THE UNIVERSITY OF READING

A Moving Mesh Method for Non-linear Parabolic Problems

by

K.W. Blake & M.J. Baines

Numerical Analysis Report 02/02

DEPARTMENT OF MATHEMATICS

A Moving Mesh Method for Non-linear Parabolic Problems

K.W.Blake & M.J.Baines
Department of Mathematics
University of Reading
P O Box 220, Reading, RG6 6AX, UK

March 2002

Abstract

A moving mesh method is described for the solution of non-linear parabolic partial differential equations, based on a conservative distribution principle. Using this principle the underlying partial differential equation may be recast as a moving mesh equation. The mesh equation is then integrated forward in time, the solution being reconstructed from the distribution principle itself. The method is described with explicit reference to the porous media equation. The distribution principle uses a monitor function which is initially taken as the dependent variable, in line with scale-invariance properties of the porous media equation with zero flux boundary conditions. Although the resulting meshes exhibit the inherent self-similar properties of the exact solution they fail to resolve the solution at the moving boundary. The procedure is therefore modified, first by incorporating mesh subdivision and then by using a gradient monitor function. In the latter case the distribution principle is no longer conservative and is therefore generalised to include time varying distributions. The procedure is applied to two other problems, one related to semi-conductor process modelling (with fixed boundaries) and the other with a non-linear source term, inducing solution blow-up.

1 Introduction

The use of adapted meshes in the numerical solution of partial differential equations (PDEs) is a valuable technique for improving existing approximation schemes. For problems in which large solution variations are common, an adapted mesh can improve the accuracy and the efficiency of existing methods by concentrating mesh points within regions of interest.

There are three main classes of mesh adaptation. The first, h-refinement, adds extra nodes to an existing mesh to improve local mesh resolution. A second technique, p-refinement, employs higher order numerical approximations to improve local accuracy as well as approximate troublesome derivatives. The third approach is r-refinement, which maintains the existing number of nodes globally but relocates them strategically over the domain. It is this latter approach that we shall use in the course of this report, although, as we shall see, r and h-refinement may be combined within this general framework.

In dynamic mesh relocation methods a mesh movement principle is needed to determine the speeds of nodes. This is designed in such a way that meshpoints remain concentrated in regions of interest as the solution evolves with time. Thus two coupled equations need to be considered, the underlying PDE and an equation to control the evolution of the mesh ([12],[25], [5], [15]). In the present work the mesh is evolved using a Conservative Distribution Principle (CDP) which moves the meshpoints so as to preserve the local integral of a monitor function in time.

In this report we describe a moving mesh method for one-dimensional parabolic PDEs of the form

$$u_t = (D(u)u_x)_x + Q(u),$$

with particular reference to problems involving non-linear diffusion, with and without moving boundaries, and solution blow-up.

The approach relies on the construction of a CDP and on recasting the PDE to yield an equation for mesh movement. As a result only a single system of ODEs for the mesh points need be integrated forward in time, the resulting solution being reconstructed algebraically from the CDP.

We begin in Section 2 by summarising the work of Budd et al ([6],[9],[10]) which argues that moving mesh methods should be designed to preserve the scale-invariance properties of the PDE in question *locally*, thus indicating a choice of monitor function. In this section we also recall the idea of equidistribution, a standard technique which underpins many existing moving mesh ideas, and review the approach of Moving Mesh Partial Differential Equations (MMPDEs) of Huang, Ren and Russell ([24],[15], [16],[17]).

In Section 3 we outline the moving mesh method used in this report in the specific case of the porous media equation (PME) with a moving boundary, and demonstrate how the PDE and the CDP are coupled together to produce a system of ODEs for the mesh coordinates. In line with the work of Budd et al, the monitor function in the CDP is chosen to be the dependent variable of the problem. These ideas are followed up in Section 4 where it is shown how mesh refinement may be included within this framework. Without mesh refinement the dependent variable monitor function gives poor resolution at the moving front and we therefore generalise the method using a more geometrical monitor function and a time-varying distribution.

Section 5 contains numerical results from the moving mesh procedure compared to an exact self-similar solution of the PME and we verify the scaling invariance of the method when using the dependent variable as the monitor function.

The procedure is then extended in Section 6 to two further problems exhibiting additional features. First we consider a model semiconductor dopant diffusion problem in which the moving boundary is replaced by a fixed boundary with a Neumann boundary condition. The mesh adaptation is required to resolve features in regions of low dopant concentration. Secondly, we apply the method to an inhomogeneous problem involving solution blow-up, using the technique to reproduce accurate times of solution combustion. The report ends with some concluding remarks.

2 Background

2.1 Scale Invariance and the Porous Media Equation

In [7],[9] and [10] Budd et al advocate that any moving mesh method should reflect the scale-invariance properties of the underlying PDE. In this section we describe the relevant properties of the PME, beginning by formally introducing the equation and highlighting some of its properties. In one dimension the PME is the non-linear diffusion equation

$$u_t = (u^m u_x)_x \tag{1}$$

where $m > 0$. The PME arises for example in the study of the diffusion of gas through a porous medium under the action of Darcy's law relating velocity to pressure gradients, or in the modelling of the swarming of insect species (see Murray [20]). There is a fairly complete existence theory for solutions of the equation which takes the form of travelling waves in a region of growing compact support $[s_-(t), s_+(t)]$, with $u = 0$ for $x \geq s_+(t)$and $x \leq s_-(t)$. Given

that such a compact solution u exists, the solution conserves two important quantities, mass (the integral of u) and centre of mass (the first moment of u). We have

$$\frac{d}{dt}\int_{-\infty}^{\infty} u dx = \int_{-\infty}^{\infty} u_t dx = \int_{-\infty}^{\infty} (u^m u_x)_x dx = (u^m u_x)|_{-\infty}^{\infty} = 0 \qquad (2)$$

so that the total mass is conserved. Also, if $\bar{x}$ is the scaled centre of mass

$$\begin{aligned}
\frac{d\bar{x}}{dt} &= \frac{d}{dt}\int_{-\infty}^{\infty} xu dx = \int_{-\infty}^{\infty} xu_t dx = \int_{-\infty}^{\infty} x(u^m u_x)_x dx \\
&= -\int_{-\infty}^{\infty} u^m u_x dx = -\frac{1}{m+1}\int_{-\infty}^{\infty} (u^m)_x dx = -\frac{1}{m+1}(u^m)|_{-\infty}^{\infty} \\
&= 0
\end{aligned}$$

so that $\bar{x}$ is conserved.

The conservation of both the total mass and centre of mass allows the speed of the moving boundary to be derived. Moreover, given symmetric initial conditions, the solution will remain symmetric throughout time. Figure 1 illustrates the behaviour of the solution, the dashed line representing the progress of the self-similar solution, for the case $m = 2$. The value of m influences both the speed and character of the solution close to the moving boundary. The larger the value of m, the steeper the evolving front and the slower the rate of displacement of the moving boundary.

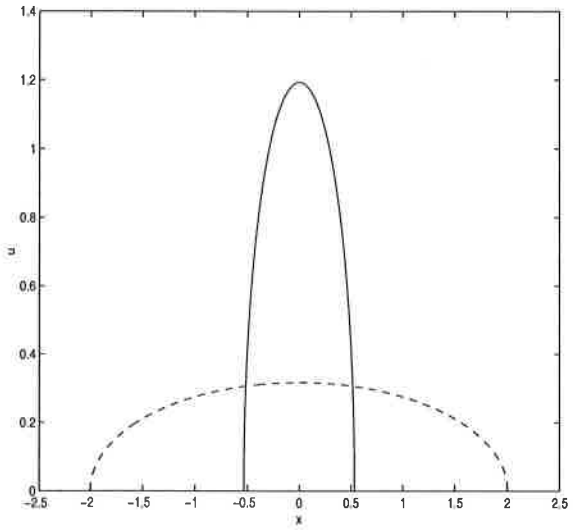

Figure 1: Evolution of solution to the PME (1), for $m = 2$.

We now outline the *scaling invariance* property which is the basis of the moving mesh ideas of Budd et al. ([7]). Given a system (u, x, t) satisfying a PDE and a mapping to a new system$(\hat{u}, \hat{x}, \hat{t})$ under the transformation

$$\hat{u} = \lambda^{\alpha} u, \quad \hat{x} = \lambda^{\beta} x, \quad \hat{t} = \lambda t \tag{3}$$

where λ is an arbitrary parameter, the original (u, x, t) system is said to be *scaling invariant* if the PDE under consideration is identical in both the original and transformed co-ordinates. Moreover, if a solution to the equation is invariant under the mappings then the solution is said to be *self-similar*. The resulting equation, written in terms of the transformed variables, is often easier to solve than the original PDE and may be solved to give a class of exact solutions. The resulting self-similar solutions are invariant in the transformed space and hence independent of time. Choosing $\hat{t}$ arbitrarily to be 1, we can write the transformed solution in terms of the original coordinates as

$$\hat{u} = u t^{\frac{1}{m+2}}, \quad \hat{x} = x t^{\frac{-1}{m+2}} \tag{4}$$

Substituting into the PME the resulting steady differential equation for $\hat{u}$ and $\hat{x}$ can be solved by techniques found in Barenblatt [2] or Dresner [13]. As a result a self-similar solution with a moving boundary can be derived (with zero flux conditions) which can be found in e.g. Murray [20], of the form

$$u(x,t) = \left\{ \begin{array}{ll} \frac{1}{\lambda(t)}[1 - \{\frac{x}{r_0\lambda(t)}\}^2]^{\frac{1}{m}} & |x| \leq r_0\lambda(t) \\ 0 & |x| > r_0\lambda(t) \end{array} \right\} \tag{5}$$

where

$$\lambda(t) = \left(\frac{t}{t_0}\right)^{\frac{1}{(2+m)}}, \quad r_0 = \frac{Q\Gamma(\frac{1}{m} + \frac{3}{2})}{\pi^{\frac{1}{2}}\Gamma(\frac{1}{m} + 1)}, \text{and} \quad t_0 = \frac{r_0^2 m}{2(m+2)}$$

with $r_0\lambda(t)$ representing the position of the moving front and Γ denoting the gamma function.

The existence of the self-similar solution allows us to validate numerical approximations of the PME. One aim is that the approximate solutions will, under the appropriate transformation, display the invariance properties of the exact solution, as is the case with the moving mesh solutions of Budd and Piggott [10].

2.2 Equidistribution and MMPDEs

Equidistribution, first introduced by De Boor [11] for use in approximation theory is now a widely used tool to produce efficient computational meshes for the numerical solution of differential equations ([26],[1],[15]). The basic idea is to construct the mesh such that a geometric property or error measure of the solution u, specified via a monitor function M, is distributed equally in each computational cell. Written formally, on a mesh consisting of $N+1$ nodes, $x_0, x_1, \ldots, x_N$, the equidistribution principle (EP) is

$$\int_{x_i}^{x_{i+1}} M dx = \frac{1}{N} \int_{x_0}^{x_N} M dx \qquad (i = 0, \ldots, N-1), \tag{6}$$

It is easy to see that by choosing a monitor involving a first or second derivative of u, nodes can be clustered in areas of large solution variation or curvature, so as to provide more accurate numerical approximations in these regions.

Alternatively the EP may be formulated as a coordinate mapping problem between the physical space x and a computational space ξ_i divided up to give equally spaced discrete points $\xi = \frac{i}{N}$ for $i = 0, 1, \ldots, N$. Then

$$\int_0^{x(\xi)} M d\tilde{x} = \xi \int_0^1 M d\tilde{x} \tag{7}$$

The EP (6,7) may be differentiated with respect to space and time in various ways to derive Moving Mesh Partial Differential Equations (MMPDEs) [15]. The EP (7) is differentiated twice with respect to ξ and then time differentiated in order to eliminate the potentially troublesome right hand side integral term and any of its derivatives, leading to MMPDEs (1-4) in [15]. The resulting equations move the mesh such that the distributive properties of the EP are preserved. Such equations for mesh movement are widely used, for example by Mackenzie et al [18],[19],[3]. The integral form (7) is used directly by Flaherty [14].

The choice of monitor function can be crucial to the solution of a specific problem. For instance, Qiu & Sloan [23] developed a specialist monitor for the solution of a reaction diffusion problem where more traditional monitors failed. In [9] Budd et al implement an MMPDE involving a simple monitor ($M = 1$) which preserves the scale-invariance of the PME, for which the resultant mesh has a uniform expanding character. For the case $m = 1$, using finite difference approximations to the pseudo-Lagrangian form of the PME and for the speed of the moving boundary, the method is applied in both the original and scaled variables. It is found that the scaling resulting from the approximate self-similar solution is identical to that in the continuous case,

so that the discrete self-similar solution has the dynamics of the underlying solution in the original variables. The scheme used has conservation of mass (and the centre of mass) built in as required and it is shown that the resulting discrete self-similar solution converges to the true self-similar solution as the number of nodes in the mesh is increased.

The density monitor $M = u$ has been used for the PME by Budd & Piggott [10], whose results show that for the PME with $m = 1$ (for which the gradient at the moving front is finite) the mass monitor preserves all the desired properties and is able to model the solution adequately. However, as we shall see, for higher values of m (for which the front at the moving boundary is infinitely steep) a different monitor is needed to resolve the front. Similar ideas have also been applied to the non-linear Schrödinger equation [8] and to problems with blow-up [7].

3 The Moving Mesh Method

In this section we describe the Conservative Distribution Principle (CDP) used in this report. Arising from the observations on scale-invariance and equidistribution above we derive, with specific reference to the PME, systems of ODEs which give the movement of mesh points under the influence of the dependent variable (or density) monitor

$$M(u) = u \tag{8}$$

However, as we shall see, this monitor leads to poor resolution at the moving front. This motivates the subsequent incorporation of mesh subdivision into the method.

3.1 A Conservative Distribution Principle (CDP)

We begin with the equidistribution principle incorporating the monitor (8). Using the notation in (6) we have

$$\int_{x_i(t)}^{x_{i+1}(t)} u dx = \frac{1}{N} \int_{x_0(t)}^{x_N(t)} u dx = \theta(t) \qquad i = 0, \ldots, N-1. \tag{9}$$

defining the integral term $\theta(t)$. Time differentiation of (9) gives

$$\frac{d}{dt} \int_{x_i(t)}^{x_{i+1}(t)} u dx = \frac{d\theta}{dt} = \dot{\theta}$$

With this monitor the mass $\theta(t)$ in (9) in each cell can be taken to be independent of time, since this is consistent with the PME problem being mass conserving. We then have

$$\frac{d}{dt}\int_{x_i(t)}^{x_{i+1}(t)} u dx = 0 \tag{10}$$

which will be referred to as the Conservative Distribution Principle (CDP).

3.2 A Mesh Movement PDE

Integrating the PME from $x_i(t)$ to $x_{i+1}(t)$ gives

$$\int_{x_i(t)}^{x_{i+1}(t)} u_t dx = \int_{x_i(t)}^{x_{i+1}(t)} (u^m u_x)_x dx = [u^m u_x]_{x_i(t)}^{x_{i+1}(t)} \tag{11}$$

Moreover, expanding (10) gives

$$\int_{x_i(t)}^{x_{i+1}(t)} u_t dx - [u\dot{x}]_{x_i(t)}^{x_{i+1}(t)} = 0$$

Substituting into (11) gives the Moving Mesh equations

$$[u\,\dot{x} + u^m u_x]_{x_i(t)}^{x_{i+1}(t)} = 0 \qquad (i = 1, \ldots, N-1) \tag{12}$$

which we refer to as the FDE or the flux form of the PDE.

The zero right hand side in (12) is the result of the specific choice of monitor $M(u) = u$, for which the mass θ in (9) is independent of time. Moreover, due to the symmetry of the porous medium solution, we have that $\dot{x}_0 = (u_x)_{x_0} = 0$ at $x_0 = 0$, so equation (12) leads to the system of ordinary differential equations (ODEs) for the mesh co-ordinates.

$$\begin{aligned} \dot{x}_0 &= 0 \\ \dot{x}_i &= -(u^{m-1}u_x)_{x=x_i} \qquad (i = 1, \ldots, N-1) \end{aligned} \tag{13}$$

Since (13) holds for any x_i arbitrarily close to x_N we also have

$$\dot{x}_N = -\lim_{x \to x_N} u^{m-1} u_x \tag{14}$$

Note that for $m > 1$, the slope u_x at $x = x_N$ must be infinite if the speed of the moving front is to be finite.

The system of ordinary differential equations (13,14) will move the mesh in such a way that, as the material diffuses, cells retain the original mass within them.

subsectionDiscretising the CDP

Using a trapezium rule approximation for the distributed mass θ we have the approximation

$$\frac{1}{2}(u_{i+1}+u_i)(x_{i+1}-x_i)=\hat{\theta} \qquad (i=1,\ldots,N-1) \tag{15}$$

where $\hat{\theta}$ is the mass under the linear interpolant. The discrepancy between the exact equidistributed mass θ used in (9) and the discrete mass $\hat{\theta}$ used in (15) is of order $(x_{i+1}-x_i)^3$.

Since $u_N = 0$ at the foot of the moving boundary we can solve these equations to give

$$u_N = 0, \tag{16}$$

$$u_i = 2\sum_{k=i+1}^{N}\hat{\theta}\frac{(-1)^{k-i-1}}{x_k-x_{k-1}} \qquad (i=N-1,\ldots,0) \tag{17}$$

yielding the interpolant values u_i in terms of the current mesh co-ordinates x_i and the discrete constant mass $\hat{\theta}$.

Initially we use a mesh that has equidistributed discrete mass (15) in all cells. This is achieved by using a linearised form of the monitor function,

$$M_{i+\frac{1}{2}}=\frac{1}{2}(u_i+u_{i+1}),$$

when generating the initial mesh, so as to be consistent with the trapezium rule approximation used in (15). Details of how to solve this initial meshing problem can be found in for example [1].

To discretise the system (13),(14) we use an upwinding approximation which respects the flow of information, i.e.

$$\dot{x}_i=-u^m u_x|_{x_i}\approx-\left[\frac{u_i+u_{i-1}}{2}\right]^{m-1}\frac{(u_i-u_{i-1})}{(x_i-x_{i-1})} \qquad (i=1,...,N) \tag{18}$$

The approximated ODE system moves the mesh in such a way that the discrete approximations to the mass will be approximately equidistributed and conserved.

3.3 Numerical Procedure

The full algorithm is as follows:

- Discretise the initial data giving the value of $\hat{\theta}$ and the x_i, u_i values.
- Solve the CDP (10) via (17) for the u_i's in terms of the x_i's.
- Substitute for the u_i's into the discretised form (18) of the FDE (12) to obtain the mesh speeds $\dot{x}_i$.
- Solve the ODE system in the x_i's by a package, in these examples using the NAG BDF routine [21].

Numerical results are given in section 5.

4 Resolution at the Moving Boundary

Despite its attractive theoretical properties the mass monitor does not naturally place nodes in regions of high solution variation, which is clearly highly desirable for the solution of the PME at the moving boundary for values of $m > 1$. However, looking at the method from a different perspective gives us two other ways of placing moving nodes in this region.

4.1 Mass Conservation and Grid Refinement

Equation (10) does not have to correspond to an equidistribution principle. In fact we may choose any initial distribution of mass, e.g.

$$\int_{x_i(t)}^{x_{i+1}(t)} u dx = \theta_{i+\frac{1}{2}} \qquad (i = 0, \ldots, N-1)$$

where the masses $\theta_{i+\frac{1}{2}}$ may be any strategically chosen masses which sum to the correct total mass. The system of ODEs remains exactly the same, the only difference in the algorithm being a slight change in the formula (17) for the recovery of the u solution from the current mesh. We now have the equations

$$\begin{aligned} u_N &= 0 \\ u_i &= 2 \sum_{k=i+1}^{N} \hat{\theta}_{i+\frac{1}{2}} \frac{(-1)^{k-i-1}}{x_k - x_{k-1}} \qquad (i = 1, \ldots, N-1). \end{aligned}$$

where $\hat{\theta}_{i+\frac{1}{2}}$ are again the masses under the linear interpolant.

To improve mesh resolution near the moving boundary we simply choose to place smaller and smaller quantities of mass in the cells near the boundary.

This gives an easy way of maintaining high mesh resolution in the desired area. The nodes are now moved by pure conservation of mass rather than by preserving equidistribution of mass. Again, results are shown in Section 5.

The next section examines the use of an alternative choice of monitor.

4.2 Using the Gradient Monitor

We now describe the use of a gradient monitor function within the existing algorithm. The motivation behind the change of monitor is to resolve high gradients without the need for mesh refinement. The gradient monitor is

$$M(u) = u_x \tag{19}$$

and requires that the solution remains monotonic, which is the case here.

When the monitor function $M = u$ was used, the value of θ in (6) was set to a constant, consistent with the mass conservation property of the PME problem. Other choices of monitor however are not consistent with mass conservation in general. Nevertheless we are still able to derive the method in much the same way as before by retaining a non-zero time derivative of θ in (12) and including an additional differential equation for θ in the ODE system expressing the conservation of the distribution of mesh pints arising from the monitor (19) We thus restate (6) in the form

$$\int_{x_i(t)}^{x_{i+1}(t)} u_x dx = \frac{1}{N}\int_{x_0(t)}^{x_N(t)} u_x dx = \theta(t) \qquad (i = 0, \ldots, N-1) \tag{20}$$

so that, differentiating with respect to time gives, over each cell, we obtain the more general form of the CDP (10) as

$$\int_{x_i(t)}^{x_{i+1}(t)} u_{xt} dx + [\dot{x}u_x]_{x_i(t)}^{x_{i+1}(t)} = \frac{d\theta}{dt} \tag{21}$$

Substituting for u_{tx} from the PME (1), differentiated w.r.t x, gives

$$\int_{x_i(t)}^{x_{i+1}(t)} (u^m u_x)_{xx} dx + [\dot{x}u_x]_{x_i(t)}^{x_{i+1}(t)} = \frac{d\theta}{dt} = \dot{\theta} \tag{22}$$

providing the generalised FDE

$$[(u^m u_x)_x + \dot{x}u_x]_{x_i(t)}^{x_{i+1}(t)} = \dot{\theta} \qquad (i = 0, \ldots, N-1) \tag{23}$$

of (12).

The right hand side term $\dot{\theta}$ but can be dealt with by using the geometric equidistribution properties of the gradient monitor for which, from (20),

$$u(x_{i+1}) - u(x_i) = \theta(t) \qquad (i = 0, \ldots, N-1) \tag{24}$$

Thus nodes will be placed such that the $u(x_i)$'s are equally distributed over the solution range of u. By summing all the equations from (24) we have

$$\theta(t) = \frac{1}{N}(u(x_0) - u(x_N))$$

which, after differentiation with respect to time, substitution into the PME and using $u_N = 0$, gives

$$\dot{\theta} = \frac{1}{N} u_t|_{x=x_0} = \frac{1}{N}(u^m u_x)_x|_{x=x_0}. \tag{25}$$

As in (15) we use (24) with approximate masses $\hat{\theta}$ as if u were its linear interpolant, corresponding to

$$u_{i+1} - u_i = \hat{\theta}(t) \tag{26}$$

and

$$\dot{\hat{\theta}} \approx \dot{u}_N \approx -2(u^n_{N-\frac{1}{2}} + \epsilon)\left[\frac{u_N - u_{N-1}}{(x_N - x_{N-1})^2}\right]. \tag{27}$$

Returning to equation (23), we then have

$$[(u^m u_x)_x + \dot{x} u_x]^{x_{i+1}(t)}_{x_i(t)} = \frac{1}{N}(u^m u_x)_x|_{x=x_0}.$$

giving the ODE system

$$\begin{aligned} \dot{x}_0 &= 0 \\ \dot{x}_i(u_x)|_{x_i} &= -(u^m u_x)_x|_{x_i} + \frac{N-i}{N}(u^m u_x)_x|_{x=x_0} \quad (i = 1, \ldots, N-1) \end{aligned} \tag{28}$$

The expression for the speed of the moving boundary (14) is still valid.

The system of ODE's (28,26) may then be numerically solved using the upwind discretisations

$$u_x|_{x_i} = \frac{1}{2}\left[\left(\frac{u_{i+1} - u_i}{x_{i+1} - x_i}\right) + \left(\frac{u_i - u_{i-1}}{x_i - x_{i-1}}\right)\right], \tag{29}$$

$$(u^m u_x)_x|_{x_i} = \frac{2}{(x_{i+1} - x_{i-1})}\left[u^m_{i+\frac{1}{2}}\left(\frac{u_{i+1} - u_i}{x_{i+1} - x_i}\right) - u^m_{i-\frac{1}{2}}\left(\frac{u_i - u_{i-1}}{x_i - x_{i-1}}\right)\right] \tag{30}$$

where

$$u_{i+\frac{1}{2}} = \frac{1}{2}\left(u_i + u_{i+1}\right)$$

Having prescribed the speed of the nodes, we relate the mass conservation property to the current solution u, the mesh x and the quantity $\theta(t)$.

4.3 Determining $\theta(t)$ from the Total Mass

From (27)

$$u_i = (N-i)\hat{\theta}(t) \qquad (i = 0, \ldots N). \tag{31}$$

(cf. (17)).

Now consider the composite trapezium rule expression for the (known) constant total conserved mass $\hat{\Theta}$,

$$\hat{\Theta} \approx \frac{1}{2}\sum_{i=0}^{N-1}(x_{i+1} - x_i)(u_{i+1}(t) + u_i(t)) \tag{32}$$

Substituting from equation (31) gives

$$\hat{\Theta} \approx \frac{\hat{\theta}(t)}{2}\sum_{j=0}^{N-1}(2(N-i)-1)(x_{i+1} - x_i)$$

which when rearranged gives the expression

$$\theta(t)\left\{\left(\sum_{j=0}^{N-1}(x_{j+1} - x_j)(N-j)\right) - \frac{1}{2}(x_N - x_1)\right\} = \Theta \tag{33}$$

for $\theta(t)$. Given the mesh x_i at any time, we can now compute $\hat{\theta}(t)$ using (33) and then recover the appropriate solution values u_i using (31).

Numerical results are shown in section 5.

4.4 A Combination Monitor

The gradient monitor gives a good solution close to the moving boundary but is less effective near the symmetry point. We therefore propose a new monitor function which combines the mass and gradient monitors with a weighting parameter ω,

$$M(u) = u - \omega u_x \tag{34}$$

The derivation of the moving mesh equations follows in the same way as before (for full details see [4]). In fact the ODE system and all other equations are just linear combinations of those used for the previous two monitors individually. Numerical results for both the gradient and combination monitors appear in the following section.

5 Numerical Results

We now show results illustrating the ability of these techniques to give approximate solutions of the PME problem. The system of ODEs (13,28) is solved using the NAG routine D02EJF [21], which uses variable order, variable step size, backwards differentiation formula (BDF) to integrate the system forward in time. Initial meshes are computed using an equidistribution algorithm (see e.g. [1]) using the discrete versions of the monitor functions. The initial conditions are specified using the self-similar solution (5) taken at an arbitrarily chosen time $t = 0.01$.

Since the approximate solution u is recovered exactly from the positions of the nodes, and the algorithm for the nodes sweeps backward from the moving boundary, it follows that the point $x = 0$ will carry the greatest truncation error. Equidistributing the mass using 20 nodes and setting $m = 1$ Figure 2 shows second order convergence of the solution at $x = 0$ at $t = 1$ as the number of nodes is increased. The right hand graph in Figure 2 shows the trajectories of the nodes. Moreover we can show how the mesh produced for $m = 1$ reproduces the scaling invariance results covered in Section 2.1. Transforming the mesh co-ordinates and porous medium solution produces the invariant mesh and solution values with respect to time shown in Figure 3.

For larger values of m, we can use the mesh refinement technique outlined in Section 4.1. An initial mesh consisting of 15 nodes is augmented by adding a further 9 nodes by repeatedly halving the mass contained in the last cell until it is below a specified tolerance 10^{-4}. Figure 4 shows the numerical solution and analytical solutions for $m = 3$, this time starting from time $t = 0.05$. Nodes are placed tightly in the region of the moving boundary without affecting the quality of the method elsewhere. Since mass is conserved locally in each cell, the scaling invariance results shown earlier hold for the refined mesh too.

We now show how the gradient monitor can be used to resolve the steep moving boundary without the need for mesh refinement. Figure 5 presents results in the same way as in Figure 4. From these results we can clearly see how the nodes arrange themselves equally over the solution of range of

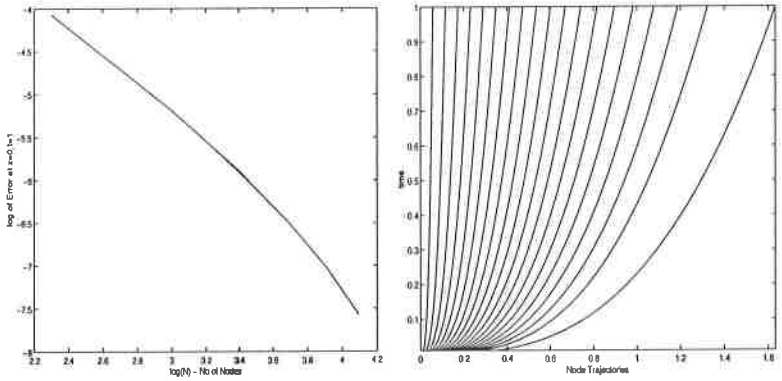

Figure 2: Error convergence and Node trajectories for $m = 1$

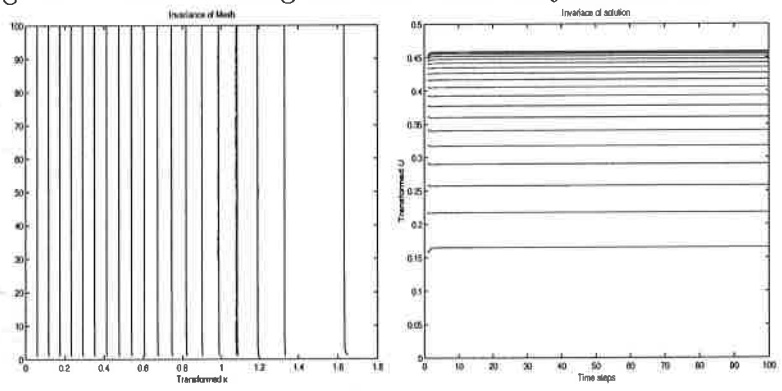

Figure 3: The invariant transformed computed mesh (Left) and PME solution (Right) for $m = 1$

u as expected and hence towards the developing front as desired. However when compared to results using the mass monitor with refinement, there are clearly inaccuracies between the generated and analytic solution.

Figure 6 shows the solution when $m = 3$, choosing the parameter $\omega = 0.2$ in the combination monitor (34). We can see clearly that the combination monitor has performed well for this value of m. Both the position of the front and the global solution are well approximated.

6 Further Applications

In previous sections the mass conservation property of the PME problem and the presence of a Dirichlet boundary condition were exploited in the moving mesh technique. We now turn our attention to two other problems, one without a Dirichlet boundary condition and one without mass conserva-

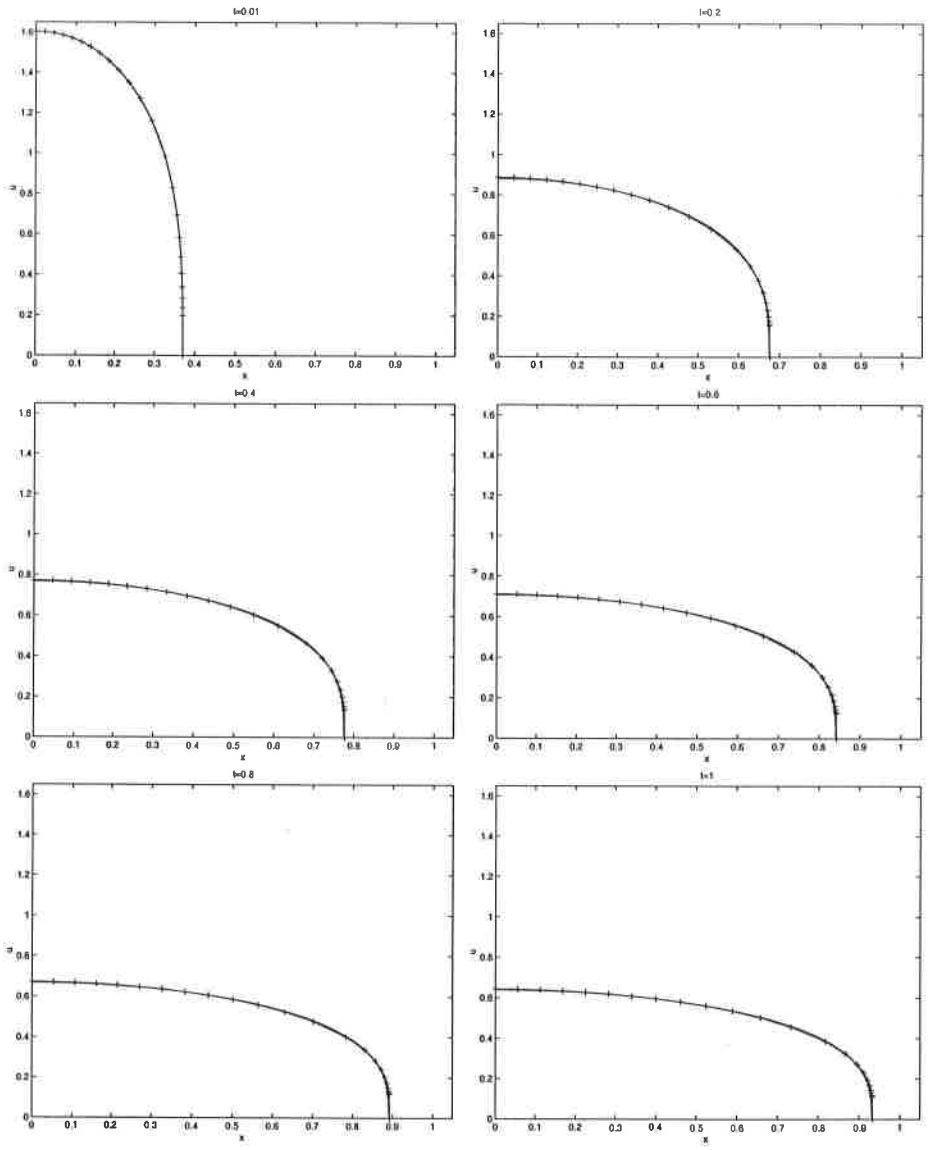

Figure 4: Approximate and reference PME solutions for $m = 3$, using mesh refinement

tion. We begin with a model problem associated with semiconductor process modelling [22].

6.1 A Semiconductor Model Problem

A simple model of semiconductor dopant diffusion is

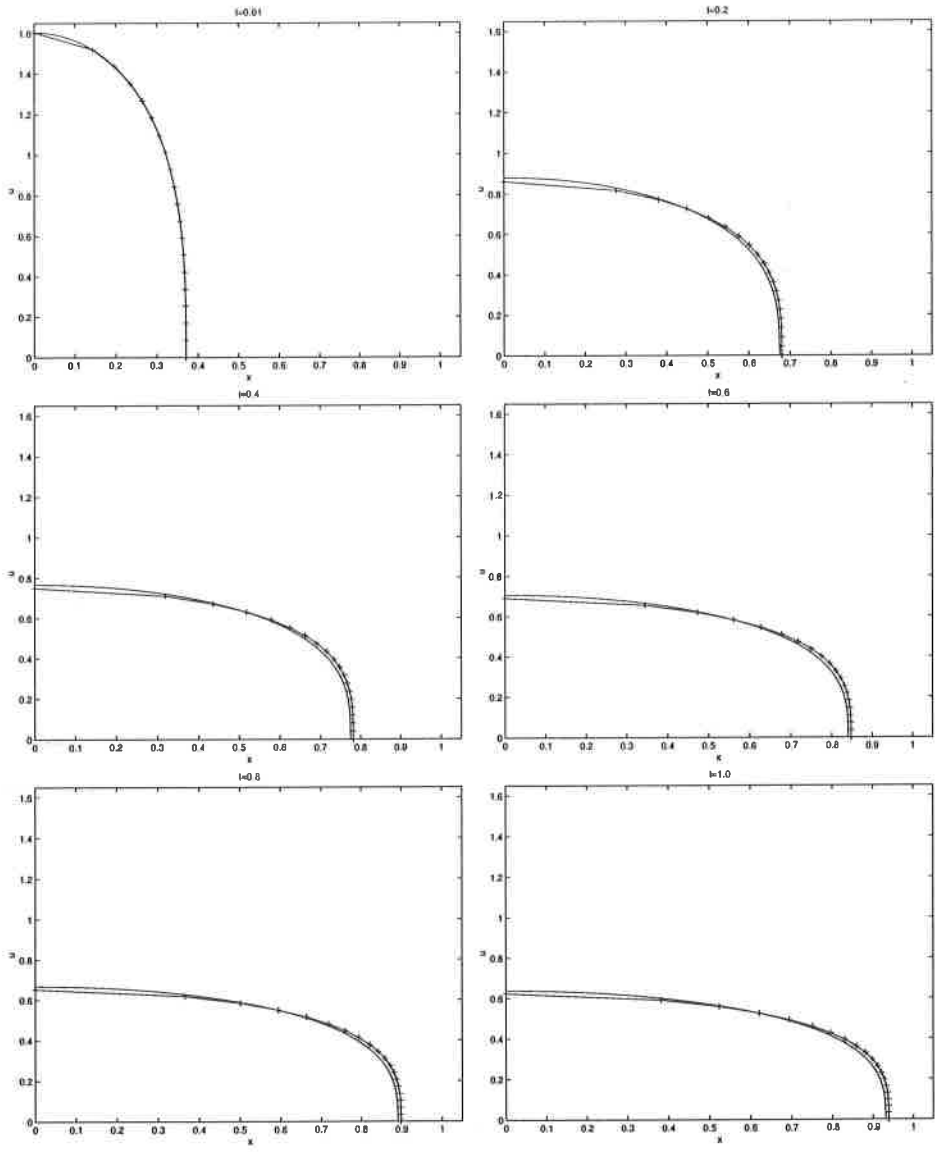

Figure 5: Approximate and reference PME solutions using gradient monitor, $m = 3$

$$u_t = ((u + \epsilon)u_x)_x, \tag{35}$$

on a fixed region $x \in [0, 1]$, ϵ being a constant of the order 10^{-2}. Homogeneous Neumann conditions are imposed at both boundaries, $x = 0$ and $x = 1$, and the dopant u has an initial Gaussian distribution

$$u(x, 0) = e^{-50x^2}.$$

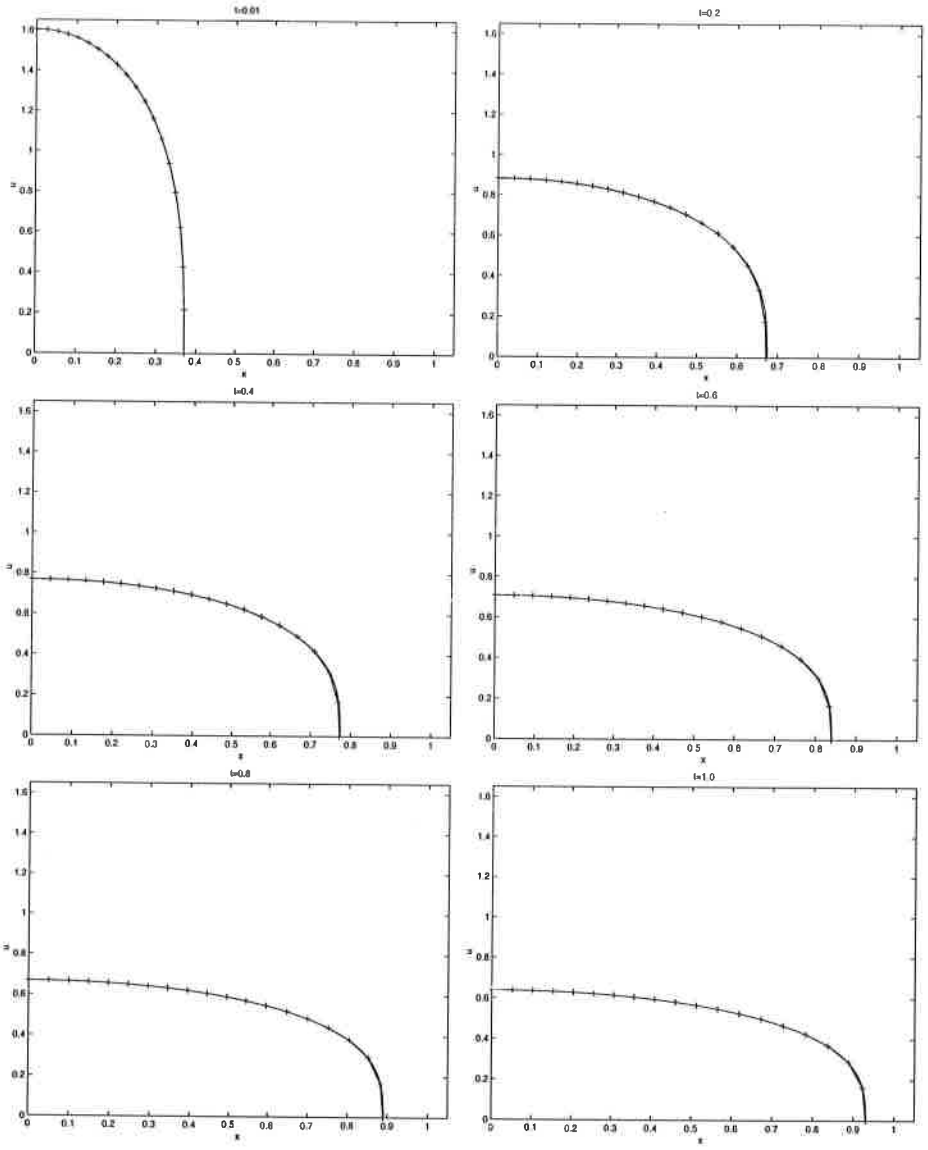

Figure 6: Approximate and reference PME solutions using the combination monitor, $m = 3$ and $\omega = 0.2$.

Since the semi-conductor problem is defined on a fixed domain, the boundary constraints on the ODE system are

$$\dot{x_0} = \dot{x_N} = 0.$$

It is clear that the solution will have a time-dependent solution value at both boundaries. For the PME the zero value of u at the moving boundary

allowed the solution to be be derived directly from the mesh and the quantity $\theta(t)$ via the algebraic relations (17) or (31) (see Sections 3.2, 4.3). In order that the method should continue to permit this reconstruction, a known solution value at one point of the mesh is required.

Here we shall derive an expression for the time derivative $\dot{u}_N$of u at $x = x_N = 1$ by considering the integral form of equation (35) in the interval between x_N and the cell midpoint $x_{N-\frac{1}{2}}$. Given the Neumann condition at $x = x_N$ we have that

$$\int_{x_{N-\frac{1}{2}}}^{x_N} u_t dx = \int_{x_{N-\frac{1}{2}}}^{x_N} ((u+\epsilon)u_x)_x \, dx = -(u+\epsilon)u_x|_{x=x_{N-\frac{1}{2}}} \tag{36}$$

Discretising in an upwind manner as in (18), we obtain an approximate expression for $\dot{u}_N$ at $x = 1$ via

$$\dot{u}_N \approx -2(u^n_{N-\frac{1}{2}} + \epsilon)\left[\frac{u_N - u_{N-1}}{(x_N - x_{N-1})^2}\right]. \tag{37}$$

Despite the low gradient of the solution at $x = 1$ an appropriate mesh spacing is required near that point during the solution to ensure accuracy of the value u_N and hence the global solution. We found that for this problem, in order to ensure an adequate mesh resolution in this area we needed to add an extra constant term α to the combination monitor, giving

$$M(u) = u - \omega u_x + \alpha. \tag{38}$$

As in section 3.1, we start by equidistributing the monitor function over the cells,

$$\theta(t) = \int_{x_i(t)}^{x_{i+1}(t)} (u - \omega u_x + \alpha)\, dx \qquad (i = 0, \ldots, N-1). \tag{39}$$

(cf. (6)).Differentiating with respect to t and using (35) yields the ODE system

$$\begin{aligned} \dot{x}_0 &= 0, \\ [(u+\epsilon)u_x - \omega\left(((u+\epsilon)u_x)_x\right) + (u - \omega u_x + \alpha)\dot{x}]_{x_i}^{x_{i+1}} &= \dot{\theta} \qquad (i = 1, \ldots, N-1), \\ \dot{x}_N &= 0 \end{aligned} \tag{40}$$

(cf. (23)), together with a differential equation for the time derivative of $\theta(t)$ which is easily found to be

$$\dot{\theta} = -\frac{\omega}{(N-1)}\left[((u+\epsilon)u_x)_x\right]_{x=0}^{x=1}, \tag{41}$$

To complete the solution set we add the integration of $\dot{u}_N$ at $x = 1$ in equation (27), as an extra equation in the ODE system. This leaves us with the system of variable

$$(\dot{x}_0, \dot{x}_1, \dot{x}_2, \ldots, \dot{x}_{N-1}, \dot{x}_N, \dot{u}_N).$$

An algebraic expression corresponding to (27) is found by considering a discrete approximation to (39), as in section 4.2. In the present case the approximate mass $\hat{\theta}(t)$ is

$$\begin{aligned}\hat{\theta}(t) &= \int_{x_i}^{x_{i+1}} (u - \omega u_x + \alpha)dx \\ &= \frac{1}{2}(x_{i+1} - xi)(u_{i+1} + u_i)(x_{i+1} - xi) - \omega(u_{i+1} - u_i) + \alpha(x_{i+1} - xi) \\ &\quad (i = 0, \ldots, N-1)\end{aligned} \tag{42}$$

The ODE system is integrated by the same NAG routine as in the PME case. For full details see [4].

Initial meshes are once again generated using a discrete form of the monitor, namely

$$M_{i+\frac{1}{2}} = \frac{u_i + u_{i+1}}{2} - \omega\frac{u_{i+1} - u_i}{x_{i+1} - x_i} + \alpha$$

In the numerical results below the parameters in the monitor function are set to be $\omega = \alpha = 1$. We analyse the performance of the moving mesh method in this application by comparing numerical solutions with a solution on a fine stationary regular mesh consisting of 10^5 nodes, using a semi-implicit scheme and adaptive time-stepping.

Figure 7 shows error convergence at both boundaries. Both convergence histories suggest that the method is again of second order accuracy. The value of ϵ has an effect, with large errors incurred for the faster diffusion rate.

Figure 8 shows the generated solutions for $\epsilon = 0.01$. Although the solutions were only generated over the domain $[0, 1]$ it is convenient to plot the profiles with their reflections in $x = 1$, to illustrate the node movement when two adjacent concentrations of dopant interact at the far boundaries, as happens in the real problem. Figure (9) shows the trajectories of the nodes.

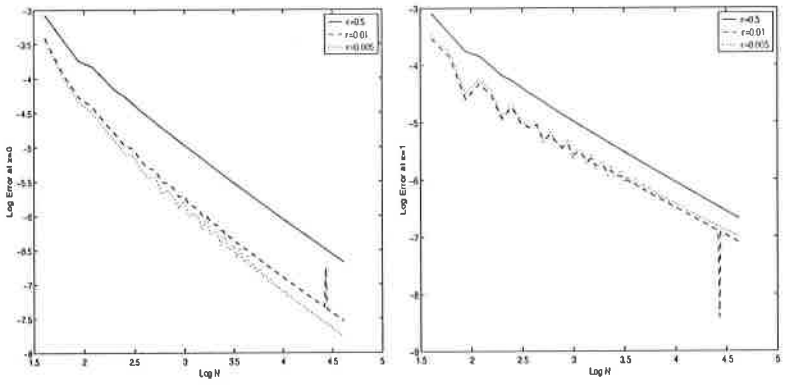

Figure 7: Absolute error at $x = 1$ and $x = 0$ (Top). Logarithmic plots of the error to show convergence (Bottom).

6.2 A Blow-up Problem

Finally we consider a problem involving solution blow-up. This illustrates that the method may be used to tackle problems which do not conserve mass. The approach is similar to that used for the semiconductor problem above, but an extra term is added to the ODE system, this time for the rate of change of the integral of mass over the domain.

The equation considered is the Fisher-type equation

$$u_t = u_{xx} + u^2 \tag{43}$$

[7] over a unit domain with initial and boundary conditions

$$u(x,0) = 20sin(\pi(\frac{1}{2} - x))$$
$$u(0,t)_x = u(1,t) = 0$$

This problem occurs in physical models which develop singularities at some finite time T. Examples of such behaviour exist in the solution of equations describing combustion in chemicals or chemotaxis, with the blow-up representing the ignition of a heated gas mixture. In particular, we seek a method to accurately reproduce the blow-up time T, given in [7] to be approximately 0.082291, with the blow-up taking place at the origin in the form of an isolated spike of increasingly narrow width.

We use the monitor function

$$M(u) = \alpha - \omega u_x. \tag{44}$$

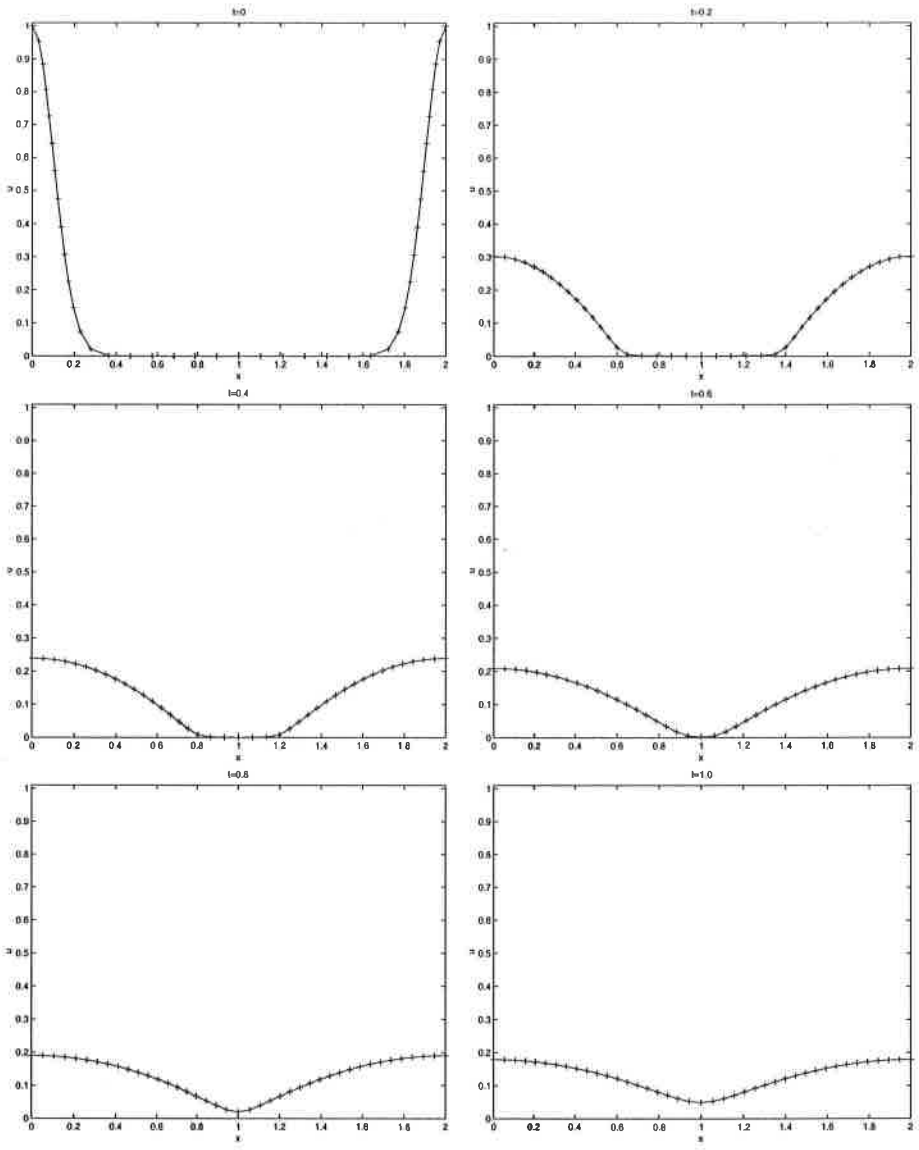

Figure 8: Approximate solution for the semiconductor problem with $\epsilon = 0.01$.

where it is anticipated that the presence of the first derivative will help to provide fine mesh spacing as the solution approaches blow-up and steep gradients develop, while the additional constant term α preserves a reasonable mesh size in the solution close to the right-hand boundary.

In deriving the appropriate moving mesh equation we still keep the same structure in the ODE system but add an extra equation to approximate to

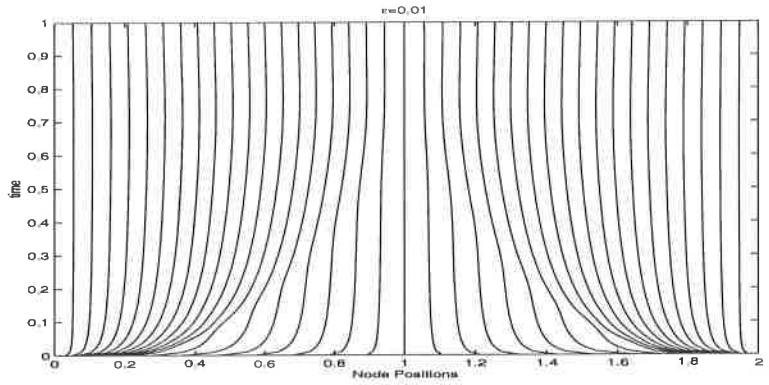

Figure 9: Node Trajectories for semiconductor moving mesh solution

the time derivative of total mass

$$\Theta_t = \int_0^1 u dx$$

This can be found by considering

$$\begin{aligned} \Theta_t &= \int_{x_0}^{x_N} u_t dx = u_x|_{x=x_N} + \int_{x_0}^{x_N} u^2 dx \\ &\approx \frac{(u_N - u_{N-1})}{(x_N - x_{N-1})} + \frac{1}{2}\sum_{j=1}^{N-1}(x_{j+1} - x_j)(u_j^2 + u_{j+1}^2) \end{aligned} \tag{45}$$

As in sections 3.2,4.2 the moving mesh equation results from differentiating the static EP with respect to time, i.e.

$$\frac{d}{dt}\int_{x_i}^{x_{i+1}} (\alpha - \omega u_x) dx = \dot{\theta} \tag{46}$$

leading to the ODE system

$$[\dot{x}(\alpha - \omega u_x)]|_{x_i}^{x_{i+1}} = \dot{\theta} + \dot{x}_i(\alpha - \omega u_x)|_{x=x_i} + \omega \left[u_{xx} + u^2\right]_{x_i}^{x_{i+1}}. \tag{47}$$

where

$$\dot{\theta} = \frac{-\omega}{N-1}\left[u_{xx} + u^2\right]_{x_1}^{x_N}.$$

with $\dot{x}_1 = \dot{x}_N = 0$.

The ODEs are solved as before. The resulting u solution again comes from discretising the stationary version of (46) and using the Dirichlet boundary condition on u at x_N.

Applying the moving mesh algorithm a satisfactory result is obtained. The integration reaches a time of $T = 0.08155173$, which is a reasonable estimate compared to the time to blow-up calculated by Budd et al for the same problem in [7]. Moreover, the value of u at $x = 0$ compares well with a value of 10^{10}. The left hand side of Figure 10 shows that the mesh trajectories tend towards the blow up region. As can be seen, the nodes move rapidly towards zero as the solution develops. To give an idea of the scale of the solution and the rapid growth, the right hand side of Figure 10 shows the maximum value of u with time.

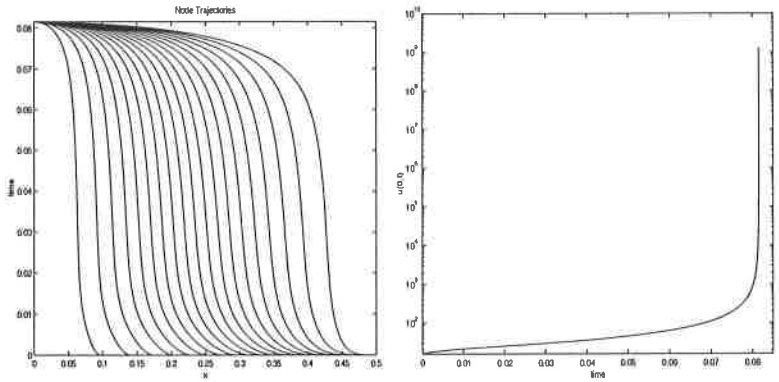

Figure 10: Mesh trajectories (Left) and evolution of $u(0,t)$ using a change of monitor function (44) with $\alpha = 1$, $\omega = 0.8$.

7 Conclusions and Further Work

In this report we have described a moving mesh method for non-linear parabolic partial differential equations with or without moving boundaries. The method relies on a monitor function whose local integral is a conserved fraction of the total integral. The resulting algebraic moving mesh equation is combined with the underlying PDE to form a differential equation system for the mesh points. Under discretisation the moving mesh equation allows the solution to be expressed in terms of the meshpoints, so only a single system of equations is required to be solved.

The method is described with reference to a porous media equation (PME) problem with a moving boundary. The choice of a density monitor function equal to the dependent variable, suggested by scale-invariance properties of the PME, is found not to resolve the solution at the moving boundary. The situation is redeemed in two ways, either by incorporating mesh subdivision into the method or by introducing alternative monitors.

For a fixed number of nodes the method is generalised to allow a gradient monitor to be used but this only partially solves the difficulty and a better solution is achieved with a combination monitor.

The method has been adapted to treat problems with Neumann conditions at fixed boundaries and also to handle problems with source terms which arise in blow-up problems, although again the monitors used are geometric rather than scale-invariant. Generally second order accuracy is achieved. The moving mesh method appears to be capable of simulating problems with non-linearities, moving boundaries and singularities in these types of problems.

References

[1] Baines, M.J. Grid Adaption via Node Movement. Applied Numerical Mathematics, vol 26, pp77-96, 1998.

[2] Barenblatt, G.I. Similarity, self-similarity, and intermediate asymptotics. New York : Consultants Bureau, 1979.

[3] Beckett, G., Mackenzie, J.A., Ramage, A. and Sloan, D.M. On the Numerical Solution of One-Dimensional Partial Differential Equations using Adaptive Methods based on Equidistribution. Technical Report No. Dept of Mathematics, University of Strathclyde, 2000 (submitted).

[4] Blake, K.W. Moving Mesh Methods for Non-linear Parabolic Partial Differential Equations. PhD Thesis, Dept of Mathematics, University of Reading, 2001.

[5] Blom, J.G., Sanz-Serna, J.M. and Verwer, J.G. On Simple Moving Grid Methods for One-Dimensional Evolutionary Partial Differential Equations. Journal of Computational Physics, vol 74, pp 191-213, 1988.

[6] Budd, C.J. and Collins, G. An Invariant Moving Mesh Scheme for the Non-linear Diffusion Equation. Technical Report No, School of Mathematics, University of Bath, 1996.

[7] Budd, C.J, Huang, W., and Russell, R.D. Moving Mesh Methods for Problems with Blow-up. SIAM Journal of Scientific and Statistical Computation, vol. pp., 1996.

[8] Budd, C.J, Chen, S. and Russell, R.D. New Self-Similar Solutions of the Non-linear Schrodinger Equation with Moving Mesh Computations. Journal of Computational Physics, vol 152, pp 756-789,1999.

[9] Budd, C.J., Collins, G.J., Huang, W., and Russell, R.D. Self-Similar Numerical Solutions of the Porous-Medium Equation using Moving Mesh Methods. Phil. Trans Roy. Soc. London, vol 357, pp 1047-1077,1999.

[10] Budd, C.J. and Piggott, M.D. The Geometric Integration of Scale-Invariant Ordinary and Partial Differential Equations. Journal of Computational and Applied Mathematics, vol 128, pp 399-422, 2001.

[11] DeBoor, C. Good Approximation by Splines with Variable Knots II. Springer Lecture Notes Series 363. Springer-Verlag, Berlin, 1973.

[12] Dorfi, E.A. and Drury, L.'O.C. Simple Adaptive Grids for 1-D Initial Value Problems. Journal of Computational Physics, vol 69, pp175-195, 1987.

[13] Dresner, L. Similarity Solutions of Non-linear Partial Differential Equations. Pitman Research Notes in Mathematics Series, London: Longman, vol 88, pp 349-388, 1983.

[14] Flaherty, J.E., Coyle, J.M. and Ludwig, R. On the Stability of Mesh Equidistribution Strategies for Time-Dependent Partial Differential Equations. Journal of Computational Physics, vol 62, pp 26-39, 1986.

[15] Huang, W. Ren, Y. and Russell R.D. Moving Mesh Partial Differential Equations (MMPDEs) based on the Equidistribution Principle. SIAM Journal on Numerical Analysis. vol 31, pp 709-730, 1994.

[16] Huang, W., Ren, Y. and Russell, R.D. Moving Mesh Methods based on Moving Mesh Partial Differential Equations. Journal of Computational Physics, vol 113, pp 279-290, 1994.

[17] Huang, W. and Russell, R.D. Analysis of Moving Mesh Partial Differential Equations with Spatial Smoothing. SIAM Journal on Numerical Analysis, vol 34, pp 1106-1126, 1997.

[18] Mackenzie, J.A. Moving Mesh Finite Volume Methods for One-Dimensional Evolutionary Partial Differential Equations. Technical Report No., Dept of Mathematics, University of Strathclyde. 1996.

[19] Mackenzie, J.A. and Robertson, M.L. A Moving Mesh Method for the Solution of the One-Dimensional Phase-Field Equations. Technical Report No., Dept of Mathematics, University of Strathclyde, 2000 (submitted).

[20] Murray, J.D. Mathematical Biology. Springer, New York, pp349-388, 1989.

[21] Numerical Algorithms Group. The NAG Fortran Library Manual. http://www.nag.co.uk/numeric/fl/manual/html/FLlibrarymanual.asp, NAG Ltd. 2001.

[22] Please, C.P. and Sweby, P.K. A Transformation to Assist Numerical Solution of Diffusion Equations. Numerical Analysis Report No., Dept of Mathematics, University of Reading, 1986.

[23] Qiu, Y. and Sloan, D.M. Numerical Solution of the Fisher's Equation using a Moving-Mesh Equation. University of Strathclyde, Dept of Mathematics, Technical Report No. 22/1997.

[24] Ren, Y and Russell, R.D. Moving Mesh Techniques based upon Equidistribution and their Stability. SIAM Journal of Scientific and Statistical Computing, vol 13, pp 1265-1286, 1992.

[25] Verwer, J.G. and Blom, J.G. and Furzeland, R.M. and Zegeling, P.A. Adaptive Methods for Partial Differential Equations. Society for Industrial and Applied Mathematics, vol. pp 160, 1989.

[26] White, A.B. On Selection of Equidistributing Meshes for Two-Point Boundary-Value Problems. SIAM Journal on Numerical Analysis, vol 16, pp 472-502, 1979.