A NEW METHOD OF SPATIAL ERROR CONTROL FOR THE FINITE ELEMENT METHOD ON CONVECTION DOMINATED PROBLEMS

P.K. Jimack
School of Computer Studies, University of Leeds, Leeds LS2 9JT, UK.

Abstract
We consider the finite element solution of a class of convection dominated partial differential equations using spatial adaptivity based upon a new strategy for the error control. This involves estimating the spatial error in the temporal derivative of the solution using the difference between the standard piecewise continuous finite element approximation and an approximation based upon a moving grid or discontinuous representation.

1 Introduction
For the efficient solution of most time-dependent partial differential equations (p.d.e.'s) in one, two or three space dimensions, some form of adaptivity is essential. In this paper we consider the use of a finite element semi-discretization in space combined with an adaptive temporal solution via standard variable order, variable step ordinary differential equation (o.d.e.) initial value problem (i.v.p.) software. In particular we focus upon the use of adaptivity in space and the error estimation that is necessary in order to establish a reliable criterion for when and where to adapt the finite element mesh.

The type of equation upon which we will concentrate is a typical convection-diffusion problem of the form

$$\frac{\partial u}{\partial t}(x,t) + ku(x,t) + \nabla \cdot (cu(x,t)) - \nu \Delta u(x,t) = f(x), \quad (1.1)$$

where $u(x,t) : \Omega \times (0,\infty) \rightarrow \mathbb{R}$ and the equation may be nonlinear with $c = c(u,x)$. For simplicity we will assume that $u$ satisfies homogeneous Dirichlet conditions on $\partial \Omega$ (a polygon in 2-d) and an initial condition of the form $u(x,0) = u^0(x) \in H^0(\Omega)$.

When $\nu$ is small the solution to (1.1) will often contain very steep fronts or boundary layers which may form from smooth initial data and/or move around the spatial domain. In order to capture these solution properties numerically a fine spatial mesh is required in appropriate regions, however these spatial regions may change. It is precisely such cases in which we are interested.

The adaptive approach which we use here is that of local refinement and derefinement of the spatial mesh according to whether some estimate of the error in the numerical solution is large or small respectively. A description of this algorithm may be found in [3] and in more detail in [4]. It is not dissimilar to the method used in [6]. Briefly, the mesh is an "unstructured" hierarchy of triangles which can be locally refined by subdividing triangles into four children, or locally derefined by combining four children back to be their original parent.

In the next section we motivate a new approach to the estimation of the spatial error in the finite element solution of equations such as (1.1). This is based upon comparing the fixed grid finite element solution on the latest mesh with a more accurate finite element solution in which the mesh would be allowed to deform or convect with the solution. This leads to the novel idea of estimating the spatial error by looking at the accuracy of the $\frac{\partial u}{\partial t}$ term in equation (1.1). Once this has been justified mathematically, a number of approximations are then made to produce an efficient algorithm for one-dimensional problems.

Section 3 discusses the extension of this error estimate and algorithm to higher spatial dimensions. A connection is made between our method and the discontinuous Galerkin method described in papers such as [10] and [11]. Finally, a difficult two-dimensional test problem is solved and the merits of the approach are discussed.

2 An Error Estimate for the Temporal Derivative

Solving (1.1) using piecewise linear finite elements in space for example, there are a number of possible methods of estimating the spatial error in order to drive the mesh adaptivity algorithm. One could use some sort
of a priori error indicator as in [9], or one could use an a posteriori estimate based upon extrapolation or estimating derivatives of the solution. The problem with the former is that it is rather ad hoc and does not reliably guarantee a good approximation of the error; it merely indicates where it is likely to be large. The second approach is usually more rigorous, however it is probably not ideally suited to convection dominated problems with large gradients. Most of the literature in this area is restricted to elliptic problems or parabolic problems dominated by diffusion ([1] or [2] for example).

We investigate an alternative approach initially based upon the Moving Finite Element (M.F.E.) method, first introduced in [8]. In this technique for solving problems such as (1.1) the finite element mesh is allowed to evolve with the solution by letting the position of the node points be degrees of freedom within the problem. This leads to a semi-discrete system of the form

\[
M(g(t)) \begin{bmatrix} \dot{u}(t) \\ \dot{\alpha}(t) \end{bmatrix} = f(g(t), g(t)),
\]

where \(g(t)\) is a vector of the conventional finite element degrees of freedom (the nodal amplitudes), \(g(t)\) is an ordered set of the positions of the finite element node points and \(M\) is a generalization of the usual Galerkin Mass matrix (see [8] for details). Since the finite element mesh can move with the solution this approach is well suited to convection dominated problems, however its major weakness is that it is difficult and expensive to effectively control this mesh movement. In our approach we do not actually allow the mesh to deform but instead, at each time-step, we use the values of \(\dot{\alpha}(t)\) and \(\dot{\alpha}(t)\) which may be calculated from (2.1) to give an estimate of \(\frac{\partial u}{\partial t}\) in equation (1.1). This is then used to form an estimate of the error in the corresponding term in the standard finite element solution, which is the one that is actually used.

We now show that by controlling the error in the finite element approximation to \(\frac{\partial u}{\partial t}\) at each time-step it is possible to control the error in \(u(x, t)\), the solution of (1.1), as integration in time proceeds. In order to do this it will be helpful to establish some notation. Firstly, we write (1.1) as

\[
\frac{\partial u}{\partial t} = \mathcal{L}(u)
\]

for an appropriate spatial operator \(\mathcal{L}\), and we represent the fully discrete solution at time \(t\) by \(u_h^k(x, t)\). Also, we assume that the integration in time is done using a one-step method of the form

\[
u_h^k(x, t_{n+1}) = u_h^k(x, t_n) + \Delta_{n+1} \phi(t_n, u_h^k(x, t_n), \Delta_{n+1})
\]

(2.2)

where \(\Delta_{n+1} = t_{n+1} - t_n\) is the time-step. Note that there is nothing in what follows which requires this scheme to be explicit and the extension to multistep methods is not difficult. The other definitions that we require are:

\(T(x, t) = \) Total error at time \(t\): \(u(x, t) - u_h^k(x, t)\),

\(u_h^k(x, t) = \) Finite element function represented by the local solution to the o.d.e. system over the \(n^{th}\) time-step (i.e. the precise solution of the o.d.e. system from \(t_n\) to \(t_{n+1}\), with \(u_h^k(x, t_n)\) as the initial data),

\(u_h^k(x, t) = \) Finite element approximation to \(u(x, t)\) over the \(n^{th}\) time-step (i.e. the projection of \(\mathcal{L}(u_h^k)\) onto the finite element test space),

\(e_h^k(x, t_{n+1}) = \) Local o.d.e. error in the \(n^{th}\) time-step.

Hence, we can now show that

\[
T(x, t_{n+1}) = u(x, t_{n+1}) - u_h^k(x, t_{n+1})
\]

\[
= u(x, t_n) + \int_{t_n}^{t_{n+1}} u(x, t) \, dt - u_h^k(x, t_n) - \Delta_{n+1} \phi(t_n, u_h^k(x, t_n), \Delta_{n+1})
\]

\[
= T(x, t_n) + \int_{t_n}^{t_{n+1}} u(x, t) \, dt - \Delta_{n+1} \phi(t_n, u_h^k(x, t_n), \Delta_{n+1})
\]

and

\[
e_h^k(x, t_{n+1}) = u_h^k(x, t_{n+1}) - u_h^k(x, t_{n+1})
\]

379
\[ T(x, t_{n+1}) - T(x, t_n) = e^{h,k}(x, t_{n+1}) + \int_{t_n}^{t_{n+1}} [u_t(x, t) - \nu u_t^h(x, t)] \, dt \]

\[ = e^{h,k}(x, t_{n+1}) + \Delta_n[u(x, \theta) - \nu u_t^h(x, \theta)] \]  

for some \( \theta \in (t_n, t_{n+1}) \).

Now, the first term on the right-hand-side of (2.3) is the local time error in the o.d.e. solver. If we require that our adaptive o.d.e. solver controls this error per unit length in time then we may ensure that

\[ |e^{h,k}(x, t_{n+1})| \leq \Delta_n \text{tol}_1, \]  

for some number \( \text{tol}_1 \) and hence also that the time global error is proportional to \( \text{tol}_1 \), see [5]. So, by demanding that the second term on the right-hand-side of (2.3) (the spatial error term), satisfies

\[ |u_t(x, \theta) - \nu u_t^h(x, \theta)| \leq \text{tol}_2 \]  

and that \( \text{tol}_1 = \epsilon \text{tol}_2 \) for some small \( \epsilon \), we can ensure (in a similar manner to that described in [5]) that the temporal error is dominated by the spatial error as given by (2.5). It is then possible to control this spatial error by remeshing.

The exact time derivative in (2.5) is not known and so the assumption is made that the M.F.E. solution provides a more accurate estimate of this than the fixed mesh solution. There is no theoretical guarantee, as yet, that this is correct, although there is plenty of experimental evidence for convection dominated problems. The term \( \nu u_t^h(x, \theta) \) in (2.5) is approximated by the computed solution. Thus the spatial error is approximated as

\[ u_t(x, t) - \nu u_t^h(x, t) \approx u_t^{(M.F.E.)}(x, t_n) - u_t^{h,k}(x, t_n), \]  

For this estimate to work the temporal error must be less than the spatial error so that the temporal error in \( \nu u_t^h(x, t) \) does not dominate the right side of (2.6) (which is why \( \text{tol}_1 = \epsilon \text{tol}_2 \)). There is also no way of knowing \( \theta \) in (2.5) and so we must try and impose this condition at the beginning and end of each step and assume this to be sufficient.

The local refinement and derefinement algorithm described in the introduction has been successfully implemented, in both one and two space dimensions, using the above error estimator. Examples may be found in [4]. In one-dimension in particular the method is extremely efficient since the M.F.E. solution can be obtained at very little cost by solving the problem in an element-by-element manner. In higher dimensions this is not the case since the matrix \( M \) in (2.1) can no longer be reduced to block diagonal form, making (2.1) far more expensive to solve.

3 Extensions to Higher Spatial Dimensions

The problem of obtaining a "local" M.F.E. solution in two or more space dimensions is discussed in [7]. The only obstacle to being able to do this is the existence of the second order term in equation (1.1). However, in most of the cases which are of interest to us, the value of \( \nu \) in this equation is small and numerical experiments in one and two dimensions seem to show that, for the purposes of the error estimate only, it is possible to ignore this second order term. This suggests that for larger values of \( \nu \) it may be possible to use an error splitting approach, where a more conventional type of estimate is used to calculate the contribution of the diffusion term to the error in \( \frac{\partial u}{\partial t} \).

When the Laplacian term in (1.1) is dropped for the purposes of calculating the approximate error, (2.6), we can again obtain a very cheap estimate, even in two or three dimensions. The expression \( u_t^{(M.F.E.)}(x, t_n) \) in (2.6), which is the M.F.E. approximation to \( \frac{\partial u}{\partial t} \) at the latest time, lies in the larger space of piecewise discontinuous functions on the current finite element mesh. In fact the "local" M.F.E. approach described in [7] consists of firstly projecting \( \mathcal{L}(u^{h,k}) \) onto the space of discontinuous piecewise linear elements and then restricting this projection to the space spanned by the Moving Finite Element functions. Hence a way of improving the efficiency of our estimate still further could be to drop this second projection stage and simply use a discontinuous piecewise linear approximation to \( \frac{\partial u}{\partial t} \). Again numerical
experiments have been performed which suggest that doing this still gives a reliable estimate of the error due to the convection term. Note that in one dimension this approach is essentially equivalent to the M.F.E. method.

The technique that we have now derived is beginning to look rather similar to the discontinuous Galerkin approach used in [10] or [11] for example. However there is at least one fundamental difference since, unlike in those papers, we always use a continuous approximation to \( u \), even for the purposes of the error estimate. It is only the approximation of \( \frac{\partial u}{\partial t} \) which we allow to become discontinuous and this is only in the error control routine. For this reason we have no need to worry about jumps in the solution across element boundaries which means that the order in which we obtain our local element-by-element approximation is unimportant. This therefore allows a much wider variety of non-linear problems to be solved than might otherwise be the case. In the example below, the error term of (2.5) is approximated using the discontinuous piecewise linear approximation to \( \frac{\partial u}{\partial t} \) rather than the local M.F.E. approximation, although the latter gives virtually identical results.

We finish this section by showing the computed solution to a non-linear problem of the form of (1.1). The equation is

\[
\frac{\partial u(x,t)}{\partial t} = \frac{u(x,t)}{\sqrt{x^2 + z^2}} \left( 1 - \frac{\partial u(x,t)}{\partial x} + x \frac{\partial u(x,t)}{\partial z} \right) + 0.005 \Delta u(x,t) \quad \text{where} \quad u(x,0) = \frac{1}{1 + e^{100(\sqrt{x^2 + z^2} - 0.3)}}, \tag{3.1}
\]

which we solve with homogeneous Neumann boundary conditions on the domain \((0,1) \times (0,1) \times (0,T)\), where, in the solution shown in figure 1, \( T = 1.0 \). This equation is chosen since it is almost identical to the polar equation

\[
\frac{\partial u}{\partial t} - u \frac{\partial u}{\partial r} = \nu \frac{\partial^2 u}{\partial r^2}, \quad \text{which has the solution} \quad u(r,t) = \frac{1}{1 + e^{100(r - 0.5t - 0.3)}}, \tag{3.2}
\]

when solved with the same initial condition. The only difference between the equations of (3.1) and (3.2) is in the small diffusion term, hence we are able to assess, at least qualitatively, the accuracy of the numerical solution to (3.1). Note that the finite element mesh is able to refine in the region where the steep front is at each time, and is then able to derefine once this front has moved on, hence an accurate solution has been obtained using a minimal number of degrees of freedom at each stage. The initial mesh is chosen so as to keep the \( L^2 \) norm of the error in the representation of the initial data below some tolerance, although if further refinement is required in order to approximate \( \frac{\partial u}{\partial t} \) sufficiently accurately, this will be performed. Also, the maximum amount of refinement allowed in the simulation has been restricted to just five levels in order to permit uncluttered plots of the finite element meshes (this also leads to a faster run-time of course). A variety of similar numerical examples can be found in [4].

4 Discussion

As is indicated by the numerical example of the last section, the approach described in this paper appears to work well when applied to the solution of a number of convection dominated problems of the form of (1.1). The idea of using the error in approximating \( \frac{\partial u}{\partial t} \) to control the refinement and derefinement algorithm has been introduced in order to allow the Moving Finite Element method to be used to form a comparison with the standard fixed grid solution. This has then been extended to work efficiently for a wider class of discontinuous approximations to \( \frac{\partial u}{\partial t} \).

We have not given a rigorous justification for our assumption that the discontinuous approximation to \( \frac{\partial u}{\partial t} \) is sufficiently more accurate than the continuous solution that we may use it as the basis for our error estimate. Clearly this is a subject for further research, especially in the light of the success of our numerical simulations. It is clear however that this estimate of the error term in (2.5) is not suitable for diffusion problems, although the approach of controlling the error in \( \frac{\partial u}{\partial t} \) is just as valid, one simply needs to use a more appropriate estimate.

Finally, all of the techniques discussed in this paper should work equally well in three space dimensions as in one or two, although the local refinement and derefinement procedure, for example, will be considerably more complicated to implement in such a case.

Acknowledgement

I would like to thank Dr. Martin Berzins for his many helpful suggestions on this work.

381
Figure 1: Grids and solutions obtained during the adaptive solution of equation (3.1) with homogeneous Neumann boundary conditions. The grids and solutions shown are those computed at times 0.0, 0.5, and 1.0.

References


