Computational Considerations For Moving Finite Element Methods

by

A. N. Hrymak, A. W. Westerberg

EDRC-06-10-86^

September 1986

Computational Considerations for Moving Finite Element Methods

Andrew N. Hrymak
Dept. of Chemical Engineering
McMaster University
Hamilton, Ontario, Canada L8S 4L7

Arthur W. Westerberg
Dept. of Chemical Engineering
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

Moving finite element methods allow grid points to follow steep fronts and provide accurate solutions with less computational expense. There are problems that must be recognized in the implementation of this numerical solution technique, but they can be systematically analyzed and overcome. Singularity or near singularity of the solution equation set provides specific information about the node motion which can be used to improve the solution. A front tracking and heat conserving solution is presented for a moving boundary problem.

Introduction

In solving sets of partial differential equations numerical methods such as finite differences or finite elements calculate the approximate value of the field variables on a discrete set of points in the computational domain of space and time. Adaptive gridding techniques allow movement of the grid points to capture physical phenomena that are on a much different scale than the overall domain. In parabolic or hyperbolic problems such phenomena include shocks, steep gradient regions, and phase change fronts. In elliptic problems the grid must be sufficiently fine to capture rapid changes in the field variables due to cracks, or high gradient regions, or be able to satisfy extra boundary conditions to determine a free surface.

In this work we discuss some features of a particular class of adaptive gridding methods, the moving finite element method (MFEM), which was originally developed by Miller et al. (1981; Gelinas et al, 1981). The solution procedure is suited to parabolic or hyperbolic problems requiring a fine grid that must follow a feature moving in time through the overall domain or where there are domain changes in time. MFEM is characterized by simultaneously calculating the node motion and the field variable approximation to minimize the least square error residual of the governing PDE's with respect to field variable and nodal coordinate trajectories. We shall present a brief overview of the 1-D method and discuss some of the details that are important in successful implementation of the method. Specific problem areas that will be addressed include: discontinuous inner products due to low order approximations, singularity and near singularity in node motion, initial grid point placement, choosing integrators for the ODE set, and how to handle moving boundary problems. Further background material is available in the references.

The MFE Method

Consider a system of p PDE's in one spatial dimension,

$$\frac{\partial u^e}{\partial t} = \dot{u}^e = L^e(u) \qquad t > 0 \qquad (1)$$

$$e = 1.2....p$$

We define an approximation to u^, v^, defined as a piecewise continuous function,

$$V^e = \sum_{j=i}^{n} a_J \, a'_J \qquad (2)$$

where aj^(t) is the value of v^(xj.t). Nodal amplitudes and nodal coordinates XJ are both functions of time, t.

The temporal variation of v is given by,

$$\dot{v}^e = \sum_{j=1}^{n} k_j^e a_j + i_j \dot{p}_j \qquad (3)$$

where,

$$a_j = \begin{cases} \dfrac{x - x_{j-1}}{x_j - x_{j-1}} & x_{j-1} \le x < x_j \\ \dfrac{x_{j+1} - x}{x_{j+1} - x_j} & x_j \le x < x_{j+1} \\ 0 & \text{elsewhere} \end{cases}$$

and,

$$\beta_j^\ell = \begin{cases} -\dfrac{(a_j^\ell - a_{j-1}^\ell)}{(x_j - x_{j-1})} a_j = -m_j^\ell a_j & x_{j-1} \le x < x_j \\ -\dfrac{(a_{j+1}^\ell - a_j^\ell)}{(x_{j+1} - x_j)} a_j = -m_{j+1}^\ell a_j & x_j \le x < x_{j+1} \\ 0 & \text{elsewhere} \end{cases}$$

for linear basis functions, as shown in Figure 1.

Ue minimize the least square functional

$$\sum_{\ell=1}^{P} (\dot{v}^\ell - L^\ell(v^\ell))^2 \tag{4}$$

with respect to a/ and x* for (= I....p and i = l,...,n and a coupled set of ODE's result.

$$A(y)\dot{y} = g$$

$$\mathbf{y} = (a_1 \; a_2 \; \dots \; a_n \; \beta_1^1 \; \beta_2^1 \; \dots \; \beta_n^1 \; \dots \; \beta_1^P \; \beta_2^P \; \dots \; \beta_n^P)$$

with submatrices of size [p 4-1 lx[p 4-1],

$$\begin{bmatrix} (a_1, a_1) & & & (a_1, p_1^1) \\ & (a_i, a_j) & & (a_i, \beta_j^2) \\ & & (a_n, a_n) & (a_n, p_n^P) \\ (\beta_1^1, a_1) & (p^2, a_2) & (p^P, a_n) & \sum_{\ell=1}^{P} (p^\ell, p^\ell) \end{bmatrix} \tag{6}$$

with subdiagonal, diagonal, and superdiagonal blocks given by j = i-l, i, and i 4-1 respectively.

The corresponding vector g segment is given by the inner product components,

$$\left\{ (a_j, f^1(v^1)), \dots, (a_i, L^P(v^P)), \sum_{\ell=1}^{P} (\beta_i^\ell, L^\ell(v^\ell)) \right\} \tag{7}$$

Note (,) denotes an inner product and the values of these products have been catalogued (Gelinas et al, 1961). A number of examples have been solved using the MFEM. In 1-D these include: viscous and inviscid Burgers' equations, quench fronts, Stefan problems (enthalpy formulation), hyperbolic color equation, Buckley-Leverett equation, Dwyer-Sanders Flame model, gas dynamic shocks, elastic-plastic deformations, and convective-diffusion equations (Gelinas et al, 1981, 1981; 1982; Johnson, 1984; Miller et al, 1981; Wathen, 1984).

Difficult Inner Products

Often L^{ℓ} (v^{ℓ}) contain $v^{\wedge\ell}$ and the inner products of ($0\pounds^{\ell}, v_{xx}^{\ell}$) are undefined in the usual sense, because the basis functions $0\pounds^{\ell}$ are discontinuous at node i, and v_{xx}^{ℓ} is formally a delta function at each node when a linear approximation basis is used within an element; low order basis functions will not give smooth estimates to higher order derivative operators. Using a higher order element would eliminate the difficulties in the inner products, but, if we wish to use linear basis functions with second order spatial derivative operators for example, then the inner products must be approximated. The main disadvantage in going directly to higher order basis functions is that there will be many more inner products to evaluate in the ODE coupling matrix.

One way of handling these terms would be to use mollification, or put more simply, treating the functions as limits of smoother functions. For example, the basis function Bj^{ℓ} would take a mean value of

$$-\frac{1}{2}(m_i^{\ell} + m_{i+1}^{\ell})$$

at node i and v_{xx}^{ℓ} is a delta function of weight $m^{\wedge}+i^{\ell}\sim m^{\wedge\ell}$. >

For the inner product (aj, v_{xx}^{ℓ}) one may use integration by parts. Note that the two basis functions are related as defined previously. In one dimension this

limit approach is quite feasible but in higher dimensions, there are difficulties in interpreting the different limits. The approach also does not lend itself to practical generalization to higher order basis functions with even higher order differential operators.

Another approach is to recognize that the solution must be smooth, and one may "recover" a smoother approximation for the second derivative, W_{xx}, from the MFE solution (Johnson, 1984). Local higher order polynomial approximations of the field variable and its gradient are made and differentiated to produce piecewise linear, but discontinuous, approximations of the second derivatives. For example, Johnson has shown that a local Hermite cubic defined by,

$$W^{H^\ell} = \begin{cases} a_i^\ell & x = x_i \\ a_{i+1}^\ell & x = x_{i+1} \end{cases}$$

$$W_x^{H^\ell} = \begin{cases} \frac{1}{2}(m_i^\ell + m_{i+1}^\ell) & x = x_i \\ \frac{1}{2}(m_{i+1}^\ell + m_{i+2}^\ell) & x = x_{i+1} \end{cases}$$

gives exactly the same inner products as the mollification approach of Miller et al.

$$(W_{xx}^H, \alpha_i) = (v_{xx}, \alpha_i) = m_{i+1} - m_i$$

$$(W_{xx}^H, \beta_i^p) = (v_{xx}, \beta_i^p) = -\frac{1}{2}(m_i^p + m_{i+1}^p)(m_{i+1} - m_i)$$

This makes intuitive sense because the mollification process is essentially smoothing the derivatives between elements by taking an average value. The Hermite basis is explicitly requiring a derivative smoothing. The Hermite form is appealing in higher dimensions since it is relatively easy to use higher dimension Hermite elements to define the inner products.

Another local approximation would define W_x on a quadratic,

$$W_x^{Q^\ell} = \begin{cases} \frac{1}{2}(m_i^\ell + m_{i+1}^\ell) & x = x_i \\ m_{i+1}^\ell & x = \frac{1}{2}(x_i + x_{i+1}) \\ \frac{1}{2}(m_{i+1}^\ell + m_{i+2}^\ell) & x = x_{i+1} \end{cases}$$

yielding the following inner products.

$$(W_{xx}^{Q^\ell}, \alpha_i) = \frac{1}{12}[(m_i^\ell + 10\, m_{i+1}^\ell + m_{i+2}^\ell) - (m_{i-1}^\ell + 10 m_i^\ell + m_{i+1}^\ell)]$$

$$(W_{xx}^{Q^\ell}, \beta_i^P) = -\frac{1}{12}[m_{i+1}^P(m_i^\ell + 10\, m_{i+1}^\ell + m_{i+2}^\ell) - m_i^P(m_{i-1}^\ell + 10 m_i^\ell + m_{i+1}^\ell)] + \frac{1}{2}(m_i^\ell + m_{i+1}^\ell)(m_{i+1}^P - m_i^P)$$

It has been reported that there is little difference in accuracy between the two approximations when used on the viscous Burgers' equation. There is some difference in the maximum time step allowed between the two methods, but the results do not favor either of the two approximations.

Therefore, the basic idea in defining the inner products, for second order differential operators using a linear approximation, is to use some type of smoothing; either use the limiting process of mollification, or explicity define a local approximation to the second order derivative terms using the solution at the nodal points. The latter approach is preferred. Higher order elements would require quadrature or more involved integration for the inner products and would make the basis functions of the nodal motion trajectories more complicated.

Singularity of the ODE Set

The MFE formulation is an optimization problem and if a value of a variable

does not affect the objective then the necessary conditions for the minimization will become singular.

If one is using a linear approximation basis the equation set is singular whenever the slope of the approximation is the same in two adjacent elements or whenever the solution is flat. The weighting functions, ϕ_4 and ϕ_i become linearly dependent at node i. Node i may lie anywhere between its neighbors and give the same residual value.

The original method of dealing with weighting function singularity was to involve penalty functions that essentially required the singular node to move at a weighted average of its neighbors' velocities (Hrymak et al, 1985). Another approach is to note that the singular node is a redundant equation. Here one can let the node motion be explicitly determined by its nonsingular neighbors' velocities rather than by using an orthogonal error criterion. Vathen suggests that, at any point in time when the ODE set becomes singular, the singular node motion equation can be replaced by an explicit node velocity averaging fomula and later, when it becomes nonsingular, replaced again by the error criterion. Hrymak et al. have shown that for the viscous Burgers' equation one need have only a subset of nodes move independently throughout the time domain and achieve good results with less computation time.

The local time decision of Vathen's approach requires a means for detecting singularity. Also, several nodes' equations may become singular at any time step. Of course, singular value decomposition (SVD) is one alternative in determining the true rank of the system, but it is expensive.

A simpler solution, and one we are considering, is to monitor the pivoting involved in solving the set of ODE's and, when a pivot becomes too small for a row an alternate node motion expression can be used. (The system is positive

definite in the nondegenerate case so the pivots would be the diagonal elements normally).

Another simple method would be to scan the slopes of the field variable approximation on the mesh, and note when the difference between two adjacent element slopes is less than some arbitrary quantity. If such a small slope difference is found, the node common to the elements may have its node motion temporarily tied to that of its neighbors.

A final alternative is to detect singular node equations and remove the nodes altogether since they are redundant. An advantage of this approach is that there would be fewer equations to integrate. However, if there are features developing in the solution, this approach would necessitate some kind of monitoring of error in the individual elements to determine when a node should be reinserted to prevent loss of accuracy. Reinsertion could be done simply by using the superelement approach of Hrymak et al. where only a subset of nodes is given independent movement throughout the time domain. As new features develop in the solution nodes can be "turned on" and given independent movement.

The important point is that an understanding of the nature of the singularity allows one to see alternate approaches to the specification of the grid motion. Because penalty functions introduce extra parameters and increase the stiffness of the ODE set, they should be less desirable as an approach to overcoming singularity.

Near Nodal Movement Singularity

It is important to understand what the MFEM is really doing. It is actually determining the evolution of the fitting of the solution with piecewise linear functions and thus a strong minimum, with respect to the grid trajectories, would

be observed with a solution containing a steep gradient or shock condition (or discontinuity). In diffusive problems the minimum residual, with respect to node position, is quite weak and there is little or no advantage in determining the node movement in this manner since the optimal grid is fairly evenly spaced (given no steep solution gradients due to point sources or material discontinuities). Additional grid points may be needed for more accuracy, but they do not have to be given independent movement.

We have observed this effect in the simulation of isothermal trace component pressure-swing adsorption where the initial adsorption transient required a fine or moving grid. Later times are essentially at steady state dominated by diffusion and adsorption (reaction) terms, and no grid refinement was necessary (Hrymak, 1984). The basic equations used were of the form,

$$\frac{\partial c}{\partial t} = \frac{1}{Pe}\frac{\partial^2 c}{\partial x^2} - \frac{\partial c}{\partial x} - f_1(c, q)$$

$$\frac{\partial q}{\partial t} = f_2(c, q)$$

where c is the concentration of the trace component in the gas phase, and q is the concentration of the trace component in the solid phase. Initially, the bed is free of the trace component, and the feed enters the bed, creating a Steep profile that is governed basically by convective effects in the initial transient. Then the diffusion and adsorption forces predominate. The solution is a very smooth profile after the initial transient, and an equidistributed mesh would be adequate. Using MFEM the nodes convected with the steep front in the initial transient leaving the upstream portion of the bed bereft of nodes. We found that the best solution was to remove independently moving nodes, leaving

only one free node for example. For the trace component case, the standard Galerkin finite elements proved to be a superior way of solving the system. However, this example did point out that there is the danger of instability and unnecessary expense in having an adaptive mesh for a relatively smooth problem. Note that one could do a singular value decomposition to test whether the grid movement equations are necessary, but observing the contributions from reaction and diffusive terms relative to convective terms will be a good first estimator.

The pressure swing adsorption simulation points out the necessity for a strategy with regard to moving grids. The initial transient and convective forces combined with a steep front moved all the grid points downstream, but also required the movement of nodes to allow an accurate solution with few nodes. However, there were no transient forces available to redistribute the nodes when a smooth solution developed. MFEM is formulated with nodal trajectories as degrees of freedom. One method of solving such a problem would be to use MFEM during the initial transient, stop and redistribute the nodes with respect to an error criterion, project the old solution onto the new grid and then use a fixed grid method. Future work involves a comparison of adaptive methods that use equations for the evolution of the nodal positions for many time steps (such as MFEM) and methods that move the grid points by essentially fitting the solution after prescribed time steps.

To detect a near singularity, similar tests could be used as in the truly singular case. For example, one could use a rough pivot test in the solution of the implicit ODE set, with the additional provision for a testing order. The center of the domain could be tested first for near singularity, and then the tests continued in a tree pattern which halves the remaining subdomains, until additional independent nodes create a nearly singular set of equations to solve.

Initial Node Placement - It Makes a Difference

Uhen solving the simple transient convective-diffusion equation,

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial x^2} \quad **$$

$$dx, 0) = o$$

$$c(0_f t) = 1$$

$$c(\infty, t) = 0$$

using finite difference or finite element methods one usually sets the value c = 1 at x ss 0_f and lets c = 0 for t = 0 for all Xj. The familiar sloping step then forms and moves from x = 0 to x = infinity with height 1. However, when these initial conditions were used with MFEM a ramp formed and did not approximate the solution at all. One must start the solution with a step inside the domain $x\pounds\{0_f\sim\}$ for the node trajectories to be correct.

A simple hand analysis may be performed on the hyperbolic limit of the above equation

$$\frac{\partial c}{\partial t} = -Pe\frac{\partial c}{\partial x}$$

leading to the usual characteristic solution v± = 0, x± = Pe, where v± is the approximated value of c at node x$ as in Figure 2. Let c be approximated by three linear elements. The MFEM equations are, for node i.

$$\begin{bmatrix} \frac{\Delta x_i}{6} & \frac{-\Delta v_i}{6} & \frac{\Delta x_i}{3} & \frac{-(\Delta v_i+\Delta v_{i+1})}{3} & \frac{\Delta x_{i+1}}{6} & \frac{-\Delta v_{i+1}}{6} \\ -\Delta v_{i-1} & m_i \Delta v_i & -(\Delta v_i+\Delta v_{i+1}) & m_i \Delta v_i + m_{i+1} \Delta v_{i+1} & -\Delta v_{i+1} & m_{i+1} \Delta v_{i+1} \end{bmatrix} \begin{Bmatrix} \dot{v}_i \\ \dot{x}_i \end{Bmatrix}$$

$$= \begin{Bmatrix} \frac{1}{2}(\Delta v_i + \Delta v_{i+1}) \\ -Pe/2(m_i \Delta v_i + m_{i+1} \Delta v_{i+1}) \end{Bmatrix}$$

$$m_i = \frac{v_i - v_{i-1}}{x_i - x_{i-1}} \qquad \Delta x_i = x_i - x_{i-1} \qquad \Delta v_i = v_i - v_{i-1}$$

$$i = 0.1, 2, 3$$

let Pe = 1 and consider two cases (both with fixed boundaries at x = 0 and x = 1).

Case i) the initial step is within the computational domain

$v_0=1$ $v_1=1$ $v_2=0$ $v_3=0.$

$x_0=0$ $x_1=0.01$ $x_2=0.02$ $x_3=1.$

and the nodal amplitude and nodal positions at i = 0 and i = 3 do not change with time. Substituting these values into the MFEM equations results in,

$\dot{v}_1=0$ $\dot{v}_2=0$

$\dot{x}_1=1$ $\dot{x}_2=1$

which is precisely the analytical solution and is shown in Figure 2. As an aside, note that the exact solution may be determined by an ODE integration method in one step for any time until the step encounters the right boundary.

Now, consider another case, using a more traditional initial value set.

Case ii)

$v_0=1$ $v_1=0$ $v_2=0$ $v_3=0.$

$x_0=0$ $x_1=0.01$ $x_2=0.02$ $x_3=1.$

Substituting these values into the MFEM equations two things happen. First, the equation set is singular; $\dot{x}_2$ is undetermined because the slopes are the same in the adjacent elements. From our earlier discussion, we note that the node motion may be determined by another criterion. Also, we find

$\dot{v}_1=0$ $\dot{v}_2=0$ $\dot{x}_1=1.5$

Thus, the solution forms a ramp with a node velocity of 1.5 as in Figure 3. Similar results are seen in cases i) and ii) regardless of position of x_1 and x_2 initially. In both cases the non-singular part of the coupling matrix is positive-definite; therefore, some minimum is being calculated. However, we see in case i) that the trajectories are free to minimize the residual error, while

in case ii) the trajectory of the corner is constrained by being on a fixed boundary. This example highlights a very important point about the moving finite element method. One must remember the method is not minimizing with respect to absolute nodal positions, but rather with respect to nodal trajectories in time given an initial grid layout.

Time Integrators

Most of the work done with MFEM has involved the use of stiff ODE solvers. However, in many cases there is no necessity for a stiff solver, if no penalty functions are used. An explicit Euler ODE method works quite well as reported by Wathen (1984) and Johnson (1984). We have found that LSODI, an implicit variable-order and variable time step ODE integrator, does work quite well, however. Stiff solvers are favored if a penalty function approach is used to control singularities in the MFE equations because of the disparate time constants introduced into the system. Recognizing that singular equations may be removed allows simpler ODE solvers to be used. Obviously, an explicit Euler method for solving the ODE set would be much faster per time step than a predictor-corrector approach and should be used unless a higher order approximation to the field variable can drastically reduce the number of time steps. This issue is still under research because the problems solved thus far have not proved conclusive in favor of either approach for general application.

Following Moving Domains

The example problems solved with MFEM have shown considerable promise for solutions which contain shocks and steep gradients which move in time through the domain. However, it would be useful to allow the spatial domain itself to change

in time. Because MFEM requires many fewer nodes than other fixed grid methods one must worry about accuracy and conservation properties. In this section, the grid movement equations are not used to minimize an error residual, but rather to follow boundary movement which is a function of time and field variables.

An example problem with a moving domain is the one-phase Stefan problem, for example, ice growing smoothly into a liquid medium with density Jb^, heat capacity $c_{\pm 9}$ and thermal conductivity K±

$$p.c. \frac{\partial T_i}{\partial t} = K. \frac{\partial^2 T_i}{\partial x^2}$$

i = 1 solid
= 2 liquid

with a surface at x=0

$$T!(O_f t) = 0 \qquad T_2(-,t) = V$$

at t=0, x>0 is liquid, with boundary conditions

(Dirichlet) $T = T_m$ at $x = X(t)$

(Stefan) $L \frac{dX}{dt} \hat{n} = \{(K \nabla T)_f - (K \nabla T)^\wedge \hat{n}$ at $x = X(t)$

where T_m is the melting temperature, L is the latent heat of solidification, and X(t) is the phase boundary position. There is an analytical similarity solution available in (Crank, 1984).

Let $V=T_m$ which implies T2=V (liquid at the melting point). We use the physical properties of Lynch that c=0.62 cal/°C/cm^3, T_m=10°C, L=17.68 cal/cm^3, and K=0.0096 cal/cm/s/°C giving the analytical boundary motion trajectory,

$$X(t) = 0.09880t^{1/2}$$

There are many numerical methods to solve the Stefan problem, however, we

show that the ttFEM approach provides a natural framework for such a problem, and more importantly is heat conserving. This observation derives from the recent work of Lynch (1985) which also uses deforming elements but decouples the grid and nodal amplitude calculations. One must be careful in determining the boundary movement because using a simple gradient calculation, i.e. making a first order estimate of the boundary flux to directly determine movement, leads to degradation in the residual error order with respect to mesh spacing. The approximation method may have second-order error with respect to the mesh spacing in the domain interior. However, if a gradient calculation is required on the boundary, and only a first order gradient approximation is used, then the entire solution will have a first-order error unless the entire Galerkin form is used which involves heat capacity terms.

In a framework where we do not have two degrees of freedom per node (in MFEM there are the trajectories of nodal amplitude and nodal position) there is a problem due to the Dirichlet boundary condition on the moving interface. If the condition is enforced by removing the Galerkin equations, for example, then the Stefan condition requires differentiation of the temperature field. If the gradient is done with respect to linear elements, there is a numerical heat imbalance leading to a first-order error in the mesh.

With MFEM, there is no problem since both Dirichlet and Stefan boundary conditions can be simultaneously used. The MFEM equations with respect to T nodal amplitude trajectories are derived as usual; however, we will specify the node motion through the Stefan condition. For simplicity, interior node motion is

$$\dot{x}_i = \left(\frac{i}{N}\right)\dot{x}_N$$

(since the transients in temperature die away quickly and the dominating time variable is the interface position there is no advantage in having independent internal phase node movement) where $\dot{x}_{JJ}$ is the velocity of Nth node on the phase change interface. The value of $\dot{T}_{JJ}$ is simply,

$$T = T_m \quad => \quad f_M = 0$$

$$\text{at } x = X$$

by the Dirichlet condition ($T_{JJ} = T_m$). The temperature is at the melting temperature at the interface. We do, however, have the Nth node amplitude equation,

$$\frac{\Delta x_{N-1}}{6}\dot{T}_{N-1} - \frac{\Delta T_{N-1}}{6}\dot{x}_{N-1} + \frac{\Delta x_N}{3}\dot{T}_N - \frac{\Delta T_N}{3}\dot{x}_N = \frac{K}{c}(a_i, T_{xx})$$

and

$$\frac{K}{c}(a_i T_{**}) = -\frac{K}{c}((a_i)_x, T_*) + \frac{K}{c}\int_{Jg} a_i T_x \, ds$$

using integration by parts. We substitute the Stefan condition into the surface integral

$$KT_x = L\frac{dx_N}{dt}$$

and the inner product is easily calculated as

$$-\frac{K}{c}((a_i)_x, T_x) = \frac{K}{c}(m_{i+1} - m_i)$$

Thua the flhal equation, in the MFE format is,

$$\frac{A x_{N-1}}{6}\dot{T}_{*N-1} - \frac{\Delta T_i}{6}\dot{x}_{N-1} + \frac{Aic_N}{3}\dot{T}_{Nr} - \left[\frac{/*T_N}{V\ 3} + \frac{LV}{c}\right]\dot{x}_N = -\frac{K A T_N}{c\ A_x N}$$

The alternative form, using the more traditional ideas, would be simply

$$\dot{T} = 0$$

$$\dot{x}_N = \frac{K}{L}\frac{*T_N}{A_{x_N}}$$

which leads to a nonconserving solution* with error decreasing in a first-order manner with mesh size (Lynch and Sullivan, 1985).

Both formulations were integrated from a starting time of 0.01 s to 1 s and the resulting solid position $\hat{X}$ was compared to the analytical solution X in Table 1. The results were consistent with those of Lynch. The difference between our approach and that of Lynch lies in the implementation and the results for this simple problem are expected to be the same. Note that even with only 5 nodes the solution for the interface position is very good, and the errors are consistently an order of magnitude less than the non-conserving form.

The decision of not directly differentiating the field variable approximation to determine the domain motion can be thought of in the context of essential and natural boundary conditions. The Dirichlet boundary condition is an essential boundary condition and must be satisfied. The MFEM equations with inner products involving the node amplitude basis functions, 04, involve conservation properties and therefore must also remain but their boundary integrals contain the natural boundary conditions which in this case is the Stefan condition. The formulation is quite natural given the additional degree of freedom at each node. Note that we have not limited the MFEM method in any way. One could have some other steep gradient phenomena occuring in one of the phases and the nodes could track such features, as shown in previous MFEM work.

Implementation

The methods described are being implemented in a code for one dimensional problems with any number of partial differential equations. Each of the problems addressed thus far is being systematically handled in coding the algorithm. Since the differences between problems involves only the right hand side inner

products, a number of common operators inner products have been provided for the user *(e.g.* convection, diffusion, reaction). Dirichlet, Neumann and Stefan boundary conditions are all easily available.

The fact that MFEM is actually doing a least squares fit would imply an "optimal" initial node distribution would be desirable. Given a function with continuous first derivatives the node redistribution algorithm of de Boor (1978) is used for modifying an initial user defined mesh. The redistribution may also be done on segments of the domain if there are first derivative discontinuities. Using super-elements, or using a subset of independent nodes, reduces the computer time by allowing far fewer Jacobian evaluations in the time integrator. Preliminary results using an algorithm which determines the independent subset by jumps in the curvature are very encouraging. Essentially the domain is being partitioned so that each region has similar curvature and only the ends of the segments are determining the node motions. Future work will be reported in this important area involving initialization.

The ODE solver currently being used is LSOIBT which takes full advantage of the implicit block structure of the ODE set.

Singularities are detected by monitoring relative slope differences so that when the equation set is nearly singular, the redundant equations are removed and temporarily replaced by an equation linking adjacent node velocities.

Two dimensional solutions using the ideas of this work and a search for an ODE solver that can be used on much larger problems are immediate priorities.

Conclusion

We have looked at some of the computational problems that were encountered in the implementation of a particular adaptive gridding method, the MFEM. Though

some of the features are unique to the particular method, the work points out the importance of recognizing redundant or singular equations, and using that knowledge to our advantage.

We have discussed simple ways of overcoming the problem of using low order basis functions for approximating higher order derivatives. Near singularity of the ODE set indicates that independent node motion may not be necessary in a local region and in the limit of a flat or parallel profile leads to a singularity and recognition of redundant equations that must be replaced. MFEM minimizes error with respect to node motion and therefore the initial conditions must not constrain the nodes from following the true solution. A front tracking and heat conserving solution is easily done with the MFEM for a moving boundary problem with no restrictions on being able to follow other steep profiles within the solution.

References

1. Crank, J. Free and Moving Boundary Problems, Clarendon Press, Oxford 1984.

2. DeBoor, C. "A Practical Guide to Splines", Springer-Verlag, New York, 1978.

3. Gelinas, R.J., S.K. Doss, and K. Miller, "The Moving Finite Element Method: Applications to General Partial Differential Equations with Multiple Large Gradients", Journal of Computational Physics, 40, 202-249, 1981.

4. Gelinas, R.J., S.K. Doss, "The Moving Finite Element Method: A Semi-Automatic Solver for Diverse PDE Applications", Fourth IMACS International Symposium on Computer Methods for Partial Differential Equations, Lehigh University, Bethlehem, PA, 1981.

5. Gelinas, R.J., S.K. Doss, "The Moving Finite Element Method: One-Dimensional Transient Flow Applications", Tenth IMACS Congress, Montreal, Canada, 1982.

6. Hrymak, A. Adaptive Grid Approaches to the Design and Analysis of Distributed Paramter Systems, Ph.D. Thesis, Department of Chemical Engineering, Carnegie-Mellon University, Pittsburgh, PA, 1984.

7. Hrymak, A. G.J. McRae, and A.W. Westerberg, "An Implementation of a Moving Finite Element Method", In Press, 1985.

8. Johnson, I.W., "The Moving Finite Element Method for the Viscous Burgers' Equation", Department of Mathematics, University of Reading, Numerical Analysis Report 3/84.

9. Lynch, D.R. and J.M. Sullivan, "Heat Conservation in Deforming Element Phase Change Simulation", Journal of Computational Physics, 57, 303-317, 1985.

10. Miller, K. and R. Miller, "Moving Finite Elements, I", SIAM Journal of Numerical Analysis, 18, 1019-1032, 1981.

11. Miller, K., "Moving Finite Elements, II", SIAM Journal of Numerical Analysis, 18, 1033-1057, 1981.

12. Wathen, A.J., "Moving Finite Elements and Oil Reservoir Modelling", Ph.D. Thesis, Department of Mathematics, University of Reading, 1984.

Table 1
Error = ($\hat{X}$/X)-1

Nodes	Conserving		Non-Conserving	
	$\hat{X}$	Error	X^A	Error
5	0.098822	0.2227E-3	0.10019	14.07E-3
10	0.098830	0.3036E-3	0.099520	7.29E-3
20	0.098832	0.3239E-3	0.099179	3.84E-3
30	0.098833	0.3340E-3	0.099064	2.67E-3

Figure 1 Finite Element Approximation and Moving Finite Element Basis Functions.

Figure 2 Case i) initial conditions. Analytic and MFEM solutions coincide exactly and X^ = (0.01 + t). Solid circle tracks point (V_1, X_1).

Figure 3 Case ii) initial conditions. MFEM forms ramp and XB = (0.01 + 1.5t). Solid circle tracks point $(V_{1(}$ XI).

Notation

a	node amplitude
A	MHEM coupling matrix
c	concentration (in gas phase for PSA)
f	arbitrary function
g	right hand side vector of inner products
K	thermal conductivity
L	latent heat
*C	spatial differential operator
m	slope
p	number of partial differential equations
Pe	Peclet number
q	concentration (in solid phase for PSA)
t	time
T	temperature
u	state variable
v	discrete approximation to u
W	local polynomial approximation to v_x
x	space coordinate
X	phase change front

Greek

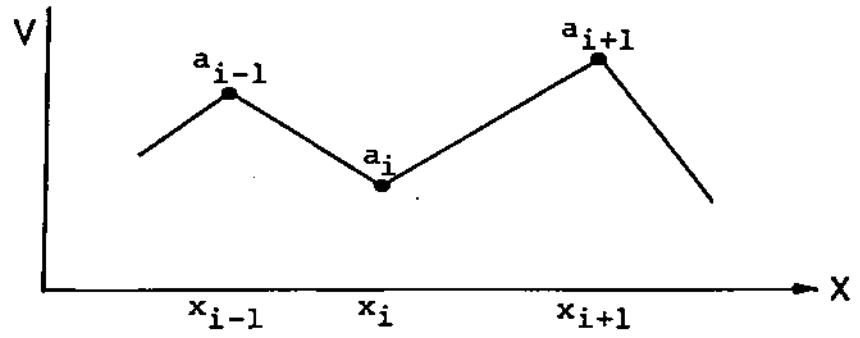
a	basis function
B	basis function
p	density

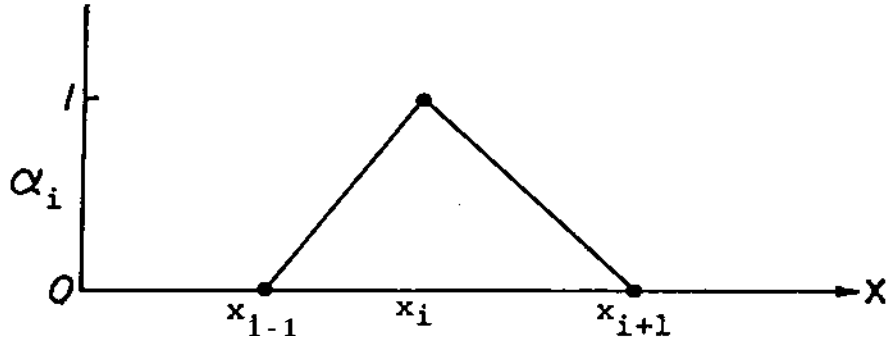
Subscripts

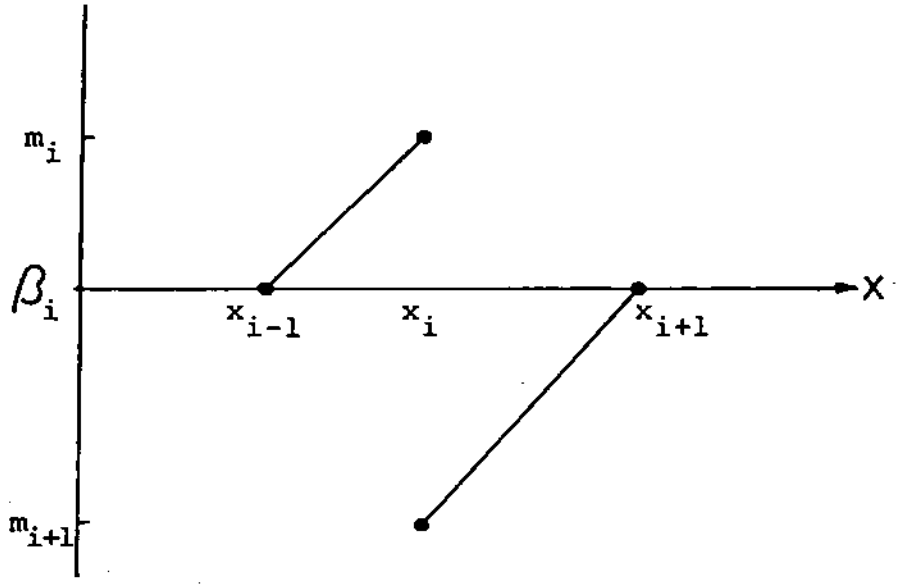
i,j	values at discrete points

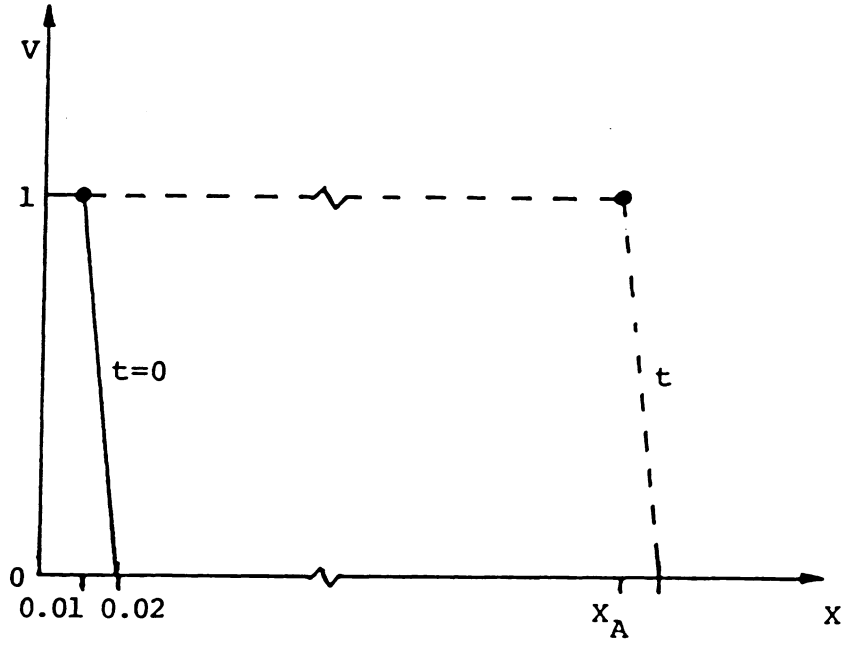
Superscripts

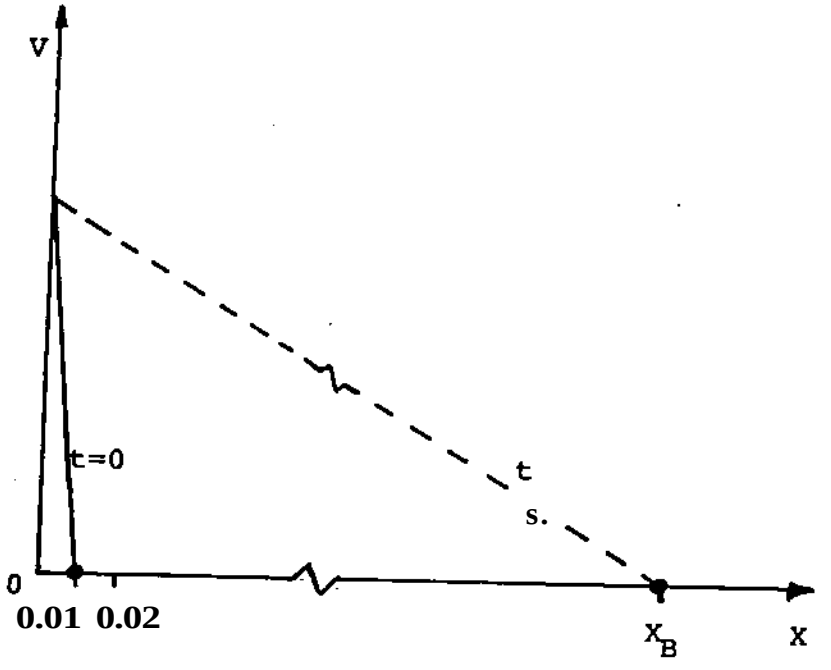
i	PDE equation number
˙	time derivative
H	Hermite
Q	Quadratic


V
a_{i-1}
a_i
a_{i+1}
X
x_{i-1}
x_i
x_{i+1}


1
α_i
0
X
x_{i-1}
x_i
x_{i+1}


m_i
β_i
m_{i+1}
X
x_{i-1}
x_i
x_{i+1}


V
1
t=0
t
0
0.01
0.02
X_A
X


V
t=0
t
s.
0
0.01
0.02
X_B
X