JOURNAL OF COMPUTATIONAL PHYSICS 58, 377–394 (1985)

Some Mixed Finite Element–Finite Difference Methods for Spherically Symmetric Relativistic Collapse

PATRICK J. MANN

Department of Astrophysics, University of Oxford, Oxford, England

Received July 27, 1983; revised February 23, 1984

Spherically symmetric perfect fluid configurations in general relativity are modelled using finite element discretizations for the radial direction and finite difference discretizations for the temporal direction. The equations are derived from a variational principle based on a Schwarschild-like metric and the relativistic enthalpy for a perfect fluid. A simple explicit two-step scheme is used in time, and the weighted residual method with various spline approximations radially. The linear spline approximation is successful, but does have an instability which becomes serious during the later stages of an evolution. Cubic Hermite splines are more stable, but the velocity near the surface shows instabilities. A special cubic Lagrange approximation for the velocity, and Hermite approximations for the other variables, proves the best scheme, but it is still necessary to introduce some smoothing algorithm near the surface.

1. INTRODUCTION

The evolution of isolated bodies in general relativity is of some interest to astrophysicists and relativists. In particular there are applications in the study of quasars, galaxies, stellar evolution and gravitational radiation. These subjects are complicated by the necessity to include "realistic" matter and asymmetric geometries. Algebraic techniques are not yet able to produce such detailed models so numerical methods must be used.

There are various numerical formulations available, but only the finite difference method (FDM) has been used for the dynamic evolution of relativistic fluids. The majority of these codes have treated spherically symmetric geometries, and those which advance to axisymmetry are not completely satisfactory. It is important, therefore, to develop alternate techniques which can model the axisymmetric configurations and act as independent checks on the FDM codes.

There are two basic avenues for the development of these alternatives: First, the underlying structure of general relativity can be utilized to develop specialized numerical techniques. This includes the methods of [12], which have only been applied to vacuum space-times, and the Regge calculus, which has recently been applied to Einstein–Rosen wave models and static spherically symmetric polytropes [9]. Second, a standard formulation can be used, but with different numerical methods for discretizing the equations introduced. This is the area to be

0021-9991/85 $3.00

investigated in this paper. In particular the general class of discretizations called Finite Element Methods (FEM) will be used.

However, there have been few FEM applications in relativity so this paper will describe some mixed FEM–FDM models of spherically symmetric polytropes. These are relatively simple, and the stability and accuracy of the FEM can be investigated.

The FDM codes for spherical symmetry which have been described in the literature fall, broadly, into three categories. The original work of [15] used co-moving coordinates with a diagonal metric. This was extended by [5], [16] and [17] to include more realistic physics and better slicing conditions, but still used co-moving co-ordinates. The second category, typified by [18] and [19], uses Eulerian coordinates, and the third category, containing only the work of [20], casts the equations in characteristic form.

Unfortunately detailed descriptions of accuracy and efficiency have not been published. The best numbers for comparison come from [17], where it is noted that the total mass for a 60-zone bounce is conserved to 0.002% after 30,000 time steps. For a complete collapse the mass is conserved to 0.01% up to the time of event horizon formation. These two numbers should be kept in mind when evaluating the FEM codes presented in this paper.

The conventions of [6] are followed, except that latin indices take the values 0, 1, 2 and 3. Relativistic units will be used, with the gravitational constant and the speed of light set to 1. The basic unit is the centimetre.

2. Equations

It is convenient to derive the equations from a variational principle. There exist many actions for Einstein's equations describing perfect fluids (e.g. [1–3] or [13]), but almost all are inadequate for our purposes, in that they do not precisely define the fluid quantities which are to be varied. However, the description derived by Schutz [10] does not have this problem.

Schutz describes the fluid by six velocity potentials but we assume that the fluid is isentropic, which allows us to write

$$U_a = (\psi_{,a} + \alpha\beta_{,a})/w$$

where U_a is the fluid four-velocity. There are four velocity potentials: ψ, α, β and the specific enthalpy w. The normalization $U_a U^a = -1$ gives one more relationship,

$$w^2 = -g^{ab}(\psi_{,a} + \alpha\beta_{,a})(\psi_{,b} + \alpha\beta_{,b}), \tag{1}$$

and leaves only three independent potentials: ψ, α and β.

The action is

$$S=\int_{\Sigma}(R+4\pi p)(-g)^{1/2}\,d^4x$$

where Σ is some four-volume of interest and p is the pressure. The quantities to be varied are the metric components g_{ab} and the fluid potentials ψ, α and β. The equation of state gives $p=p(w)$, and $w=w(g_{ab}, \psi, \alpha, \beta)$ by Eq. (1), so the variation is well defined.

Many metrics have been used for spherically symmetric models [11]. We feel that a metric based on the well-known and well-understood Schwarzschild form is of greatest interest, so we choose

$$\begin{aligned} ds^2 = &-B^2(1-2m/r)\,dt^2+(1-2m/r)^{-1}\,dr^2 \\ &+r^2(d\theta^2+\sin^2\theta d\varphi^2) \end{aligned}$$

where B and the gravitational mass m are functions of r and t. This metric has the added advantage of producing an extremely simple action;

$$\begin{aligned} S=&\int_0^R\int_0^T (m_{,r}B+4\pi r^2pB)\,dr\,dt \\ &-\int_0^T f(t)\,m(r=R,t)\,dt \end{aligned}$$

where the volume of integration is the sphere of radius R which is extended in time between $t=0$ and $t=T$, and f is an arbitrary function.

The variation can be further simplified by noting that one of the velocity potentials may be arbitrarily defined on the initial slice [10]. We choose $\alpha=0$. However, the fluid equation for α [10] is $U^a\alpha_{,a}=0$ so, unless a singularity develops, we have $\alpha=0$ always. Also $U^\theta=U^\varphi=0$ by the symmetry of the problem, so we can write

$$w^2=\frac{\psi_{,t}^2}{B^2(1-2m/r)}-\psi_{,r}^2(1-2m/r),$$

with $U_t=\psi_{,t}/w$ and $U_r=\psi_{,r}/w$. Schutz warns that α and β may become singular but still leave the quantity $\alpha\beta_{,a}$ well behaved. We feel this is not an important consideration unless real singularities ("perfect" shocks) form, and ignore the problem.

The action can now be varied to give

$$\begin{aligned} \delta S=&\int_0^T\int_0^R [m_{,r}-4\pi r^2(\rho+\psi_{,r}^2(1-2m/r)(p+\rho)/w^2)]\,\delta B \\ &+[B_{,r}-4\pi rB(p+\rho)(1-2m/r)^{-1}(1+2\psi_{,r}^2(1-2m/r)/w^2)]\,\delta m \\ &+4\pi\Bigg[-\left(r^2\frac{(p+\rho)}{w^2}\frac{\psi,t}{B(1-2m/r)}\right)_{,t} \end{aligned}$$

$$+\left(r^2\frac{(p+\rho)}{w^2}B(1-2m/r)\,\psi_{,r}\right)_{,r}\Bigg]\delta\psi\,dr dt$$

$$+\int_0^T\left[(B-f(t))\,\delta m\right]_{r=R}dt+4\pi\int_0^T r^2\frac{(p+\rho)}{w^2}\frac{\psi_{,t}}{B(1-2m/r)}\,\delta\psi\,|_{r=0}^{r=R}\,dt$$

$$-4\pi\int_0^T r^2\frac{(p+\rho)}{w^2}B(1-2m/r)\,\psi_{,r}\delta\psi\,|_{t=0}^{t=T}\,dr$$

where ρ is the total energy density.

The last three terms are boundary terms and should vanish. The first one does so if we take $B(r=R,\,t)=f(t)$, thereby setting the scale of t. The second boundary term vanishes at $r=0$ trivially and at $r=R$ if we assume that R is in a vacuum. The last term vanishes at $t=0$ if $\psi_{,r}$ (equivalently U_r) vanishes on the initial slice. We are not concerned with the $t=T$ boundary, as it can always be taken as being beyond the domain of numerical integration.

Therefore the Euler equations are

$$m_{,r}=4\pi r^2(\rho+\psi_{,r}^2(1-2m/r)(p+\rho)/w^2)$$

$$B_{,r}=4\pi r\frac{B(p+\rho)}{1-2m/r}(1+2\psi_{,r}^2(1-2m/r)/w^2)$$

$$\left(r^2\frac{(p+\rho)}{w^2}\frac{\psi_{,t}}{B(1-2m|r)}\right)_{,t}=\left(r^2\frac{(p+\rho)}{w^2}B(1-2m/r)\,\psi_{,r}\right)_{,r}.$$

We note that only the derivatives of ψ appear as physical quantities, which necessitates using a higher-order approximation for ψ. To avoid this we define new variables:

$$X=-\frac{(p+\rho)}{w^2}\frac{\psi_{,t}}{B(1-2m/r)},\qquad V=\psi_{,r}\qquad\text{and}\qquad U=\frac{\psi_{,t}}{B}.$$

Then the equations become

$$V_{,t}=(UB)_{,r} \tag{2}$$

$$X_{,t}=r^{-2}(Br^2V(1-2m/r)(p+\rho)/w^2))_{,r} \tag{3}$$

$$X=-\frac{(p+\rho)}{w^2}(V^2+w^2/(1-2m/r))^{1/2} \tag{4}$$

$$m_{,r}=4\pi r^2(\rho+V^2(1-2m/r)(p+\rho)/w^2) \tag{5}$$

$$B_{,r}=4\pi r\frac{B(p+\rho)}{1-2m/r}(1+2V^2(1-2m/r)/w^2) \tag{6}$$

where

$$U = -[(w^2 + V^2(1-2m/r))(1-2m/r)|^{1/2}. \tag{7}$$

We can now see the usual split into evolution equations ((2) and (3)) for the variables X and V, and into constraint equations ((4), (5) and (6)) for the variables m, B and w.

The boundary conditions are:

(1) $m=0$ at the centre. This can be derived from the condition that the Riemann tensor of $t=$ constant slices be well behaved everywhere.

(2) $V=0$ at the centre. This is the usual symmetry condition on the velocity.

(3) $X_{,r}=0$ at the centre. Again this is a symmetry condition.

(4) $B=1$ at the centre. This is numerically advantageous, and implies t is the proper time for an observer at the centre, but we can always change the normalization if necessary. The Schwarzschild time (proper time for an observer at spatial infinity) takes $B(\text{surface})=1$ and satisfies the variational principle. It is this time which we will use to describe the results.

(5) $X=0$ at the surface (vacuum).

As initial conditions we specify $V=0$ (stationary configuration) and $X=X(r)$ (the enthalpy distribution).

The variation of the action has isolated a complete set of evolution and constraint equations from the Einstein equations. In this spherically symmetric situation only the G_{rt} equation is missing. It is

$$m_{,t} = 4\pi r^2 B(1-2m/r)\, XV$$

and is important as a check on the results. It implies that the total mass, $m(R)$, is conserved, as is to be expected in spherical symmetry [6].

2a. *The Equation of State*

We intend to test various finite element discretizations so we will use a simple polytropic equation of state. This can be stated in terms of w as

$$p = K(w-1)^{\gamma/(\gamma-1)} \qquad \text{and} \qquad \rho = p\frac{(w+\gamma-1)}{(w-1)(\gamma-1)}$$

where γ is the usual polytropic index. This is equivalent to $p=K_0 n^\gamma$ where n is the baryon number density. We take $\gamma=5/3$ and $K=1.075\times 10^{-12}\,\text{cm}^{-2}$, which is based on the work of Miller [5].

3. Method I: Linear Splines

3a. *The Discretization*

The first discretization is a mixed FEM–FDM using linear splines and the Galerkin method for the space (r) dimension [7]. We will use m as a typical example. The approximation is $m(r, t) = \sum_{i=1}^{n} m_i(t)\, N_i(r, r_j(t))$ where the r_i form a set of suitable nodes. The N_i are the usual "hat" basis functions [7], although we allow the nodes r_i to be functions of t. It is important to have this freedom as it allows the nodes to follow the evolution. Later in this paper we will discuss nodal positions in more detail, but for the present the r_i are assumed to be known functions of t.

In general we will use the Einstein summation convention for the summation. Also a prime ($'$) will indicate the partial derivative with respect to r, and a dot the derivative with respect to t.

The Galerkin method then gives, for the typical constraint equation $m' = f$,

$$\int_0^R m_i N_i' N_j dr = \int_0^R f N_j\, dr \qquad \text{for } j = 1, 2, \ldots, n.$$

For the typical evolution equation, $\dot{V} = g$, we have

$$\int_0^R \left(\dot{V}_i(t)\, N_i + V_i(t) \frac{\partial N_i}{\partial r_k} \dot{r}_k \right) N_j\, dr = \int_0^R g N_j\, dr \qquad \text{for } j = 1, 2, \ldots, n.$$

Note the natural appearance of the grid velocity $\dot{r}_k$, which follows from the dependence of the N_i on t.

Much of the integration can be performed by hand, and some is amenable to integration by parts. For the evolution equation (3) we are left with:

$$\begin{aligned}
&\dot{X}_{i-1} h_i/6 + \dot{X}_i (h_i + h_{i+1})/3 + \dot{X}_{i+1} h_{i+1}/6 \\
&\quad = \int_{r_{i-1}}^{r_{i+1}} \left[2B \frac{(p+\rho)}{w^2} \frac{V}{r} (1 - 2m/r)\, N_i \right. \\
&\qquad \left. - B \frac{(p+\rho)}{w^2} V (1 - 2m/r)\, N_i' - \sum_{j=i-1}^{i+1} X_j \dot{N}_j N_i \right] dr \\
&\dot{X}_1 h_2/3 + \dot{X}_2 h_2/6 \\
&\quad = \int_{r_1}^{r_2} \left[2B \frac{(p+\rho)}{w^2} (1 - 2m/r)(V/r)\, N_1 \right. \\
&\qquad \left. - B \frac{(p+\rho)}{w^2} V (1 - 2m/r)\, N_1' - (X_1 \dot{N}_1 + X_2 \dot{N}_2)\, N_1 \right] dr \\
&\dot{X}_s = 0
\end{aligned}$$

where $h_i = r_i - r_{i-1}$. The remaining integrals will have to be evaluated numerically.

A similar set holds for the V equation (2), except that the known boundary condition is $\dot{V}_1 = 0$.

The discretizations of the constraint equations are naturally partitioned into two sets. The first is

$$X_i = -\frac{(p_i + \rho_i)}{w_i^2} \left[V_i^2 + w_i^2/(1 - 2m_i/r_i) \right]^{1/2} \tag{8}$$

$$m_{i+1} = m_{i-1} + 2 \int_{r_{i-1}}^{r_{i+1}} 4\pi r^2 (\rho + V^2(1 - 2m/r)(p + \rho)/w^2)\, N_i dr \tag{9}$$

$$m_2 = m_1 + 2 \int_{r_1}^{r_2} 4\pi r^2 (\rho + V^2(1 - 2m/r)(p + \rho)/w^2)\, N_1 dr \tag{10}$$

$$m_1 = 0$$

where B does not occur. The second set is

$$B_{i+1} = B_{i-1} + 2 \int_{r_{i-1}}^{r_{i+1}} 4\pi r \frac{B(p + \rho)}{1 - 2m/r} (1 + 2V^2(1 - 2m/r)/w^2)\, N_i\, dr \tag{11}$$

plus the N_1 equation, and $B_1 = 1.0$.

3b. *The Constraint Solver*

We assume that values for V and X are known on the time slice of interest, either from the initial data or by evolution from a previous time slice, and that values for m, w and B are required.

First we evaluate (8) at $i = 1$ (the centre) to get

$$X_1 = \frac{p_1 + \rho_1}{w_1}$$

which can be solved, given X_1 and the equations of state, for w_1. Then we solve Eq. (8) (at $i = 2$) and Eq. (10) for the two unknowns w_2 and m_2. This gives enough data to "evolve" outwards using (8) and (9), solving for m_{i+1} and w_{i+1} at each step, until the surface is reached.

These values are used in the "B" equation (11) to again "evolve" outwards, this time solving for the B_i. Note that these equations are linear in the B_i so the solution is trivial.

Unfortunately the m and w equations are highly non-linear. In particular (8) is very difficult to solve because w appears in the form $(w-1)^{\gamma/\gamma-1}$ through the equations of state. It is necessary to use an iterative scheme for the solution of such equations.

The standard solvers we tried were not satisfactory as none of them would give good convergence rates for all w. For instance, the secant method is prohibitively slow when w approaches 1.

The best method we could devise uses a regular falsi iteration on the m equation,

with Eq. (8) solved for w by a separate iteration whenever needed. The w iteration uses the formula

$$w = 1 + \left(\frac{(\gamma - 1)\, X}{K\gamma}\right)^{(\gamma - 1)} (V^2/w^2 + 1/(1 - 2m/r))^{(1-\gamma)/2}$$

which can be derived from Eq. (8) using the polytropic equations directly. At any given step of the m solution an approximate m is known, and a new w is calculated by inserting a guess for w on the right side of this equation and iterating until convergence occurs.

This method is specific to polytropic equations of state, and a new method would have to be designed for other equations of state.

3c. *The Evolution Equations*

The discretized evolution equations form a set of first-order ordinary differential equations in time, of the form

$$A_X \dot{\mathbf{X}} = \mathbf{b}_X \qquad \text{and} \qquad A_V \dot{\mathbf{V}} = \mathbf{b}_V$$

where A_X and A_V are tridiagonal matrices. These matrix equations are easy to solve (on the computer) to give a set of equations

$$\dot{X}_i = g_{Xi} \qquad \text{and} \qquad \dot{V}_i = g_{Vi}$$

which we will discretize with the explicit two-step finite difference method [8] as follows. The superscript refers to the time slice $t^{(a)}$.

Assume a consistent and complete set of data is known on $t = t^{(a-1)}$. First evolve to $t^{(a-1/2)}$ by

$$X_i^{(a-1/2)} - X_i^{(a-1)} = (dt/2)\, g_{Xi}^{(a-1)}$$

where $dt = t^{(a)} - t^{(a-1)}$, with a similar equation for V. Then solve the constraints to give a complete set of data at $t^{(a-1/2)}$. The final step uses

$$X_i^{(a)} - X_i^{(a-1)} = (dt)\, g_{Xi}^{(a-1/2)}$$

to give X (similarly V) at $t^{(a)}$, and the constraint solver is again applied to complete an evolution to $t^{(a)}$.

4. Method IIa: Cubic Hermite Splines

Our experience with the equations for static space-times [4] suggests that cubic Hermite splines will produce a better approximation. This takes the form (again using m as a typical example)

$$m = \sum_{i=1}^{n} [m_{1i}(t)\, N_{1i}(r, r_j(t)) + m_{2i}(t)\, N_{2i}(r, r_j(t))]$$

where the N_{ai} are piecewise cubic, with support $[r_{i-1}, r_{i+1}]$. They are normalized to give $m(r_i) = m_{1i}$ and $m'(r_i) = m_{2i}$ [7].

To give better stability to the constraint solver (the constraints are hyperbolic) we have decided to use the weighted residual FEM, which allows us to choose weight functions. For the general constraint $m' = f$ we have

$$\int m' W_{ai}\, dr = \int f W_{ai}\, dr$$

where the W_{ai} are $2n$ weights. Unfortunately it is impossible to produce a good solver from these discretizations [4]. Instead we choose one set of weighted residual equations, with weight N_{1i},

$$\int_{r_{i-1}}^{r_i} m' N_{1i}\, dr = \int_{r_{i-1}}^{r_i} f N_{1i}\, dr$$

and one set derived from the direct application of the differential equation,

$$m_{2i} = f(r_i).$$

The evolution equations are discretized in a similar manner, although different weights are used. For $\dot{V} = g$ we note that in all cases g contains at most first derivatives in r. Therefore we can take

$$(V_{ai} N_{ai})\dot{}_{r = r_i} = g(r_i)$$

because the r-derivatives of the approximations are continuous. This gives

$$\dot{V}_{1i} = g(r_i) + V_{2i}\dot{r}_i \qquad \text{(no sum on } i\text{)}. \tag{12}$$

For the V_{2i} evolution we use a weight N_{2i} on $[r_{i-1}, r_{i+1}]$ which gives

$$\begin{aligned} &-\dot{V}_{2i-1} h_i^3/140 + \dot{V}_{2i}(h_i^3 + h_{i+1}^3)/105 - \dot{V}_{2i+1} h_{i+1}^3/140 \\ &\quad = -\int_{r_{i-1}}^{r_{i+1}} \left[g - \sum_{j=i-1}^{i+1} (V_{1j}\dot{N}_{1j} + V_{2j}\dot{N}_{2j}) \right] N_{2i}\, dr \\ &\qquad - 13(\dot{V}_{1i+1} h_{i+1}^2 - \dot{V}_{1i-1} h_i^2)/420 - 11\dot{V}_{1i}(h_{i+1}^2 - h_i^2)/210. \end{aligned} \tag{13}$$

The explicit two-step evolution scheme can be applied to the V_{1i} and V_{2i}. First the $\dot{V}_{1i}$ can be evaluated using (12), and this inserted in (13). The resulting tridiagonal matrix can be solved for the $\dot{V}_{2i}$, and then the two-step differencing applied.

The X equation is simple to solve this way because we have boundary conditions $X_{21} = 0$ and $X_{2s} = 0$. Unfortunately there are no known boundary conditions for V'. Applying the FEM method to both ends (weights N_{21} and N_{2s}) results in an ill-conditioned matrix, as there is no condition to "nail down" any value. We have used the simple difference equation $V_{2s} = (V_{1s} - V_{1s-1})/h_s$ but we might expect some difficulties with the V evolution.

The constraints are solved as in the linear case. First we solve for m_{ai} and w_{ai} from the centre out, and then for B_{ai}. Equation (4) for w must be differentiated to calculate w_{2i}, and this introduces the quantity $d\rho/dw$, which is calculated from the equation of state.

5. Method IIb. Cubic Lagrange Splines for V

To alleviate the problems involved with approximating V' we decided to try cubic Lagrange splines. These splines are awkward to use because it is not possible to have basis functions with support $[r_{i-1}, r_{i+1}]$. Following [7] we choose basis functions which extend over $[r_{i-2}, r_{i+2}]$, although this means that the function values at each node can no longer be used as the coefficients of the approximation. Instead we write $V = \sum_{i=0}^{n+1} \gamma_i(t)\, Q_i(r, r_j(t))$ where the Q_i are cubics in r, with support $[r_{i-2}, r_{i+2}]$. The formulae for the Q_i are given in [4] and ensure that the approximation has continuous first derivatives. In the limit of constant h the second derivatives are also continuous.

This approximation has the property

$$V(r_i) = (\gamma_{i-1} h_{i+1}^2 + \gamma_i (h_i + h_{i+1})^2 + \gamma_{i+1} h_i^2)/(h_i + h_{i+1})^2.$$

Note that the quantities h_1 and h_{n+1} occur in a non-trivial fashion. They can be chosen arbitrarily, as long they do not vanish. We choose $h_1 = h_2$ and $h_{n+1} = h_n$.

For a typical evolution equation $\dot{V} = g$, the Galerkin method gives

$$\int \sum_{j=i-3}^{i+3} (\dot{\gamma}_j Q_j)\, Q_i \, dr = \int f Q_i \, dr - (\text{grid velocity terms}).$$

This produces a seven-diagonal matrix with special boundary conditions for the top three and bottom three rows. To simplify this the weights are changed to the linear basis functions N_i, which restricts the integration to $[r_{i-1}, r_{i+1}]$ and therefore produces a five-diagonal matrix. However, there are only n equations in the $n+2$ variables γ_i. The same problem, in disguised form, occurs with the Galerkin weight because the matrix is badly ill-conditioned, and can be singular.

The usual solution is to apply boundary conditions. Only one is available here, $V_1 = 0$, and some other equation is needed. To get this V_s is calculated directly, which is feasible because the approximations have continuous first derivatives, and only first derivatives occur on the right side of the V equation. This will lower the precision attained because the derivatives are approximated with lower accuracy, and is not a completely satisfactory solution.

6. Grid Velocity and Time Step

The grid must move to follow the collapse. In the previous discussion the r_i have been assumed to be known functions of t. They are constrained by only one condition, that r_{i-1} is less than r_i. A simple choice is a linear spline approximation in t, given by $r_i(t) = r_i^{(a-1)} N^{(a-1)}(t) + r_i^{(a)}(t)$ where t is in $[t^{(a-1)}, t^{(a)}]$. Then the $r_i^{(a)}$ are the nodes on time slice $t = t^{(a)}$, and we can write

$$r_i^{(a)} = r_i^{(a-1)} + \dot{r}_i (t^{(a)} - t^{(a-1)})$$

where $\dot{r}_i$ is the (constant) grid velocity between the two time levels. The set of $r_i^{(a-1)}$ is known, either from the initial data or a previous evolution, so we are left with the problem of calculating suitable $\dot{r}_i$ on $t^{(a-1)}$, and a suitable time step $dt = t^{(a)} - t^{(a-1)}$.

First we assume that the outermost node stays on the surface. Then $\dot{r}_s$ is given by the three-velocity of the surface, $(U^r/U^t)_s$.

Ideally the inner nodes should be chosen to give greatest precision to the numerical evolution, but useful estimates for the precision related to nodal position are not yet available. As an interim solution the r_i are chosen to follow lines of constant mass, which gives

$$\dot{r}_i = \frac{-B_i^2 V_i (1 - 2m_i/r_i)^2 X_i}{\rho_i + V_i^2 B_i (1 - 2m_i/r_i) X_i / U_i}.$$

This at least will ensure constant precision in the m evolution.

The time step was limited by allowing a predetermined relative change (ε) in the central X value, and by testing for the Courant condition. The Courant stability criterion gives

$$dt < d_c \min[|C_i h_i|, |C_i h_{i+1}|]$$

where C_i is the three-velocity of sound in the fluid, and d_c a parameter which should include all the details of a full stability analysis. We will choose d_c experimentally.

Note that $r_i < 1$ (the speed of light) and $C_i < 1$, so the dt choice ensures that nodes will not cross and no time-consuming tests are required.

7. Initial Conditions

As discussed in Section 2, we start with $V = 0$ (and $V' = 0$) and specify some initial matter distribution X (and X'). Then the constraint solver can be applied to calculate m, B and w.

The matter distribution is calculated by multiplying a static solution (X_{old}) by a factor

$$1 + \alpha - \alpha r^2 ((r_s^2 - 2r_m^2) r + 2r_m^3 - r_s^3)/(r_s^2 r_m^2 (r_m - r_s)).$$

This is a simple cubic which ensures that X_{new} vanishes at r_s and that X'_{new} vanishes at r_1 and r_s. It has two arbitrary parameters: α and r_m. We choose α, and then find r_m by solving the constraint equations and requiring that the new mass (m_s) be identical to the static mass.

The new model has $X_{\text{new}} = (1+\alpha) X_{\text{old}}$ at the centre, and $X_{\text{new}} = X_{\text{old}}$ at $r = r_m$, which, for positive α, results in an initial model with a central peak in the density.

The original static data are supplied by various FEM codes for the static equations, which are described in [4]. The linear code uses a linear spline based static solver, and the cubic Hermite code a cubic Hermite based solver. The Lagrange modification only affects V, which vanishes, so the cubic Hermite data are sufficient with the γ_i set to zero.

The different static solvers made it impossible to give identical initial data to all the codes. However, the data sets never varied by more than a relative difference of 1% in the surface radius and mass. More seriously, the number and position of nodes varied, due to a more efficient subroutine using derivative information in the cubic Hermite static solver. Therefore the number of nodes and some information on their position is given in the tables.

8. Some Runs and Comparisons

The codes exist on a CDC 7600 machine at the University of Manchester Regional Computing Centre, and calculations are performed in 60–bit precision. All numerical integration is performed using Simpson's rule with 11 nodes in every interval $[r_{i-1}, r_i]$. More nodes make no difference within the machine precision. For the same reason all iterations were allowed to continue until successive estimates had a relative separation of less than 10^{-12}, which is approximately the machine precision.

The first tests ran static ($\alpha = 0$) initial data with $w_1 = 1.178$. This model is physically of marginal stability [14] so we would expect the physical stabilizing forces to be small. Therefore any numerical instabilities should become obvious quite rapidly.

The linear code ran well, but the grid velocity soon developed a large oscillation. It was necessary to average the grid velocities to smooth out this wobble.

There was also an oscillation in adjacent m and B values near the surface. This has been noted in the static solvers [4] and is due to the instabilities inherent in the linear FEM when applied to the initial value equations for m and B. As long as the time step was small ($d_c < 0.01$) the oscillation did not increase with time.

Previous runs with the static solvers have also indicated that the accuracy near the centre is poor. To test this we set up special initial data which artificially flattened X near the centre (and therefore changed the total mass).

The inaccuracy near the centre indeed appeared, but as a wave in V which was amplified catastrophically as the evolution progressed. However, it was heavily damped radially, and never extended beyond the sixth node of a 66-node run. As all

quantities behaved well outside this wave we tried to remove it by fitting a straight line to the quantities near the centre. This was successful, making almost no difference to results outside the central nodes.

The cubic Hermite code (IIa) did not need the grid velocity averaging, and was stable with $d_c = 0.05$. However, it was evident that the evolution of V_{21} was not adequate, and that this was dragging the other variables. The mass, for instance, was slowly increasing (a 0.9% change after 540 evolutions). As the V_{11} were behaving satisfactorily we decided to take $V_{21} = V_{12}/r_2$, a linear fit in the central element. This helped, but the mass was still increasing and the V_{2i} were not evolving well.

The cubic Lagrange code (IIb) did not exhibit any central instabilities, but did have an oscillation in V near the surface. To smooth the oscillation we took

$$\dot{V}(r_s) = \tfrac{1}{2}(V_{1s} + V_{1s-1}) + \tfrac{1}{2}(V_{2s} + V_{2s-1})\,\dot{r}_s$$

where the V_{1i} are calculated directly and the V_{2i} from the (known) Lagrange approximation. With $d_c = 0.1$ the evolution was quite satisfactory. After 1080 steps ($t = 63M$) the central enthalpy had changed by 0.002%, and the total mass by 0.9%.

It is of interest to approximate X with cubic Lagrange splines also. This is quite simple to code, but the results are poor. The central enthalpy is not static, and (with $d_c = 0.1$) had decreased by 7.4% after 1080 evolutions. A smaller time step may stabilize these results, but we did not continue developing this code.

To compare the codes they were given almost identical static initial data, and allowed to run with identical parameters. The results are summarized in Table I. All the cubic codes are noticeably better than the linear code, and code IIb (Lagrange splines for V) is the best of the cubic codes.

TABLE I

A Comparison of Evolutions from Static Initial Data ($w_1 = 1.178$)

Code	Linear (I)		Cubic(IIa)	Cubic (IIb)	Cubic (IIc)
Nodes	34	34	32	32	32
d_c	0.01	0.05	0.1	0.1	0.1
Evolution Number	500	—	100	100	100
Time	7.01M	—	7.02M	7.02M	7.02M
dw/w	$8.6 \times 10^{-2}\%$	—	$-4.0 \times 10^{-6}\%$	$-8.6 \times 10^{-5}\%$	$5.8 \times 10^{-3}\%$
dm/m	0.0094%	—	0.22%	0.0055%	−0.46%
V_{surface}	0.40×10^{-2}	—	0.32×10^{-3}	-0.98×10^{-4}	0.42×10^{-2}

Note. M is the initial surface mass, dm/m is the relative change in mass and dw/w is the relative change in central enthalpy. Note that dw/w, dm/m and V_{surface} should be zero. The linear code with $d_c = 0.05$ is unstable and did not reach $t = 7.01M$. Code IIc uses cubic Lagrange splines for both the X and V evolutions.

9. Two Dynamic Evolutions

We started from static solutions with $w_1 = 1.178$ and 25.0, and ran the initial value code with $\alpha = 0.4$. The two sets of initial data are summarized in Table II.

The $L1$ evolution required $d_c = 0.01$ which seriously increases the computational time required. It evolved 14,000 times, but by $t^{(11000)} = 219M$ (where M is the surface mass on the initial slice) the mass had increased by an amount equal to the two-step surface oscillation (2.3%). Also the central enthalpy was beginning to increase to an extent which seemed unphysical although there were no obvious numerical defects.

Unfortunately the $C1$ run with code IIb exhibited large instabilities in V near the surface, which seemed to be independent of time step size. Trial and error eventually produced a method which successfully smoothed the wobble. It fits two quadratics to V, one at r_s and r_{s-2} and one at r_{s-1} and r_{s-2}, and then averages the two curves.

This code an extremely well. With $d_c = 0.1$ the run continued to $t^{(4000)} = 415M$, at which time the evolution seemed to have settled down to a static situation. The mass was conserved to 0.6%, which is very good for a 27-node run.

Figure 1 compares the central enthalpy in the $L1$ and $C1$ evolutions. It is apparent that the linear code, as was suspected, becomes inaccurate near the centre. In Fig. 2 the final enthalpy configuration reached by $C1$ is compared with a static model with central enthalpy equal to 1.178. The agreement is good, although it does point out some interesting behaviour near the centre. This is partly due to the rebound which occurred near $t = 120M$ (Fig. 1), but we feel that the first five points exhibit numerical inaccuracies.

In Table III we further compare the $L1$ and $C1$ evolutions. Note especially that the $C1$ code is marginally faster per evolution. This is due to the constraint solver,

TABLE II

A Summary of the Four Sets of Initial Data Used To Test Dynamic Evolutions

Data set	Number of nodes	w (centre)	B (centre)	m (surface) (cm)	r_s (cm)	r_{s-2} (r_s)	r_2 (r_s)
$L1$ (linear)	28	1.223	0.7466	72,372	817,820	0.95	0.037
$C1$ (cubic)	27	1.223	0.7466	72,527	820,472	0.84	0.034
$L2$ (linear)	59	31.04	0.00318	49,681	395,358	0.69	0.00047
$C2$ (cubic)	64	31.04	0.00318	49,684	397,013	0.76	0.0033

Note. The $L1$ and $C1$ data sets are based on $w_{\text{centre}} = 1.178$ static data, and the other two sets on $w_{\text{centre}} = 25.0$ static data.

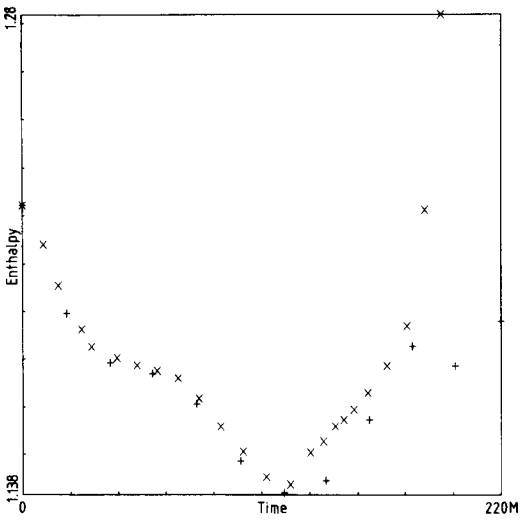

FIG. 1. The central enthalpy is graphed against time for some representative points during the $L1$ and $C1$ evolutions. The X's are from the $L1$ evolution and the $+$'s are from a $C1$ evolution using the mixed cubic Lagrange–cubic Hermite code.

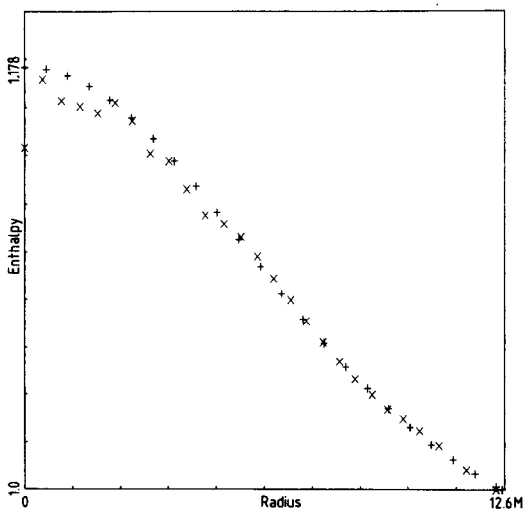

FIG. 2. The enthalpy versus radius points (X's) at $t^{(4000)} = 415M$ from the mixed cubic code for the $C1$ initial data. As a comparison a static configuration with $w_{\text{centre}} = 1.178$ is graphed with the $+$ signs. The $C1$ initial data are a modification of this data, indicating that the model has relaxed to the static configuration.

TABLE III

A Comparison of Evolutions from the $L1$ and $C1$ Initial Data at a Late Time

Code	Evolution number	Time	dm/m	w (centre)	r (surf)	Computer time	Average computer time per evolution
Linear ($L1$) code I $d_c = 0.01$	11000 28 nodes	$185.9M$	2.2%	1.221	$15.1M$	1681 sec	0.15 sec
Cubic ($C1$) code 11b $d_c = 0.1$	1045 27 nodes	$185.3M$	−0.15%	1.174	$12.2M$	136 sec	0.13 sec

Note. M is the initial surface mass and dm/m is the relative change in the surface mass.

which is a one-step method in the cubic code and a two-step method in the linear code.

The initial configurations with $w_1 = 31.04$ were expected to collapse quickly, as they are highly relativistic, and based on static models which are unstable. This is exactly what happened. Indeed, the collapse was so fast that much larger time steps could be used, as the instabilities did not become significant.

Two $L2$ runs were made, one with $d_c = 0.01$ (14,000 evolutions) and one with $d_c = 0.2$ (255 evolutions). The behaviour was almost identical in both cases,

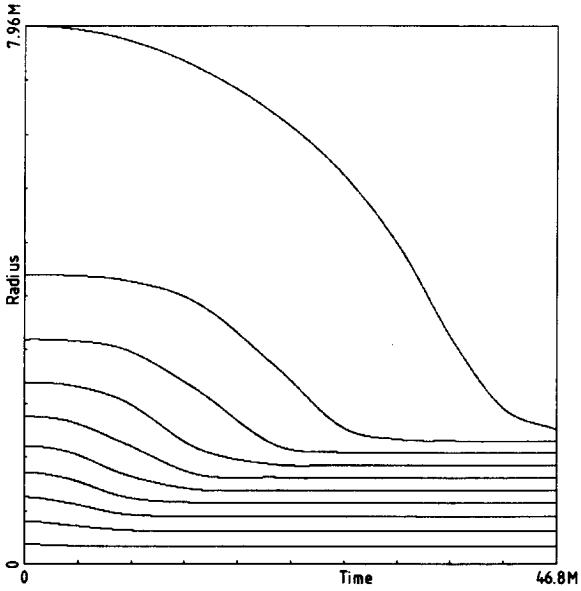

FIG. 3. Lines of constant mass at $0.1M$ intervals, graphed in the r, t plane. The data are from the $L2$ evolution with $d_c = 0.2$, and indicate the development of an event horizon.

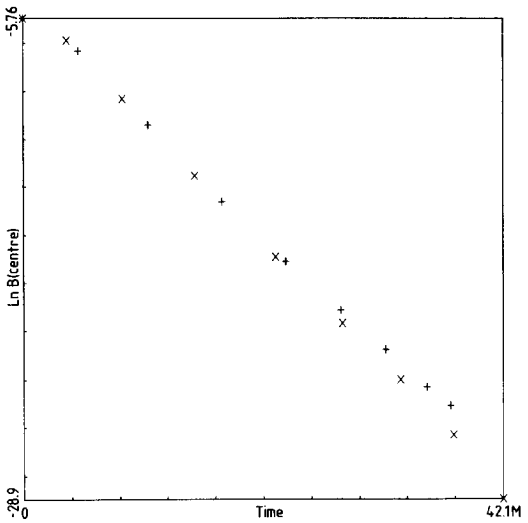

FIG. 4. The X's are representative of the central B versus time for the linear $L2$ evolution. The + signs are from the equivalent cubic run. These central B values indicate the ratio of proper time at the centre with Schwarzschild time.

although the $d_c = 0.2$ run seemed a bit better. At $t^{(175)} = 37.7M$ ($d_c = 0.2$) the mass had decreased by 1.8%, while at $t^{(9200)} = 36.8M$ ($d_c = 0.01$) the mass had increased by 2.3%. At this time $2m/r$ had almost become unity at a point near the surface, indicating the development of a horizon. Only five or six nodes were available for that part of the star which was still "outside," and this number is insufficient for an accurate model. The overall behaviour is summarized in Fig. 3, which graphs curves of equal mass in r and t. The cubic Lagrange code (IIb) produced adequate models with $d_c = 0.5$, but did not conserve the mass as well as the linear code, the mass having changed by 3% at $t^{(72)} = 37.6M$. This is probably due to the time step difference.

A comparison showing the agreement between the runs is given by Fig. 4, which graphs the logarithm of central B against time. The collapse is almost a free fall so this graph is very close to a straight line.

10. CONCLUSIONS

We can say with some confidence that the mixed FEM–FDM is successful in modelling spherical symmetric collapse. There are still problems involved with stability, but we note that an artificial viscosity has not been included in any of the codes.

The FEM as applied here has required quite a lot of work, both in the discretization and in the programming. This stems from the generality of the

calculations, as the whole mechanism of the FEM must be used. However, it is not necessary to include many extra calculations when the method is extended to more dimensions. This is notable in the element size and geometry, and in the inclusion of grid velocities.

It is difficult to give quantitative comparisons with the FDM because of the different computers involved, but we feel that the FEM codes are significantly slower. This is due to the integration which requires special calculations at each integration node. We expect that this can be greatly improved, but more work remains to be done.

We conclude that the FEM is not a good alternative to the FDM for one-dimensional codes, but that it is sufficiently successful to warrant attacking two- or three-dimensional problems where the FEM's particular advantages may become evident.

ACKNOWLEDGMENTS

I would like to thank Professor D. W. Sciama and Dr. J. C. Miller for their help and encouragement during the development of this paper.

REFERENCES

1. J. M. BARDEEN, *Astrophys. J.* **162** (1970), 71.
2. J. B. HARTLE AND D. H. SHARP, *Astrophys. J.* **147** (1967), 317.
3. S. W. HAWKING AND G. F. R. ELLIS, "The Large-Scale Structure of Space-Time," Cambridge Univ. Press, Cambridge, 1973.
4. P. J. MANN, *Comput. Phys. Comm.* **30** (1983), 127.
5. J. C. MILLER, "Computer Calculations of the Gravitational Collapse of Stars in a Full General Relativistic Treatment," D. Phil. thesis, University of Oxford, 1974.
6. C. W. MISNER, K. S. THORNE, AND J. A. WHEELER, "Gravitation," Freeman, San Francisco, 1973.
7. A. R. MITCHELL AND R. WAIT, "The Finite Element Method in Partial Differential Equations," Wiley, Chichester, 1977.
8. D. POTTER, "Computational Physics," Wiley, London, 1973.
9. J. D. PORTER, "Regge Calculus," D. Phil. thesis, University of Oxford, 1982.
10. B. F. SCHUTZ, *Phys. Rev. D* **2**(1970), 2762.
11. L. SMARR, *in* "Sources of Gravitational Radiation" (L. Smarr, Ed.), p. 245, Univ. of Cambridge Press, Cambridge, 1979.
12. J. M. STEWART AND H. FRIEDRICH, Numerical relativity. I. The characteristic initial value problem, Max-Plank-Institute preprint, 1982.
13. A. H. TAUB, *Phys. Rev.* **94** (1954), 1468.
14. S. WEINBERG, "Gravitation and Cosmology," Wiley, New York, 1972.
15. M. MAY AND R. H. WHITE, *Math. Comput. Phys.* **7** (1967), 219.
16. J. R. WILSON, *Astrophys. J.* **232** (1975), 558.
17. K. A. VAN RIPER, *Astrophys. J.* **232** (1979), 558.
18. J. R. WILSON, *in* "Sources of Gravitational Radiation" (L. Smarr, Ed.), p. 423, Univ. of Cambridge Press, Cambridge, 1979.
19. S. L. SHAPIRO AND S. A. TEUKOLSKY, *Astrophys. J.* **235** (1980), 199.
20. G. B. BICKNELL AND R. N. HENRIKSEN, *Astrophys. J.* **232** (1979), 670.