Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

P.A. Zegeling, J.G. Blom

An evaluation of the gradient-weighted moving-finite-element method in one space dimension

Department of Numerical Mathematics Report NM-R9006 March

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

An Evaluation of the Gradient-Weighted Moving-Finite-Element Method in One Space Dimension *)

P.A. Zegeling, J.G. Blom
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

Moving-grid methods are becoming increasingly popular for solving several kinds of parabolic and hyperbolic partial differential equations involving fine scale structures such as steep moving fronts and emerging steep layers. An interesting example of such a method is provided by the moving-finite-element (MFE) method. A difficulty with MFE, as with many other existing moving-grid methods, is the threat of grid distortion which can only be avoided by the use of penalty terms. The involved parameter tuning is known to be very important, not only to provide for a safe automatic grid-point selection, but also for efficiency in the time-stepping process. When compared with MFE, the gradient-weighted MFE (GWMFE) method has some promising properties to reduce the need of tuning. To investigate to what extent gradient-weighted MFE can be called robust, reliable and effective for the automatic solution of time-dependent PDEs in one space dimension, we have tested this method extensively on a set of five relevant example problems with a different solution behavior. All tests have been carried out using the BDF time integrator **SPGEAR** of the existing method-of-lines software package **SPRINT**.

1980 Mathematics subject classification: Primary:65M60. Secondary:65M20, 65M50.
1987 CR Categories: G.1.8.
Key Words & Phrases: partial differential equations, time-dependent problems, moving finite elements, method of lines.

Note: This report will be submitted for publication elsewhere.

*) This work has been carried out in connection with a joint CWI/Shell project on 'Adaptive Grids'. For this project Paul Zegeling has received support from the 'Netherlands Foundation for the Technical Sciences' (STW), future Technical Science Branch of the Netherlands Organization for the Advancement of Pure Research (NWO) (Contract no. CWI 59.0922).

1. Introduction

Moving-grid methods are becoming increasingly popular for several kinds of parabolic and hyperbolic partial differential equations (PDEs) involving fine scale structures such as steep moving fronts, emerging steep layers, pulses, shocks, etc. Moving-grid methods use nonuniform space grids and, like Lagrangian methods, move the grid continuously in the space-time domain while the discretization of the PDE and the grid selection are intrinsically coupled. Well-known examples are provided by the moving-finite-element (MFE) method originally proposed by Miller & Miller[15] and Miller[11], and by the moving-finite-difference (MFD) method discussed in Verwer, Blom, Furzeland & Zegeling[19] (see also references therein). The MFD method is restricted to problems in one space dimension and is strongly based on ideas to Dorfi & Drury[5].

However, because of the intrinsic coupling between the discretization of the PDE and the grid selection, the application of moving-grid methods is not without difficulties, not even in the relatively simple case of one space dimension. The main difficulty we are referring to is the threat of grid distortion which can only be avoided by using penalty terms which, to some extent, are artificial and invariably involve parameter tuning. The parameter tuning is known to be very important, not only to provide for a safe automatic grid-point selection, but also for efficiency in the time-stepping process. Another difficulty is that the automatic grid-point selection introduces nonlinear equations which may appear troublesome if handled with standard Newton solvers as commonly in use in implicit, stiff ODE solvers.

Due to these specific difficulties, the question arises as to how moving-grid methods combined with implicit, stiff ODE solvers (method-of-lines (MOL) approach) do compare with common fixed-grid MOL procedures concerning the important issues of efficiency and, in particular, robustness, reliability and ease of use. This is a natural question because, on the one hand, fixed-grid MOL procedures are known to become more and more popular, but, on the other hand, their use is limited when steep moving transitions must be resolved, since in such situations too many points in space and time may be needed.

In a previous evaluation report, see Furzeland, Verwer & Zegeling[8], we have attempted to provide insight in this question. There we have tested three moving-grid methods for time-dependent PDE problems in one space dimension, including the MFE and the above mentioned MFD method. On account of this investigation a moving-grid interface was developed meant for automatic use in combination with the MFD method and a stiff ODE integrator (see Blom & Zegeling[4]). The interface provides the possibility of letting grid points move in time and performs the spatial discretization of the PDE problem under consideration without additional programming effort for the user, completely similar as in standard, fixed-grid interfaces like those of the SPRINT package [2, 3] and of Sincovec & Madsen[16, 17].

In [8] we have also reported rather severe difficulties in applying MFE and announced a further examination of the gradient-weighted MFE (GWMFE) of Miller[12, 13]. The current evaluation report is completely devoted to the GWMFE method, again for the one-dimensional case. The gradient-weighting amounts to the use of weighting functions in the finite-element formulation that depend on the gradient u_x of the solution. This treatment results in a more robust process in that the parameter tuning becomes easier and also less critical. A second improvement, specifically concerning the implicit solution of the nonlinear system required in the time-stepping process, results from a particular block-diagonal preconditioning of the fully discretized equations (Miller[14]). One of the goals of the current examination therefore, is to find out whether GWMFE is suited to be included in the above mentioned moving-grid interface. In that case a user of the interface has the freedom to choose between two moving-grid methods. This way we hope to be able to cover a wider class of difficult problems from practice. For that matter an interesting point is to see how GWMFE handles solutions of strongly convection dominated convection-diffusion problems containing sharp corners and near-shocks, since these are very hard to solve with the already implemented MFD method.

The paper is divided into four sections and one appendix. In Section 2 we describe the main ideas of MFE and GWMFE and the implementation of the latter. Section 3 contains the results of extensive numerical experiments on a set of five test models. In this test set are included Burgers' equation with a small diffusion coefficient, a scalar diffusion problem describing a shifting pulse, a system of two nonlinear convection-reaction equations, a flame-propagation model with a heat source at the boundary, and the standard shocktube problem of gasdynamics. In Section 4 our conclusions and recommendations are summarized. Finally, the Appendix gives a catalogue of worked-out innerproducts used in Sections 2 and 3.

2. Description of the method

In this section an outline is given of GWMFE. Miller derived the method from his own moving-finite-element method (MFE). Since many basic properties of GWMFE are related very naturally to MFE properties, we first give a description of the MFE method.

2.1. MFE

Consider the scalar PDE problem

$$u_t = L(u), \quad x_L < x < x_R, \quad t > 0, \tag{2.1}$$

where L represents a differential operator involving only spatial derivatives up to second order. The space interval is supposed to be fixed for all times $t > 0$ under consideration. Corresponding to the common method-of-lines approach, we consider N time-dependent grid points

$$x_L = X_0 < \cdots < X_i(t) < X_{i+1}(t) < \cdots < X_{N+1} = x_R \tag{2.2}$$

On such a grid, MFE approximates the solution $u(x,t)$ of (2.1) by

$$u \approx U = \sum_{j=1}^{N} U_j(t)\alpha_j(x, \{X_i(t)\}) = \sum_{j=1}^{N} U_j(t)\alpha_j(x, X_{j-1}(t), X_j(t), X_{j+1}(t)) \tag{2.3}$$

where α_j are the standard piecewise linear basis functions

$$\alpha_j = \begin{cases} \dfrac{x - X_{j-1}}{\Delta X_j} & \text{for } X_{j-1} \leqslant x \leqslant X_j \\ \dfrac{X_{j+1} - x}{\Delta X_{j+1}} & \text{for } X_j \leqslant x \leqslant X_{j+1} \\ 0 & \text{elsewhere} \end{cases}$$

with $\Delta X_j := X_j - X_{j-1}$ and U_j the amplitudes at X_j. Differentiating U w.r.t. t and applying the chain rule gives

$$U_t = \sum_{j=1}^{N} \dot{U}_j\alpha_j + \dot{X}_j\beta_j \tag{2.4}$$

where

$$\beta_j = \begin{cases} -m_j\alpha_j & \text{for } X_{j-1} \leqslant x \leqslant X_j \\ -m_{j+1}\alpha_j & \text{for } X_j \leqslant x \leqslant X_{j+1} \\ 0 & \text{elsewhere} \end{cases}$$

and $m_j = \Delta U_j / \Delta X_j$ is the slope of the semi-discrete approximation U on $[X_{j-1}, X_j]$; ΔU_j is defined similarly as ΔX_j. It must be noted that β_j is piecewise linear discontinuous. The equations determining the semi-discrete unknowns U_j and X_j are now obtained in the standard Galerkin way by minimizing $\|R(U)\|_2^2$ w.r.t. $\dot{U}_i$ and $\dot{X}_i$, where

$$R(U) := U_t - L(U). \tag{2.5}$$

This minimization gives a system of $2N$ Ordinary Differential Equations in the $2N$ unknowns U_i and X_i:

$$\sum_{j=1}^{N} < \alpha_i, \alpha_j > \dot{U}_j + < \alpha_i, \beta_j > \dot{X}_j = < \alpha_i, L(U)>, \quad i = 1, \cdots, N, \tag{2.6a}$$

$$\sum_{j=1}^{N} < \beta_i, \alpha_j > \dot{U}_j + < \beta_i, \beta_j > \dot{X}_j = < \beta_i, L(U)>, \quad i = 1, \cdots, N, \tag{2.6b}$$

where $<,>$ denotes the usual L_2-innerproduct. It is clear that (2.6a) without the $\dot{X}$-innerproducts is just the standard Galerkin method applied to (2.1) using piecewise linear basis and test functions on a nonuniform grid. The time dependency of the grid is reflected in the $\dot{X}$-innerproducts in (2.6a) and the complete equation (2.6b).

Working out the innerproducts and defining the vector

$$Y := (U_1, X_1, \cdots, U_i, X_i, \cdots, U_N, X_N)^T,$$

we arrive at the semi-discrete MFE system

$$\mathscr{A}(Y)\,\dot{Y} = G(Y), \quad t > 0, \quad Y(0) \text{ given}, \tag{2.7}$$

where $\mathscr{A}(Y)$ is a block-tridiagonal matrix, the so-called mass-matrix, containing quantities from the left-hand sides of (2.6), whereas the only problem-specific terms are contained in the vector $G(Y)$.

This ODE-system must be integrated numerically to obtain the required fully discretized solution. Before starting to integrate in time, we must ask ourselves whether (2.7) represents a well-defined system. The minimization of $\|R(U)\|_2^2$ (cf. (2.5)) has a unique solution if and only if the basis functions $\{\alpha_j\}$ and $\{\beta_j\}$ are linearly independent, which is only the case as long as $m_j \neq m_{j+1}$. But even if the solution exists and is unique the question remains whether (2.7) is 'easily' solvable. A natural requirement for that is regularity of the mass-matrix $\mathscr{A}(Y)$ to avoid the problem of solving a DAE system of index 1 or higher. Concerning this, it can be shown that $\mathscr{A}(Y)$ is singular in exactly two situations (cf. Wathen[21]).

The first singularity is caused by the same reason as above and is called parallelism, which means that the approximation U has zero second differences at some node ($m_k = m_{k+1}$ for some $k \in \{1, \cdots, N\}$). This implies that the determinant of $\mathscr{A}$ is zero. In other words, system (2.7) becomes singular whenever a straight line can be drawn through the three neighboring points (X_{i-1}, U_{i-1}), (X_i, U_i) and (X_{i+1}, U_{i+1}). In physical terms this means that, in absence of curvature ($u_{xx} = 0$ locally), the method has no way to determine in which direction the grid points should be moved.

The second degeneracy of $\mathscr{A}$ arises whenever two nodes are coming too close together. $\mathscr{A}$ will then become very ill-conditioned and numerically singular. In fact there is nothing in the method which will prevent nodes from crossing. Hence one will need some mechanism to control the grid-point motion.

Furthermore, the nonlinear steady-state system $G(Y) = 0$ may exhibit degeneracies as well, for instance, in the case of parallelism.

To overcome these problems, Miller[12] introduces the following regularization terms (penalty functions) in the residual minimization. Instead of $\|R(U)\|_2^2$ the minimization is carried out for

$$\|R(U)\|_2^2 + \sum_{j=1}^{N+1} (\epsilon_j \Delta\dot{X}_j - S_j)^2, \tag{2.8}$$

where

$$\epsilon_j^2 = \frac{C_1^2}{\Delta X_j - \delta}, \quad \epsilon_j S_j = \frac{C_2^2}{(\Delta X_j - \delta)^2} \tag{2.9}$$

with C_1, C_2 and δ small, user-chosen, constants. In particular, δ serves as a user-defined minimum node distance. The modifications involved are only made to the grid-point equations (2.6b) and the combined effect is to add

$$\epsilon_i^2 \Delta\dot{X}_i - \epsilon_{i+1}^2 \Delta\dot{X}_{i+1} \quad \text{and} \quad \epsilon_i S_i - \epsilon_{i+1} S_{i+1}$$

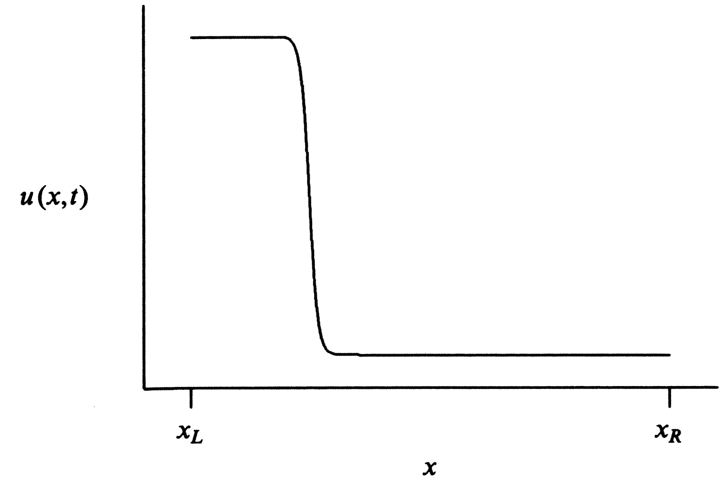
to the left- and right-hand side, respectively. The ϵ-terms serve to avoid parallelism. It can be shown that the addition of these terms renders the mass-matrix $\mathscr{A}$ positive definite [15], and thus regular. They represent a form of 'internodal' viscosity, since they penalize relative motion between the nodes and, provided the penalty is sufficiently large to take over before the mass-matrix becomes numerically singular, result in the degenerate nodes being carried along with the rest of the solution. The ϵ-terms do prevent node overtaking in a dynamic way since the internodal viscosities become infinite as ΔX tends to zero; however over longer time intervals degenerate nodes (those caught in straight line segments where they are unneeded) may still slowly drift together. The S-terms, sometimes called internodal spring forces, serve to prevent this long term numerical drift. For a clarification of the effect of the internodal spring forces, we refer to Herbst et al.[9].

As for any other method, the regularization is somewhat heuristic and necessarily problem-dependent. For example, if C_1 is chosen too large, the grid movement is restricted ($C_1 \rightarrow \infty$ gives a non-moving grid) with the result that there may not be sufficient refinement in regions of large spatial activity (a typical phenomenon is then that the grid moves slower than a front region). On the other hand, if C_1 is too small, the mass-matrix $\mathscr{A}$ may become numerically singular. Also of great importance is that the minimum node distance δ be small enough in relation to the anticipated small-scale structure. However, too small values of δ and C_2 may allow numerical errors to lead to near node overtaking (or even worse), which is a source of severe numerical difficulties in the time integration, even for the most robust stiff solver. When nodes drift extremely close together, the sets of nonlinear algebraic equations to be solved at each time step are likely to become badly conditioned.

As can be seen in the numerical experiments in [8] (using a straightforward implementation without the features mentioned in Section 2.3), it is not possible to give a problem-independent interval for the parameters C_1, C_2 and δ, for which the MFE method solves the PDE properly, proper in the sense of reliability of the obtained solution w.r.t. the user-chosen penalty parameters, and time-integrational aspects, respectively. Among others, for this reason the gradient-weighted MFE method has been developed.

2.2. GWMFE

Let us consider a PDE problem of the form (2.1) with a steep front solution $u(x,t)$ as pictured below for a given point of time t.



Outside the front region the solution is nearly constant resulting in nearly zero residual values (2.5). Because the grid points are allowed to move, the minimization procedure thus will send almost all points into the steepest part of the solution to fit the nonzero residual there. Apart from the fact that points are then wasted, this leads to numerical problems since the grid points may come very close to one another. The penalty terms introduced in (2.8) will partly remedy the situation, but this may require subtle tuning and, as already mentioned in the previous section, the practical experience with MFE is that tuning alone is not always sufficient for a good performance.

The gradient-weighting, as incorporated in GWMFE, aims at de-emphasizing the steep parts of the solution and, as a positive side result, at reducing the need for tuning. For this purpose the minimization of the residual (2.5) is now carried out using a weighted L_2-norm. For a functional ϕ of the solution $u(x)$ (suppressing the time dependence) this norm is defined by

$$|||\phi||| := \left[\int_{x_L}^{x_R} \frac{\phi^2(x)}{\sqrt{1+u_x^2}} dx \right]^{1/2} \tag{2.10}$$

and the new minimization w.r.t. $\dot{U}_i$, $\dot{X}_i$ is then applied to

$$|||R(U)|||^2 = \int_{x_L}^{x_R} \frac{(R(U))^2}{\sqrt{1+U_x^2}} dx. \tag{2.11}$$

Outside the steep front the weighting factor has little effect, since in those regions $U_x \approx 0$. However, inside the front region the spatial gradient admits very large values so that the integrand in (2.11) is much smaller than the residual itself. Due to this fact, the new minimization is supposed to send considerably less grid points into the steep front. In [12] Miller gives a geometrical interpretation of this idea, to which the interested reader is referred.

The minimization gives, as before, a system of $2N$ ODEs in the $2N$ unknowns U_i and X_i :

$$\sum_{j=1}^{N} < \alpha_i, \alpha_j w > \dot{U}_j + < \alpha_i, \beta_j w > \dot{X}_j = < \alpha_i, L(U)w>, \quad i = 1, \cdots, N, \tag{2.12a}$$

$$\sum_{j=1}^{N} < \beta_i, \alpha_j w > \dot{U}_j + < \beta_i, \beta_j w > \dot{X}_j = < \beta_i, L(U)w>, \quad i = 1, \cdots, N, \tag{2.12b}$$

where the weighting function $w = w(U_x)$ is defined by

$$w(U_x) = \frac{1}{\sqrt{1+U_x^2}}. \tag{2.13}$$

A nice property of w, due to the piecewise linear approximation (2.3), is the fact that it is constant on each cell, i.e.,

$$w_i = \frac{1}{\sqrt{1+m_i^2}}, \qquad \text{for } X_{i-1} \leqslant x \leqslant X_i. \tag{2.14}$$

A catalogue of worked-out innerproducts is given in an appendix. Like before, insertion of all innerproducts yields the semi-discrete GWMFE system of the form

$$\mathscr{A}_g(Y)\,\dot{Y} = G_g(Y). \tag{2.15}$$

Also in this case, the mass-matrix $\mathscr{A}_g$ may become singular. It is known that singularity occurs if we have parallelism. It is also known that in case of parallelism the steady-state system $G_g(Y) = 0$ has at least two linearly dependent equations. In order to prevent these singularities, Miller[13] has suggested to carry out the minimization for the penalized expression

$$|||R(U)|||^2 + \sum_{i=1}^{N+1} (\epsilon_i \dot{l}_i - S_i)^2, \tag{2.16}$$

where $\epsilon_i^2 := \dfrac{A^2}{l_i}$, $\epsilon_i S_i := \dfrac{B^2}{l_i^2}$, with A and B user-chosen constants, and

$$l_i := \sqrt{(\Delta X_i)^2 + (\Delta U_i)^2}. \tag{2.17}$$

In contrast with MFE, the modifications involved are made to all variables $\dot{U}_i$ and $\dot{X}_i$ in both equation (2.12a) and (2.12b). The combined effect is to add

$$\begin{aligned} &- \epsilon_i^2 \frac{(\Delta U_i)^2}{l_i^2} \dot{U}_{i-1} - \epsilon_i^2 \frac{\Delta X_i \Delta U_i}{l_i^2} \dot{X}_{i-1} + (\epsilon_i^2 \frac{(\Delta U_i)^2}{l_i^2} + \epsilon_{i+1}^2 \frac{(\Delta U_{i+1})^2}{l_{i+1}^2}) \dot{U}_i \\ &+ (\epsilon_i^2 \frac{\Delta X_i \Delta U_i}{l_i^2} + \epsilon_{i+1}^2 \frac{\Delta X_{i+1} \Delta U_{i+1}}{l_{i+1}^2}) \dot{X}_i - \epsilon_{i+1}^2 \frac{(\Delta U_{i+1})^2}{l_{i+1}^2} \dot{U}_{i+1} - \epsilon_{i+1}^2 \frac{\Delta X_{i+1} \Delta U_{i+1}}{l_{i+1}^2} \dot{X}_{i+1} \\ &- \epsilon_i S_i \frac{\Delta U_i}{l_i} + \epsilon_{i+1} S_{i+1} \frac{\Delta U_{i+1}}{l_{i+1}} \end{aligned} \tag{2.18a}$$

to the left part of equation (2.12a) and

$$- \epsilon_i^2 \frac{\Delta X_i \Delta U_i}{l_i^2} \dot{U}_{i-1} - \epsilon_i^2 \frac{(\Delta X_i)^2}{l_i^2} \dot{X}_{i-1} + (\epsilon_i^2 \frac{\Delta X_i \Delta U_i}{l_i^2} + \epsilon_{i+1}^2 \frac{\Delta X_{i+1} \Delta U_{i+1}}{l_{i+1}^2}) \dot{U}_i$$
$$+ (\epsilon_i^2 \frac{(\Delta X_i)^2}{l_i^2} + \epsilon_{i+1}^2 \frac{(\Delta X_{i+1})^2}{l_{i+1}^2}) \dot{X}_i - \epsilon_{i+1}^2 \frac{\Delta X_{i+1} \Delta U_{i+1}}{l_{i+1}^2} \dot{U}_{i+1} - \epsilon_{i+1}^2 \frac{(\Delta X_{i+1})^2}{l_{i+1}^2} \dot{X}_{i+1}$$
$$- \epsilon_i S_i \frac{\Delta X_i}{l_i} + \epsilon_{i+1} S_{i+1} \frac{\Delta X_{i+1}}{l_{i+1}} \tag{2.18b}$$

to the right part of equation (2.12b), respectively. It is clear that, with these modifications, GWMFE produces equations that are even more complicated and nonlinear than the penalized MFE equations (2.6) (see also Section 2.3).

As for MFE, the 'internodal viscosity' terms ϵ_i^2 serve to avoid parallelism. This means that the parameter A provides for the regularity of the mass-matrix $\mathcal{A}_g$ in the near degenerate situation of an almost flat solution. Likewise, the 'internodal spring' terms $\epsilon_i S_i$ take over to regularize the semi-discrete system in the steady-state case $G_g = 0$ whenever parallelism occurs. In applications, it is often possible to put B equal to zero so that only the parameter A remains. A third penalty parameter, such as the δ in MFE, is not considered in the present form of ϵ_i or $\epsilon_i S_i$. The direct analogue $l_i - \delta$ is redundant: it is unlikely that l_i tends to zero because this would require that both $\Delta X_i \rightarrow 0$ *and* $\Delta U_i \rightarrow 0$. Leaving out the penalty parameter to refrain ΔX_i from becoming zero might be defended by noting that GWMFE is supposed to send considerably less points in the steep front of the solution. It is interesting to remark that l_i may be associated with the arclength of the solution u on the i-th cell. Since u is approximated by a piecewise linear function U, this can be derived by using the fact that U_x is constant on each cell:

$$l_i = \sqrt{(\Delta X_i)^2 + (\Delta U_i)^2} = \Delta X_i \sqrt{1+(\frac{\Delta U_i}{\Delta X_i})^2} = \int_{X_{i-1}}^{X_i} \sqrt{1+(\frac{\Delta U_i}{\Delta X_i})^2}\, dx$$
$$= \int_{X_{i-1}}^{X_i} \sqrt{1+m_i^2}\, dx = \int_{X_{i-1}}^{X_i} \sqrt{1+U_x^2}\, dx.$$

It must be noted that we derived MFE and GWMFE for scalar PDEs. However, the foregoing can be generalized very naturally to a system of PDEs by replacing the residual (2.16) by

$$\sum_{k=1}^{NPDE} \{ W_k \,|||\, U_t^k - L^k(U) \,|||_k^2 + \sum_{j=1}^{N+1} (\epsilon_{jk} \dot{l}_{jk} - S_{jk})^2 \}, \tag{2.19}$$

where k denotes the k-th PDE component, $NPDE$ equals the total number of PDEs, and

$$|||\phi|||_k^2 := \int_{x_L}^{x_R} \frac{\phi^2(x)}{\sqrt{1+(U_x^k)^2}} dx,$$
$$(\epsilon_{jk})^2 := \frac{(A_k)^2}{l_{jk}},\ \epsilon_{jk} S_{jk} := \frac{(B_k)^2}{(l_{jk})^2},\ l_{jk} := \sqrt{(\Delta U_j^k)^2 + (\Delta X_j)^2}.$$

Here W_k represents a weighting factor to emphasize, if wanted, a particular PDE component. In our tests we have taken $W_k = 1$ for all components. Likewise, A_k and B_k have been chosen to be independent of k.

2.3. Implementation

The test results with GWMFE in [13] were obtained with the GWMFE1DS code developed by N. Carlson and K. Miller. In that code a second order Diagonally Implicit Runge-Kutta method (DIRK2) has been used as time integrator for the ODE system (2.15). Miller conjectured [14] that it would be profitable to use a higher order stiff ODE solver like the SPGEAR module in SPRINT. We therefore disconnected the modules of GWMFE1DS which compute the residual and coupled them directly to SPRINT, using the stiff BDF code SPGEAR as time integrator.

In this subsection we will discuss some of the 'implementation tricks' in GWMFE1DS which we feel to contribute significantly to the performance of the code and which are not previously described in the open literature by the authors Carlson and Miller.

But firstly we would like to give the reader an idea of the complexity of the ODE system (2.15). To that intent we work out equation (2.12) + penalty terms for the scalar PDE

$$u_t = \epsilon u_{xx} + (f(t,x,u))_x + g(t,x,u).$$

Let w_i be defined by (2.14) and l_i by (2.17). Then (2.12a) + (2.18a) yield for $i = 1, \cdots, N$,

$$(\frac{w_i}{6}\Delta X_i - \frac{A^2}{l_i^3}(\Delta U_i)^2)\dot{U}_{i-1} + (-\frac{w_i}{6}\Delta U_i - \frac{A^2}{l_i^3}\Delta X_i\Delta U_i)\dot{X}_{i-1} +$$

$$(\frac{w_i}{3}\Delta X_i + \frac{A^2}{l_i^3}(\Delta U_i)^2 + \frac{w_{i+1}}{3}\Delta X_{i+1} + \frac{A^2}{l_{i+1}^3}(\Delta U_{i+1})^2)\dot{U}_i +$$

$$(-\frac{w_i}{3}\Delta U_i + \frac{A^2}{l_i^3}\Delta X_i\Delta U_i - \frac{w_{i+1}}{3}\Delta U_{i+1} + \frac{A^2}{l_{i+1}^3}\Delta X_{i+1}\Delta U_{i+1})\dot{X}_i +$$

$$(\frac{w_{i+1}}{6}\Delta X_{i+1} - \frac{A^2}{l_{i+1}^3}(\Delta U_{i+1})^2)\dot{U}_{i+1} + (-\frac{w_{i+1}}{6}\Delta U_{i+1} - \frac{A^2}{l_{i+1}^3}\Delta X_{i+1}\Delta U_{i+1})\dot{X}_{i+1} =$$

$$\frac{B^2}{l_i^3}\Delta U_i - \frac{B^2}{l_{i+1}^3}\Delta U_{i+1}$$

$$+ \epsilon(-\ln(m_i + \sqrt{m_i^2 + 1}) + \ln(m_{i+1} + \sqrt{m_{i+1}^2 + 1}))$$

$$+ w_i\left[f(t,X_i,U_i) - \frac{1}{\Delta X_i}\int_{X_{i-1}}^{X_i} f(t,x,U(x))dx\right]$$

$$+ w_{i+1}\left[-f(t,X_i,U_i) + \frac{1}{\Delta X_{i+1}}\int_{X_i}^{X_{i+1}} f(t,x,U(x))dx\right]$$

$$+ w_i\int_{X_{i-1}}^{X_i}\alpha_i g(t,x,U(x))dx + w_{i+1}\int_{X_i}^{X_{i+1}}\alpha_i g(t,x,U(x))dx. \qquad (2.20a)$$

(2.12b) together with (2.18b) gives for $i = 1, \cdots, N$,

$$(-\frac{w_i}{6}\Delta U_i - \frac{A^2}{l_i^3}\Delta X_i\Delta U_i)\dot{U}_{i-1} + (\frac{w_i}{6}m_i\Delta U_i - \frac{A^2}{l_i^3}(\Delta X_i)^2)\dot{X}_{i-1} +$$

$$(-\frac{w_i}{3}\Delta U_i + \frac{A^2}{l_i^3}\Delta X_i\Delta U_i - \frac{w_{i+1}}{3}\Delta U_{i+1} + \frac{A^2}{l_{i+1}^3}\Delta X_{i+1}\Delta U_{i+1})\dot{U}_i +$$

$$(\frac{w_i}{3}m_i\Delta U_i + \frac{A^2}{l_i^3}(\Delta X_i)^2 + \frac{w_{i+1}}{3}m_{i+1}\Delta U_{i+1} + \frac{A^2}{l_{i+1}^3}(\Delta X_{i+1})^2)\dot{X}_i +$$

$$(-\frac{w_{i+1}}{6}\Delta U_{i+1} - \frac{A^2}{l_{i+1}^3}\Delta X_{i+1}\Delta U_{i+1})\dot{U}_{i+1} + (\frac{w_{i+1}}{6}m_{i+1}\Delta U_{i+1} - \frac{A^2}{l_{i+1}^3}(\Delta X_{i+1})^2)\dot{X}_{i+1} =$$

$$\begin{aligned}
&\frac{B^2}{l_i^3}\Delta X_i - \frac{B^2}{l_{i+1}^3}\Delta X_{i+1} \\
&+ \epsilon(\sqrt{m_i^2+1} - \sqrt{m_{i+1}^2+1}) \\
&- w_i m_i \left[f(t,X_i,U_i) - \frac{1}{\Delta X_i}\int_{X_{i-1}}^{X_i} f(t,x,U(x))dx \right] \\
&- w_{i+1}m_{i+1}\left[-f(t,X_i,U_i) + \frac{1}{\Delta X_{i+1}}\int_{X_i}^{X_{i+1}} f(t,x,U(x))dx \right] \\
&- w_i m_i \int_{X_{i-1}}^{X_i} \alpha_i g(t,x,U(x))dx - w_{i+1}m_{i+1}\int_{X_i}^{X_{i+1}} \alpha_i g(t,x,U(x))dx.
\end{aligned} \tag{2.20b}$$

It is obvious that the resulting system is extremely nonlinear.

Note, that for the gradient-weighted MFE method the evaluation of *both* the innerproducts $< \alpha_i, u_{xx}w >$ and $< \beta_i, u_{xx}w >$ has to be interpreted in the sense of 'mollification', i.e., the piecewise linear function U is smoothed at the nodal points (c.f. Miller[12, 15]). The ϵ terms in the right-hand side of (2.20a) and (2.20b) are the limits obtained for the 'mollified' innerproducts if the mollification parameter tends to zero.

The implementation of the 'u_{xx}-terms' has to be done carefully because both the formulae $-\ln(m_i + \sqrt{m_i^2+1}) + \ln(m_{i+1} + \sqrt{m_{i+1}^2+1})$ and $\sqrt{m_i^2+1} - \sqrt{m_{i+1}^2+1}$ are susceptible to loss of accuracy if m_i and m_{i+1} are small and the first formula also if m_i or m_{i+1} is large and negative. In **GWMFE1DS** $< \beta_i, u_{xx}w >$ is evaluated as

$$\sqrt{m_i^2+1} - \sqrt{m_{i+1}^2+1} = \frac{m_i^2}{1 + \sqrt{m_i^2+1}} - \frac{m_{i+1}^2}{1 + \sqrt{m_{i+1}^2+1}}$$

which gives automatically the correct expression even for small values of m_i. In $< \alpha_i, u_{xx}w >$ $\ln(m_i + \sqrt{m_i^2+1})$ is evaluated as

$$\text{sign}(m_i)\ln(|m_i| + \sqrt{m_i^2+1})$$

to avoid the problems for large and negative m_i, and in case $\eta = m_i / \sqrt{m_i^2+1}$ is small as a truncated Taylor series, viz.,

$$\ln(m_i + \sqrt{m_i^2+1}) = \tfrac{1}{2}\ln(\frac{1+\eta}{1-\eta}) \approx \eta + \tfrac{1}{3}\eta^3 + \tfrac{1}{5}\eta^5 + \tfrac{1}{7}\eta^7.$$

A second problem which arises if one would implement the method straightforwardly within the method-of-lines context is that a tolerance of, say, 10^{-4} for both the time error and the convergence to the solution of the nonlinear system is quite insufficient if the distance of two nodes is also of order 10^{-4}. Therefore we have, following the **GWMFE1DS** implementation, used as acceptance criterion for both the time error and the convergence of the Newton process that as well

$$\|v / tol\| < 1$$

should hold as

$$\max_i \frac{|v(X_{i+1}) - v(X_i)|}{(X_{i+1} - X_i)\,\rho} < 1, \tag{2.21}$$

where v is a vector either containing an estimate of the time error or the last correction in the Newton process, and ρ a user-defined parameter to indicate what weight should be given to the relative error tolerance on node distance. This implies that for $0 < \rho \leqslant 1$ the 'uncertainty' in ΔX_i will not be larger than ΔX_i itself. Note, that for $\rho \rightarrow \infty$ the old criterion is obtained.

Another feature that is implemented in `GWMFE1DS` and which we also adopted is the block-diagonal preconditioning of the residual system

$$\tilde{R}_g(Y) := \mathscr{A}_g(Y) \frac{Y - Z}{\Delta t\, d} - G_g(Y) = 0, \tag{2.22}$$

Where $(Y - Z)/(\Delta t\, d)$ is in our case the BDF substitute for $\dot{Y}$, with Z a vector depending on information from previous time steps and d a parameter that depends on the integration method in use.

This preconditioning is prompted by the results of Wathen[21] for the MFE mass-matrix $\mathscr{A}$ in (2.7). He proved that premultiplying $\mathscr{A}$ by a block-diagonal matrix $\mathscr{D}^{-1}(Y)$ where

$$D_i = A_{ii} = \begin{bmatrix} <\alpha_i, \alpha_i> & <\beta_i, \alpha_i> \\ <\alpha_i, \beta_i> & <\beta_i, \beta_i> \end{bmatrix}$$

results in a matrix $\mathscr{D}^{-1}(Y)\mathscr{A}(Y)$ which is very well-conditioned, in fact the condition number is even independent of the grid and the solution. Although the effects of preconditioning the residual system (2.22) with $\mathscr{D}_g^{-1}(Y)$ (the analogue of $\mathscr{D}^{-1}(Y)$ in case of gradient-weighting) has not yet been analytically shown, numerical results suggest that it has a considerable influence on the condition number of the Jacobian of the nonlinear system (2.22) too. Therefore we solve, unless otherwise indicated, not (2.22) but instead

$$\mathscr{D}_g^{-1}(Y)\, \tilde{R}_g(Y) = 0. \tag{2.22'}$$

3. Numerical experiments

In this section we discuss test results obtained with our implementation of the GWMFE method for five example problems, viz., (I) Burgers' equation, a scalar model for nonlinear convection-diffusion phenomena; for this PDE we took two different initial solutions, (II) a linear heat conduction problem with a shifting and oscillating pulse as solution, (III) a system of two nonlinear convection-reaction equations representing two opposite traveling pulses, (IV) a flame-propagation model with a heat source at the boundary, and (V) the well-known shocktube problem of gasdynamics. With these five problems we test the performance of the GWMFE method on a wide variety of solutions having a high degree of spatial activity, ranging from steep moving wave fronts to pulses and emerging and dying layers.

To get an impression of the dependence of the GWMFE method on the penalty parameters, the first Burgers' problem was tested for a large set of penalty parameter values A^2 and B^2. Moreover, for this problem the robustness of GWMFE was compared with respect to that of MFE as tested in [8]. All other problems were run with a smaller, and for every problem the same, range of penalty parameter values. Note that Miller suggests to take for B^2 a small value, say 10^{-10}. We chose the values 0 and 10^{-8}; 0 because we think that in general this is the best choice, and 10^{-8} to show the behavior of the GWMFE method for a somewhat larger value of B^2.

For all runs the 'cell-width'-tolerance parameter ρ from (2.21) was taken 0.1 and block-diagonal preconditioning was used in solving the nonlinear system with Newton. For a few cases we tried the effect of these 'implementation tricks'. The relative error tolerance on cell widths was, as can be expected, especially effective for the problems with a steep moving wave as solution; e.g., without this feature Burgers' problem often broke at the point where the shock reaches the boundary due to node crossing. Block-diagonal preconditioning was of great benefit for the condition number of the Jacobian of the nonlinear system. Without preconditioning the condition number was frequently of the order of the inverse of the machine precision (say 10^{14} with a machine precision of $\approx 10^{-16}$).

Preconditioning reduced it to $\approx 10^7$. The actual speedup was in view of these numbers not even that large, but it is clear that preconditioning makes the method much more robust.

The integrals resulting from the innerproducts $< \alpha_i, L(U)w >$ and $< \beta_i, L(U)w >$ were evaluated exactly (cf. Appendix) unless indicated otherwise. If numerical quadrature was used Boole's rule was applied (closed Newton-Cotes with error $O(h^7)$). We have also tried Simpson's rule ($O(h^5)$) but this gave, e.g., for problem IV, far worse results. This difficulty with numerical quadrature has already been mentioned by Miller[11] in his extensive testing of the MFE method.

The results will be presented in tables and for a few parameter choices in plots wherein marks will indicate the GWMFE approximation and the solid line the exact solution. If no exact solution was available, we used a very accurate numerical reference solution.

In the description of the experiments the following notation has been used:

Δt_0 initial step size,
TOL time-tolerance value (absolute and relative) for the SPGEAR integrator,
NPTS number of grid points,
STEPS number of successful time steps,
JACS number of Jacobian evaluations,
CTF number of correction time failures,
ETF number of error time failures,
CPU normalized CPU-time, i.e., CPU := CPU-secs / CPU-secs$_{\min}$, where CPU-secs$_{\min}$ is the minimum number of CPU seconds used for the problem under consideration,
ORD average order used by the time integrator measured over the whole time range.

Finally, we give marks for the quality of the computed solution and the (smoothness of the) grid: ++ (very good), + (good), □ (reasonable), − (bad), and −− (very bad). × indicates that GWMFE broke down during the run.

3.1. Problem I: Burgers' equation

This model, which can be considered as the simplest, non-trivial 1-D case for the Navier-Stokes equations, possesses a nonlinear convection term combined with a very small diffusion term

$$u_t = \epsilon u_{xx} - u u_x, \qquad 0 < x < 1, \quad t > 0, \qquad 0 < \epsilon << 1. \tag{3.1}$$

We make a distinction between two specific problems (both stemming from Miller):

a) the initial condition is the smooth function

$$u\,|_{t=0} = \sin(2\pi x) + 0.5\sin(\pi x), \qquad 0 \leqslant x \leqslant 1,$$

accompanied by homogeneous Dirichlet boundary conditions.
In this case the solution is a wave that first develops a very steep gradient, depending on the size of ϵ, and subsequently moves towards the right boundary $x = 1$. Next, for increasing time t the amplitude u decreases due to the Dirichlet boundary conditions. Finally, for $t \to \infty$ the steady-state solution $u = 0$ is reached. While the choice $\epsilon = 10^{-3}$ yields a problem having all properties for testing a moving-grid method, we take the even smaller value $\epsilon = 10^{-4}$ as a more severe test case. The problem is solved on the time interval [0,2]. (See also [8].)

b) the initial condition is the trapezoid

$$u\,|_{t=0} = \begin{cases} 0.2 & 0 \leqslant x \leqslant 0.1 \\ 8x - 0.6 & 0.1 \leqslant x \leqslant 0.2 \\ 1 & 0.2 \leqslant x \leqslant 0.5 \\ -10x + 6 & 0.5 \leqslant x \leqslant 0.6 \\ 0 & 0.6 \leqslant x \leqslant 1 \end{cases},$$

with the boundary conditions

$$u\,|_{x=0} = 0.2, \quad u\,|_{x=1} = 0, \qquad t > 0.$$

For this case the course of the amplitude u is roughly the same as for case a), with the understanding that the solution now possesses several sharp features unlike the sinusoidal pulse which is very smooth outside the shock region. Again we consider the case $\epsilon = 10^{-4}$ and the time interval [0,2].

		$t = 1.4$				$t = 2.0$				qual.	qual.
A^2	B^2	STEPS	JACS	STEPS	JACS	CTF	ETF	CPU	ORD	sol.	grid
1E-9	1E-8	278	205	429	326	91	16	2.3	1.27	++	+
1E-8	1E-8	274	183	396	280	79	27	2.1	1.18	++	+
1E-7	1E-8	197	144	266	197	56	13	1.4	1.37	++	+
1E-6	1E-8	174	140	232	186	56	7	1.3	1.41	++	++
1E-5	1E-8	140	105	191	146	49	2	1.0	1.40	++	++
1E-4	1E-8	135	111	179	147	55	1	1.0	1.40	++	++
1E-3	1E-8	162	138	201	171	62	3	1.2	1.37	−	+
1E-9	0	305	224	424	306	84	31	2.2	1.26	++	+
1E-8	0	275	195	351	253	70	16	1.8	1.23	++	+
1E-7	0	222	161	292	215	70	11	1.5	1.41	++	+
1E-6	0	225	196	291	255	87	8	1.7	1.36	++	++
1E-5	0	206	189	×		301	6		1.34	++	++
1E-4	0	156	136	×		266	1		1.36	++	++
1E-3	0	170	138	208	167	63	3	1.2	1.45	−	+

TABLE 3.1. Problem Ia. Integration history.

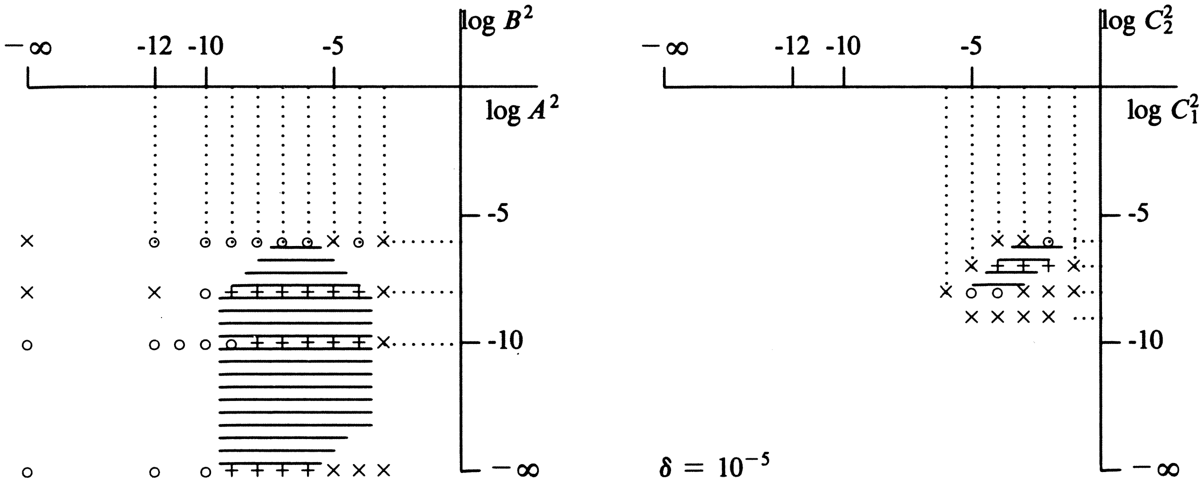


FIGURE 3.1. Problem Ia. Penalty parameter dependence for GWMFE (left) and MFE (right) Results at $t = 1.4$ are: +: good, o: dubious, ×: unacceptable.

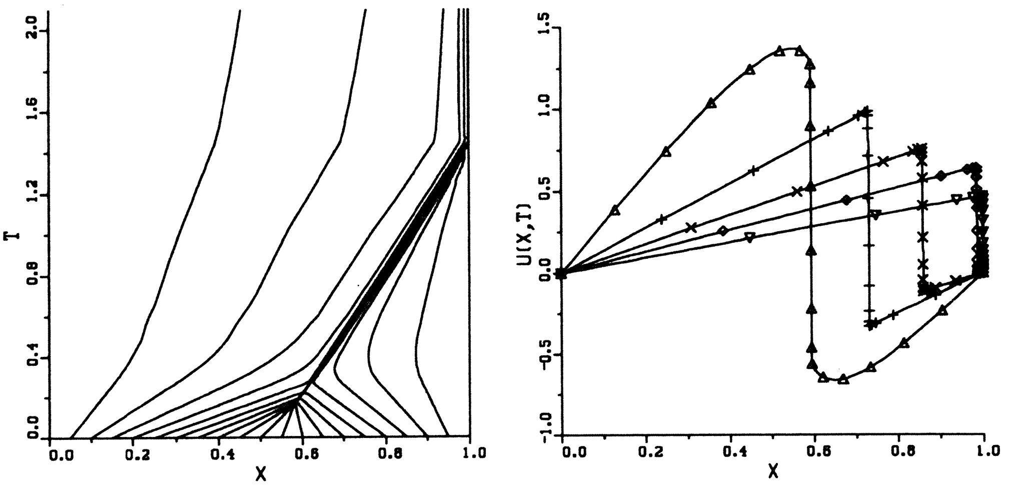


FIGURE 3.2. Problem Ia. Grid and solution at times $t = 0.2, 0.6, 1.0, 1.4, 2.0$ $(\Delta, +, \times, \diamond, \nabla)$ for $A^2 = 1E-5$ and $B^2 = 1E-8$.

3.1.1. Numerical results for Problem Ia.

Starting on a uniform grid with the number of grid points NPTS = 21, and as time-integration parameters TOL = 10^{-3} and $\Delta t_0 = 10^{-5}$, we obtain a series of test results by choosing $B^2 = 10^{-8}$, resp., 0 and by letting A^2 increase from 10^{-9} to 10^{-3}. The results are given in Table 3.1.

It can be seen that except for the largest value of A^2 the results are very satisfying. For $A^2 = 10^{-3}$ the speed of the shock was much too slow. There was not much difference between the grids and the solutions for the other values of A^2, but for small A^2 the ODE system and the resulting nonlinear system were much harder to solve, which made the computation more expensive. For $B^2 = 0$ the grid points were concentrated in the shock and no grid points were lying in the curvature. This makes the behavior of GWMFE more erratically. The computation broke down only twice, for $B^2 = 0$, both times because of (near) node crossing. This robustness is strikingly compared with the MFE method as tested in [8], as can be seen from the plots in Fig. 3.1 where the acceptable range of penalty parameter values is graphically represented for both the GWMFE and the MFE method. One should remember, however, that in the MFE implementation neither some form of preconditioning nor relative error tolerance on cell widths was available. We therefore incorporate only the results up to time $t = 1.4$ in these plots.

In Fig. 3.2 we give a plot of the typical grid behavior and solution. One can see that the solution is accurate up to plot resolution.

The method-of-lines approach does not seem to pay off for this problem. The number of Jacobians almost equals the number of (successful) steps. Even if we take into account the number of step failures (ranging from $\approx 50 - 100$) the number is still quite large. Also the observed average order turns out to be rather low. In fact SPGEAR almost never uses a third order method, not even in the time intervals where the problem is smooth and no step rejection or convergence failure occurs ($t \in [0.4, 1.2]$ and $t \in [1.5, 2.0]$). The fact that the order is not increased in these regions is somewhat amazing since plots of the $X_i(t)$ and the $U_i(t)$ show that both are reasonably smooth curves. However, in these areas the step size is drastically increased (only 10% of the steps is used in the smooth parts) and it could be that this is more efficient than an increase of the order. Most of the computational work is done where the shock is formed (at $t \approx 0.2$) and when the shock reaches the boundary (at $t \approx 1.3$). In these regions no high order method will be used because of the continual (true or near) node crossings within the iterative Newton process which result in convergence problems.

		$t = 0.9$		$t = 2.0$						qual.	qual.
A^2	B^2	STEPS	JACS	STEPS	JACS	CTF	ETF	CPU	ORD	sol.	grid
1E-7	1E-8	282	202	368	274	78	11	1.5	1.40	++	+
1E-6	1E-8	200	168	248	203	64	8	1.1	1.44	++	+
1E-5	1E-8	178	160	223	191	68	1	1.0	1.38	++	++
1E-4	1E-8	189	161	228	189	65	7	1.0	1.36	+	++

TABLE 3.2. Problem Ib. Integration history.

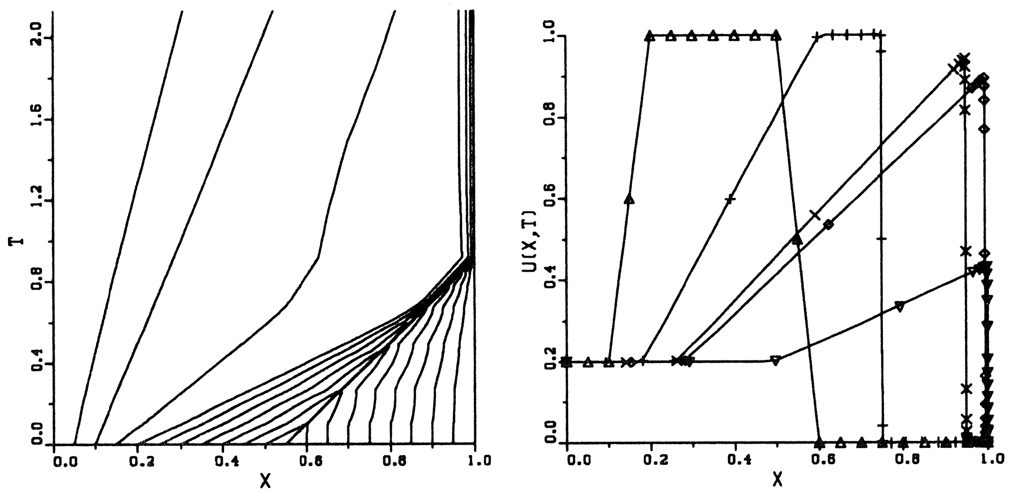


FIGURE 3.3. Problem Ib. Grid and solution at times $t = 0.0, 0.4, 0.8, 0.9, 2.0$ ($\Delta, +, \times, \Diamond, \nabla$) for $A^2 = 1E-5$ and $B^2 = 1E-8$.

3.1.2. Numerical results for Problem Ib.

For this problem, which is of the same nature as the above, we used only a small range of penalty parameter values. The integration parameters were chosen the same, i.e., NPTS = 21, TOL = 10^{-3}, and $\Delta t_0 = 10^{-5}$. The results are given in Table 3.2 and Fig. 3.3. The performance is comparable with that of problem Ia. We have also run this problem with $B^2 = 0$ and the same A^2-values. Again the results are comparable *if* the method does not break down, but it seems advisable to take B^2 slightly larger than zero if the problem under consideration exhibits a very steep wave.

3.2. Problem II: A shifting pulse

The ideas for this problem stem from Adjerid & Flaherty[1], who constructed a model (in 2-D) of a rotating cone using an exact solution. The PDE reads as follows:

$$u_t = u_{xx} + f(x,t), \quad 0 < x < 1, \quad t > 0, \tag{3.2}$$

where f is chosen in such a way that

$$u_{\text{exact}} := e^{-\alpha(x - r_\beta(t))^2}(1 - \sin(\gamma\pi t)), \qquad r_\beta(t) := \frac{1}{4}(2 + \sin(\beta\pi t))$$

satisfies (3.2). The boundary conditions at $x = 0$ and $x = 1$, being of Dirichlet type, and the initial condition, being a Gaussian pulse, are derived from the exact solution u_{exact}. The three parameters

$\alpha > 0$, $\beta > 0$ and $\gamma \geqslant 0$ each have their own meaning in the model. Choosing $\gamma > 0$ means that the pulse will decrease and rise again with a period of $2/\gamma$. The steepness of the solution is controlled by the parameter α in the exponential function and β represents the speed of the pulse which moves periodically from the left to the right boundary and back again in a period of $2/\beta$. We have chosen the values $\alpha = 320$, $\beta = 1$ and $\gamma = 2$. The PDE is integrated over one period, i.e., until $t = 2.0$.

The integrals stemming from $< \alpha_i, fw >$ and $< \beta_i, fw >$ were evaluated by numerical quadrature using Boole's rule.

					$t = 2.0$				qual.	qual.
A^2	B^2	STEPS	JACS	CTF	ETF	$\Vert err \Vert_\infty$	CPU	ORD	sol.	grid
1E-7	0	381	274	73	12	5.7E-2	1.2	1.40	□	+
1E-6	0	310	215	52	9	5.1E-2	1.0	1.33	□	+
1E-5	0	608	504	173	0	4.2E-2	2.2	1.34	□	++
1E-4	0	988	912	348	2	5.3E-2	3.8	1.25	□	++
1E-7	1E-8	672	499	133	13	5.7E-1	2.2	1.42	−−	□
1E-6	1E-8	559	411	112	13	2.3E-1	1.9	1.39	−	□
1E-5	1E-8	461	351	112	3	5.5E-2	1.6	1.29	□	++
1E-4	1E-8	967	846	305	2	4.8E-2	3.6	1.23	□	++

TABLE 3.3. Problem II. Integration history.

3.2.1. Numerical results for Problem II.

For this problem we start on a nonuniform grid with NPTS = 41 and all but the two boundary points concentrated around the pulse, uniformly distributed between 0.35 and 0.65. If one starts with a uniform grid the results are slightly worse. The time-integration parameters were again TOL $= 10^{-3}$ and $\Delta t_0 = 10^{-5}$.

The performance of the GWMFE method for this problem is significantly less satisfying than for the convection dominated Burgers' problem of the previous section. The oscillating character of the solution makes that GWMFE loses track of the movement of the pulse if it is nearly zero and picks it up again only if the pulse is already at some height, thereby losing accuracy. This is especially true for the, admittedly large, value of 10^{-8} for B^2 which appeared to be such a good choice for the Burgers' problem. One can see from Table 3.3 that, especially for the smaller values of A^2, the solution is bad at a considerable amount of work. In the grid plots it shows that after the solution has become zero (at $t = 0.25$ and 1.25) the grid points do not return fast enough to their position around the pulses to get a correct approximation of the right-hand side of the PDE. The fact, that GWMFE does not adjust itself fast enough to an emerging pulse, can also be shown by starting the problem at $t_0 = 0.25$ and on a uniform grid (since $u \equiv 0$) (cf. Fig. 3.4). The initial performance is roughly the same as that on the whole interval for $B^2 = 10^{-8}$ started at $t_0 = 0$, but after $t = 1.0$ it recovers.

The efficiency of the GWMFE method is for this problem strongly dependent on the penalty parameter choice; for approximately the same accuracy the amount of work varies rather capriciously with a factor 3 or 4 for different choices of A^2 (cf. Table 3.3).

The time integrator reacts on this problem in a similar way as on the previous one. Again we see that for all parameter choices the number of Jacobian updates is large relative to the number of time steps even if we add the number of rejected steps. Also the order behavior is more or less the same. In regions which are supposed to be easy for GWMFE, i.e., a moving pulse which is significantly larger than zero, SPGEAR rather increases the step size than the order.

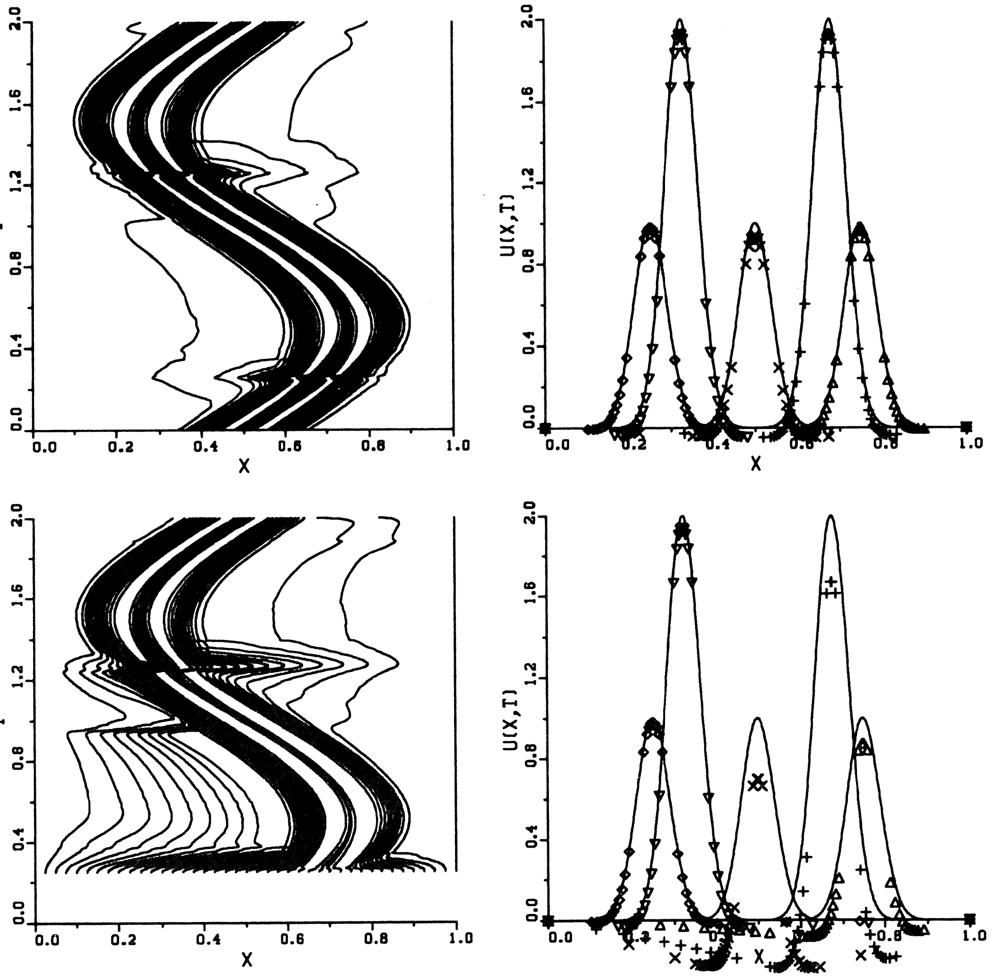


FIGURE 3.4. Problem II. Grid and solution at $t = 0.5, 0.75, 1.0, 1.5, 1.75$ ($\Delta,+,\times,\diamond,\nabla$) for $A^2 = 1E-6$ and $B^2 = 0$ starting at $t_0 = 0.0$ (above) and $t_0 = 0.25$ (below).

3.3. Problem III: Pulses traveling in opposite directions

Our third example problem is a two-component, semi-linear hyperbolic system, the solution of which is given by two pulses traveling in opposite directions (copied from [10], see also [8, 19, 20]). The system is given by

$$\begin{aligned} u_t &= -u_x - 100uv \\ v_t &= v_x - 100uv \end{aligned} \qquad -0.5 < x < 0.5, \quad t > 0, \tag{3.3}$$

and the solution is subjected to homogeneous Dirichlet boundary conditions and the initial condition

$$u\,|_{t=0} = \begin{cases} 0.5\,(1 + \cos(10\pi x)), & -0.3 \leqslant x \leqslant 0.1 \\ 0, & \text{elsewhere} \end{cases},$$

$$v\,|_{t=0} = \begin{cases} 0.5\,(1 + \cos(10\pi x)), & 0.1 \leqslant x \leqslant 0.3 \\ 0, & \text{elsewhere} \end{cases}.$$

Note that these are functions with a mere C^1 continuity, which represent wave pulses located at $x = -0.2$ and $x = 0.2$, respectively. Initially, while the pulses are separated, the nonlinear term $100uv$ vanishes, so that for $t > 0$ these pulses start to move with speed 1 and without change of shape, u to the right and v to the left. At $t = 0.1$ they collide at $x = 0$ and the nonlinear term becomes nonzero, resulting in a nonlinear interaction leading to changes in the shapes and speeds of the pulses. Specifically, the crests of the pulses collide a little beyond $t = 0.25$ and they have separated again at $t \approx 0.3$, so that from this time on the solution behavior is again dictated by the linear advection terms. At the nonlinear interaction, the pulses lose their symmetry and experience a decrease in amplitude.

				$t = 0.5$				qual.	qual.
A^2	B^2	STEPS	JACS	CTF	ETF	CPU	ORD	sol.	grid
1E-7	0	285	221	35	34	1.4	1.37	+	□
1E-6	0	210	166	38	14	1.0	1.30	+	□
1E-5	0	267	221	60	10	1.3	1.39	+	+
1E-4	0	345	297	98	12	1.8	1.35	−−	−
1E-7	1E-8	509	402	98	29	2.5	1.37	−−	−
1E-6	1E-8	367	287	77	16	1.7	1.40	−−	−
1E-5	1E-8	218	161	46	5	1.0	1.44	−−	−
1E-4	1E-8	304	255	90	5	1.5	1.34	−−	−

TABLE 3.4. Problem III. Integration history.

3.3.1. Numerical results for Problem III.

In contrast with our experience with the MFE method, GWMFE is not able to solve this problem without addition of (artificial) diffusion. Therefore, we added to both equations a diffusion term ϵu_{xx}, resp. ϵv_{xx}. The tests as described below are done with $\epsilon = 10^{-4}$; we also have tried $\epsilon = 10^{-5}$ but then GWMFE broke down.

Again we start on a nonuniform grid with NPTS $= 41$ and all but the two boundary points concentrated uniformly around the pulses. In this case too a uniform start implies slightly worse results. The time-integration parameters were the same as before: TOL $= 10^{-3}$, and $\Delta t_0 = 10^{-5}$. The results are given in Table 3.4 and Figs. 3.5 and 3.6. Note that the solid (u), resp. dashed (v), line in the plots represents an accurate reference solution of the original problem *without* diffusion term.

One can see from Table 3.4. that for this problem as well it is advisable to choose $B^2 = 0$. The choice $B^2 = 10^{-8}$ yields a very bad performance after the pulses have collided, as is illustrated in Fig. 3.6. For $B^2 = 0$ the plots look much better (cf. Fig. 3.5), but the computation is still quite expensive.

3.4. Problem IV: The Dwyer-Sanders flame-propagation model

Our fourth problem (see [7] for more details and also [19, 20]) serves as a useful test example for the simulation of several basic features which occur in physical flame models. The two PDEs for mass density u and temperature v are given by

$$\begin{aligned} u_t &= u_{xx} - u f(v) \\ v_t &= v_{xx} + u f(v) \end{aligned}, \quad 0 < x < 1, \quad 0 < t \leqslant 0.006, \tag{3.4}$$

where $f(v) = 3.52\,10^6\, e^{-4/v}$. The initial functions are

$$\begin{aligned} u\,|_{t=0} &= 1.0 \\ v\,|_{t=0} &= 0.2 \end{aligned} \quad 0 \leqslant x \leqslant 1,$$

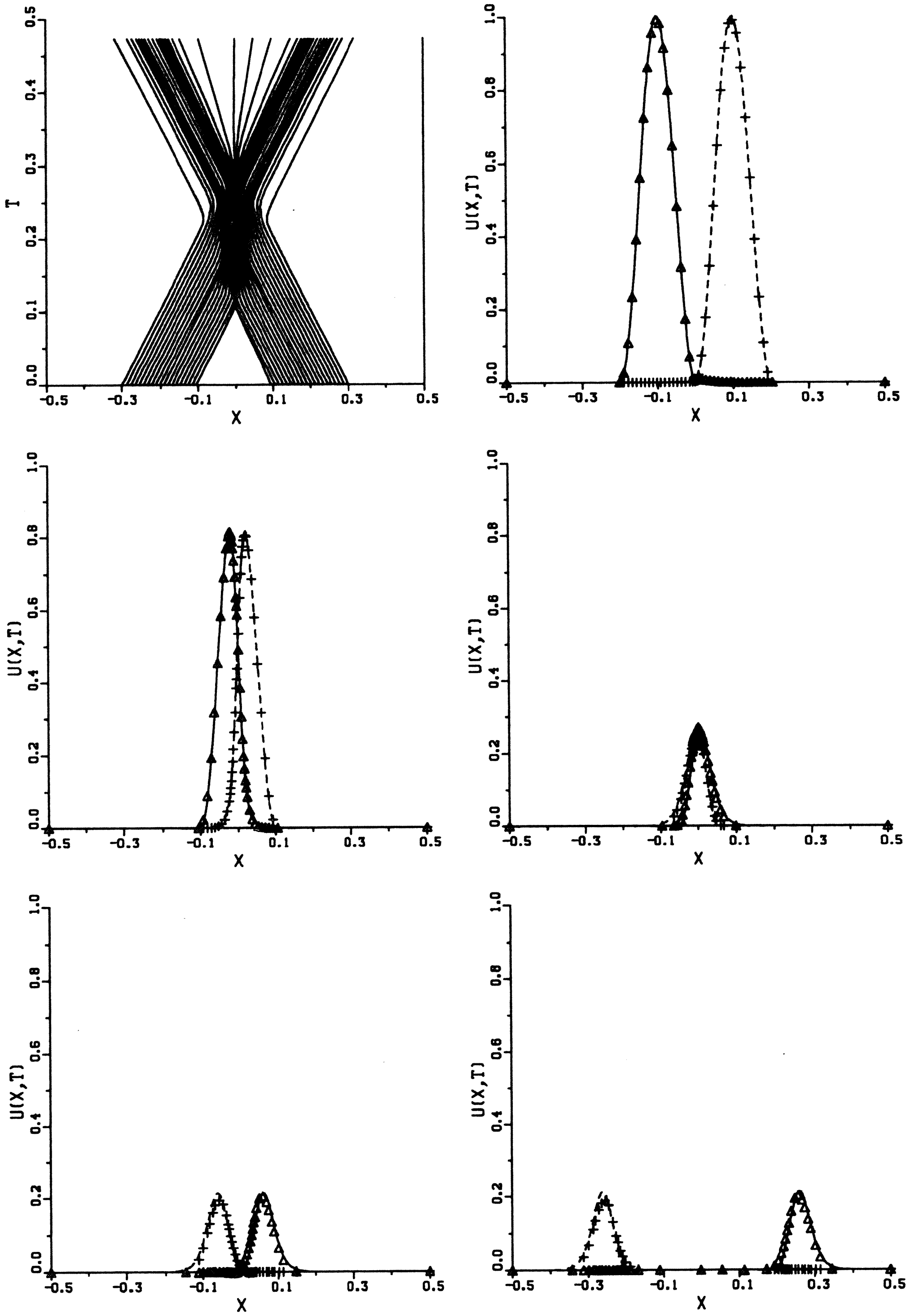


FIGURE 3.5. Problem III. Grid and solution at times $t = 0.1, 0.2, 0.25, 0.3, 0.5$ for $A^2 = 1E-5$ and $B^2 = 0$.

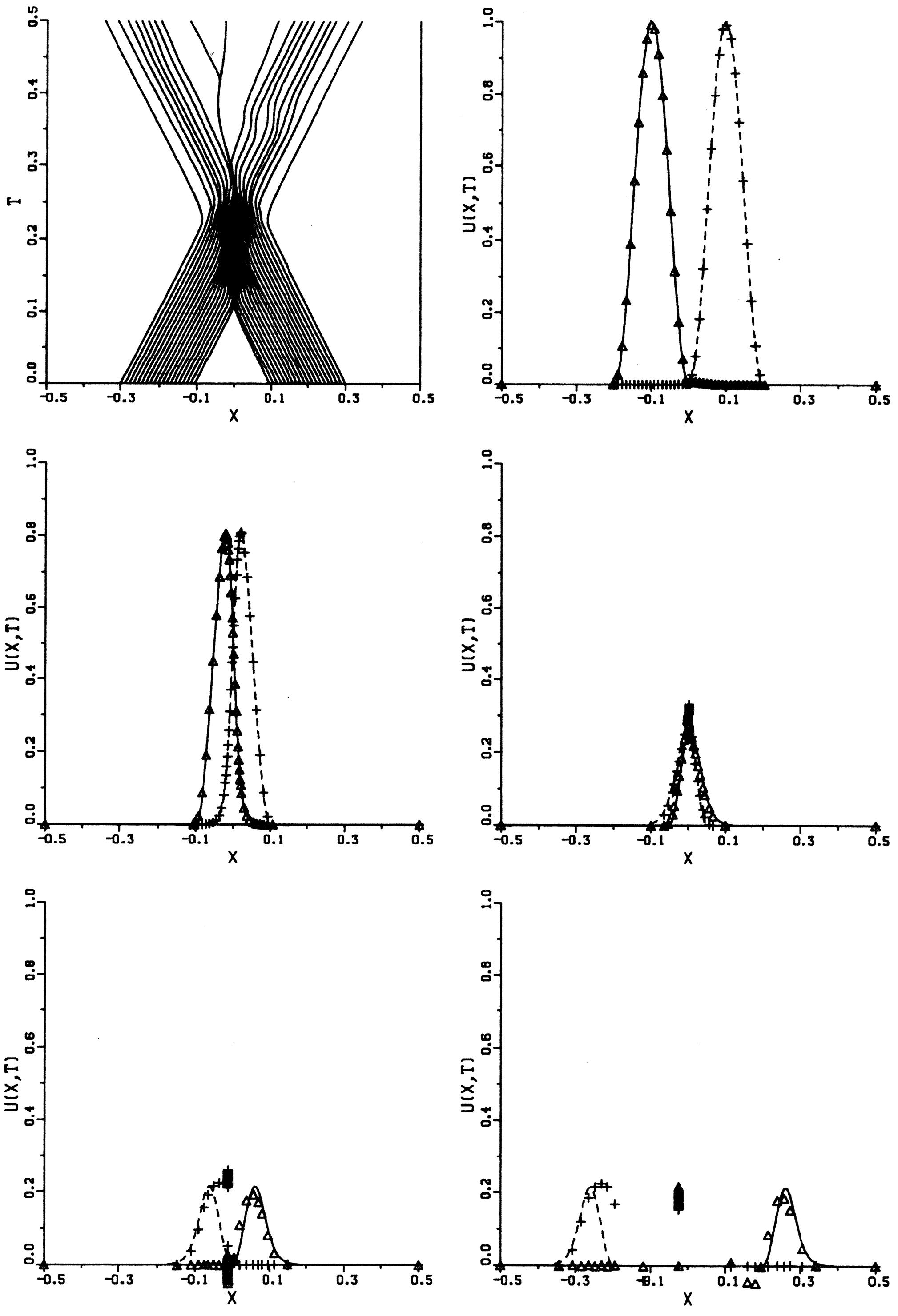


FIGURE 3.6. Problem III. Grid and solution at times $t = 0.1, 0.2, 0.25, 0.3, 0.5$ for $A^2 = 1E-5$ and $B^2 = 1E-8$.

and the boundary conditions read

$$u_x|_{x=0} = 0, \quad v_x|_{x=0} = 0, \quad t > 0,$$

and

$$u_x|_{x=1} = 0, \quad v|_{x=1} = \begin{cases} 0.2 + \dfrac{t}{0.0002}, & 0 < t \leqslant 0.0002 \\ 1.2, & 0.0002 \leqslant t \leqslant 0.006 \end{cases} .$$

The time-dependent forcing function for the temperature at the right boundary represents a heat source which generates a flame front. As soon as the temperature $v|_{x=1}$ reaches its maximum value 1.2 at $t = 0.0002$, this flame front starts propagating to the left at a relatively high (almost constant) speed ≈ 150. For $t = 0.006$ the front has nearly reached the left boundary.

The integrals stemming from $< \alpha_i, ufw >$ and $< \beta_i, ufw >$ were evaluated by numerical quadrature using Boole's rule.

		$t = 0.006$						qual.	qual.
A^2	B^2	STEPS	JACS	CTF	ETF	CPU	ORD	sol.	grid
1E-7	0	400	181	19	55	1.1	1.71	□	+
1E-6	0	361	164	25	27	1.0	1.79	+	+
1E-5	0	752	379	62	57	2.2	1.88	□	□
1E-4	0	789	508	122	56	2.7	1.56	□	□

TABLE 3.5. Problem IV. Integration history.

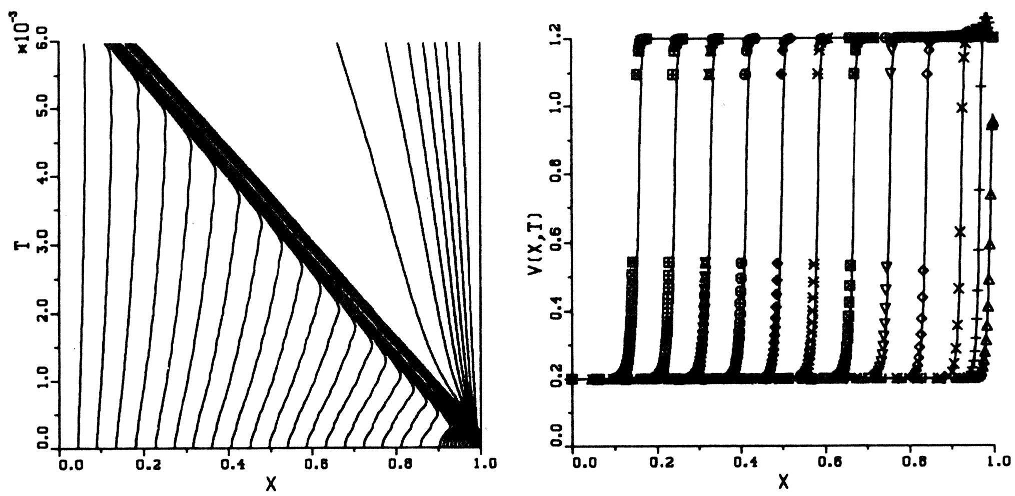


FIGURE 3.7. Problem IV. Grid and temperature component at $t =$ .15E-3, .3E-3, .6E-3 (.6E-3) .6E-2 for $A^2 = 1E-6$ and $B^2 = 0$.

3.4.1. Numerical results for Problem IV.

For this problem a strongly nonuniform initial grid was needed with NPTS $= 41$: 20 uniformly distributed grid points in [0.0,0.9], 10 in [0.9,0.99] and 10 in [0.99,1.0]. The time-integration parameters were TOL $= 10^{-4}$ and $\Delta t_0 = 10^{-5}$. We only present data for $B^2 = 0$. For $B^2 = 10^{-8}$ comparable results have been observed.

If we start on a uniform grid the flame front at the right boundary starts at the wrong time, but the solution has more or less the correct speed. If one approximates the innerproducts with Simpson quadrature instead of Boole's rule (with a nonuniform starting grid) the solution is initially the same, but the flame propagates much too fast. It is possible that even the seventh order quadrature rule is not accurate enough to approximate the integral over the source term and that this causes the flame to propagate slightly too fast as can be seen in Fig. 3.7.

The obtained average order is higher than for the previous problems, but unfortunately here the step size behaves very erratically. A plot of the step sizes shows a saw-tooth: the step size is increased, say 4 times in a row, then a convergence error occurs whereupon the step size is decreased by a factor of 4. Then the time error is found to be very small, so the step size is increased, etc., etc.. It is possible however, that this behavior results from the fact that **SPGEAR** is not tuned to the strongly nonlinear problems arising from PDEs discretized on a grid which moves continuously in time.

3.5. Problem V: The shocktube problem

Consider a shocktube containing a gas separated by a thin membrane, and assume the gas is at rest on both sides of the membrane, but with different constant pressures and densities on each side. At time $t = 0$, the membrane is broken, for example by a laser beam, and the problem is to determine the ensuing motion of the gas. The equations of the standard shocktube problem as presented by Sod[18] are the one-dimensional Euler equations of gasdynamics in conservative form

$$\begin{aligned} u_t &= -\, v_x \\ v_t &= -\frac{\partial}{\partial x}\{\, (\gamma-1)w \,-\, 0.5(\gamma-3)\frac{v^2}{u}\}\,, \qquad 0 < x < 1, \quad t > 0, \\ w_t &= -\frac{\partial}{\partial x}\{\, (\gamma w \,-\, 0.5(\gamma-1)\frac{v^2}{u})\frac{v}{u}\} \end{aligned} \tag{3.5}$$

where u, v and w are the density, momentum and total energy per unit volume, respectively, and γ is the ratio of specific heats ($\gamma = 1.4$ in the case of a perfect gas). The initial conditions satisfy

$$\begin{aligned} u\,|_{t=0} &= \begin{cases} 1, & 0 \leqslant x \leqslant 0.5 \\ 0.125, & 0.5 \leqslant x \leqslant 1 \end{cases} \\ v\,|_{t=0} &= \quad 0, \qquad 0 \leqslant x \leqslant 1 \quad . \\ w\,|_{t=0} &= \begin{cases} 2.5, & 0 \leqslant x \leqslant 0.5 \\ 0.25, & 0.5 \leqslant x \leqslant 1 \end{cases} \end{aligned}$$

The boundary conditions for u and w are of Neumann-type

$$u_x\,|_{x=0} = w_x\,|_{x=0} = 0, \quad \text{resp.,} \quad u_x\,|_{x=1} = w_x\,|_{x=1} = 0;$$

v is subjected to homogeneous Dirichlet boundary conditions. Briefly the course of the solution is as follows: at $t = 0$ the membrane in the tube bursts, with the consequence that the initial discontinuity breaks up into two discontinuities, a contact-discontinuity and a shock wave, which move to the right boundary, and a rarefaction wave moving to the left. If the shock wave has reached the right boundary, it reflects from the wall.

The integrals resulting from the innerproducts in the right-hand sides of the second and third PDE (cf. Appendix) were evaluated by numerical quadrature using Boole's rule.

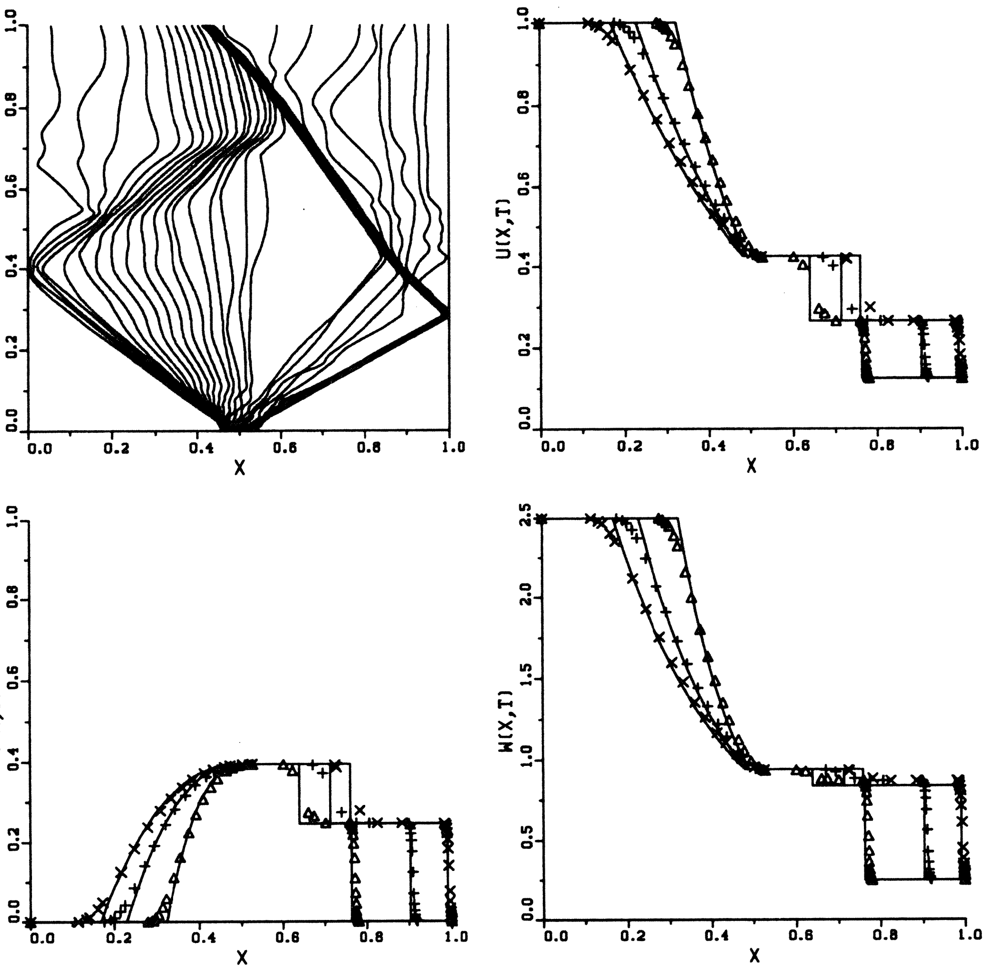


FIGURE 3.8. Problem V. Grid and PDE components at t = .15, .23, .28 (Δ,+,×) for $A^2 = 1E-6$ and $B^2 = 0$.

3.5.1. *Numerical results for Problem V.*

This problem was only solvable using artificial diffusion. In the experiment described below we added to all three equations a second order diffusion term, ϵu_{xx}, resp. ϵv_{xx}, resp. ϵw_{xx}, with $\epsilon = 10^{-3}$. Two other diffusion choices were tried, viz., $\epsilon = 10^{-4}$ for all equations, and no diffusion for the continuity equation with $\epsilon = 10^{-3}$ for the second and third equation. All resulted in a failure of GWMFE because the stepsizes taken by the integrator were much too small to reach the endpoint due to convergence problems.

We started on a nonuniform grid with NPTS = 41 and all but the two boundary points in the interval [0.45,0.55]. The time integration parameters were TOL = 10^{-3} and $\Delta t_0 = 10^{-5}$ and the penalty parameters $A^2 = 10^{-6}$ and $B^2 = 0$. The integration interval was [0,1].

The integration statistics at the endpoint were STEPS = 586, JACS = 486, CTF = 131, ETF = 32, and ORD = 1.40. To give some insight where GWMFE experienced most trouble: 73 steps were needed to reach t = 0.01, 170 for t = 0.1 and only 41 to go from t = 0.1 to the wall at t = 0.28.

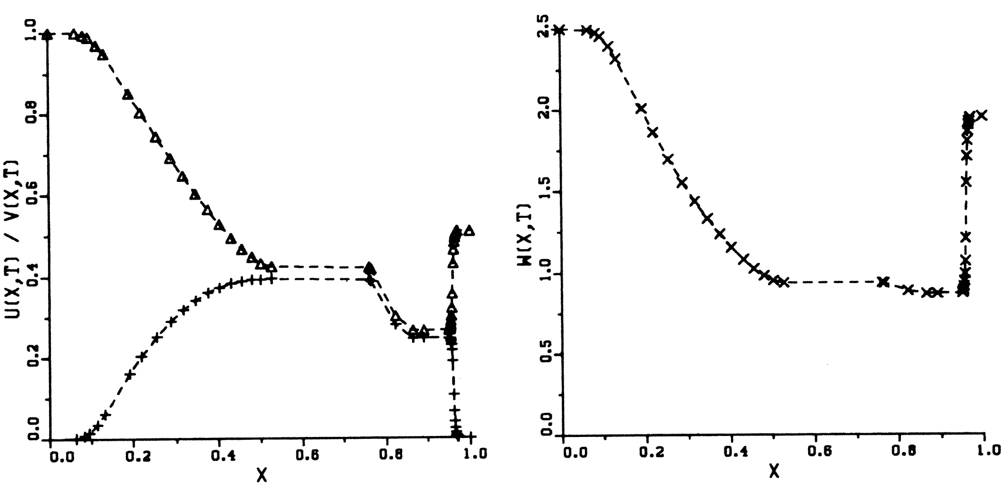


FIGURE 3.9. Problem V. PDE components at $t = .32$ for $A^2 = 1E-6$ and $B^2 = 0$ u (Δ) and v ($+$) left, and w right (no reference solution given).

The reflection phase, $t = 0.28$ until $t = 0.29$, took 153 (successful) steps. For the graphical representation of the results we refer to Figs. 3.8 and 3.9. The reference solution in Fig. 3.8 was calculated with the ICASE Riemann Solver [6], so the solid line in the solution plots represents the solution of the Euler equations. One can see that the artificial diffusion induces considerable smearing, but on the other hand the speed of the shock is approximated satisfactorily. The results at $t = 0.32$ in Fig. 3.9 are optically in agreement with the results as shown in [5] except for the steepness of the contact discontinuity. The grid movement is not really satisfying. The grid follows the shock wave quite well and also the rarefaction wave can be clearly seen in Fig. 3.8; but there are very few points in the region of the contact discontinuity, presumably because of the artificial diffusion. And on the whole the grid movement is not very smooth.

This is a very hard problem and we therefore consider the result as satisfying although the smearing of the shocks and the rarefaction wave is considerably in consequence of the need for a rather large diffusion coefficient. For this problem GWMFE showed itself more sensitive to the choice of the penalty parameters and the initial grid than for the previous ones; small changes in A^2 (say 5.10^{-6}), B^2 (10^{-8}) or a uniform starting grid resulted in a failure or in strongly oscillating solutions.

4. CONCLUSIONS

In this paper we have tested the gradient-weighted MFE method in 1-D on five difficult problems with steep moving fronts from different areas of application. A first observation is the robustness of the preconditioned GWMFE method compared with the MFE method as used in [8]. Our experience has been that for GWMFE the range of penalty parameters is much wider. Our recommendation is to take as standard choice $(A^2, B^2) = (10^{-6}, 0)$ which has worked quite well for most problems. The relative error tolerance on node distance (cf. (2.21)) meant an improvement especially when the nodes were concentrated in a small band; $\rho = 0.1$ appeared to be a good choice. We strongly advise to use the block-diagonal preconditioning of the residual. Although we as yet do not precisely understand why, it brings down the condition number of the Jacobian of the nonlinear system with several orders of magnitude.

It is dissuasive to use GWMFE as a general purpose method for all kinds of evolutionary problems. The disadvantage is not only the much more complex nonlinear system resulting from the addition of the strongly nonlinear grid equation, but also the fact that GWMFE does not get on with the method-of-lines approach. Compared to a fixed grid integration the number of Jacobians needed is much larger, say 1 Jacobian per 10 steps versus 2 every 3 steps, which means a factor 6. Although GWMFE solves Burgers' equation quite satisfactorily and the, also convection dominated, shocktube problem of Sod reasonably, it has its difficulties with problems having an emerging solution. So our advise here is to use GWMFE only when the solution is known to have steep moving parts over the whole integration interval.

In the context of the possible incorporation of the GWMFE method in the moving-grid user interface of SPRINT we will now compare the behavior of GWMFE with that of the MFD method [4, 8, 19] which is already a part of the interface.

For Problem Ia, the Burgers' equation with the sinusoidal initial condition, both methods are comparable (cf. Table 3.1 and the results given in [8]). But it should be noted that the MFD method has considerable difficulties with Problem Ib (the trapezoid initial condition). This can be explained by the fact that MFD applies a grid-point movement based on the equidistribution of the arclength and accordingly puts most of the points in the shock ignoring the less steep slope at the left of it. As a consequence, the space derivatives in that region cannot be approximated well enough by finite differences resulting in large oscillations.

For Problem III, on the contrary, MFD performs better. Although GWMFE, with $B^2 = 0$, gives a good solution, the computation is still quite expensive (STEPS $\approx$ 250 and JACS $\approx$ 200) in comparison with the data obtained with the MFD-method in [8], viz., STEPS = 105 and JACS = 58.

Also for Problem IV GWMFE needed much more time steps and Jacobians ($\approx$ 500 and $\approx$250) than the MFD method which gave an accurate solution at the cost of STEPS = 148 and JACS = 52 (cf. [19]).

We did not try the MFD method on Sod's problem, but it is likely that GWMFE will perform better on this problem because of its resemblance to Problem Ib.

Although GWMFE solves shock-like problems better than MFD we do not think the overall advantage large enough to justify the incorporation in our user interface, the more so as GWMFE does not seem to exploit the high order of the BDF method (the average order is approximately 1.5). We therefore see no advantage of the use of the SPGEAR integrator of SPRINT vs. DIRK2. We have decided to postpone the implementation of the gradient-weighted moving-finite-element method in the interface until as well the method itself as its performance in a method-of-lines context is better understood.

ACKNOWLEDGEMENTS

The authors wish to acknowledge N. Carlson and K. Miller for placing at our disposal their GWMFE1DS code. They would like to thank J.G. Verwer for his critical reading of this manuscript and his many constructive remarks. Finally, thanks are due to B. Koren and B. van Leer for their help in constructing the reference solution of Problem V.

REFERENCES

1. S. ADJERID and J.E. FLAHERTY (1988). A Local Refinement Finite Element Method for Two-Dimensional Parabolic Systems, *SIAM J. Sci. Stat. Comput.*, 9, 792-811.
2. M. BERZINS and R.M. FURZELAND (1985). *A User's Manual for SPRINT - A Versatile Software Package for Solving Systems of Algebraic, Ordinary and Partial Differential Equations: Part 1 - Algebraic and Ordinary Differential Equations,* Report TNER.85.058, Thornton Research Centre, Shell Research Ltd., U.K..
3. M. BERZINS and R.M. FURZELAND (1986). *A User's Manual for SPRINT - A Versatile Software Package for Solving Systems of Algebraic, Ordinary and Partial Differential Equations: Part 2 - Solving Partial Differential Equations,* Report No. 202, Department of Computer Studies, The University of Leeds.

4. J.G. Blom and P.A. Zegeling (1989). *A Moving-Grid Interface for Systems of One-Dimensional Time-Dependent Partial Differential Equations,* Report NM-R8904, Centre for Mathematics and Computer Science (CWI), Amsterdam.
5. E.A. Dorfi and L. O'C. Drury (1987). Simple Adaptive Grids for 1-D Initial Value Problems, *J. Comput. Phys.*, 69, 175-195.
6. M. Dunn and B. van Leer (1981). *The ICASE Riemann Solver,* ICASE Internal Report, Document No. 19, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, Virginia.
7. H.A. Dwyer and B.R. Sanders (1978). *Numerical Modeling of Unsteady Flame Propagation,* Report SAND77-8275, Sandia National Laboratories, Livermore, USA.
8. R.M. Furzeland, J.G. Verwer, and P.A. Zegeling (1988). *A Numerical Study of Three Moving Grid Methods for One-Dimensional Partial Differential Equations which are based on the Method of Lines,* Report NM-R8806, Centre for Mathematics and Computer Science (CWI), Amsterdam (to appear in J. Comput. Physics (1990)).
9. B.M. Herbst, S.W. Schoombie, and A.R. Mitchell (1983). Equidistributing Principles in Moving Finite Element Methods, *J. Comp. Appl. Math.*, 9, 377-389.
10. N.K. Madsen (1984). MOLAG: A Method of Lines Adaptive Grid Interface for Nonlinear Partial Differential Equations, in *PDE Software: Modules, Interfaces and Systems*, 207-223, ed. B. Engquist and T. Smedsaas, North-Holland.
11. K. Miller (1981). Moving Finite Elements II, *SIAM J. Numer. Anal.*, 18, 1033-1057.
12. K. Miller (1983). Alternate Modes to Control the Nodes in the Moving Finite Element Method, in *Adaptive Computational Methods for PDEs*, 165-182, ed. I. Babuška, J. Chandra and J.E. Flaherty, SIAM, Philadelphia.
13. K. Miller (1986). Recent Results on Finite Element Methods with Moving Nodes, in *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, 325-338, ed. I. Babuška, O.C. Zienkiewicz, J. Gago and E.R. de A. Oliveira, John Wiley & Sons Ltd..
14. K. Miller (1988). *Private Communication.*
15. K. Miller and R.N. Miller (1981). Moving Finite Elements I, *SIAM J. Numer. Anal.*, 18, 1019-1032.
16. R.F. Sincovec and N.K. Madsen (1975). Algorithm 494, PDEONE, Solutions of Systems of Partial Differential Equations, *ACM Trans. Math. Software 1*, 261-263.
17. R.F. Sincovec and N.K. Madsen (1975). Software for Nonlinear Partial Differential Equations, *ACM Trans. Math. Software 1*, 232-260.
18. G.A. Sod (1978). A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws, *J. Comput. Phys.*, 27, 1-31.
19. J.G. Verwer, J.G. Blom, R.M. Furzeland, and P.A. Zegeling (1989). A Moving-Grid Method for One-Dimensional PDEs based on the Method of Lines, in *Adaptive Methods for Partial Differential Equations*, 160-175, ed. J.E. Flaherty, P.J. Paslow, M.S. Shephard and J.D. Vasilakis, SIAM, Philadelphia.
20. J.G. Verwer, J.G. Blom, and J.M. Sanz-Serna (1989). An Adaptive Moving Grid Method for One-Dimensional Systems of Partial Differential Equations, *J. Comput. Phys.*, 82, 454-486.
21. A.J. Wathen (1986). Mesh-Independent Spectra in the Moving Finite Element Equations, *SIAM J. Numer. Anal.*, 23, 797-814.

APPENDIX

A CATALOGUE OF INNERPRODUCTS FOR (2.12)

Let

$$\Delta X_i := (X_i - X_{i-1}), \qquad \Delta U_i := (U_i - U_{i-1}),$$
$$m_i := \Delta U_i / \Delta X_i,$$

and

$$w_i := 1/\sqrt{1 + m_i^2}, \qquad \text{for } X_{i-1} \leqslant x \leqslant X_i.$$

Then the innerproducts read as follows:

$$< \alpha_j, \alpha_i w > = \begin{cases} \frac{1}{6}\Delta X_i w_i\ , & j = i-1 \\ \frac{1}{3}\Delta X_i w_i + \frac{1}{3}\Delta X_{i+1} w_{i+1}\ , & j = i \\ \frac{1}{6}\Delta X_{i+1} w_{i+1}\ , & j = i+1 \end{cases}$$

$$< \beta_j, \alpha_i w > = \begin{cases} -\frac{1}{6}\Delta U_i w_i\ , & j = i-1 \\ -\frac{1}{3}\Delta U_i w_i - \frac{1}{3}\Delta U_{i+1} w_{i+1}\ , & j = i \\ -\frac{1}{6}\Delta U_{i+1} w_{i+1}\ , & j = i+1 \end{cases}$$

$$< \alpha_j, \beta_i w > = \ < \beta_j, \alpha_i w > \quad \forall\ i,j$$

$$< \beta_j, \beta_i w > = \begin{cases} \frac{1}{6}m_i \Delta U_i w_i\ , & j = i-1 \\ \frac{1}{3}m_i \Delta U_i w_i + \frac{1}{3}m_{i+1}\Delta U_{i+1} w_{i+1}\ , & j = i \\ \frac{1}{6}m_{i+1}\Delta U_{i+1} w_{i+1}\ , & j = i+1 \end{cases}$$

For different choices of the operator L we give $< \alpha_i, L(U)w >$ and $< \beta_i, L(U)w >$:

I. $L(u) = u_{xx}$

$$< \alpha_i, L(U)w > \ = -\ln(m_i + \sqrt{m_i^2 + 1}) + \ln(m_{i+1} + \sqrt{m_{i+1}^2 + 1})$$
$$< \beta_i, L(U)w > \ = \sqrt{m_i^2 + 1} - \sqrt{m_{i+1}^2 + 1}$$

II. $L(u) = u_x$

$$< \alpha_i, L(U)w > \ = \frac{1}{2}\Delta U_i w_i + \frac{1}{2}\Delta U_{i+1} w_{i+1}$$
$$< \beta_i, L(U)w > \ = -\frac{1}{2}m_i \Delta U_i w_i - \frac{1}{2}m_{i+1}\Delta U_{i+1} w_{i+1}$$

III. $L(u) = uu_x$

$$< \alpha_i, L(U)w > \ = \Delta U_i w_i(\frac{1}{6}U_{i-1} + \frac{1}{3}U_i) + \Delta U_{i+1} w_{i+1}(\frac{1}{3}U_i + \frac{1}{6}U_{i+1})$$
$$< \beta_i, L(U)w > \ = -m_i \Delta U_i w_i(\frac{1}{6}U_{i-1} + \frac{1}{3}U_i) - m_{i+1}\Delta U_{i+1} w_{i+1}(\frac{1}{3}U_i + \frac{1}{6}U_{i+1})$$

IV. $L(u) = (f(x,u))_x$

$$< \alpha_i, L(U)w > \; = f(U_i)(w_i - w_{i+1}) - \frac{w_i}{\Delta X_i} \int_{X_{i-1}}^{X_i} f dx + \frac{w_{i+1}}{\Delta X_{i+1}} \int_{X_i}^{X_{i+1}} f dx$$

$$< \beta_i, L(U)w > \; = -f(U_i)(m_i w_i - m_{i+1} w_{i+1}) + \frac{m_i w_i}{\Delta X_i} \int_{X_{i-1}}^{X_i} f dx - \frac{m_{i+1} w_{i+1}}{\Delta X_{i+1}} \int_{X_i}^{X_{i+1}} f dx$$

V. $L(u) = f(x,u)$

$$< \alpha_i, L(U)w > \; = w_i \int_{X_{i-1}}^{X_i} \alpha_i f dx + w_{i+1} \int_{X_i}^{X_{i+1}} \alpha_i f dx$$

$$< \beta_i, L(U)w > \; = \; - m_i w_i \int_{X_{i-1}}^{X_i} \alpha_i f dx - m_{i+1} w_{i+1} \int_{X_i}^{X_{i+1}} \alpha_i f dx$$

For the numerical integration of the integrals in IV and V we used Boole's rule (closed Newton-Cotes of order 7) resulting in:

$$\int_{X_{i-1}}^{X_i} f dx = \frac{\Delta X_i}{90}(7f_0 + 32f_{1/4} + 12f_{1/2} + 32f_{3/4} + 7f_1)$$

$$\int_{X_{i-1}}^{X_i} \alpha_i f dx = \int_{X_{i-1}}^{X_i} \frac{x - X_{i-1}}{\Delta X_i} f dx = \frac{\Delta X_i}{90}(0 + 8f_{1/4} + 6f_{1/2} + 24f_{3/4} + 7f_1)$$

and analogously

$$\int_{X_i}^{X_{i+1}} \alpha_i f dx = \int_{X_i}^{X_{i+1}} \frac{X_{i+1} - x}{\Delta X_{i+1}} f dx = \frac{\Delta X_i}{90}(7f_1 + 24f_{1+1/4} + 6f_{1+1/2} + 8f_{1+3/4} + 0)$$

with

$$f_0 = f(X_{i-1}, U_{i-1}), \quad f_1 = f(X_i, U_i),$$

$$f_{1/4} = f(\frac{3X_{i-1} + X_i}{4}, \frac{3U_{i-1} + U_i}{4}),$$ (NB. $U(x,t)$ is a piecewise linear function)

etc., etc..