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

P. A. ZEGELING AND 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 GWMFE 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 various solution characteristics. All tests have been carried out using the BDF time integrator **SPGEAR** of the existing method-of-lines software package **SPRINT**. © 1992 Academic Press, Inc.

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 and Miller [16] and Miller [11], and by the moving-finite-difference (MFD) method discussed in Verwer *et al.* [20] (see also references therein). The MFD method is restricted

to problems in one space dimension and is strongly based on ideas due to Dorfi and Drury [6].

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 *et al.* [8]) we have attempted to provide insight in this question. There we 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 and Zegeling [5]). 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 [3, 4] and of Sincovec and Madsen [17, 18].

In [8] we 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 to a great extent 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 [15]). One of the goals of the current examination, therefore, is to find out to which extent GWMFE is a general purpose method. While most tests in the literature of (GW) MFE refer to strongly convection-dominated convection–diffusion problems, in this paper we test GWMFE also on true parabolic equations.

The paper is divided into five sections. 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 a problem from gasdynamics with a small diffusion term. Section 4 is devoted to a concise comparison between GWMFE and the MFD method from [5, 20]. In Section 5 our conclusions and recommendations are summarized.

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

$$u_t = L(u), \qquad x_L < x < x_R, \qquad t > 0, \qquad (2.1)$$

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

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

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

$$u \approx U = \sum_{j=1}^{N} U_j(t)\, \alpha_j(x, \{X_i(t)\})$$

$$= \sum_{j=1}^{N} U_j(t)\, \alpha_j(x, X_{j-1}(t), X_j(t), X_{j+1}(t)), \qquad (2.3)$$

where α_j is the standard piecewise linear basis function which is 1 at the jth node and 0 at the other nodes. Differentiating U with respect to t and applying the chain rule gives

$$U_t = \sum_{j=1}^{N} \dot{U}_j \alpha_j + \dot{X}_j \beta_j, \qquad (2.4)$$

where $\beta_j = -U_x \alpha_j$. 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 the L_2-norm $\|R(U)\|_2^2$ with respect to $\dot{U}_i$ and $\dot{X}_i$, where

$$R(U) := U_t - L(U) \qquad (2.5)$$

is the PDE residual. This minimization gives a system of $2N$ ordinary differential equations in the $2N$ unknowns $\dot{U}_i$ and $\dot{X}_i$,

$$\sum_{j=1}^{N} \langle \alpha_i, \alpha_j \rangle \dot{U}_j + \langle \alpha_i, \beta_j \rangle \dot{X}_j$$

$$= \langle \alpha_i, L(U) \rangle, \qquad i = 1, ..., N, \qquad (2.6a)$$

$$\sum_{j=1}^{N} \langle \beta_i, \alpha_j \rangle \dot{U}_j + \langle \beta_i, \beta_j \rangle \dot{X}_j$$

$$= \langle \beta_i, L(U) \rangle, \qquad i = 1, ..., N, \qquad (2.6b)$$

where $\langle , \rangle$ 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

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

we arrive at the semi-discrete MFE system

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

where $\mathscr{A}(Y)$ is a block-tridiagonal matrix, the so-called mass-matrix, containing the innerproducts of the basisfunctions $\{\alpha_j\}$ and $\{\beta_j\}$, whereas the only problem-specific terms are contained in the vector $G(Y)$. Note that the

boundary conditions are assumed to be incorporated in (2.7).

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. This is only the case as long as $m_j \neq m_{j+1}$ at every node, where m_j is the slope of the semi-discrete approximation U on $[X_{j-1}, X_j]$. 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 [22]).

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 $k \in \{1, ..., N\}$). 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. 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

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

where

$$\varepsilon_j^2 = \frac{C_1^2}{\Delta X_j - \delta}, \qquad \varepsilon_j S_j = \frac{C_2^2}{(\Delta X_j - \delta)^2},$$

$$\Delta X_j := X_j - X_{j-1}, \qquad (2.9)$$

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

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

to the left- and right-hand side, respectively. The ε-terms serve to avoid the degeneracy caused by parallelism. It can be shown that the addition of these terms renders the mass-matrix $\mathscr{A}$ positive definite [16], and thus regular. They represent a form of "internodal" viscosity, since they penalize relative motion between the nodes and result in the degenerate nodes being carried along with the rest of the solution, provided the penalty is sufficiently large to take over before the mass-matrix becomes numerically singular. The ε-terms do prevent node overtaking in a dynamic way since the internodal viscosities become infinite as ΔX tends to δ; 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 ($C_1 \to \infty$ 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 with respect to 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

An important class of PDE problems may be represented by the well-known Burgers' equation

$$u_t = \varepsilon u_{xx} - u u_x \qquad (2.10)$$

with a steep moving front solution $u(x, t)$ as pictured below for two given points in time.
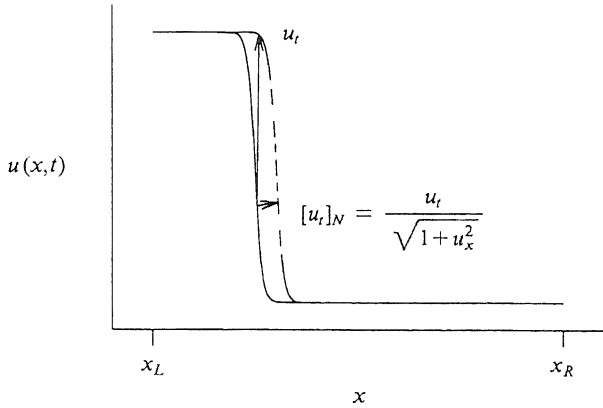
In such a front u_t is a near delta function and in case of a true shock not an L_2-function. To use the L_2-norm in the minimization of the residual $u_t - L(u)$ is therefore for such problems not appropriate. Since the normal component of u_t, $[u_t]_N$, remains bounded even in an arbitrarily steep front, it is preferable to minimize the residual of the PDE for the normal motion of the solution. So, instead of using the L_2-norm, GWMFE uses the weighted L_2-norm

$$\||R(U)\||^2 = \int [U_t - L(U)]_N^2 \, ds = \int (R(U))^2 \, w \, dx, \qquad (2.11)$$

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

$$w(U_x) = \frac{1}{\sqrt{1 + U_x^2}}. \qquad (2.12)$$

Baines [2] has proved that for (2.10), with $\varepsilon = 0$, MFE is identical to the method of characteristics and therefore will gradually concentrate most of the grid points into the front. It is likely that such a grid movement will also occur for (2.10) with $0 < \varepsilon \ll 1$. Apart from the fact that points are then wasted in the steepest part of the front, 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. The grid points will be concentrated more near the corners of the front (but still *in* the front). For scalar truly hyperbolic PDEs, however, both MFE and GWMFE will be (mathematically) equivalent to the method of characteristics, so in this case the gradient-weighting will not provide a remedy.

For GWMFE the minimization of (2.11) with respect to $\dot{U}_i$ and $\dot{X}_i$ gives, as before, a system of $2N$ ODEs in the $2N$ unknowns U_i and X_i

$$\sum_{j=1}^{N} \langle \alpha_i, \alpha_j w \rangle \, \dot{U}_j + \langle \alpha_i, \beta_j w \rangle \, \dot{X}_j$$

$$= \langle \alpha_i, L(U)w \rangle, \qquad i = 1, ..., N, \qquad (2.13a)$$

$$\sum_{j=1}^{N} \langle \beta_i, \alpha_j w \rangle \, \dot{U}_j + \langle \beta_i, \beta_j w \rangle \, \dot{X}_j$$

$$= \langle \beta_i, L(U)w \rangle, \qquad i = 1, ..., N, \qquad (2.13b)$$

where the weighting function $w = w(U_x)$ is defined by (2.12). The only difference with (2.6) is that the inner products are replaced by weighted inner products. A nice property of w, due to the piecewise linear approximation (2.3), is the fact that it is constant on each cell. Like before, insertion of all innerproducts yields the semi-discrete GWMFE system of the form

$$\mathscr{A}_g(Y) \, \dot{Y} = G_g(Y). \qquad (2.14)$$

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

$$\||R(U)\||^2 + \sum_{i=1}^{N+1} (\varepsilon_i \dot{l}_i - S_i)^2, \qquad (2.15)$$

where $\varepsilon_i^2 := A^2/l_i$, $\varepsilon_i S_i := B^2/l_i^2$, with A and B user-chosen constants, and l_i is the length of the ith segment. In contrast with MFE, the modifications involved induce changes to both equations (2.13a) and (2.13b). The combined effect is that each ith segment adds a "viscous" penalty force of magnitude $\varepsilon_i^2 \dot{l}_i = A^2 \dot{l}_i/l_i$, and a "spring" penalty force of magntidue $\varepsilon_i S_i = B^2/l_i^2$ to the two nodes at its ends, both penalty forces working in the tangential direction. 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 "segment viscosity" terms ε_i^2 serve to avoid parallelism. This means that the parameter A provides for the regularity of the mass-matrix $\mathscr{A}_g$ in the near degenerate situation of an almost flat solution. Likewise, the "internodal spring" terms $\varepsilon_i S_i$ 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 $\varepsilon_i S_i$. The direct analogue $l_i - \delta$ is redundant: it is unlikely that l_i tends to zero because this would require that both $\Delta X_i \to 0$ and $\Delta U_i \to 0$. 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 parts of the solution.

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 (2.15) by

$$\sum_{k=1}^{NPDE} \{ W_k \, \| U_t^k - L^k(U) \|_k^2 + \sum_{j=1}^{N+1} (\varepsilon_{jk} l_{jk} - S_{jk})^2 \}, \quad (2.16)$$

where k denotes the kth PDE component, NPDE equals the total number of PDEs, and

$$\| \phi \|_k^2 := \int_{x_L}^{x_R} \frac{\phi^2(x)}{\sqrt{1+(U_x^k)^2}} \, dx,$$

$$(\varepsilon_{jk})^2 := \frac{(A_k)^2}{l_{jk}}, \qquad \varepsilon_{jk} S_{jk} := \frac{(B_k)^2}{(l_{jk})^2},$$

$$l_{jk} := \sqrt{(\Delta U_j^k)^2 + (\Delta X_j)^2}.$$

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.

Carlson and Miller use in their code GWMFE1DS a shared set of x positions for the nodes of all the approximating functions U^k. Although it is possible to use more than one grid this seems only advisable for vary specific systems of PDEs, since the number of equations would be increased and it would complicate the implementation considerably.

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 (2.14). Miller [15] conjectured 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 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 first, we would like to give the reader an idea of the complexity of the ODE system (2.14). To that intent we work out Eq. (2.13) + penalty terms for the scalar PDE

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

Let w_i be defined by $w_i := 1/\sqrt{1+m_i^2}$ and $\Delta U_i := U_i - U_{i-1}$ for $X_{i-1} \leq X \leq X_i$. Then (2.13a) plus penalties yield for $i = 1, ..., N$,

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

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

$$+ \left(\frac{w_i}{3} \Delta X_i + \frac{A^2}{l_i^3} (\Delta U_i)^2 + \frac{w_{i+1}}{3} \Delta X_{i+1} \right.$$

$$+ \left. \frac{A^2}{l_{i+1}^3} (\Delta U_{i+1})^2 \right) \dot{U}_i$$

$$+ \left(-\frac{w_i}{3} \Delta U_i + \frac{A^2}{l_i^3} \Delta X_i \Delta U_i - \frac{w_{i+1}}{3} \Delta U_{i+1} \right.$$

$$+ \left. \frac{A^2}{l_{i+1}^3} \Delta X_{i+1} \Delta U_{i+1} \right) \dot{X}_i$$

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

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

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

$$+ \varepsilon(-\ln(m_i + \sqrt{m_i^2+1})$$

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

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

$$+ w_{i+1} \left(-f(t, X_i, U_i) \right.$$

$$+ \left. \frac{1}{\Delta X_{i+1}} \int_{X_i}^{X_{i+1}} f(t, x, U(x)) \, dx \right)$$

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

$$+ w_{i+1} \int_{X_i}^{X_{i+1}} \alpha_i g(t, x, U(x)) \, dx. \quad (2.17a)$$

Equation (2.13b) together with the penalties gives a similar expression as (2.17a) except for the diffusion term, which reads

$$\varepsilon(\sqrt{m_i^2+1} - \sqrt{m_{i+1}^2+1}). \quad (2.17b)$$

It is obvious that the resulting system is extremely nonlinear.

Note, that for the gradient-weighted MFE method the evaluation of both the innerproducts $\langle \alpha_i, u_{xx} w \rangle$ and

$\langle \beta_i, u_{xx}w \rangle$ has to be interpreted in the sense of "mollification," i.e., the piecewise linear function U is smoothed at the nodal points (cf. Miller [12, 16]). The ε-terms in (2.17a) and (2.17b) are the limits obtained for the "mollified" inner-products 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 by roundoff error 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, $\langle \beta_i, u_{xx}w \rangle$ 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 $\langle \alpha_i, u_{xx}w \rangle$, $\ln(m_i + \sqrt{m_i^2 + 1})$ 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.,

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

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 horizontal distance between 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

$$\| v/\text{tol} \| < 1$$

should hold as

$$\max_i \frac{|v(X_{i+1}) - v(X_i)|}{\Delta X_{i+1}\rho} < 1, \qquad (2.18)$$

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.

Another feature that is implemented in GWMFE1DS and which we also adopted is the block-diagonal preconditioning of the highly nonlinear implicit BDF equations

$$\tilde{R}_g(Y) := \mathscr{A}_g(Y) \frac{Y - Z}{\Delta t\, d} - G_g(Y) = 0, \qquad (2.19)$$

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 formula in use.

This preconditioning is prompted by the results of Wathen [22] for the MFE mass-matrix $\mathscr{A}$ in (2.7). He proved that premultiplying $\mathscr{A}$ by the inverse of its block-diagonal $\mathscr{D}$ 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. Miller [14] showed that this holds also for $\mathscr{D}_g^{-1}(Y) \mathscr{A}_g(Y)$ (the analogue of $\mathscr{D}^{-1}\mathscr{A}$ in case of gradient-weighting). Although the effects of preconditioning system (2.19) with $\mathscr{D}_g^{-1}(Y)$ 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 (2.19) too. Therefore we solve not (2.19) but instead

$$\mathscr{D}_g^{-1}(Y) \tilde{R}_g(Y) = 0. \qquad (2.19')$$

Note that the $\mathscr{D}_g^{-1}$ in (2.19') includes also that part of the penalty functions that occurs in the left-hand side of (2.17).

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) Sod's problem from gasdynamics with a small diffusion term. 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.

In Ref. [12] Miller gives a rationale of the penalty choice, based on a remedy of the degeneracies in both $\mathscr{A}$ and the residual system (see also Section 2.2). This results in a "standard choice" coupled with the time-tolerance TOL.

The parameter in the viscous penalty force should be $A^2 \gtrsim TOL^2$, say $TOL < A < 10TOL$. The standard choice for the B in the spring penalty force is $B^2 = 0$, unless it concerns a problem approaching steady-state with possible geometrical parallelism degeneracies. In this case the balancing of penalty contributions and true terms lead to $B^2 \approx 0.1\varepsilon TOL^2$, where ε is the coefficient of the diffusion term, cf. (2.10). To get an impression of the dependency 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 range of penalty parameter values based on the standard choice.

For all runs the "cell-width" relative error tolerance parameter ρ from (2.18) was taken 0.1 and block-diagonal preconditioning was used in solving the nonlinear system with Newton. For a few cases we evaluated the effects 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 down 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 non-linear 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 so large, but it is clear that preconditioning makes the method much more robust.

In some cases, for example in Problems I, III, and V, vertical rescaling of the PDE-system could result also in a better performance of GWMFE [15]. Such a vertical rescaling, say by a factor M (replacing "u" by "Mu" everywhere in the PDE), could allow a larger range of successful values of A^2. However, to choose the value of M, some insight into the solution behavior is needed, which makes it difficult to incorporate this parameter in an automatic code. For this reason we do not present results for rescaled PDE-systems.

In Ref. [23] we have given a catalogue of worked-out innerproducts. The integrals resulting from the inner-products $\langle \alpha_i, L(U)w \rangle$ and $\langle \beta_i, L(U)w \rangle$ were evaluated exactly 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, for Problem IV, far worse results. This difficulty with numerical quadrature on certain types of problems has already been mentioned by Miller [11] in his extensive testing of the MFE method.

The results are presented in tables and for a few parameter choices in plots wherein marks 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, i.e., no convergence of the Newton process after three iterations with a new Jacobian, or node crossing,

ETF   number of times the ODE integrator rejected a step,

CPU   normalized CPU-time, i.e., $CPU := CPU\text{-}secs/CPU\text{-}secs_{min}$, where $CPU\text{-}secs_{min}$ 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 (compared to either the exact solution or (in plots) to the numerical reference solution) and the quality of the grid (distribution and the smoothness of the motion in time): $++$ (very good), $+$ (good), $\square$ (reasonable), $-$ (bad), and $--$ (very bad). $\times$ 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 analogue to the Navier–Stokes equations, possesses a nonlinear convection term combined with a very small diffusion term,

$$u_t = \varepsilon u_{xx} - u u_x, \quad 0 < x < 1, \quad t > 0, \quad 0 < \varepsilon \ll 1. \quad (3.1)$$

We make a distinction between two specific problems (both stemming from Miller [15]):

(a) the initial condition is the smooth function

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

accompanied by homogeneous Dirichlet boundary conditions.

In this case the solution is a wave that first develops a very steep gradient, with a shock width proportional to ε, and subsequently moves towards the right boundary $x = 1$. It then collides with the boundary and forms a very thin boundary layer. This collision is a difficult part of the computation. Next, for increasing time t the amplitude u

decreases due to the Dirichlet boundary conditions. Finally, for $t \to \infty$ the solution dies out towards the steady-state solution $u = 0$. While the choice $\varepsilon = 10^{-3}$ yields a problem having all properties for testing a moving-grid method, we take the even smaller value $\varepsilon = 10^{-4}$ 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

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

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 $\varepsilon = 10^{-4}$ and the time interval $[0, 2]$.

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 $B^2 = 10^{-8}$, 10^{-11} (standard choice), 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 extreme values of A^2 the ODE system and the resulting nonlinear system sometimes were harder to solve, which made the computation more expensive. For $B^2 = 0$ and to some extent also for the standard choice $B^2 = 10^{-11}$ the grid points were concentrated in the shock and no grid points were lying in the curvature. This makes the behavior of GWMFE more precarious. Choosing a spring penalty value of $B^2 = 10^{-8}$, which is too large from the view of a reasonable balance of penalty contributions and terms of the system (2.14), results in a case like this in a very efficient performance, especially with the standard choice for A^2. This efficiency is likely to be caused by the fact that the grid points are pushed out of the front into the curvature by the large spring forces (see also Fig. 3.2). The computation broke down only twice, for $B^2 = 0$, both times because of (near) node crossing. This robustness is strikingly compared

TABLE 3.1

Problem Ia: Integration History

| A^2 | B^2 | $t = 1.4$ | | $t = 2.0$ | | | | | | Qual. sol. | Qual. grid |
		STEPS	JACS	STEPS	JACS	CTF	ETF	CPU	ORD		
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	1E-11	303	214	377	261	80	18	1.9	1.28	+ +	+
1E-8	1E-11	276	200	356	250	70	23	1.9	1.24	+ +	+
1E-7	1E-11	243	184	300	229	69	16	1.6	1.31	+ +	+
1E-6	1E-11	255	227	315	279	98	5	1.9	1.38	+ +	+ +
1E-5	1E-11	266	253	334	315	106	11	2.1	1.35	+ +	+ +
1E-4	1E-11	150	126	400	386	146	0	2.6	1.38	+ +	+ +
1E-3	1E-11	163	135	206	175	64	3	1.2	1.42	−	+
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	−	+
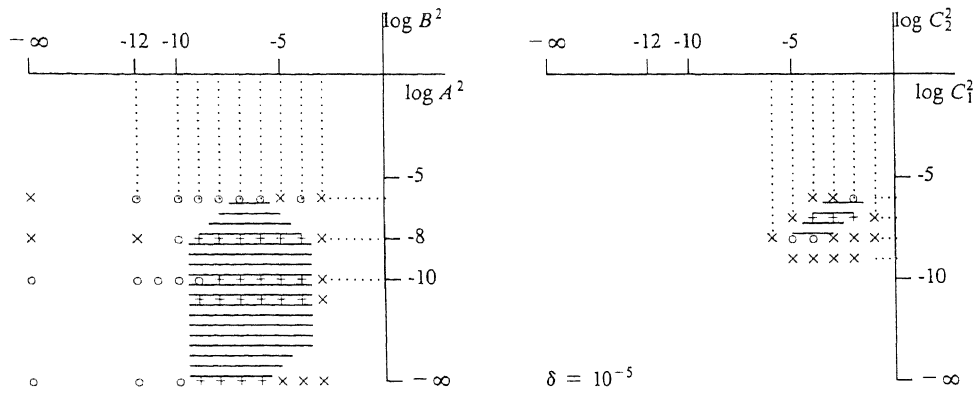
FIG. 3.1. Problem Ia. Penalty parameter dependence for GWMFE (left) and MFE (right). Results at $t = 1.4$ are: (+) good, (○) dubious, (×) unacceptable.

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 (by a shaded area) 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 plots of the typical grid behavior and solution. One can see that in both cases illustrated the solution is accurate up to plot resolution.

The time-integration process is not really satisfying 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 ≈ 50–100) the number is still quite large. Also the observed average order
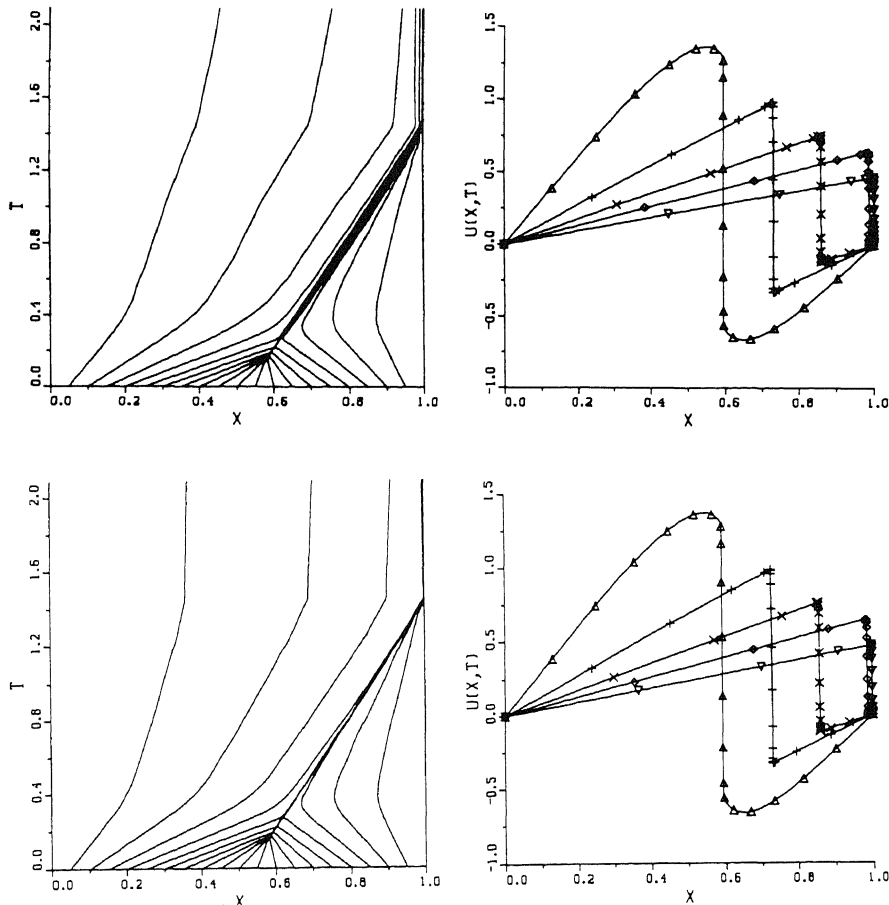

FIG. 3.2. Problem Ia. Grid and solution at times $t = 0.2, 0.6, 1.0, 1.4, 2.0$ ($\triangle$, +, ×, $\diamond$, ∇) for $A^2 = 1\text{E-}5$ and $B^2 = 1\text{E-}8$ (above), $1\text{E-}11$ (below).

TABLE 3.2

Problem Ib: Integration History

		$t = 0.9$		$t = 2.0$						Qual. sol.	Qual. grid
A^2	B^2	STEPS	JACS	STEPS	JACS	CTF	ETF	CPU	ORD		
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	+	+ +
1E-6	1E-11	177	144	227	184	70	3	1.0	1.39	+ +	+
1E-5	1E-11	191	176	276	247	91	3	1.3	1.37	+ +	+ +
1E-4	1E-11	206	179	403	373	138	9	2.0	1.37	+	+ +

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 for this coarse time tolerance. 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.

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, viz., $A^2 = 10^{-6}$, 10^{-5}, 10^{-4}, and $B^2 = 10^{-8}$, 10^{-11}. 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 for this problem to handle the degeneracies in the near steady-state situation. Although Miller's standard choice results in $B^2 = 10^{-11}$, the larger value of $B^2 = 10^{-8}$ seems both for this and for the previous problem to lead to more efficiency.

Problem II: A Shifting Pulse

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

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

where f is chosen in such a way that

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

$$r_\beta(t) := \tfrac{1}{4}(2 + \sin(\beta \pi t))$$

satisfies (3.2). The boundary conditions at $x = 0$ and $x = 1$, being of Dirichlet type, and the initial condition, being a
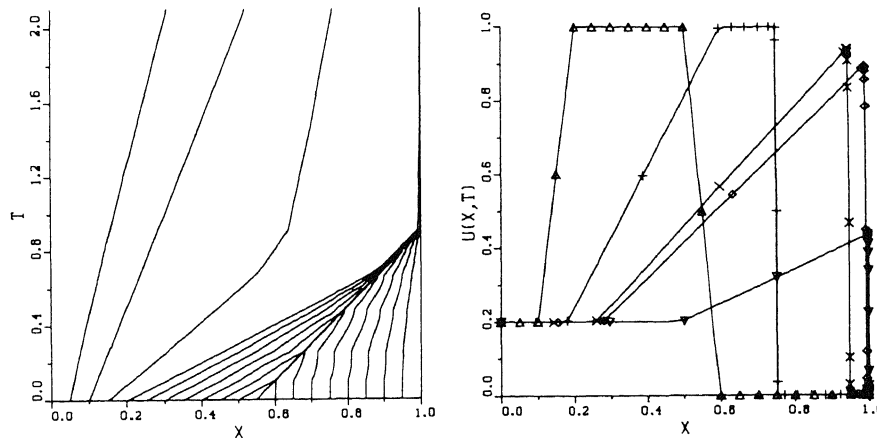

FIG. 3.3. Problem Ib. Grid and solution at times $t = 0.0, 0.4, 0.8, 0.9, 2.0$ ($\triangle$, +, ×, $\lozenge$, ∇) for $A^2 = $ 1E-5 and $B^2 = $ 1E-11.

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 $\langle \alpha_i, fw \rangle$ and $\langle \beta_i, fw \rangle$ were evaluated by numerical quadrature using Boole's rule.

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 $\Delta t_0 = 10^{-5}$. Since there is no steady-state involved in this problem we use the standard choice for the spring force penalty, $B^2 = 0$ and for A^2 the range 10^{-6}, 10^{-5}, 10^{-4}.

The performance of the GWMFE method for this problem is significantly less satisfying than for the convec-

tion dominated Burgers' problem of the previous section. The oscillating character of the solution makes that GWMFE loses track of the movement of the pulse when the amplitude goes to zero and picks it up again only if the pulse is already at some height, thereby losing accuracy. The grid plots show that after the solution has become zero (at $t = 0.25$ and 1.25) the grid point 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 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 to 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
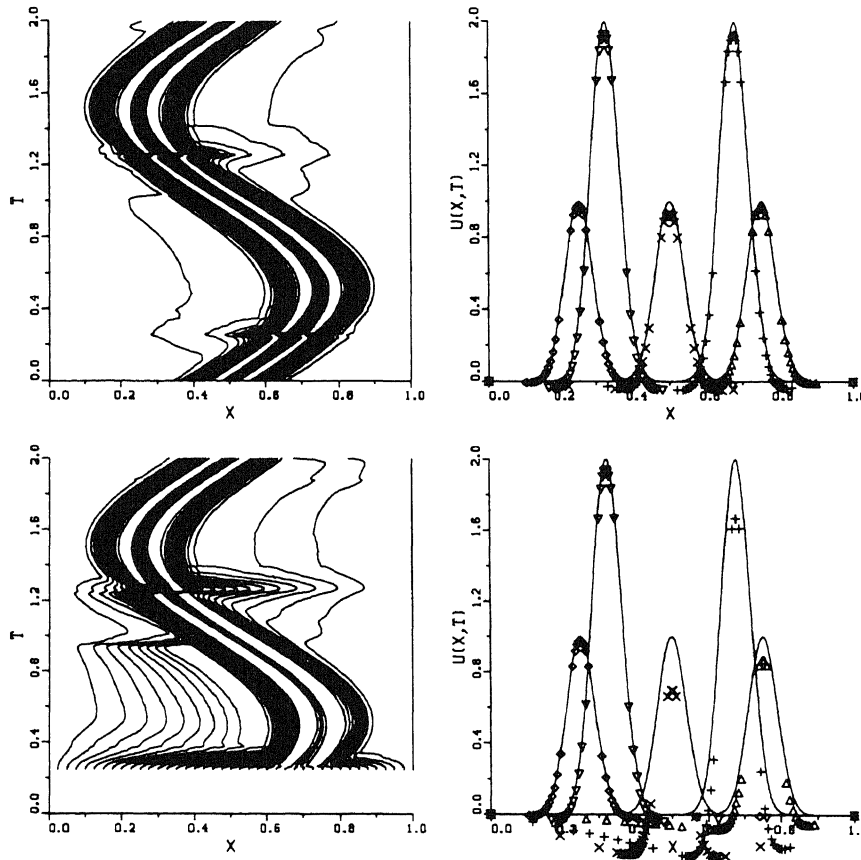
FIG. 3.4. Problem II. Grid and solution at $t = 0.5, 0.75, 1.0, 1.5, 1.75$ ($\triangle$, $+$, $\times$, $\diamond$, ∇) for $A^2 = 1$E-6 and $B^2 = 0$ starting at $t_0 = 0.0$ (above) and $t_0 = 0.25$ (below).

TABLE 3.3

Problem II: Integration History

A^2	B^2	STEPS	JACS	CTF	ETF	$\|err\|_\infty$	CPU	ORD	Qual. sol.	Qual. grid
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	□	+ +

(header row spans $t = 2.0$)

than zero, SPGEAR rather increases the step size than the order.

3.2. 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, 20, 21]). The system is given by

$$u_t = -u_x - 100uv$$
$$v_t = v_x - 100uv \quad -0.5 < x < 0.5, \quad t > 0, \quad (3.3)$$

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 non-zero, 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.

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 $\varepsilon = 10^{-4}$; we also have tried $\varepsilon = 10^{-5}$ 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 initial grid led to slightly worse results. The time-integration parameters TOL and Δt_0, and the GWMFE penalty parameters A^2 and B^2 have the same values as in Problem II. 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.

For this problem a correct choice of A^2 is of importance. $A^2 = 10^{-6}$ or 10^{-5} yields a satisfactory approximation, but $A^2 = 10^{-4}$ results in a very bad performance after the pulses have collided, as is illustrated in Fig. 3.6. For the other values of A^2 the approximation is much better (cf. Fig. 3.5), but the computation is still quite expensive.

Miller [15] showed that for this problem a "vertical rescaling" of the PDEs by a large factor (say 1000) could help to improve the performance of GWMFE. Rescaling the PDEs gives the method much lower traveling pulses to deal with. In fact it means that GWMFE is replaced by the

TABLE 3.4

Problem III: Integration History

A^2	B^2	STEPS	JACS	CTF	ETF	CPU	ORD	Qual. sol.	Qual. grid
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	– –	–

(header row spans $t = 0.5$)

original MFE method, which is profitable since for this problem no gradient weighting is needed.

3.3. Problem IV: The Dwyer–Sanders Flame-Propagation Model

Our fourth problem (see [7] for more details and also [20, 21]) 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

$$\begin{array}{l} u_t = u_{xx} - uf(v) \\ v_t = v_{xx} + uf(v) \end{array}, \quad 0 < x < 1, \quad 0 < t \leqslant 0.006, \quad (3.4)$$

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

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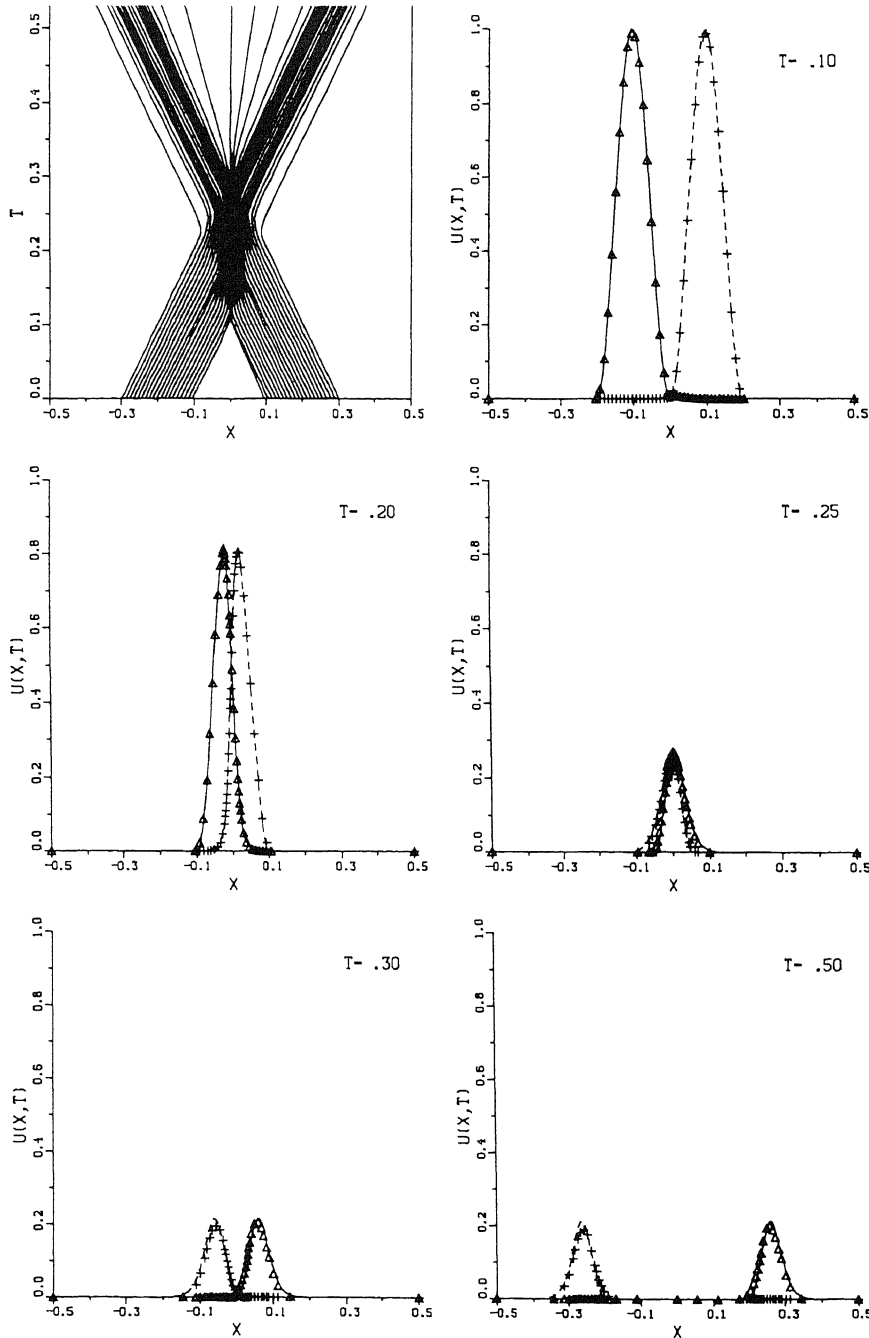

FIG. 3.5. Problem III. Grid and solution at times $t = 0.1, 0.2, 0.25, 0.3, 0.5$ for $A^2 = 1\text{E-}5$ and $B^2 = 0$.
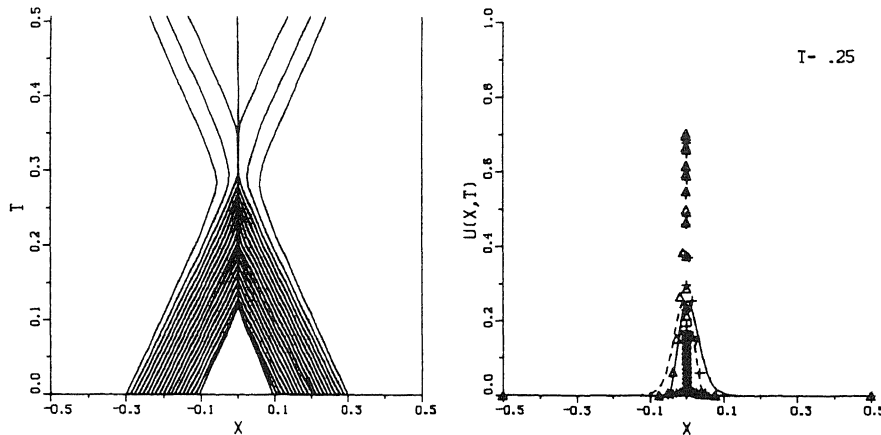
FIG. 3.6. Problem III. Grid and solution at $t = 0.25$ for $A^2 = 1\text{E-}4$ and $B^2 = 0$.

and the boundary conditions read

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

and

$$u_x\big|_{x=1} = 0,$$

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

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 $v\big|_{x=1}$ 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 $\langle \alpha_i, u fw \rangle$ and $\langle \beta_i, u fw \rangle$ were evaluated by numerical quadrature using Boole's rule.

Numerical Results for Problem IV. For this problem a strongly nonuniform initial grid was needed with $\text{NPTS} = 41$: 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 $\text{TOL} = 10^{-4}$ and $\Delta t_0 = 10^{-5}$. We only present data for the standard penalty parameter choices $A^2 = 10^{-8}, 10^{-7}, 10^{-6}$; $B^2 = 0$ (see Table 3.5).

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. This behavior is conform our observation in Problem II that GWMFE can not detect and resolve an emerging pulse. If one approximates the inner-products 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 (plot marks are centered). This problem would probably benefit from an appropriate adaptive quadrature method. The grid behavior can be explained for the lower band that points are absorbed in the front and cannot pass through a zero curvature (cf. Baines [2]). Note that the gap in the X-T diagram above the upper band is desirable because the solution is nearly constant there.

The obtained average order is higher than for the previous problems (probably because of the tighter tolerance), 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 four 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

TABLE 3.5

Problem IV: Integration History

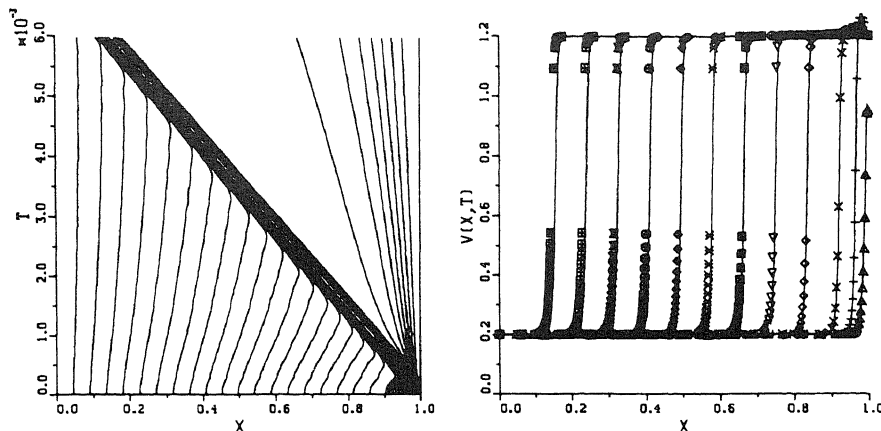|				$t = 0.006$				Qual. sol.	Qual. grid
A^2	B^2	STEPS	JACS	CTF	ETF	CPU	ORD		
1E-8	0	1566	627	82	166	3.9	2.07	−	□
1E-7	0	400	181	19	55	1.1	1.71	□	+
1E-6	0	361	164	25	27	1.0	1.79	+	+

FIG. 3.7. Problem IV. Grid and temperature component for $A^2 = 1E-6$ and $B^2 = 0$ at $t = 0.15E-3, 0.3E-3, 0.6E-3, 1.2E-3, ..., 6E-3$ (from right to left).

the strongly nonlinear problems arising from PDEs discretized on a grid which moves continuously in time. Another explanation, given by Miller [15], is that each time a new node runs into the front this results in small residual oscillations set up in the nodes just outside the lip of the shock as the nodes readjust. The 20 widely spaced nodes placed in the initial grid ahead of the front are not really needed in the present problem and it would probably be more efficient to use fewer of them.

3.4. Problem V: A Gasdynamics Problem with a Small Diffusion Term

The system of equations for this problem are the one-dimensional Euler equations of gasdynamics in conservative form supplemented by a small diffusion term

$$u_t = -v_x + \varepsilon u_{xx}$$

$$v_t = -\frac{\partial}{\partial x}\left\{(\gamma - 1)w - 0.5(\gamma - 3)\frac{v^2}{u}\right\} + \varepsilon v_{xx},$$

$$0 < x < 1, \qquad t > 0, \tag{3.5}$$

$$w_t = -\frac{\partial}{\partial x}\left\{\left(\gamma w - 0.5(\gamma - 1)\frac{v^2}{u}\right)\frac{v}{u}\right\} + \varepsilon w_{xx}$$

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 are linear ramps

$$u|_{t=0} = \begin{cases} 1, & 0 \leqslant x \leqslant 0.5 - 5\varepsilon \\ \text{linear}, & 0.5 - 5\varepsilon \leqslant x \leqslant 0.5 + 5\varepsilon \\ 0.125, & 0.5 + 5\varepsilon \leqslant x \leqslant 1 \end{cases}$$

$$v|_{t=0} = 0, \qquad 0 \leqslant x \leqslant 1$$

$$w|_{t=0} = \begin{cases} 2.5, & 0 \leqslant x \leqslant 0.5 - 5\varepsilon \\ \text{linear}, & 0.5 - 5\varepsilon \leqslant x \leqslant 0.5 + 5\varepsilon \\ 0.25, & 0.5 + 5\varepsilon \leqslant x \leqslant 1. \end{cases}$$

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

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

v is subjected to homogeneous Dirichlet boundary conditions. For $\varepsilon = 0$ there is no classical solution for this problem, but we are interested in the weak solution, which is the limiting solution as $\varepsilon \to 0$. This is the so-called shock-tube problem (cf. Sod [19]) and the problem and its weak solution are briefly described as follows. Consider a long thin cylindrical tube 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 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. For $0 < \varepsilon \ll 1$ the course of the solution is expected to be approximately the same, but now without true discontinuities. In fact the contact discontinuity will be rather smeared in comparison with the inviscid case. Of course, the shock wave and the rarefaction wave will also be smoothed depending on the size of ε.

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

Numerical Results for Problem V. In the experiment described below we used a diffusion coefficient $\varepsilon = 10^{-3}$. We have also tried $\varepsilon = 10^{-4}$. This 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 with 33 points on the linear ramp between $[0.5 - 5\varepsilon, 0.5 + 5\varepsilon]$, three points in an interval of length 10ε on both sides of the ramp, and the boundary points. The time-integration parameters were TOL = 10^{-3} and $\Delta t_0 = 10^{-5}$ and the (standard) penalty parameter values $A^2 = 10^{-6}$ and $B^2 = 0$. The time-integration interval was $[0, 1]$.

The integration statistics at the endpoint were STEPS = 698, JACS = 522, CTF = 120, ETF = 53, and ORD = 1.48. To give some insight where GWMFE experienced most trouble: 33 steps were needed to reach $t = 0.01$, 60 for $t = 0.1$ and only 35 to go from $t = 0.1$ to the wall at $t = 0.28$. The reflection phase, $t = 0.28$ until $t = 0.29$, took 102 (successful) steps. Until $t = 0.41$, when the rarefaction wave has reached the left boundary and the contact discontinuity has crossed the reflected shock, another 102 steps were needed. The last phase from $t = 0.41$ until 1.0 took surprisingly many steps, 399. This can be only explained by the oscillations both in the grid movement and in the solution itself, the latter caused by a too coarse grid around the reflected shock. In another run we used the true discontinuities for u and w as initial conditions and all but the two boundary points uniformly distributed over the interval

$[0.45, 0.55]$. In this case the initial phase gave, as could be expected, more difficulties. On a total of 586 steps 170 were used to reach $t = 0.1$, but from $t = 0.41$ until 1.0 only 134 steps were needed, the solution remained without oscillations and the grid points stayed in the shock band. This difference in behavior shows that for this problem GWMFE should be applied with care. For the graphical representation of the results we refer to Figs. 3.8 and 3.9. The reference solution in Fig. 3.8 was obtained by using 81 grid points and a time tolerance of 10^{-5}. A comparison with the solution for $\varepsilon = 0$ shows that the added diffusion induces considerable smearing, but on the other hand the speed of the shock is approximated satisfactorily. It should be noted, however, that more carefully chosen diffusion terms could be used to decrease the smearing of the contact discontinuity in particular. The grid movement is not really optimal. 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. And on the whole the grid movement is not very smooth, although for the last part of the integration this is probably due to the inaccurate and oscillating approximation of the solution.

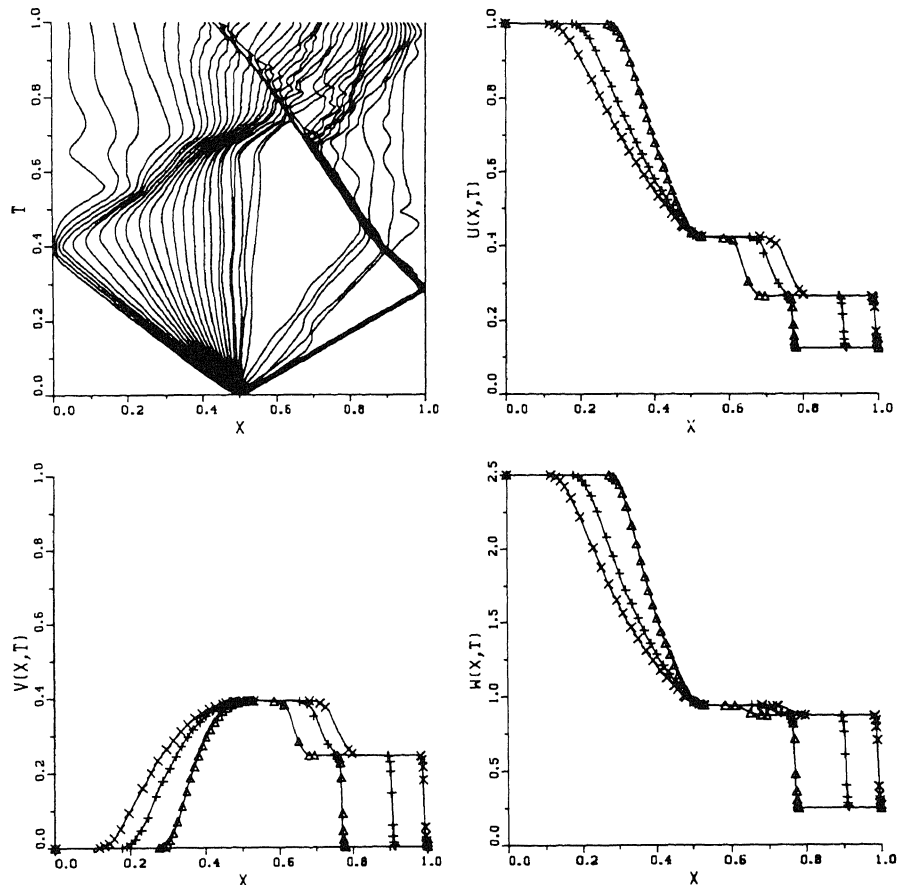This is a very hard problem for a nonspecialized code and

FIG. 3.8. Problem V. Grid and PDE components at $t = 0.15, 0.23, 0.28$ ($\triangle$, $+$, $\times$) for $A^2 = 1\text{E-}6$ and $B^2 = 0$.
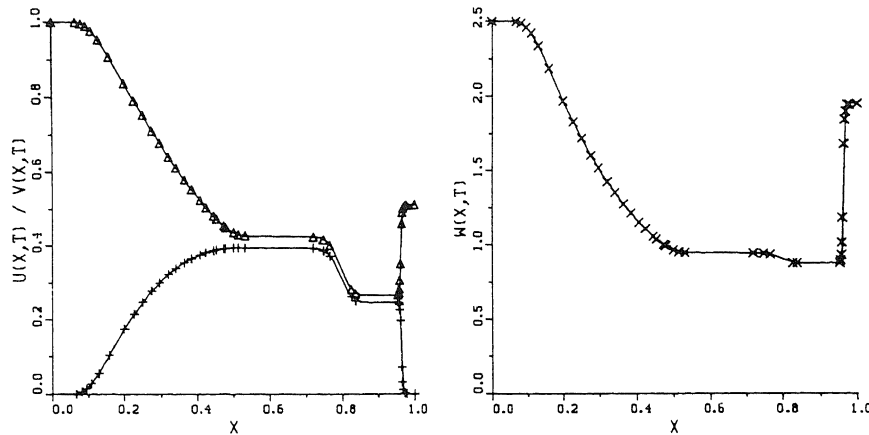
FIG. 3.9. Problem V. PDE components at $t = 0.32$ for $A^2 = $ 1E-6 and $B^2 = 0$, $u(\triangle)$ and $v(+)$ [left], and w[right].

we therefore consider the result as satisfying although GWMFE showed itself more sensitive to the choice of the penalty parameters and the initial grid or solution than for the previous problems; small changes in A^2 (say 10^{-5}, with $B^2 = 0$ or 10^{-11}) resulted in a failure and changes in the initial grid more than once caused strongly oscillating solutions at a later time.

4. A COMPARISON WITH A MOVING-FINITE-DIFFERENCE METHOD

In Ref. [8] a numerical comparison was made, a.o., between MFE (i.e., without gradient-weighting) and a moving-finite-difference method MFD (see also [5, 20]). The methods were tested extensively on three test problems. One of the conclusions was that MFD performed favorably with respect to efficiency and robustness compared to MFE. In this section we update that test work with the comparison of MFD versus GWMFE on the current set of test problems which has more variety (e.g., sharp moving corners) than the previous. For the sake of completeness we first give a short description of the MFD method.

4.1. The Moving-Finite-Difference Method

The MFD method is based on the Lagrangian discretization approach where the grid is moved continuously along with the solution with the aim of reducing the rapid transitions in space and in time that occur when a moving front passes a (fixed) grid point. The PDE (2.1) is transformed to its Lagrangian form

$$\dot{u} - u_x \dot{x} = L(u), \tag{4.1}$$

where $\dot{u}$ denotes the total time derivative. This PDE is discretized in space using N time-dependent grid points (cf. (2.2)) to obtain

$$\dot{U}_i - \frac{(U_{i+1} - U_{i-1})}{(X_{i+1} - X_{i-1})} \dot{X}_i = L_i, \quad t > 0, \quad 1 \leqslant i \leqslant N. \tag{4.2}$$

Here, U_i represents the semi-discrete approximation to the exact PDE solution u and L_i is the (centered) finite-difference replacement for the differential operator L, both at the point $(x, t) = (X_i(t), t)$. To solve the ODE system (4.2) additional equations are required for the time-dependent grid points X_i. The moving-grid technique that controls the spatial grid-movement in time is due to Dorfi and Drury [6]. For the theoretical background and some analytical aspects of the method we refer to [20], whereas a description of an MOL interface using this technique can be found in [5]. The underlying idea behind this grid movement is the spatial equidistribution of some monitor function. The grid equation reads

$$\frac{\tilde{n}_{i-1} + \tau \dot{\tilde{n}}_{i-1}}{M_{i-1}} = \frac{\tilde{n}_i + \tau \dot{\tilde{n}}_i}{M_i}, \quad 1 \leqslant i \leqslant N, \tag{4.3}$$

where $\tilde{n}_i := n_i - \kappa(\kappa + 1)(n_{i+1} - 2n_i + n_{i-1})$ and n_i stands for the, so-called, point concentration $n_i := (\Delta X_i)^{-1}$ of the grid. κ and τ are smoothing parameters; $\kappa \geqslant 0$ denotes a spatial smoothing parameter and $\tau \geqslant 0$ is a time-smoothing parameter. M_i is a monitor function, viz., the semi-discrete representation of the first derivative solution functional

$$m(u) = \sqrt{\alpha + \|u_x\|^2}.$$

The parameter α should regularize the transformation in regions where u is flat; its magnitude determines the number of grid points in flat regions. In a sophisticated implementation α could be related to the total integral over $m(u)$ with $\alpha = 0$, but until now we just chose a constant related to the average magnitude (over the time-integration interval) of the first spatial derivative of the solution.

In the grid equation the parameter κ determines the level of clustering of the grid points and the arclength monitor M_i determines the shape of the X_i-distribution. The parameter τ prevents the grid movement from adjusting immediately to new values of the monitor function M_i, therefore trying

to avoid temporal oscillations in the grid which may cause relatively large errors, when applied to solutions with steep gradients. A standard choice for the spatial smoothing parameter is $\kappa = 2$ and a typical choice for the temporal smoothing parameter $\tau = 10^{-3}$. Equations (4.2) and (4.3) are combined to yield a (stiff) system of ODEs.

4.2. MFD versus GWMFE

In this section we compare the performance of the MFD method and GWMFE. As far as the results have been published before, notably in [8, 20], we refer to those papers for the precise results of MFD and restrict ourselves here to some remarks.

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]). For Problem III the over-all performance of MFD is better. Although

GWMFE, with $A^2 = 10^{-6}$ or 10^{-5} and $B^2 = 0$, gives a good solution, the computation is still quite expensive (STEPS ≈ 250 and JACS ≈ 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 (≈ 500 and ≈ 250) than the MFD method which gave an accurate solution at the cost of STEPS $= 148$ and JACS $= 52$ (cf. [20]).

However, MFD has considerable difficulties with Problem Ib (the trapezoid initial condition). We did two experiments, with 21, resp., 41 grid points. The first was roughly twice as expensive as the GWMFE run (STEPS $= 447$ and JACS $= 224$) and the results were extremely bad (cf. Fig. 4.1). With 41 grid points MFD performs much better (STEPS $= 165$ and JACS $= 115$), but the sharp corners at $t = 0.4$ are still not very well resolved. This can be explained by the fact that MFD applies a grid-
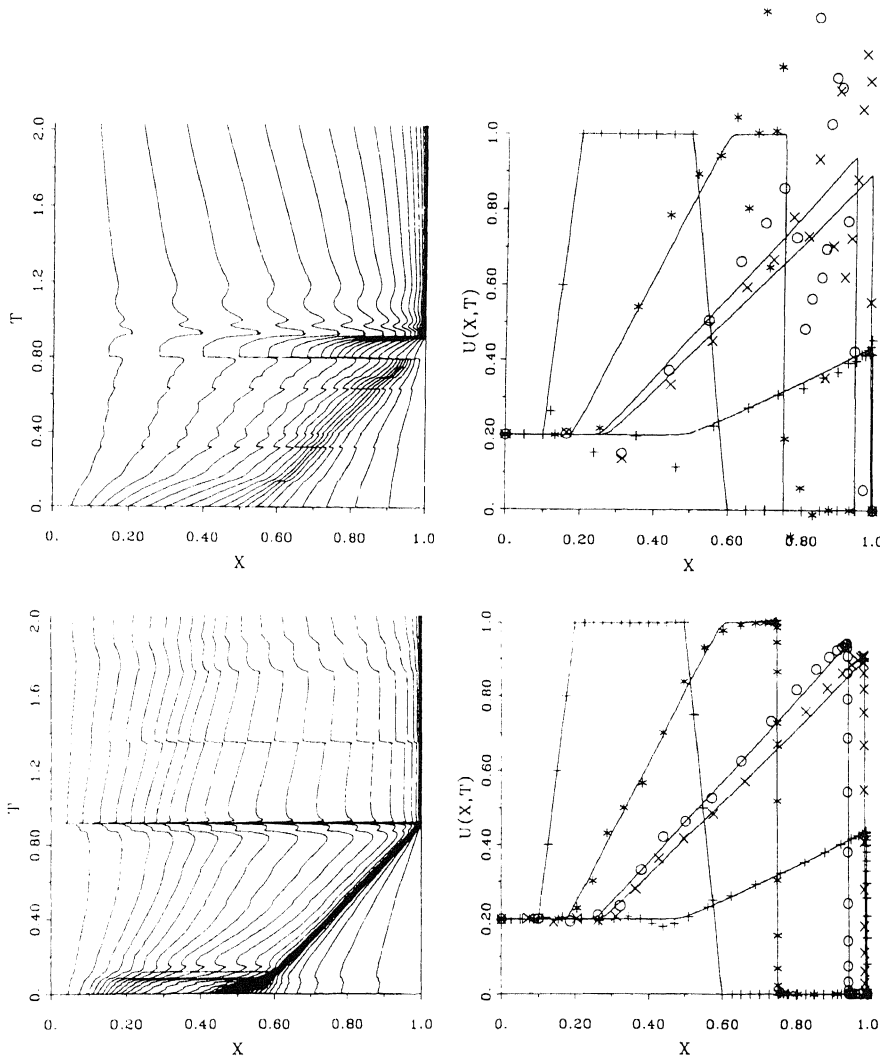

FIG. 4.1. Problem Ib. Grid and solution at times $t = 0.0, 0.4, 0.8, 0.9, 2.0$ (+, *, O, ×, +) MFD. 21 grid points (above) and 41 grid points (below).

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, using a total number of 21 grid points, cannot be approximated well enough by finite differences resulting in large oscillations.

As could be expected the difference in performance between both methods on Problem II is similar to that on Problem IV. With comparable results MFD (STEPS = 158, JACS = 80) is much cheaper than GWMFE. Even more important is the fact that there is no difference in performance of MFD if one starts at $t = 0.25$, indicating that MFD has less problems than GWMFE with emerging pulses.

Undoubtedly, GWMFE will perform better than MFD on Problem V because of its resemblance to Problem Ib, although we did no actual experiments with the MFD method on this problem.

It is obvious from the data above that neither of the two methods is a general purpose method. MFD has problems with solutions having discontinuous derivatives (sharp corners) (resulting in smearing and/or oscillations), largely different monitor values in different parts (oscillations), or near-shocks (small time steps caused by (temporary) node-crossing). Adding more grid points improves almost always the total performance (including the time stepping), but this makes the method less efficient of course. GWMFE has its problems with solutions with emerging structures; it is, in contrast to MFD, dependent on the initial placement of the nodes. Moreover, it results in a strongly nonlinear ODE system which is difficult to solve and, in the framework of MOL methods, most ODE solving packages, to our experience, will not efficiently solve the system.

5. 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 concerns the robustness of the preconditioned GWMFE method compared with the MFE method as used in [8]. Our experience has been, for one of the five problems at least, that for GWMFE the range of penalty parameters is much wider. Miller's rational choice for the values of the penalty parameters A^2 and B^2 has worked quite well for most problems, but there is some indication that for a tighter time tolerance the value of the viscous penalty parameter A^2 should be taken relatively larger than for a more coarse time tolerance (cf. Problem IV). 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.

We do not advocate 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 gasdynamics problem (with diffusion) reasonably, it has its difficulties with problems having an emerging solution. Our advise is to use GWMFE mainly when the solution is known to have steep moving fronts (not true shocks) over the whole time-integration interval.

ACKNOWLEDGMENTS

REFERENCES

1. S. Adjerid and J. E. Flaherty, *SIAM J. Sci. Stat. Comput.* **9**, 792 (1988).

2. M. J. Baines, *SIAM J. Numer. Anal.* **28**, 1323 (1991).

3. M. Berzins and R. M. Furzeland, Report TNER.85.058, Thornton Research Centre, Shell Research, Ltd., United Kingdom, 1985 (unpublished).

4. M. Berzins and R. M. Furzeland, Report No. 202, Department of Computer Studies, The University of Leeds, 1986 (unpublished).

5. J. G. Blom and P. A. Zegeling, Report NM-R8904, CWI, Amsterdam, 1989 (submitted for publication).

6. E. A. Dorfi and L. O'C. Drury, *J. Comput. Phys.* **69**, 175 (1987).

7. H. A. Dwyer and B. R. Sanders, Report SAND77-8275, Sandia National Laboratories, Livermore, California, 1978 (unpublished).

8. R. M. Furzeland, J. G. Verwer, and P. A. Zegeling, *J. Comput. Phys.* **89**, 349 (1990).

9. B. M. Herbst, S. W. Schoombie, and A. R. Mitchell, *J. Comput. Appl. Math.* **9**, 377 (1983).

10. N. K. Madsen, in *PDE Software: Modules, Interfaces and Systems*, edited by B. Engquist and T. Smedsaas (North-Holland, Amsterdam, 1984), p. 207.

11. K. Miller, *SIAM J. Numer. Anal.* **18**, 1033 (1981).

12. K. Miller, in *Adaptive Computational Methods for PDEs*, edited by I. Babuŝka, J. Chandra, and J. E. Flaherty (SIAM, Philadelphia, 1983), p. 165.

13. K. Miller, in *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, edited by I. Babuŝka, O. C. Zienkiewicz, J. Gago, and E. R. de A. Oliveira (Wiley, New York, 1986), p. 325.

14. K. Miller, *SIAM J. Numer. Anal.* **29**, 89 (1992).

15. K. Miller, Center for Pure and Applied Mathematics, University of California, Berkeley, private communication.

16. K. Miller and R. N. Miller, *SIAM J. Numer. Anal.* **18**, 1019 (1981).

17. R. F. Sincovec and N. K. Madsen, *ACM Trans. Math. Software* **1**, 261 (1975).

18. R. F. Sincovec and N. K. Madsen, *ACM Trans. Math. Software* **1**, 232 (1975).

19. G. A. Sod, *J. Comput. Phys.* **27**, 1 (1978).

20. J. G. Verwer, J. G. Blom, R. M. Furzeland, and P. A. Zegeling, in *Adaptive Methods for Partial Differential Equations*, edited by J. E. Flaherty, P. J. Paslow, M. S. Shephard, and J. D. Vasilakis (SIAM, Philadelphia, 1989), p. 160.

21. J. G. Verwer, J. G. Blom, and J. M. Sanz-Serna, *J. Comput. Phys.* **82**, 454 (1989).

22. A. J. Wathen, *SIAM J. Numer. Anal.* **23**, 797 (1986).

23. P. A. Zegeling and J. G. Blom, Report NM-R9006, CWI, Amsterdam, 1990 (unpublished).