem in the form

$$u_t = D(u)u_{xx} \quad , \quad u_t = D'(u)u_x^2 \tag{6.13}$$

and, if $D(u)$ is positive, a maximum principle approach can be used for the first of these equations. For the second, we find that

$$\frac{dm}{dt} = m^3 \; D''(\eta) \qquad\qquad \eta \in (v_1,v_2) \tag{6.14}$$

(c.f. (3.7) and (3.8)) so that

$$-\frac{1}{m^2} = \int^t D''(\eta)\, dt. \tag{6.15}$$

If $D''(\eta)$ is small m will not change much, but if $D''(\eta)$ is positive it will increase.

The effects of the two equations (6.13) on v will often therefore be contrary to one other, the interaction being simulated by the splitting.

If $D(u) = u$, equation (6.15) gives m = constant in time: here the nodes may re-adjust but the pattern of slopes will be preserved.

Another possibility for the first of (6.13) is cubic spline recovery of u_{xx}.

Moving to the convection-diffusion equation (5.14) we can adapt the shock strategy described above using (5.15) as a guide. This equation shows that $\sum_k m_k^2$ is equal to a certain constant, in fact

$$\sum_k m_k^2 = \frac{J^{3/2}}{(12\varepsilon)^{\frac{1}{2}}} \quad , \tag{6.16}$$

where J is the jump present in the shock structure. Suppose that we carry out the MFE solution of (5.14) by splitting into

$$\text{(a)} \quad u_t = -uu_x \qquad\qquad \text{(b)} \quad u_t = \varepsilon u_{xx} \tag{6.17}$$

Suppose also that in solving (6.17(a)) we restrict the time step to be less than that which would take the solution up to the point where (6.16) was satisfied. Then, in solving (6.17(b)), using the same time step, let us use an MFE method which possesses a maximum principle for m and employ time stepping implicitly in the manner (6.10)-(6.11).

The effect is to successively steepen and diffuse the solution in such a way that $\sum m_K^2$ never exceeds the right hand side of (6.16), since this quantity will reduce during the diffusion and be restricted as it increases during the convection.

So far we have discussed restrictions on time-stepping due to physical limitations but have not discussed time-stepping for a given accuracy. This aspect cannot be seen in isolation. We need to embed the problem into the question of how many nodes should be used, how to deploy nodes initially and when and where to add or delete nodes during the evolution.

In the matter of the initial placement of nodes Herbst (1982) suggested that this should be done in such a way as to equidistribute the quantity

$$|u''|^{\frac{1}{2}} \quad . \tag{6.18}$$

The idea comes from truncation error considerations and represents an equidistribution of this error. An alternative may be to equidistribute the residual minimised in the MFE method when a best fit to $L(v)$ is found for v_t. The square of the residual concerned is

$$<v_t - L(v), \ v_t - L(v)> \tag{6.19}$$

(c.f. (1.2)) or, using (2.19),

$$\underline{w}^T C\underline{w} - 2\underline{w}^T C\underline{b} + \|L(v)\|^2 \tag{6.20}$$

with $\underline{b}$ given by (4.9). Since $\underline{w} = C^{-1}\underline{b}$ the square root of (6.20) becomes

$$\{\underline{b}^T C^{-1}\underline{b} - 2\underline{b}^T\underline{b} + \|L(v)\|^2\}^{\frac{1}{2}} \tag{6.21}$$

and this quantity, being a measure of the error in the differential equation, is a good candidate for equidistribution. Note that, unlike (6.18), the measure (6.21) is an elementwise local quantity. It evolves with the solution and is always associated with a particular element.

A possible strategy for error control is to monitor the quantity (6.21) as time-stepping proceeds, both in respect of choosing a time step and, possibly, in deciding (in combination with other residuals) on the introduction or deletion of nodes. (See also Section 10).

This procedure is akin to using the truncation error to determine the time step. To control the global error, the best fit of v to the exact solution over the whole region, it is necessary to connect the two errors. Mueller (1984) has made important strides in this direction but in one dimension only. A more direct error monitoring procedure is to concentrate attention on the global norm of v, keeping changes in this norm small in comparison with some tolerance, related to $\|u - v\|$ initially perhaps.

Other criteria for the addition and deletion of nodes are related to boundary conditions, both internal and external. This aspect of node control is discussed in the next section concerning boundary conditions in general.

7. BOUNDARY CONDITIONS

This section discusses boundary conditions. Taking the elementwise approach of Section 4 we have within each element (including those adjacent to boundaries) that (4.13) holds for the calculation of the intermediate vector $\underline{w}$. Evaluation of $\dot{\underline{y}}$ from (4.14) depends on knowing $\underline{w}$ in adjacent elements, which breaks down at boundaries. Referring back to (4.4) we see that, at the left hand boundary s_0, we have the second of (4.4) but not the first (see Fig. 7.1).

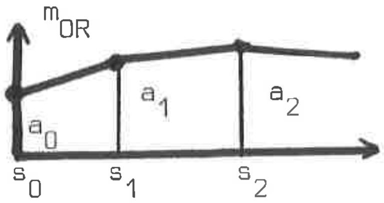


FIG. 7.1 : Boundary nodes

To find $\dot{a}_0$ and $\dot{s}_0$ from the single equation

$$\dot{a}_0 - m_{OR}\dot{s}_0 = c_{OR} \tag{7.1}$$

we need a boundary condition. The simplest to impose is the condition

$$\dot{s}_0 = 0 \tag{7.2}$$

corresponding to fixing the boundary s_0. Then a_0 is left free to find its own level and the condition models a Neumann boundary condition at s_0 as a natural boundary condition (see Wathen (1984)).

Another condition easy to impose is

$$\dot{a}_0 = 0 \tag{7.3}$$

which fixes a_0 and allows s_0 to find its own position. Such a condition is useful in modelling free boundary problems (see below). A mixed condition

$$p\dot{s}_0 + q\dot{a}_0 = 0 \tag{7.4}$$

can also be imposed if the ratio of the constants q/p is not $- m_{OR}$.

A Dirichlet condition is harder to impose in this context because it overdetermines the system. We may think of fixing the boundary s_0 by the use of a dummy element to the left of s_0 mirroring s_0s_1. Then (7.2) is replaced by

$$\dot{a}_0 + m_{0R}\dot{s}_0 = c_{0R} \tag{7.5}$$

with the same effect as (7.3). But if $\dot{a}_0 = 0$, as in a Dirichlet condition, this is inconsistent unless $c_{0R} = 0$. If c_{0R} is forced to zero then neither (7.5) nor (7.3) is consistent with (7.1) and this is a constraint on the projection (4.7) for $\underline{c}$. Thus we do not expect uniform accuracy from the MFE method near the boundaries when a Dirichlet condition is to be modelled.

Solid boundaries can be modelled by the technique (7.5) while transparent boundary conditions may be approximated by regarding the contribution c_{-1L} sent rightwards in the dummy element adjacent to the boundary to be null, i.e.

$$\dot{a}_0 - m_{0L}\dot{s}_0 = 0 \quad , \tag{7.6}$$

where m_{0L} is arbitrary in the dummy element. There are interesting comparisons here with the cell-based finite difference schemes of Roe (1981) which also make full use of the flow of information in a cell.

Finally, where the data has compact support and spreads or convects the method provides new support after a time step. To see this consider (7.3) where $a_0 = 0$ at the edge of the support s_0. After a time step s_0 will move to the new position where $a_0 = 0$ giving the new support. In this way diffusing data can be followed or a moving wave tracked.

Internal boundaries, such as the shock interface in the previous section, can be tracked by abutting problems with appropriate boundary conditions. The shock interface prescribes $\dot{s}$ and finds $\dot{a}$ at the interface but other conditions are also possible which find $\dot{s}$ and hence track the interface.

In higher dimensions boundary conditions are more complex. The basic idea of augmenting equations equivalent to (7.1) is the same but since there are at least two variants on the basic method they need to be discussed in the context of the methods themselves.

For problems of parabolic type with two space derivatives in the differential equation an extra boundary condition is supplied either at s_0 or at the other end s_{N+1}. Here the simplest case is the imposition of a Neumann condition at each end although it must be remembered that, if a recovered function is used, that too must accord with the boundary conditions.

A typical moving boundary problem would have two conditions, perhaps Dirichlet and Neumann, imposed at the same moving (but unlocated) point, s_0 say. Here we may set $\dot{a}_0 = 0$ and locate the boundary as the intersection of the new solution with the Dirichlet value.

Finally, we consider a more radical approach to Dirichlet conditions. The philosophy behind any boundary condition is that the outside world can conveniently be modelled at an interface by isolated pieces of information imposed there. Usually there is a prior assumption that the boundary is fixed. It may be that in setting up a problem the outside world can as conveniently (or more conveniently) be modelled by boundary conditions on a moving boundary. In particular a zero Dirichlet condition at a fixed boundary may be replaced by an equivalent zero condition at a boundary located by the method. Thus for moving mesh methods it could be useful to formulate the boundary conditions in a fresh way.

We turn now to systems of equations in 1-D where, owing to coupling of variables, special problems may arise in the behaviour of the components.

8. SYSTEMS OF EQUATIONS

We consider systems of equations of the form

$$u_t^{(m)} = L^{(m)}(\underline{u}) \qquad (m = 1,2,\ldots,M) \tag{8.1}$$

where $\underline{u} = [u^{(1)},\ldots,u^{(M)}]^T$. Included are the linear wave equation written as a system and the Euler equations for compressible flow, for example.

In extending the MFE ideas to systems we are faced with an immediate decision. Do we work with separate nodal coefficients and a common mesh or give each component of the system its own finite element mesh with individual nodal coefficients and co-ordinates? Where discontinuous features are expected to occur simultaneously for all components m there is a strong argument for using a common mesh. However, a nicer algebraic structure (although a nastier quadrature) is obtained if each component is given its own finite element mesh. We shall discuss both strategies, here in one dimension.

In the first of these strategies, called method (A), we seek finite element approximations $v^{(m)}$ to $u^{(m)}$ of the form

$$v^{(m)} = \sum_j a_j^{(m)} \alpha_j \tag{8.2}$$

so that

$$v_t^{(m)} = \sum_j (\dot{a}_j^{(m)} \alpha_j + \dot{s}_j \beta_j^{(m)}) \tag{8.3}$$

where

$$\beta_j^{(m)} = -v_x^{(m)} \alpha_j \tag{8.4}$$

(c.f. (2.24) and (2.23)). Also

$$v_t^{(m)} = \sum_k (w_{k1}^{(m)} \phi_{k1} + w_{k2}^{(m)} \phi_{k2}). \tag{8.5}$$

A global minimisation of the weighted residual

$$\sum_m \theta_m \, \| v_t^{(m)} - L^{(m)}(\underline{v}) \|^2 \quad , \tag{8.6}$$

where $\underline{v} = [v^{(1)},\ldots,v^{(m)}]^T$, over $\dot{a}_j^{(m)}$ and s_j leads to the equations

$$\langle \alpha_i,\ v_t^{(m)} - L^{(m)}(\underline{v})\rangle = 0 \qquad (m = 1,2,\ldots,M) \tag{8.7}$$

$$\sum_m \theta_m \langle \beta_i^{(m)},\ v_t^{(m)} - L^{(m)}(\underline{v})\rangle = 0 \tag{8.8}$$

$(i = 1,2,\ldots,N)$, the latter also written as

$$\sum_m \theta_m \langle v_x^{(m)} \alpha_j,\ v_t^{(m)} - L^{(m)}(\underline{v})\rangle = 0. \tag{8.9}$$

The system (8.7) with (8.9) can be written

$$A(\underline{y})\dot{\underline{y}} = \underline{g}(\underline{y}) \tag{8.10}$$

where now $\underline{y} = [a_1^{(1)}, a_1^{(2)}, \ldots, a_1^{(M)}, s_1; \ldots; a_N^{(1)}, a_N^{(2)}, \ldots, a_N^{(M)}, s_N]^T$ (8.11)

and $A(\underline{y})$ is block tri-diagonal with $(M+1) \times (M+1)$ blocks of the form

$$\begin{bmatrix} \langle \alpha_i, \alpha_j \rangle I_M & \langle \underline{\beta}_i, \alpha_j \rangle \\ \langle \alpha_i, \underline{\theta\beta}_j^T \rangle & \langle \beta_i^{(M)}, \beta_i^{(M)} \rangle \end{bmatrix} \tag{8.12}$$

where $\underline{\beta}_i = (\beta_i^{(1)}, \ldots, \beta_i^{(M)})^T$, $\underline{\theta\beta}_j = (\theta_1\beta_j^{(1)}, \ldots, \theta_M\beta_j^{(M)})^T$ and I_M is the $M \times M$ identity matrix. The right hand side of (8.10) is the obvious extension of (1.5).

If the weights θ_m are chosen such that $\theta_1 = 1$, $\theta_m = 0$ $(m \neq 1)$ then the first component $v^{(1)}$ drives the nodes. In that case the single component MFE method can be used to find $\dot{a}_j^{(1)}$, $\dot{s}_j$ and these values fed back into the rest of (8.10) to obtain $\dot{a}_j^{(m)}$ $(m \neq 1)$, the latter operation involving a traditional FFE mass matrix.

Some care is needed over the choice of θ_m. For instance, consider the wave equation written as a system

$$\left.\begin{aligned} u_t^{(1)} &= u_x^{(2)} \\ u_t^{(2)} &= u_x^{(1)} \end{aligned}\right\} \tag{8.13}$$

If $\theta_1 = 1$, $\theta_2 = 0$, the first of (8.13) is used to solve for $\dot{a}^{(1)}$ and $\dot{s}$ as a single component MFE system. But $v_x^{(2)}$ in the approximate form of this equation is piecewise constant, and using e.g. (4.29) it follows that

$$\frac{d}{dt} (v_x^{(1)}) = 0 \quad . \tag{8.14}$$

Now it is possible that $u^{(1)} = 0$ initially but is driven away from zero (in the exact solution) by $u^{(2)}$, as in the case of the solution

$$\left.\begin{aligned} u^{(1)} &= \cos x \sin t \\ u^{(2)} &= -\sin x \cos t . \end{aligned}\right\} \tag{8.15}$$

Because of (8.14) this will never happen in the approximate solution. For this reason a weighting which relies on one component is not recommended. The same difficulty occurs with the Euler equations if the density ρ is used to drive the nodes, since the density equation

$$\rho_t + m_x = 0 , \tag{8.16}$$

where m is the momentum, also has the form of the first of (8.13).

An alternative version of method (A) which leads effectively to an optimal choice of θ_m is based on elementwise considerations and is as follows. A straightforward application of the elementwise best fit procedure (see Section 4) gives

$$C^{(m)} \underline{w}^{(m)} = \underline{b}^{(m)} \tag{8.17}$$

(c.f. (8.5) and (4.8) where

$$b_{k1}^{(m)} = \langle \phi_{k1}^{(m)} , L(\underline{v}) \rangle , \quad b_{k2}^{(m)} = \langle \phi_{k2}^{(m)} , L(v) \rangle \tag{8.18}$$

and $C^{(m)}$ is the block 2x2 diagonal matrix with blocks

$$\begin{bmatrix} \langle \phi_{k1}^{(m)} , \phi_{k1}^{(m)} \rangle & \langle \phi_{k1}^{(m)} , \phi_{k2}^{(m)} \rangle \\ \langle \phi_{k2}^{(m)} , \phi_{k1}^{(m)} \rangle & \langle \phi_{k2}^{(m)} , \phi_{k2}^{(m)} \rangle \end{bmatrix} \tag{8.19}$$

Inversion of (8.19) is trivial in general: the difficulty comes in retrieving $\dot{a}_j^{(m)}$, $\dot{s}_j$ from the resulting information, since the matrix M of (4.14) is now rectangular diagonal with blocks

$$\begin{bmatrix} 1 & 0 & -(v_x^{(1)})_{jL} \\ 1 & 0 & -(v_x^{(1)})_{jR} \\ 0 & 1 & -(v_x^{(2)})_{jL} \\ 0 & 1 & -(v_x^{(2)})_{jR} \end{bmatrix} \tag{8.20}$$

(c.f. (4.11)). Thus (4.14) gives sets of four equations for only three unknowns $\dot{a}_j^{(1)}$, $\dot{a}_j^{(2)}$, $\dot{s}_j$. The best solution in the L_2 norm is to solve

$$M^T M\underline{\dot{y}} = M^T\underline{w} \tag{8.21}$$

rather than (4.14).

Inversion of (8.21) is easy since the left hand side matrix is block diagonal with 2 x 2 blocks as in the single component method. Note that parallelism will generally not occur unless both components go collinear simultaneously. The overall effect is of a double projection, first that of $L^{(m)}(\underline{v})$ into $S_\phi(m)$ space, giving $w_{k1}^{(m)}$, $w_{k2}^{(m)}$ and then that of $\underline{w}$ into $S_{\alpha\beta}(m)$ space (a smaller space), ultimately providing a best approximation of $u_t^{(m)}$ to $L^m(\underline{u})$ in a particular sense. Call this method (A^1).

A similar device is used in Section 9 to deal with a mismatch in dimensions of function spaces when higher dimensional physical spaces are considered.

Turning now to the other strategy for systems, called method (B), which uses separate finite element bases for each component, the corresponding finite element approximation to (8.2) is

$$v^{(m)} = \sum a_j^{(m)}\alpha_j^{(m)} \tag{8.22}$$

so that

$$v_t^{(m)} = \sum_j \left(a_j^{(m)} \alpha_j^{(m)} + \dot{s}_j^{(m)} \beta_j^{(m)}\right) \tag{8.23}$$

where

$$\beta_j^{(m)} = - v_x^{(m)} \alpha_j^{(m)} \ . \tag{8.24}$$

Also

$$v_t^{(m)} = \sum_k \left(w_{k1}^{(m)} \phi_k^{(m)} + w_{k2}^{(m)} \phi_k^{(m)}\right) \ , \tag{8.25}$$

different components requiring completely separate bases.

As a result of the isolation of each basis we easily form the MFE equations

$$A^{(m)}(\underline{y}^{(m)})\, \dot{\underline{y}}^{(m)} = \underline{g}^{(m)} \ , \tag{8.26}$$

where $A^{(m)}$ and $y^{(m)}$ are exactly as for the single component method for each component m while $\underline{g}^{(m)}$ has components

$$\begin{aligned} g_{2j}^{(m)} &= \langle \alpha_j^{(m)}, L(\underline{v}) \rangle \\ g_{2j+1}^{(m)} &= \langle \beta_j^{(m)}, L(\underline{v}) \rangle \end{aligned} \tag{8.27}$$

Thus the only new feature is the quadrature in (8.27) which links the components through the evaluation of $\underline{g}^{(m)}$. This presents no difficulty in one dimension since the elements can be subdivided suitably with quadrature over each sub-element, but in higher dimensions the subdivision is more tricky.

There are no other obvious difficulties in method (B) except when it comes to shock modelling. A feature of a shock is that every component shocks at once and, because of the approximations involved (both in space and in time) this is not guaranteed for the moving element approximation. A device has therefore to be used to arrange that when a shock appears it is simultaneous in the appropriate components. This involves some post-processing and additional approximation.

Methods (A) and (B) have been applied to the well-known Sod problem (1979) by Baines & Wathen (1985).

This brings us to the end of this sequence of topics arising from the MFE method with the exception of phenomena present in higher dimensional problems. Aspects of these have been touched upon in previous sections but we now devote a complete section to some of the difficulties peculiar to the implementation of the MFE method in two dimensions or more.

9. HIGHER DIMENSIONS

We consider now the form of the MFE method in higher dimensions. In preceding sections we have often embarked upon a general description of a particular topic, subsequently specialising to the one-dimensional case. Thus many of the basic points have already been covered and we merely recall them here. We shall take examples from two dimensions but the principles are the same for two or more dimensions.

In Section 1 we stated the basic MFE equations as derived by Miller, namely (1.3), and noted that they can be written

$$A(\underline{y})\dot{\underline{y}} = M^T CM\dot{\underline{y}} = N^T Q^T CQN\dot{\underline{y}} = \underline{g}(\underline{y}) \tag{9.1}$$

where N $(= Q^{-1}M)$ is rectangular block diagonal and Q is a permutation matrix. In Section 2 several basic formulae were given, primarily (2.3), (2.4) and (2.14) for v, v_t and $\underline{\beta}_j$, respectively. Taking an elementwise formulation the corresponding formulae in Section 2 were (2.15), (2.16) and (2.17).

In Section 3 we considered "exact" solutions of (1.1), i.e. where the operator function $L(v)$ lies in the space $S_{\alpha\underline{\beta}}$ containing v_t. In particular (3.1) included the right hand side of the generalised inviscid Burgers' equation

$$u_t = -uu_x - uu_y \tag{9.2}$$

and hence an exact match between v_t and $L(v)$ is possible in this case. We obtain from (2.22) and (2.23)

$$\sum_j [\dot{a}_j\alpha_j + \dot{\underline{s}}_j \bullet \underline{\beta}_j] = \sum_j [\dot{a}_j - \dot{\underline{s}}_j \bullet \underline{\nabla}_{\sigma_j} v]\ \alpha_j = -\sum_j \begin{bmatrix}1\\1\end{bmatrix} \bullet \underline{\nabla}_{\sigma_j} va_j\alpha_j \tag{9.3}$$

from which it follows that

$$\dot{a}_j = 0, \qquad \dot{\underline{s}}_j = a_j\begin{bmatrix}1\\1\end{bmatrix} . \tag{9.4}$$

From an elementwise point of view, consider the generalisation of (3.4)

$$u_t = uf(\underline{\nabla}u) + g(\underline{\nabla}u) \tag{9.5}$$

giving (c.f. (3.5)

$$\sum_k \sum_i w_{ki}\phi_{ki} = \sum_k \sum_i \{a_{ki} f(\underline{\nabla}_{\underline{\sigma}_{ki}} v) + g(\underline{\nabla}_{\underline{\sigma}_{ki}} v)\}\phi_{ki} \tag{9.6}$$

so that

$$w_{ki} = \dot{a}_{ki} - \underline{\dot{s}}_{ki} \cdot \underline{\nabla}_{\underline{\sigma}_{ki}} v = a_{ki} f(\underline{\nabla}_{\underline{\sigma}_{ki}} v) + g(\underline{\nabla}_{\underline{\sigma}_{ki}} v) \quad (i = 1,2,\ldots,I) \tag{9.7}$$

(see (2.20)). Dropping the suffix k this reads

$$\dot{a}_i - \underline{\dot{s}}_i \cdot \underline{\nabla}_{\underline{\sigma}_i} v = a_i f(\underline{\nabla}_{\underline{\sigma}_i} v) + g(\underline{\nabla}_{\underline{\sigma}_i} v) \quad . \tag{9.8}$$

Take a node i and choose $\sigma_i = \sigma$ to be the co-ordinate along the axis which is "coplanar" with the axis of a (i.e. v) and the normal to the linear approximation v. In the (σ,a) plane let θ be the angle, in two dimensions, shown in Fig. 9.1,

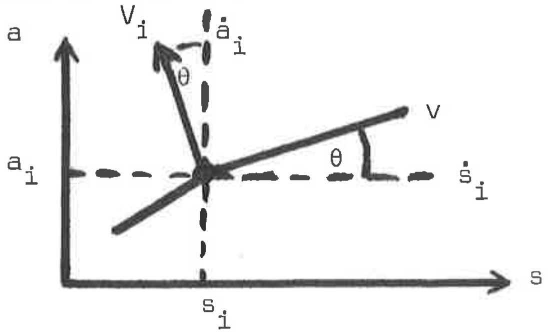


FIG. 9.1 : Gradient angle and elementwise velocity

for which

$$\tan\theta = |\underline{\nabla}_{\underline{\sigma}} v| \tag{9.9}$$

with the appropriate sign. Then the component of velocity of the element corner i in the direction normal to v is

$$\dot{V}_i = \dot{a}_i \cos\theta - \dot{s}_i \sin\theta \tag{9.10}$$

which, by comparison with (9.8) and (9.9) gives,

$$\dot{V}_i = [\dot{a}_i f(\underline{\nabla}_\sigma v) + g(\underline{\nabla}_\sigma v)] \cos \theta_i \tag{9.11}$$

$$= [\dot{a}_i f(\underline{\nabla}_\sigma v) + g(\underline{\nabla}_\sigma v)]/[1 + (\underline{\nabla}_\sigma v)^2]. \tag{9.12}$$

Thus we can deduce from (9.8) the velocities of the element corners normal to the linear solution v. To obtain the nodal velocities we have simply to put together the element corner velocities from adjacent elements.

In one dimension two element corner velocities from adjacent elements give two nodal velocity components $\dot{a}_i$, $\dot{s}_i$ uniquely, but in higher dimensions this is not so, since generally a node has more adjacent elements than degrees of freedom. As a result the elementwise approach leads to overdetermined nodal velocity components and a uniquely determined method can only be obtained by some form of projection (see below).

In Section 4 we discussed minimisation of the residual of (1.1) in $S_{\alpha\beta}$ and projection of $L(v)$ into S_ϕ and showed that these were the same in one dimension. By virtue of the above argument we no longer have equivalence of these approaches in higher dimensions. Let us consider them in turn. If we take the minimisation of the residual of (1.1) first, we have as the residual squared

$$\|v_t - L(v)\|^2 = \langle v_t - L(v),\ v_t - L(v)\rangle$$

$$= \langle v_t, v_t\rangle - 2\langle v_t,\ L(v)\rangle \tag{9.13}$$

apart from terms which do not involve v_t. Using (2.8) we obtain

$$\underline{\dot{y}}^T A\, \underline{\dot{y}} - 2\underline{y}^T\, \underline{g} \tag{9.14}$$

where we have used the $\underline{y}$, A and $\underline{g}$ of Section 1. Minimisation of (9.14) over $\underline{\dot{y}}$ yields the standard form of the MFE equations

$$A\underline{\dot{y}} = \underline{g} \quad . \tag{9.15}$$

As shown in Wathen & Baines (1985), this may also be written (c.f. (4.18))

$$M^T C M \dot{\underline{y}} = \underline{g} \tag{9.16}$$

where C is square and block diagonal and M is rectangular and similar to a block rectangular diagonal matrix. The non-square nature of M reflects the non-uniqueness of the elementwise approach, as we shall see below, and prevents straightforward inversion of $M^T CM$.

Consider now the elementwise approach. We begin by solving (4.13), namely,

$$C\underline{w} = \underline{b} \tag{9.17}$$

which is straightforward since C is square block diagonal. The difficulty only arises when we attempt to solve (4.14), i.e.

$$M\dot{\underline{y}} = \underline{w} \tag{9.18}$$

since this is an overdetermined set of equations for $\dot{\underline{y}}$. Thus for the elementwise approach to give a unique solution we must seek a $\underline{w}$ which lies in the range space of M.

One option available is to carry out a constrained minimisation of (9.13) where the solution is forced to lie in the range space of M. Let the columns of the matrix Z span the orthogonal complement of the range space of M. Then

$$Z^T M = M^T Z = 0 \tag{9.19}$$

and the constraint we impose is that

$$Z^T \underline{w} = 0 \quad . \tag{9.20}$$

We now use the elementwise residual in the form

$$\underline{w}^T C\underline{w} - 2\underline{w}^T b \tag{9.21}$$

(omitting terms which do not involve v_t) and apply (9.20) as a constraint.

Taking $\underline{\lambda}$ as a vector of Lagrange multipliers we have to minimise

$$\tfrac{1}{2}\underline{w}^T C\underline{w} - \underline{w}^T\underline{b} + \underline{\lambda}^T Z^T \underline{w} \tag{9.22}$$

which gives

$$\left.\begin{aligned} C\underline{w} - \underline{b} + Z\underline{\lambda} &= 0 \\ \text{and} \qquad Z^T\underline{w} &= 0 \end{aligned}\right\} \tag{9.23}$$

which in turn gives (from (9.19))

$$\left.\begin{aligned} M^T(C\underline{w} - \underline{b}) &= 0 \\ Z^T\underline{w} &= 0 \end{aligned}\right\} \tag{9.24}$$

the first of which provides

$$M^T CM\underline{\dot{y}} = \underline{g} \tag{9.25}$$

and brings us to the result of the nodewise approach. This analysis shows how the projection of $L(v)$ into S_ϕ has to be constrained so that it lies in $S_{\underline{\alpha\beta}}$.

Another way of obtaining the same equations is to accept the solution of (9.17) and do a least squares projection of equation (9.18) having first preconditioned the equation with the matrix $C^{\frac{1}{2}}$. We then minimise

$$\|C^{\frac{1}{2}}M\underline{\dot{y}} - C^{\frac{1}{2}}C^{-1}\underline{b}\| \tag{9.26}$$

in a least squares sense (over $\underline{\dot{y}}$), giving

$$(C^{\frac{1}{2}}M)^T(C^{\frac{1}{2}}M)\underline{\dot{y}} = (C^{\frac{1}{2}}M)^T C^{-\frac{1}{2}}\underline{b} \tag{9.27}$$

or

$$M^T CM\underline{\dot{y}} = M^T\underline{b} = \underline{g} \quad , \tag{9.28}$$

as before.

This is an example of applying a general preconditioning matrix P to (9.18) and minimising in a least squares sense. The minimisation is then of

$$\|P(M\underline{\dot{y}} - C^{-1}\underline{b})\| \tag{9.29}$$

and the result is

$$M^T P^T PM\underline{\dot{y}} = M^T P^T C^{-1}\underline{b} \quad . \tag{9.30}$$

If P is I, we obtain alternative MFE equations

$$M^T M \dot{\underline{y}} = M^T C^{-1} \underline{b} \tag{9.31}$$

which are readily solved since $M^T M$ is a square block diagonal matrix (3 x 3 blocks in two dimensions).

The difference between solving (9.28) and (9.31) is the difference between the preconditioner $P = C^{\frac{1}{2}}$ and $P = I$ in (9.29). Since C is block diagonal with the blocks proportional to the area or volume of the elements, the use of the preconditioner $P = C^{\frac{1}{2}}$ weights the larger elements more heavily with the smaller elements playing little part in the minimisation. (Note that $C^{-1}\underline{b}$ in (9.29) is independent of the size of element).

There is of course no difference in the methods in one dimension (where everything is square and the preconditioner has no effect), but in higher dimensions (9.31) is certainly easier to solve, giving

$$\dot{\underline{y}} = (M^T M)^{-1} M^T C^{-1} \underline{b} \quad . \tag{9.32}$$

Note that, as remarked before, the elementwise approach using $\underline{b}$ gives more information that the nodewise approach using $\underline{g}$ in these circumstances, as evidenced from

$$M^T \underline{b} = \underline{g} \tag{9.33}$$

with M not square.

A similar analysis to the above is applicable to the projection argument in Section 8 where the matrix M in the equivalent equation to (9.18) in the case of a system of equations in one dimension is not square. Different preconditionings prior to a least squares solution of the equation yield different MFE equations, including both (9.25) and (9.31) (the same as (8.15)).

A technical difficulty, so far ignored, is the similarity transformation between M and its block rectangular diagonal counterpart N (see Wathen & Baines (1985)). With the nodewise approach it is not trivial in higher dimensions to switch from elementwise numbering (to set up (9.17)) to nodewise numbering (for (9.18) or its equivalent). The permutation matrix needed is called Q here and M uses elementwise numbering while N uses nodewise numbering (see Wathen & Baines (1985)). With

$$M = QN \tag{9.34}$$

it makes sense to use $P = Q^{-1}$ in (9.29) rather than $P = I$ since this immediately gives

$$N^T N\underline{\dot{y}} = N^T C^{-1}\underline{b} \tag{9.35}$$

and of course since P is simply a permutation there is no effect on the least squares minimisation in (9.29).

The MFE equations (9.35) demonstrate that this method is local in the same way that the one-dimensional method was local, only links with adjacent elemets playing a part. Had we preconditioned with $C^{\frac{1}{2}}Q^{-1}$, however, we would have had

$$N^T Q^T CQN\underline{\dot{y}} = \underline{g} \tag{9.36}$$

and the rows and columns of C when permutated under the Q's would provide wider links between different parts of the solution and make for a less local method. This is likely to be an advantage for diffusion problems, of course, so the two approaches should perhaps be tailored to the type of problem.

It is clear that the matrix $N^T N$ of (9.35) will suffer from the phenomenon of parallelism in the same way as N. There are substantial problems with N when it loses rank (see Wathen & Baines (1985)) since this may happen in different ways. The remedy is the same as in one-dimension, namely to remove the equations causing the parallelism and solve the resulting system, re-introducing the absent nodes in a suitable way at the end. Some care is required with the larger blocks occurring in N (or $N^T N$) in higher dimensions: it is safest to transform the local system to upper triangular form and reduce the system on that basis. This avoids problems of ill-conditioning which can arise if an arbitrary equation is left out.

In Section 4 we also discussed conservation properties of the MFE method. The corresponding properties, for the unconstrained nodewise approach which gives (9.25), are

$$\frac{d}{dt}\left[\int_{\Omega} v d\ \Omega\right] = \int_{\Omega} L(v) d\ \Omega \tag{9.37}$$

(c.f. (4.22)) and

$$\frac{d}{dt}\left[\frac{1}{2}\int_{\Omega} v^2 d\Omega\right] = \int_{\Omega} vL(v) d\Omega \tag{9.38}$$

(c.f. (4.24)), where Ω is the (fixed) domain involved.

For the elementwise approach the minimisation of $v_t - L(v)$ over the variables $\underline{c}$, which leads to Galerkin equations (c.f. (4.1) and (4.7)).

$$<\phi_{ki}, v_t - L(v)> = 0, \tag{9.39}$$

also yields (9.37) and (9.38) since the ϕ_{ki} (as well as the α_j) are a partition of unity.

In Section 5 the treatment of second order terms in (5.1) is either governed by (5.9) or by a generalisation of the recovery technique discussed subsequent to (5.9).One way of generalising the technique is to seek a quadratic function Q_1 which matches v_x (a piecewise constant function) at suitable points. In two dimensions there are four unknown coefficients in Q_1 (which is of the form

$$Q_1(x,y) = ax^2 + bxy + cx + d\ \ ) : \tag{9.40}$$

we can take the slope at the centroid together with either the means of the slopes v_x in the elements around each of the three corner nodes or the means of the slopes across each of the three mid-points of the sides. Levine [1984] has pointed out enhanced convergence properties for the latter for fixed elements. Working with a Hermite cubic recovery in the same situation involves finding a cubic of the form

$$W_1(x,y) = ax^3 + bx^2y + cx^2 + dxy + ex + f \tag{9.41}$$

with six coefficients. Here we can take the three nodal values and three values of the slope in either of the two manners above. Spline fits, either pointwise or in a least squares sense, are also feasible.

In Section 6 an important question was the modelling of shocks. In one dimension this is signalled in a hyperbolic problem by node overtaking causing the slope of an element to go infinite. In many dimensions the same criterion of an infinite slope can be used and this will occur when the size of an element goes to zero after a time step. (In two dimensions a triangular element stands up on its end). The technique is discussed in Wathen & Baines (1985) and here we confine the discussion to general principles using the elementwise approach.

Except when C is singular the solution of (9.17) for each element gives the vector $\underline{w}$ which in turn provides the quantities on the left hand side of (9.8), i.e. the components of the velocity of the corner nodes of the element in a direction normal to the element. The use of this information to drive the nodes is then an overdetermined problem and as we have seen (9.35) may be used to obtain a least squares solution. Similarly, with shocked elements present (together with their specified shocked velocity components in the direction of the shock movement) the problem of finding the nodal velocities is an overdetermined one. It is now necessary to preserve the shock speeds and exclude them from the least squares solution, but in all other respects (9.35) provides a solution. There are technical problems of course but the principles are clear.

Initial placement of nodes in higher dimensions may be generalised from the one-dimensional case as that initial arrangement which equidistributes the quantity $\{\nabla^2 u\}^{\frac{1}{2}}$. How this is to be actually done in practice is not obvious, however.

Error control is as discussed at the end of section 6, including the use of (6.21).

Finally, on the subject of boundary conditions the main new feature in higher dimensions is the occurrence of boundaries in the form of a line or surface. Taking the two dimensional case as example, it is clear that three components of velocity are needed at every node to determine its motion (see Fig. 9.2, in which the points B_1 B_2 B_3 are boundary nodes and I_1 I_2 are interior nodes).

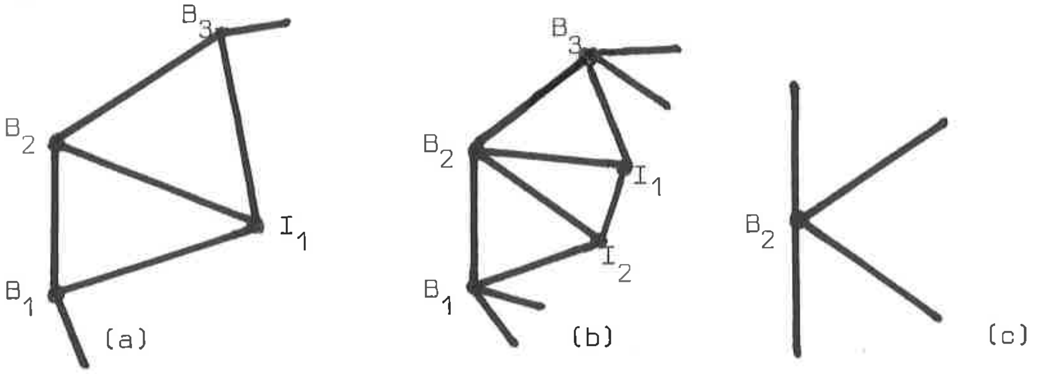


FIG. 9.2 : Boundary and adjacent nodes

The elementwise MFE method provides two velocities at B_2 in Fig. 9.2(a) (from the triangles B_2 B_3 I_1 and B_2 B_1 I_1) and hence to determine the behaviour of B_2 one boundary condition is required. On the other hand, for the pattern of elements shown in Fig. 9.2(b) three velocities are provided from the triangles adjoining B_2, and the behaviour of B_2 is determined. No boundary condition is apparently needed in this case ! (but see below). If a boundary condition is to be imposed in the case shown in Fig. 9.2(b), or more than one condition is to be imposed as in Fig. 9.2(a), constraints are put on the elementwise projection.

The cases in which precisely one condition may be imposed are shown in Fig. 9.2(a), when $\dot{a}$ is set at the point B_2 and the nodal co-ordinates are left free, and in Fig. 9.2(c) (which is the same as Fig. 9.2(a) but with a straight boundary) when one speed is set (perpendicular to the boundary) and the other movements left free.

A transparent boundary condition can be conveniently modelled by accepting that no information on the movement of a node, say B_2 in Fig. 9.2, comes from outside the system. Thus the behaviour of B_2 is determined entirely from elements present in the system. This will in general result in movement of the node B_2.

The nodewise MFE method gives rather different rules. Here the quantity $\dot{a}$ and two speeds (along the axes) are determined by the algorithm, rather than three nodal corner velocities. If the speeds are set to zero and $\dot{a}$ is left free a natural Neumann boundary condition is modelled. A Dirichlet conditions involves overwriting a and the nodal positions as in one dimension. With a straight boundary as in Fig. 9.2(c) one speed (normal to the boundary) may be set and the other allowed to be determined.

In discussing systems of equations in higher dimensions the question of giving each component its own moving mesh or using a common mesh again arises. The arguments for and against are as in Section 8, except that there are substantial technical difficulties in carrying out the right hand side quadratures when several meshes are present. The situation is sketched in Fig. 9.3, where functions on the dotted mesh have to be

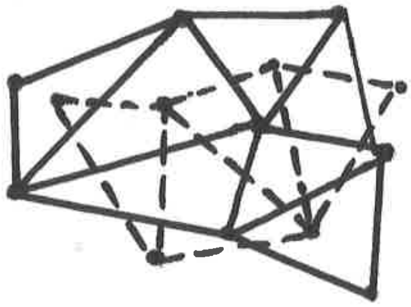


FIG. 9.3 : Meshes for two different components

integrated over elements on the full-line mesh. Since these are non-smooth functions over such elements some care is necessary over their quadrature.

In the final section of this report we summarise what we consider the most important points raised in previous sections.

10. CONCLUSION

In this last section we summarise some of the theory given earlier and bring together some of the more important results so as to be able to make recommendations on the use of the Moving Finite Element method or its derivatives.

The most striking outcome of this study is that the basic method in one dimension, as described in Section 1, can be built up from the solution of diagonal matrices with 2x2 blocks. We also note that there is a simple extension to two dimensions involving diagonal matrices with 3x3 blocks. A second far-reaching fact is that the nodal velocities can be found from "element velocities" which arise immediately from straight line best fits to the (approximate) driving function in the differential equation.

To summarise these results, for the differential equation

$$u_t = L(u) \quad , \tag{10.1}$$

and the piecewise linear continuous approximation v to u, we evaluate $L(v)$ and calculate the best straight line fit to $L(v)$ in each element. Using the L_2 norm this gives in each element an approximation to v_t, namely,

$$v_t = w_{k1}\phi_{k1} + w_{k2}\phi_{k2} \quad , \tag{10.2}$$

where

$$C_k \begin{pmatrix} w_{k1} \\ w_{k2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{3} \end{pmatrix} \Delta s \begin{pmatrix} w_{k1} \\ w_{k1} \end{pmatrix} = \begin{pmatrix} b_{k1} \\ b_{k2} \end{pmatrix} \tag{10.3}$$

and

$$b_{k1} = \langle \phi_{k1}, L(v) \rangle \quad , \qquad b_{k2} = \langle \phi_{k2}, L(v) \rangle \quad . \tag{10.4}$$

Inversion of (10.3) gives

$$\begin{pmatrix} w_{k1} \\ w_{k2} \end{pmatrix} = \frac{1}{\Delta s} \begin{pmatrix} 4 & -2 \\ -2 & 4 \end{pmatrix} \begin{pmatrix} b_{k1} \\ b_{k2} \end{pmatrix} \tag{10.5}$$

for each element, and to get from (10.2) (over all elements) to the form

$$v_t = \sum (\dot{a}_j\alpha_j + \dot{s}_j\beta_j) \tag{10.6}$$

we have to solve

$$M_j \begin{bmatrix} \dot{a}_j \\ \dot{s}_j \end{bmatrix} = \begin{bmatrix} 1 & -m_{j-\frac{1}{2}} \\ 1 & -m_{j+\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \dot{a}_j \\ \dot{s}_j \end{bmatrix} = \begin{bmatrix} w_{j-\frac{1}{2},2} \\ w_{j+\frac{1}{2},1} \end{bmatrix} \quad , \tag{10.7}$$

where

$$m_{j+\frac{1}{2}} = \frac{a_{j+1} - a_j}{s_{j+1} - s_j} \quad . \tag{10.8}$$

Inversion of (10.7) (when $m_{j-\frac{1}{2}} \neq m_{j+\frac{1}{2}}$) gives

$$\begin{bmatrix} \dot{a}_j \\ \dot{s}_j \end{bmatrix} = \frac{-1}{\Delta m_j} \begin{bmatrix} -m_{j+\frac{1}{2}} & m_{j-\frac{1}{2}} \\ -1 & 1 \end{bmatrix} \begin{bmatrix} w_{j-\frac{1}{2},2} \\ w_{j+\frac{1}{2},1} \end{bmatrix} \quad . \tag{10.9}$$

Combining (10.9) with (10.5) gives the nodal velocities.

Even without the inversion of (10.7) however we can interpret elementwise pairs of equations of (10.7), namely,

$$\begin{aligned} \dot{a}_j - m\,\dot{s}_j &= w_{j+\frac{1}{2},1} \\ \dot{a}_{j+1} - m\,\dot{s}_{j+1} &= w_{j+\frac{1}{2},2} \quad , \end{aligned} \tag{10.10}$$

where $m = m_{j+\frac{1}{2}}$, as velocities of the piecewise linear segment in the element $j+\frac{1}{2}$ as follows. If θ is the angle between the solution v and the axis,

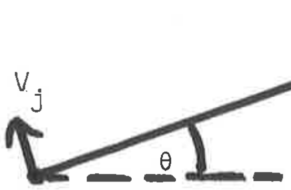


FIG. 10.1 : Elementwise corner velocities

from (10.8) we have

$$m_{j+\frac{1}{2}} = \tan\theta = m \tag{10.11}$$

and (10.10) can be written

$$\left.\begin{aligned} \dot{a}_j \cos\theta - s_j \sin\theta &= \frac{1}{\sqrt{(1+m^2)}}\, w_{j+\frac{1}{2},1} \\ \dot{a}_{j+1} \cos\theta - \dot{s}_{j+1} \sin\theta &= \frac{1}{\sqrt{(1+m^2)}}\, w_{j+\frac{1}{2},2} \quad . \end{aligned}\right\} \tag{10.12}$$

The left hand sides of (10.12) are the velocities V_j, V_{j+1} (due to the single element only) of the ends of the element in Fig. 10.1 at right angles to the element. The complete velocity of a node will be a combination of two such element end velocities from adjacent segments.

These end velocities can be written

$$\begin{pmatrix} V_j \\ V_{j+1} \end{pmatrix} = \frac{1}{(1\ +\ m^2)} \begin{pmatrix} w_{j+\frac{1}{2},1} \\ w_{j+\frac{1}{2},2} \end{pmatrix} = \frac{1}{\Delta s\sqrt{(1+m^2)}} \begin{pmatrix} 4 & -2 \\ -2 & 4 \end{pmatrix} \begin{pmatrix} b_{j+\frac{1}{2},1} \\ b_{j+\frac{1}{2},2} \end{pmatrix} \tag{10.13}$$

using (10.5) or, by (10.4),

$$\begin{pmatrix} V_j \\ V_{j+1} \end{pmatrix} = \frac{1}{\Delta s\sqrt{(1+m^2)}} \left< \begin{pmatrix} 4\phi_{j+\frac{1}{2},1} - 2\phi_{j+\frac{1}{2},2} \\ -2\phi_{j+\frac{1}{2},1} + 4\phi_{j+\frac{1}{2},2} \end{pmatrix} \quad , \quad L(v) \right> \tag{10.14}$$

Subtraction and addition of components gives

$$V_{j+1} - V_j = \frac{6}{\Delta s\sqrt{(1+m^2)}} \left<\phi_{j+\frac{1}{2},2} - \phi_{j+\frac{1}{2},1},\ L(v)\right> \tag{10.15}$$

and

$$V_{j+1} + V_j = \frac{2}{\Delta s\sqrt{(1+m^2)}} \left<\phi_{j+\frac{1}{2},1} + \phi_{j+\frac{1}{2},2},\ L(v)\right> \tag{10.16}$$

$$= \frac{2}{\Delta s\sqrt{(1+m^2)}} \left<1, L(v)\right> = \frac{2}{\Delta s\sqrt{(1+m^2)}} \int_{s_j}^{s_{j+1}} L(v)dx. \tag{10.17}$$

(10.17) gives the normal velocity of the mid-point of the segment in Fig. 10.1. In particular, if

$$L(v) = -\, f_x \tag{10.18}$$

(10.17) yields

$$\tfrac{1}{2}(V_{j+1} + V_j) = \frac{-1}{\sqrt{(1+m^2)}} \frac{\Delta f}{\Delta s} = \frac{-m}{\sqrt{(1+m^2)}} \frac{\Delta f}{\Delta a} = - \frac{\Delta f}{\Delta a} \sin\theta . \qquad (10.19)$$

Thus the speed of the mid-point of the segment in Fig. 10.1 in the direction normal to the segment is consistent with the average wave speed in the element.

A more significant companion to (10.17) is obtained by subtracting the equations (10.10) giving,

$$\frac{dm}{dt} = \frac{1}{\Delta s} (w_{j+\frac{1}{2},2} - w_{j+\frac{1}{2},1}) \qquad (10.20)$$

$$= \frac{6}{\Delta s} <\phi_{j+\frac{1}{2},2} - \phi_{j+\frac{1}{2},1}, L(v)> \qquad (10.21)$$

$$= \frac{6}{\Delta s} \int_{s_j}^{s_{j+1}} \frac{(2x - \overline{s_j + s_{j+1})}}{(s_{j+1} - s_j)} L(v)\, dx . \qquad (10.22)$$

If $L(v)$ is as in (10.18) we obtain

$$\frac{dm}{dt} = \frac{-6}{(\Delta s)^2} \int_{s_j}^{s_{j+1}} (2x - \overline{s_j + s_{j+1}}) f_x \, dx \qquad (10.23)$$

$$= \frac{-6}{(\Delta s)} \left[f_{j+1} + f_j - \frac{2}{\Delta s} \int_{s_j}^{s_{j+1}} f \, dx \right]$$

$$= \frac{-12}{\Delta s} (\bar{f} - \hat{f}) \quad , \qquad (10.24)$$

where $\bar{f} = \tfrac{1}{2}(f_j + f_{j+1})$, $\hat{f} = \dfrac{1}{\Delta s} \displaystyle\int_{s_j}^{s_{j+1}} f \, dx$, as in Section 4.

Equation (10.24) gives the result that the rate of change of slope of the solution in an element is proportional to the second derivative of f. Put another way the solution segment rotates in response to the local convexity of f.

The results (10.19) and (10.24) go a long way towards explaining why the method is so good for one-dimensional scalar conservation laws.

Even in higher dimensions the result (10.17) holds and, if

$$L(v) = -\,\mathrm{div}\,\underline{f} \tag{10.25}$$

we find that the normal velocity of the mid-point of a linear element is

$$\overline{V}_k = \frac{-\sin\theta}{\Delta_k}\int_{\text{element } k} \mathrm{div}\,\underline{f}\,d\tau = \frac{\sin\theta}{\Delta_k}\,\mathcal{F}_k\,, \tag{10.26}$$

where

$$\tan\alpha = |\underline{\nabla} v| \tag{10.27}$$

and $\mathcal{F}_k$ is the inward flux of $\underline{f}$ through the boundary of the element. If that flux were to drive a local wave velocity in a direction coplanar with the normal to the element then the velocity $\overline{V}_k$ would be consistent with this wave velocity.

It appears that a result corresponding to (10.24) for the rate of change of maximum slope can also be generated but this has not been done here.

Having discussed the central features of the space approximation, we now summarise some of the other points made.

For certain equations $L(v)$ is already a piecewise linear discontinuous function and the space approximation is exact. This allows certain non-linear equations to be solved very accurately.

There are two circumstances in which the procedure outlined above does not go through. One is when $\Delta s = 0$ and the matrix C_k of (10.3) cannot be inverted. The other is when $m_{j-\frac{1}{2}} = m_{j+\frac{1}{2}}$ and the matrix M_j of (10.7) cannot be inverted. The former is considered below. The latter is circumvented by temporarily fixing the node responsible and relocating it in an averaged position after the approximation has been carried out.

In the elementwise approach, however, the way in which this is to be done is not obvious. Clearly the offending equation is (10.7) but, whereas in the nodewise approach the presence of parallelism leads to two equations which are identical and hence consistent, the pair of equations (10.7) become inconsistent (different right hand sides) when parallelism occurs. Thus we cannot simply delete one equation and set $\dot{s} = 0$ to obtain a solution to the reduced system since there is in fact no solution to (10.7) in this case.

However we can conveniently return to the α,β basis by combining the equations (10.3) in staggered pairs giving blocks of the form

$$\begin{bmatrix} 1 & 1 \\ -m_{jL} & -m_{jR} \end{bmatrix} \begin{bmatrix} \frac{1}{6}\Delta_L s & \frac{1}{3}\Delta_L s & 0 & 0 \\ 0 & 0 & \frac{1}{3}\Delta_R s & \frac{1}{6}\Delta_R s \end{bmatrix} \begin{bmatrix} w_{L1} \\ w_{L2} \\ w_{R1} \\ w_{R2} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 \\ -m_{jL} & -m_{jR} \end{bmatrix} \begin{bmatrix} b_{jL} \\ b_{jR} \end{bmatrix} \tag{10.28}$$

where L,R refer to the elements to the left and right of node j. As in the nodewise approach in the event of parallelism we keep the first of these equations and replace the second by $\dot{s}_j = 0$. This gives

$$\frac{1}{6}\Delta_L s\, w_{L1} + \frac{1}{3}\Delta_L s\, w_{L2} + \frac{1}{3}\Delta_R s\, w_{R1} + \frac{1}{6}\Delta_R s\, w_{R2} = b_{jL} + b_{jR} \tag{10.29}$$

and since now $\dot{s}_j = 0$ and

$$\left.\begin{aligned} w_{L2} &= \dot{a}_j - m_L\dot{s}_j = \dot{a}_j \\ w_{R1} &= \dot{a}_j - m_R\dot{s}_j = \dot{a}_j \end{aligned}\right\} \tag{10.30}$$

with $m_L = m_R$, we have from (10.29)

$$\frac{1}{6}\Delta_L s\, w_{L1} + \frac{1}{3}(\Delta_L s + \Delta_R s)\dot{a}_j + \frac{1}{6}\Delta_R s\, w_{R2} = b_{jL} + b_{jR} \tag{10.31}$$

which yields

$$\begin{aligned} \dot{a}_j &= \{b_{jL} + b_{jR} - \frac{1}{6}\Delta_L s(w_{L1} + \Delta_R s\, w_{L2})\}/\{\tfrac{1}{3}(\Delta_L s + \Delta_R s)\} \\ \dot{s}_j &= 0 \end{aligned} \tag{10.32}$$

as the solution for the reduced equation for a parallel node.

The null space is spanned by the vector $[m \quad 1]^T$ as before (where $m = m_L = m_R$) and an appropriate multiple of this vector can be added to satisfy an externally imposed averaged velocity or position.

If several nodes are parallel at once a number of equations of the type (10.29) will arise and it may be necessary to solve a tri-diagonal system for the unknown $\dot{a}_j$ if the relevant j's are adjacent to one another.

For diffusion equations $L(v)$ exists only in the sense of distributions and evaluation of $\underline{b}$ in (10.4) needs some care. Recovery has been proposed in Section 5 as a mechanism for evaluation of these quantities. In fact it is the evaluation of the integrals in (10.17) and (10.22) which are required and with a recovery mechanism the behaviour of m and $\frac{1}{2}(V_j + V_{j+1})$ can readily be predicted.

Time stepping in the present method is grafted on to the space approximation. It is at present the weakest aspect of the method and not much is known about the choice of time step. Both this aspect and the related problem of node insertion and deletion need more attention. It has however been argued in Section 6 that simple explicit time integration is sufficient.

There is one situation where a time stepping strategy is clear and that is when C_k goes singular as a result of node overtaking in a hyperbolic problem. In this circumstance the differential equation is abandoned in favour of the corresponding jump conditions which may easily be applied. We note that, as $m \to \infty$, the average speed of (10.19) tends smoothly to the shock speed while $\frac{dm}{dt}$ of (10.24) tends smoothly to zero.

Algorithms for accuracy are not well developed but in view of the simplicity and fast nature of the method we can afford to be generous in taking a trial and error approach. A possible algorithm is as follows:-

```
  DELT = DELT0
1 CALL MFE (A,S,DELT)        yielding  A1, S1
  DELT = 0.5*DELT
  CALL MFE (A,S,DELT)        yielding  AH, SH
  CALL MFE (AH,SH,DELT)      yielding  A2, S2
  IF  $\|(A2,S2) - (A1,S1)\| < TOL$ GOTO 2
  SET A1 = AH, S1 = SH,GOTO 1
2 DELT = DELT0, A = A1, S = S1,GOTO 1
```

This algorithm compares the result of one MFE step with that of two MFE half-steps and continues having the step until the difference between the two is acceptable.

With the elementwise view of MFE taken here boundary conditions are imposed in a consistent way which is somewhat different to that of the traditional approach. In particular a radical view of Dirichlet conditions is taken and a possible way of doing transparent conditions is proposed in Section 7.

Systems of equations in one dimension (with a common mesh) and the extension to higher dimensions have a common feature. In each case there is a mismatch when solving (**4.14**) between the size of the vector $\underline{w}$ and that of $\dot{\underline{y}} = \{(\dot{a}_j,\dot{s}_j)^T\}$. In both cases a least squares approach to the solution of (10.7) is proposed, although this takes the method away from the traditional approach. For systems in one dimension with individual meshes for individual components the difficulties are those of preserving simultaneous features such as shocks and the technical problems of carrying out quadrature on unstructed meshes in many dimensions.

To conclude, we have discussed in some detail in this report recent advances and some alternatives in the MFE method and sketched its potential and possible limitations. The elementwise view developed throughout is based on an observation of Herbst [1982] and Morton [1982]. Although there are still several unanswered questions and more work needs to be done to make the method reliable, the approach here has already shown significant results (Wathen (1984), Johnson (1984), Wathen, Baines & Morton (1984), Wathen & Baines (1985), Baines & Wathen (1985)). Moreover in the form sketched at the beginning of this section the program which implements the method in one dimension can be run on a small micro, which shows in a dramatic way how far the method has come from its original implementation.

REFERENCES

1. BAINES, M.J. & WATHEN, A.J. (1985) Moving Finite Element Modelling of Compressible Flow. Numerical Analysis Report 4/85, University of Reading.

2. HERBST, B.M. (1982) Moving Finite Element Methods for the solution of Evolution Equations. Ph.D. Thesis, University of the Orange Free State.

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

4. LEVINE, N.D. (1984) Pointwise logarithm-free error estimates for finite elements on linear triangles. Numerical Analysis Report 6/84, University of Reading.

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

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

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

8. MORTON, K.W. (1982) Private communication.

9. MORTON, K.W. (1983) Finite element methods for non-self-adjoint elliptic and hyperbolic problems: Optimal approximations and optimal recovery techniques. Numerical Analysis Report 7/83, University of Reading.

10. MUELLER, A. (1983) Ph.D. Thesis, University of Texas at Austin.

11. ROE, P.L. (1981) Approximate Riemann Solvers, Parameter Vectors and Difference Schemes, J. Comput. Phys. 43, 357.

12. SOD, G.A. (1978) A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws. J. Comput. Phys. 27, 1-31.

13. WATHEN, A.J. (1982) Moving Finite Elements and Applications to Some Problems in Oil Reservoir Modelling. Numerical Analysis Report 4/82, University of Reading.

14. WATHEN, A.J. (1984) Ph.D. Thesis, University of Reading.

15. WATHEN, A.J. & BAINES, M.J. (1985) On the Structure of Moving Finite Element Matrices, IMA J. Num. An. (to appear): also Numerical Analysis Report 5/83, University of Reading,

16. WATHEN, A.J., BAINES, M.J. & MORTON, K.W. (1984) Moving Finite Element Methods for the Solution of Evolutionary Equations in One and Two Dimensions, Proc. of MAFELAP Conference, Brunel, (to appear).