International Journal of Pure and Applied Mathematics

Volume 4 No. 1 2003, 1-29

NUMERICAL EXPERIMENTS WITH THE MOVING FINITE ELEMENT METHOD

A. Vande Wouwer[1] §, P. Saucez[2], W.E. Schiesser[3]

[1]Laboratoire d'Automatique
Faculté Polytechnique de Mons
Boulevard Dolez, 31B-7000, Mons, BELGIUM
e-mail: vdw@autom.fpms.ac.be

[2]Service de Mathématique
Faculté Polytechnique de Mons, BELGIUM

[3]Mathematics and Engineering
Lehigh University, USA

Abstract: Since its invention by K. Miller, the moving finite element (MFE) method has been the subject of continuing investigations and developments. However, because of the usual lag between research and applications, the MFE method, and its several variants, can still be sensitive to the approach and use by the non-specialist.

This paper presents the results of a series of numerical experiments aimed at assessing, from a user's point of view, the influence of several of the method's features. These include the choice of regularizing parameters, the selection of the initial node distribution, the use of gradient weighting (GWMFE), as well as some issues relating to the numerical evaluation of the PDE residuals, time integration and the use of diagonal preconditioning.

A few classical 1-D test examples, which can be solved successfully with the MFE and GWMFE methods, are used to highlight these several points. In addition, some more difficult PDE systems are used to illustrate limitations

Received: March 3, 2002

§Correspondence author

and potential failures of the method.

AMS Subject Classification: 65M20, 65M50, 65M60
Key Words: partial differential equations, method of lines, combustion, gas dynamics, Schroedinger equation

1. Introduction

The numerical study of systems of evolutionary PDEs, whose solutions display steep moving fronts, shocks or boundary layers has demonstrated the need for numerical solution procedures with time and space adaptation. Over the past several years, numerous adaptive grid methods have been proposed; see for instance [1] for a survey of recent results. Among these, the acclaimed moving finite element (MFE) method introduced by K. Miller and his co-workers in the early eighties [2], [3], [4] has proved very efficient in solving a variety of difficult example problems.

In this method, the solution is expressed in terms of piecewise linear basis functions in which both nodal amplitudes and nodal positions move continuously with time, so as to satisfy a system of ODEs, which minimize the PDE residuals. To penalize the relative motions between nodes and to avoid parametrization degeneracy, regularization terms are introduced in the minimization problem. Hence, as with every adaptive grid techniques, a certain amount of parameter tuning is required to obtain satisfactory node motions.

Since its invention, the MFE method has aroused considerable interest and has benefited from the investigations of many researchers. Some important steps in the method development can be summarized, as follows.

A gradient-weighted moving finite element (GWMFE) version, which involves an evaluation of the PDE residuals using a gradient-weighted L_2 norm, was originally introduced by K. Miller [5]. The use of this new norm definition deemphasizes the steep parts of the solution, and results in an easier and less sensitive parameter tuning.

Grid distortion (and ultimately, parametrization degeneracy) results in extremely ill-conditioned residual equations. A diagonal preconditioning of the MFE mass matrix, that results in a condition number independent of the grid and the solution and so, in much more well-behaved preconditioned equations, has been proposed by Wathen [6], [7]. This technique has been extended to GWMFE by K. Miller [8].

Further, the GWMFE method is amenable to an elegant geometrical - mechanical interpretation, as a balance of forces on the nodes [9], that allows

important simplifications of the regularization terms to be operated.

Recently N. Carlson and K. Miller, [10], [11], have reported on the design of new 1-D and 2-D GWMFE codes which implement these method extensions, as well as some special numerical features, including roundoff error control in computing residuals, relative error tolerance on the internodal distance, and the use of a nonlinear Krylov subspace accelerator for Newton's method, which contribute significantly to increased robustness and efficiency.

This work is aimed at assessing, from a practical, user-oriented, point of view, the effects of some of these method extensions and improvements. To this end, the results of a set of numerical experiments with several variations of the MFE method are presented. At this stage, it is important to mention the previous work of P. Zegeling [12], who carried out an extensive comparison of MFE with two other adaptive grid codes, as well as an evaluation of an earlier version of GWMFE. In this study, an attempt is made to extend these results by considering new code features and additional test examples, including some situations in which the method is pushed to its limits.

Motivated by the observation made in [10] that there are many possible pitfalls in the design of (GW)MFE codes, a step by step procedure, based on numerical trial and error, is used to gain a feeling of the importance of the several method features. These include the choice of regularizing parameters, the selection of the initial node distribution, the use of gradient weighting, as well as some issues relating to the numerical evaluation of the PDE residuals, time integration and the use of diagonal preconditioning.

This analysis is restricted to a few classical and some more challenging 1-D problems. For problems in higher space dimensions, the reader is deferred to [11] and the references therein.

This paper is organized as follows. Section 2 briefly presents the MFE and GWMFE methods. In Section 3, a set of 1-D test-examples, i.e. two models from combustion theory, and the Euler equations, are solved using several variations of the (GW)MFE method. These numerical results allow several important method features to be highlighted. Subsection 3.4 deals with another test-example, i.e., the cubic Schrödinger equation, for which the MFE methods experience serious difficulties. Finally, Section 4 is devoted to some conclusions.

2. The MFE and GWMFE Methods

Consider the scalar PDE in one space dimension

$$\partial u/\partial t = L(u) = f(x, t, u, \partial u/\partial x, \partial^2 u/\partial x^2),\ x_L < x < x_R,\ t > 0\,, \qquad (1)$$

with initial and boundary conditions. In this expression, it is assumed that the partial differential operator $L(u)$ involves spatial derivatives which are, at most, second order.

In the original MFE method proposed by K. Miller and his co-workers [4], [2], [3], the solution to (1) is approximated using a finite element formulation with piecewise linear basis functions α_j:

$$u(x,t) \approx U(x,t) = \sum_{j=1}^{N} U_j(t)\ \alpha_j(x, X(t)), \tag{2}$$

in which both the nodal amplitudes $U_j(t)$, $j = 1, ..., N$, and the nodal positions $x_L = X_1(t) < X_2(t) < .... < X_N(t) = x_R$ are unknown functions of time.

Partial differentiation of (2) with respect to time yields

$$\partial U(x,y)/\partial t = \sum_{j=1}^{N} \dot{U}_j(t)\ \alpha_j(x, X(t)) + \dot{X}_j(t)\ \beta_j(x, X(t)), \tag{3}$$

in which $\beta_j = \partial U/\partial X_j$ can be considered as a second type of basis function. Both α_j and β_j are piecewise linear and have the same support (i.e., two elements). In general, β_j displays a simple discontinuity at X_j, as $\beta_j = -m_j\alpha_j$ on $[X_{j-1}, X_j]$ and $\beta_j = -m_{j+1}\alpha_j$ on $[X_j, X_{j+1}]$ with $m_j = (U_j - U_{j-1})/(X_j - X_{j-1})$.

The $2N$ unknown functions $U_j(t)$ and $X_j(t)$ are determined by minimizing the L_2 norm of the PDE residual $\|R(U)\|_2^2 = \|\partial U/\partial t - L(U)\|_2^2$ with respect to $\dot{U}_j(t)$ and $\dot{X}_j(t)$, which results in a system of $2N$ ODEs:

$$\sum_{j=1}^{N} \langle \alpha_i, \alpha_j \rangle \dot{U}_j + \langle \alpha_i, \beta_j \rangle \dot{X}_j = \langle \alpha_i, L(U) \rangle\,, \tag{4}$$

$$\sum_{j=1}^{N} \langle \beta_i, \alpha_j \rangle \dot{U}_j + \langle \beta_i, \beta_j \rangle \dot{X}_j = \langle \beta_i, L(U) \rangle\,, \qquad i = 1, ..., N\,. \tag{5}$$

Although the choice of piecewise linear approximations has the advantage of simplicity, it makes particularly delicate the evaluation of the inner products $\langle \alpha_i, L(U) \rangle$ and $\langle \beta_i, L(U) \rangle$, which involve second-order differential operators $\partial^2/\partial x^2$. In those cases, $L(U)$ is a sum of Dirac functions with weights $m_{j+1} - m_j$ at the node X_j, and the inner-products must be interpreted in the sense of mollification introduced by Miller [2], [3]. Loosely speaking, the mollification technique amounts to considering that β_j assumes its mean value

$-(m_{j+1}+m_j)/2$ at the node X_j and to computing the inner products involving Dirac functions in the usual sense of distributions.

By reordering the unknown variables in a column vector $Y = (U_1, X_1, U_2, X_2, ..., U_N, X_N)^T$, it is possible to write (4-5) in a compact form

$$A(Y)\dot{Y} = g(Y), \tag{6}$$

where $A(Y)$ is an $N \times N$ block-tridiagonal matrix

$$A(Y) = \begin{bmatrix} A_1 & B_1 & & & & & 0 \\ C_2 & A_2 & B_2 & & & & \\ & C_3 & A_3 & B_3 & & & \\ & & & ... & ... & ... & \\ 0 & & & & & C_N & A_N \end{bmatrix}. \tag{7}$$

In this matrix, each block A_j, B_j, C_j is a 2×2 matrix consisting of inner products of the basis functions α_j and β_j (in the case of a system of *npde* partial differential equations, each block is a $(npde+1) \times (npde+1)$ matrix).

Integrating (6) in time can become problematic for two reasons. First, the mass matrix $A(Y)$ becomes ill-conditioned, when some nodes drift very close together and extremely nonuniform grids are generated. Second, the mass matrix $A(Y)$ becomes singular, when parallelism occurs, i.e., when at a particular node X_j the solution curvature vanishes ($m_j = m_{j+1}$, which occurs, when three neighboring nodes move to a straight line portion of the solution). In this case, the MFE method intrinsically fails to determine the direction in which the node X_j should be moved (note also that the equilibrium system $g(Y) = 0$ degenerates in the case of parallelism).

To avoid these problem degeneracies, Miller [2], [3] introduced regularization terms in the residual minimization, which penalize the relative motions between nodes. The new minimization problem can be written as follows:

$$\min_{\dot{U}_j, \dot{X}_j} \|\partial U/\partial t - L(U)\|_2^2 + \sum_{j=2}^{N} (\varepsilon_j \Delta \dot{X}_j - S_j)^2 \,, \tag{8}$$

where $\Delta X_j = X_{j+1} - X_j$.

While the ODEs (4) remain unchanged, the ODEs (5), which govern the node motion, become

$$\sum_{j=1}^{N} \langle \beta_i, \alpha_j \rangle \dot{U}_j + \langle \beta_i, \beta_j \rangle \dot{X}_j + \varepsilon_i^2 \Delta \dot{X}_i - \varepsilon_{i+1}^2 \Delta \dot{X}_{i+1}$$
$$= \langle \beta_i, L(U) \rangle + \varepsilon_i S_i - \varepsilon_{i+1} S_{i+1} \,. \tag{9}$$

The left-hand side terms, which involve the internodal viscosities ε_j, regularize the dynamic internodal node movements and keep the resulting mass-matrix positive definite, while the right-hand side terms, which contain the internodal spring functions S_j, allows a regularization of the long term or equilibrium system. The internodal viscosities induce forces that tend to discourage relative node motion and degenerate nodes (at which the graph of the solution becomes straight) are driven along essentially in step with the neighboring nodes.

According to the original algorithm formulation [3], the regularizing functions are given by

$$S_j = (\frac{c_1}{\Delta X_j - \delta} - c_2(\Delta X_j - \delta))\ (1 + \frac{\delta}{\Delta X_j - \delta})^2, \tag{10}$$

$$\varepsilon_j = (\frac{c_3}{\Delta X_j - \delta} + c_4)\ (1 + \frac{\delta}{\Delta X_j - \delta})^2, \tag{11}$$

in which c_1, c_2 (in most cases, $c_2 = 0$) , c_3, c_4 and δ are tuning parameters. In particular, δ can be interpreted as a minimum permissible internodal separation. Both the viscosities and the internodal spring functions becomes infinite as the internodal separation approaches δ. In each of the expressions (10, 11), the first factor incorporates a short range term and a long range term; the second factor emphasizes the short range forces or viscosities. As is apparent on inspection of (9), it is in fact ε_j^2 and $\varepsilon_j S_j$ that play a regularizing role and, with the expressions (10, 11), a relatively sophisticated behavior of the regularization terms can be achieved, i.e. (for $c_2 = 0$) $\varepsilon_j S_j$ has inverse sixth power behavior for $0 < d_j = \Delta X_j - \delta < \delta$, inverse square for $\delta < d_j < c_3/c_4$ and inverse linear for $d_j > c_3/c_4$; the corresponding behaviors of ε_j^2 are inverse sixth, inverse square, and constant (for more details see [3]). Several variations and simplifications of these expressions have been proposed in the literature. In particular, it might be more convenient to directly work with expressions for ε_j^2 and $\varepsilon_j S_j$ of the form [5],

$$\varepsilon_j^2 = \frac{A^2}{\Delta X_j - \delta}\ , \tag{12}$$

$$\epsilon_j S_j = \frac{B^2}{(\Delta X_j - \delta)^2}, \tag{13}$$

in which A, B and δ are the tuning parameters.

As with every adaptive grid technique, a certain amount of parameter tuning is required to obtain satisfactory node motions and good solution performance. However, in some instances, the regularization terms alone might be insufficient

to achieve this objective, and in particular to avoid clustering most of the nodes in the steepest part of the solution. To remedy the situation, Miller [5] suggested using a gradient-weighted L_2 norm for the evaluation of the PDE residual. This weighted norm can be interpreted as the L_2 norm of the "normal component" of the residual, integrated with respect to the arclength s of the solution, i.e.,

$$\begin{aligned} |||R(U)|||^2 &= \int [\partial U/\partial t - L(U)]_N^2 \, ds \\ &= \int \left[(\partial U/\partial t - L(U))(1 + (\partial U/\partial x)^2)^{-1/2} \right]^2 \\ &\quad \times (1 + (\partial U/\partial x)^2)^{1/2} dx \\ &= \int [(\partial U/\partial t - L(U))]^2 \, (1 + (\partial U/\partial x)^2)^{-1/2} dx \end{aligned} \tag{14}$$

The main advantage of this new norm definition is that $[\partial U/\partial t]_N = \partial U/\partial t \, (1 + (\partial U/\partial x)^2)^{-1/2}$, which represents the "normal speed" of the graph of the solution, remains bounded even in an arbitrarily steep front (while $\partial U/\partial t$ itself, which represents the "vertical speed" of the graph of u as a function of x, is a near delta function in the case of a true shock). The minimization of (14) with respect to $\dot{U}_j(t)$ and $\dot{X}_j(t)$ yields a set of $2N$ ODEs that keep the same form as the ODEs (4, 5) with the inner products replaced by weighted inner products. Since the weighting function $w = (1 + (\partial U/\partial x)^2)^{-1/2}$ is a constant on each element, these changes are easily implemented (in other words, α_j and β_j are replaced by $w\alpha_j$ and $w\beta_j$).

As a result of further studies, Miller [13] proposed alternative regularization terms in the residual minimization

$$\min_{\dot{U}_j, \dot{X}_j} |||R(U)|||^2 + \sum_{j=2}^{N} (\varepsilon_j \dot{l}_j - S_j)^2 \,, \tag{15}$$

in which $\varepsilon_j^2 = A^2/l_j$, $\varepsilon_j S_j = B^2/l_j^2$, and $l_j = \sqrt{\Delta U_j^2 + \Delta X_j^2}$ is the arclength of the j^{th} solution segment. This latter definition of the regularization terms does not include a minimum permissible internodal separation δ any longer since it is unlikely that l_j tends to zero (both ΔU_j and ΔX_j would tend to zero). While this formulation appears simpler and more natural, it yields a system of $2N$ equations, in which both the ODEs of the type (4) and (5) are affected by the regularization terms, and so, which is more complex and nonlinear than the corresponding regularized MFE ODEs (4, 9) (for more details see [12]).

In all the previous developments, the PDE residual minimization yields a system of ODEs in the form $A(Y)\dot{Y} = g(Y)$. Considering the case of the

MFE ODEs (4, 5), Wathen [7] demonstrated that the preconditioned matrix $D(Y)^{-1}A(Y)$, in which $D(Y)$ is the block-diagonal of the mass matrix $A(Y)$ (7)

$$D(Y) = \begin{bmatrix} A_1 & & 0 \\ & ... & \\ 0 & & A_N \end{bmatrix} \tag{16}$$

is well-conditioned and has a condition number independent of the grid and the solution. Miller [13] extended these results to the GWMFE ODEs.

In a recent paper [10], Carlson and Miller describe a new 1-D code, called GWMFE1DS, which implements a state-of-the-art version of the GWMFE method (Eq-s. (14, 15)). In this implementation, the GWMFE ODEs $A(Y)\dot{Y} = g(Y)$ are integrated in time using a tailored implementation of a variable time step Backward Differentiation Formula (BDF) method of second order. In this algorithm, preconditioned Krylov subspace methods, with $D(Y)$ as preconditioner, are used to solve the nonlinear algebraic system $R(Y) = 0$ arising in each implicit time step (with the approximation of $\partial[D(Y)^{-1}R(Y)]/\partial Y$ by $D(Y)^{-1}\partial R(Y)/\partial Y$). The initial guess for Newton's method is given by second order extrapolation of previous values. The computed predictor error is used as a measure of the local error, the max norm of which is kept less than a prescribed absolute tolerance *ptol* (which can be set independently for each solution component and the grid). The error tolerances for Newton's method are set ten times smaller.

In addition, this code includes additional improvements such as a careful round off error control in computing the inner products in the PDE residuals (a catalog of inner products is given in [10]), and a relative error tolerance on the internodal separation.

An elegant geometrical - mechanical interpretation of the GWMFE method [9], as a balance of forces on the nodes, allows simplification of the regularizations to be achieved, especially in 2- and 3-D. In addition, a new expression for the spring forces in (15) is introduced, which is no longer in the form B^2/l_j^2, but instead in the form $B^2 l_j^2$. However, for most problems, $B^2 = 0$ gives satisfactory results, so that the only remaining tuning parameter is A^2.

Following this interpretation, the MFE method can be seen as the "small-slope limit" of the GWMFE method, i.e., MFE can be obtained from GWMFE by vertical rescaling of the graph of u by a large factor M. This observation suggests another method feature, which contributes to improved robustness, i.e., the use of a vertical rescaling factor M, which "flattens" the vertical dimensions of the graph of u and yields smaller apparent front slopes. This is accomplished

by replacing the dependent variable u by Mu in the PDE and by reducing the initial values $u(x,0)$ by a factor of M.

3. Numerical Experiments

In this section, the results of a set of numerical experiments with four test-problems, i.e., two models from combustion theory, the Euler equations, and the cubic Schrödinger equation, are presented. These numerical experiments are carried out with two codes:

- Our implementation of the original MFE method with the regularization terms given by (10, 11), in which the ODEs $A(Y)\dot{Y} = g(Y)$ (4) and (9) are integrated in time using the Backward Differentiation Formula (BDF) solver LSODI [14]. Starting with this basic version, a step-by-step procedure is used to illustrate the influence of the choice of regularizing parameters (internodal viscosities and spring functions), the selection of the initial node distribution, the method used to compute the inner products (either based on an analytical expression or using numerical quadrature), and the use of block-diagonal preconditioning. In this latter case, it is required to solve the preconditioned ODE system $D(Y)^{-1}A(Y)\dot{Y} = D(Y)^{-1}g(Y)$ or $A^*(Y)\dot{Y} = g^*(Y)$, where

$$A^*(Y) = \begin{bmatrix} I & A_1^{-1}B_1 & & & & & 0 \\ A_2^{-1}C_2 & I & A_2^{-1}B_2 & & & & \\ & A_3^{-1}C_3 & I & A_3^{-1}B_3 & & & \\ & & & & \ddots & & \\ 0 & & & & & A_N^{-1}C_N & I \end{bmatrix} \quad (17)$$

  using LSODI. The computation of the blocks $A_j^{-1}B_j$, $A_j^{-1}C_j$, and $A_j^{-1}g_j$ is performed by solving linear systems of $npde + 1$ equations using LU-factorization (e.g., the $npde + 1$ linear equations $A_jZ = B_j$ have the solution $Z = A_j^{-1}B_j$). In both cases (unpreconditioned and preconditioned ODE systems), the Newton matrix is treated as a banded matrix, which is evaluated numerically.

- The new code GWMFE1DS developed by Carlson and Miller [10], which incorporates many sophisticated features, and allows the influence of gradient weighting and vertical rescaling to be highlighted.

For each test example, a table presents a summary of computational statistics: the number of nodes N, the number of time steps STEPS needed to complete the solution, the number of function evaluations FNS, the number of Jacobian evaluations JACS, as well as the normalized CPU time, $CPU = CPU(s)/CPU_{\min}(s)$, where $CPU_{\min}(s)$ is the smallest computation time for the problem under consideration. For illustration purposes, a few graphs are also given, in which circled points are used for the numerical results while solid lines represent a reference solution computed with a large number of nodes or the analytical solution, when available. The location of the moving nodes is indicated at the bottom of the figures.

Example 1. A model of a single-step reaction with diffusion. A reaction-diffusion problem from combustion theory, first proposed as a test-example in [15], is given by

$$\partial T/\partial t = \partial^2 T/\partial x^2 + D(1+\alpha-T)e^{-\delta/T}, \quad 0 < x < 1, \ t > 0\,, \tag{18}$$

where $D = R\ e^{\delta}/\alpha\delta$.

The boundary and initial conditions are:

$$\partial T/\partial x(0,t) = 0, \qquad T(1,t) = 1, \qquad T(x,0) = 1, \qquad 0 \le x \le 1\,. \tag{19}$$

This test problem is solved on a time interval $(0, 0.29)$ and for the parameter values $\alpha = 1$, $\delta = 20$, $R = 5$.

For short times, the temperature gradually increases from unity with a maximum at $x = 0$. Suddenly, ignition occurs and the temperature at $x = 0$ rapidly increases to $1 + \alpha$. A steep front then forms and quickly propagates towards $x = 1$, where the problem reaches a steady state.

One of the main difficulties in this problem is to accurately capture the start of the ignition, which occurs at $t \approx 0.25$ and proceeds very rapidly. Errors at this stage of the solution may result in significantly larger errors later on. Hence, stringent error tolerances for the time integrator LSODI must be imposed, i.e., $atol = rtol = 10^{-7}$.

The initial node distribution is also of primary importance in the MFE method, particularly when using a small number of nodes. In fact, the best solution is obtained by selecting an initial nonuniform grid with more nodes concentrated in a small region near $x = 0$ and by preventing these nodes from moving away from this region before the ignition time. A series of numerical experiments was carried out to justify this observation and to illustrate the sensitivity of the numerical solution to the initial node distribution:

- Trial 1: the problem is solved using an initial grid with $N = 31$ uniformly distributed nodes, and the regularization parameters $c_1 = 10^{-3}$, $c_3 = 10$,

$c_4 = 10^{-3}$, $\delta = 10^{-6}$. In fact, with this large value of the parameter c_3, the nodes are rooted to the spot so that the grid remains uniform on the complete solution interval. In this case, the start of ignition is not accurately captured so that, at subsequent times, the computed flame front is largely ahead of the exact solution.

- Trial 2: an initial nonuniform grid with 10 uniformly distributed nodes in $[0, 0.0001]$ and 21 uniformly distributed nodes in $[0.0001, 1]$ is used with the same regularization parameters as in Trial 1. The concentration of nodes in a small region near $x = 0$ allows the start of the ignition to be satisfactorily captured, but the large value of the parameter c_3 prevents the nodes from following the flame front, which is not fully satisfactorily reproduced.

- Trial 3: the problem is solved under the same conditions as in Trial 2, but using a small value for the viscosity coefficient $c_3 = 0.001$. The nodes, which are initially concentrated near $x = 0$, can now move before the start of the ignition, which is not satisfactorily captured. This leads to large subsequent errors in the computed flame front, as shown in Figure 1. In fact, this figure suggests that, at $t = 0.26$, the grid would be quite capable of resolving the smooth solution profile, *if the spatial nodes were fixed*. In turn, it seems that *an inaccurate node movement* would be responsible for the huge errors observed in the solution. Attempts have been made to tighten the error tolerances imposed on the nodal positions, but without success.

- Trial 4: a compromise solution between Trials 2 and 3 consists of setting $c_3 = 10$ for $t \leq 0.258$ and $c_3 = 0.001$ for $t > 0.258$. In this case, the initial concentration of nodes near $x = 0$ is maintained until the ignition occurs, so that the start of the ignition is accurately detected. Then, the nodes are released so that they can follow the flame front, which is satisfactorily reproduced.

- Trial 5: using the somewhat tricky parameter tuning of Trial 4, the number of nodes can be further reduced to $N = 15$. At the initial time, 10 nodes are uniformly distributed nodes in $[0, 0.0001]$ and 5 nodes are uniformly distributed nodes in $[0.0001, 1]$. Both the start of the ignition and the flame propagation are very satisfactorily reproduced, as illustrated in Figure 2 (note that a slight retuning of c_1 to 10^{-4} is required to get these results, which otherwise are a little bit less satisfactory at the ignition time).

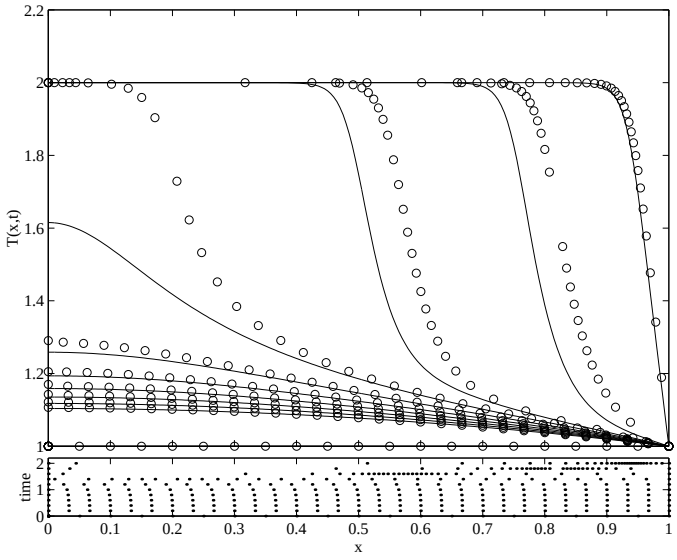

Figure 1: Example 1 - Trial 3: MFE with a nonuniform initial distribution of $N = 31$ nodes, $c_1 = 10^{-3}$, $c_3 = 10$, $c_4 = 10^{-3}$, $\delta = 10^{-6}$ - temperature from $t = 0.2$ to 0.29 at time intervals of 0.01

These numerical experiments show that the problem (18-19) cannot be solved efficiently with the MFE method, unless ad-hoc initial node distribution and parameter tuning are adopted. Of course, the sensitivity of the solution to these choices is decreased by increasing the number of nodes, but, in this case, the effectiveness of a moving grid strategy becomes questionable. This observation is supported by the results presented in [12], in which this problem is solved with the MFE method using an initial uniform grid with 40 nodes. In these experiments, although the ignition is not accurately detected, the flame front propagation is well reproduced.

The computational statistics for trials 1-5 are summarized in Table 1. It is apparent (compare Trials 3-4 to Trial 2) that the use of a small viscosity coefficient c_3, which allows the nodes to follow the flame front, results in higher computational costs. As explained in Section 2, the origin of this increasing computational effort is in the structure of the MFE equations. Hence, expected savings due to the reduction in the number of nodes (as compared to a solution obtained on a non moving grid) could be obscured by larger costs for the time integration. In Trials 6-7, a block-diagonal matrix preconditioning of the equations is used. The Newton matrix $\partial[D(Y)^{-1}R(Y)]/\partial Y$ from the preconditioned ODE system is more slowly varying than the one $(\partial R(Y)/\partial Y)$ from the

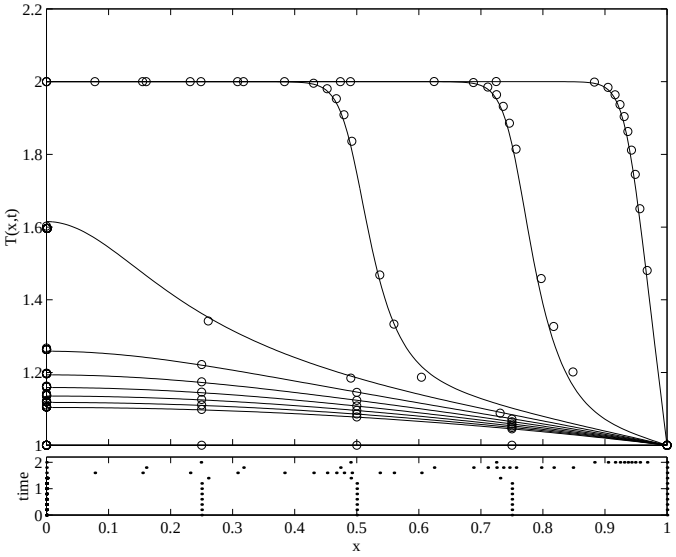

Figure 2: Example 1 - Trial 5: MFE with a nonuniform initial distribution of $N = 15$ nodes, $c_1 = 10^{-4}$, $c_3 = 10$ for $t \leq 0.258$ and $c_3 = 10^{-3}$ for $t > 0.258$, $c_4 = 10^{-3}$, $\delta = 10^{-6}$ - temperature from $t = 0.2$ to 0.29 at time intervals of 0.01

original systems. Hence, this matrix needs to be evaluated and LU-factored less frequently, resulting in improved performance of time integration, as reflected in Table 1.

In all these numerical experiments, the inner products of the basis functions α_i and β_i with the nonlinear source term $D(1+\alpha-T)e^{-\delta/T}$ in (18) are evaluated by numerical quadrature using a 6-point closed Newton-Cotes' rule (or Boole's rule).

Both the influence of matrix preconditioning and of the evaluation of the inner products by numerical quadrature will be further discussed in the next

Trial #	Initial grid	Tuning	Precond.	FNS	JACS	STEPS	CPU
1	unif. (31)	$c_3 = 10$	no	967	47	477	1.9
2	nonunif. (31)	$c_3 = 10$	no	1286	72	543	2.5
3	nonunif. (31)	$c_3 = 10^{-3}$	no	5978	398	1252	11.3
4	nonunif. (31)	time adjusted	no	4359	289	892	8.2
5	nonunif. (15)	time adjusted	no	4738	367	891	4.0
6	nonunif (31)	time adjusted	yes	2389	140	801	5.7
7	nonunif. (15)	time adjusted	yes	1686	97	590	1.7
8	unif. (31)	$A^2 = 10^{-6}$	yes	2469	262	339	4.5
9	unif. (15)	$A^2 = 10^{-6}$	yes	1445	112	278	1.0

Table 1: Computational statistics for example 1

example (flame propagation model).

Finally, the problem is solved, in a very straightforward way, with the code GWMFE1DS using an initial grid with $N = 31$ (Trial 8) or $N = 15$ (Trial 9) uniformly distributed nodes, the regularization parameter $A^2 = 10^{-6}$ ($B = 0$), and the predictor error tolerances for the solution and the grid $ptol = 10^{-3}, 10^{-3}$. In both cases, the computed solution is very satisfactory both in terms of accuracy and computational expense, which demonstrates the efficiency and ease of use of this code. Neither a nonuniform initial grid nor a specific parameter tuning are required in order to accommodate the two solution phases, i.e., ignition and flame propagation. Trial 9 leads to the smallest CPU time ($0.44s$ with a Pentium II 233), which is used as a reference in Table 1.

Example 2. A flame propagation model. We consider a model of flame propagation proposed by Dwyer and Sanders [16], which consists of two coupled equation for mass density and temperature

$$\begin{aligned} \partial\rho/\partial t &= \partial^2\rho/\partial x^2 - N_{DA}\rho\,, \\ \partial T/\partial t &= \partial^2 T/\partial x^2 + N_{DA}\rho, \qquad 0 < x < 1,\ t > 0\,, \end{aligned} \tag{20}$$

where $N_{DA} = 3.52 \times 10^6\ e^{-4/T}$.

The initial conditions are given by

$$\rho(x,0) = 1, \qquad T(x,0) = 0.2, \qquad 0 \le x \le 1\,, \tag{21}$$

and the boundary conditions are

$$\begin{aligned} &\partial\rho/\partial x(0,t) = 0, \qquad \partial T/\partial x(0,t) = 0, \\ &\partial\rho/\partial x(1,t) = 0, \qquad T(1,t) = f(t), \qquad t \ge 0\,, \end{aligned} \tag{22}$$

with

$$\begin{aligned} f(t) &= 0.2 + t/2 \times 10^{-4}, \qquad t \le 2 \times 10^{-4} \\ &= 1.2 \qquad\qquad\qquad\quad t \ge 2 \times 10^{-4}\,. \end{aligned} \tag{23}$$

This problem is solved on the time interval $(0, 0.006)$. The heat source located at $x = 1$ generates a flame front, which propagates from right to left at an almost constant speed ≈ 142. Besides the comparison with a reference solution, the reproduction of this speed of propagation can serve as a criterion to evaluate the quality of a numerical scheme (note at this stage that the value of 150 given in [12] appears to be not very accurate).

First, several numerical experiments are performed to illustrate the influence of the initial node distribution and the use of block-diagonal preconditioning (the computational statistics are summarized in Table 2):

Trial #	Initial grid	Precond.	Scaling	FNS	JACS	STEPS	CPU
1	unif. (74)	no	no	16354	1092	1267	284
2	nonunif. (74)	no	no	35267	2411	2336	614
3	nonunif. (74)	yes	no	26079	1783	2038	561
4	nonunif. (21)	yes	no	4349	283	468	24.2
5	unif.(74)	yes	no	859	74	188	8.5
6	nonunif. (74)	yes	no	993	92	190	10.2
7	nonunif. (21)	yes	20	403	31	112	1.0

Table 2: Computational statistics for example 2

- Trial 1: the problem is solved using an initial grid with $N = 74$ uniformly distributed nodes, and the MFE method with the regularization parameters $c_1 = 10^{-5}$, $c_3 = 10^{-3}$, $c_4 = 10^{-2}$, $\delta = 10^{-4}$. The error tolerances for the time integrator LSODI are $atol = rtol = 10^{-4}$. At the first few output times $(0.006, 0.012, 0.018)$, the nodes concentrate mostly in left half of the front, and the speed of propagation is not accurately reproduced. Later, the nodes are more evenly distributed around the front and, although the location of the front is not correct, the speed of propagation is now satisfactorily reproduced.

- Trial 2: an initial nonuniform grid with an increasing density of nodes from left to right is used, i.e., $N = 74$ nodes are distributed according to:

  6 uniformly distributed nodes in $[0, 0.5]$

  8 uniformly distributed nodes in $(0.5, 0.7]$

  10 uniformly distributed nodes in $(0.7, 0.8]$

  20 uniformly distributed nodes in $(0.8, 0.9]$

  30 uniformly distributed nodes in $(0.9, 1]$

  The problem is solved using the same regularization parameters and tolerances as in Trial 1. In this case, the flame front and its speed of propagation are very accurately reproduced. However, the higher grid distortion makes the problem more computationally expensive.

- Trial 3: a block-diagonal preconditioning (using the inverse of the block-diagonal of the MFE mass matrix) of the MFE ODEs (4) and (9) is performed. Together with the regularization parameters, the matrix preconditioning allows the ODEs' stiffness to be reduced, and better computational performance to be achieved (see Table 2).

- Trial 4: starting with the positive results obtained in Trial 3, the number of nodes can be further reduced up to $N = 21$ (10 nodes in $[0, 0.9]$ and 11

nodes in $(0.9, 1]$) ! This reduced number of nodes still allows a very good reproduction of the flame front, and, together with matrix preconditioning, considerable computational savings.

Then, the same numerical experiments are performed using GWMFE1DS with the regularization parameter $A^2 = 10^{-7}$ ($B = 0$) and the predictor error tolerances for the solution and the grid $ptol = 10^{-3}$, $10^{-3}, 10^{-3}$. The same qualitative conclusions hold, i.e., the formation, and the subsequent propagation, of the flame front is not accurately reproduced, when using an initial uniform grid (Trial 5), while an initial nonuniform grid allows much better results (Trials 6-7 and Figures 3-4).

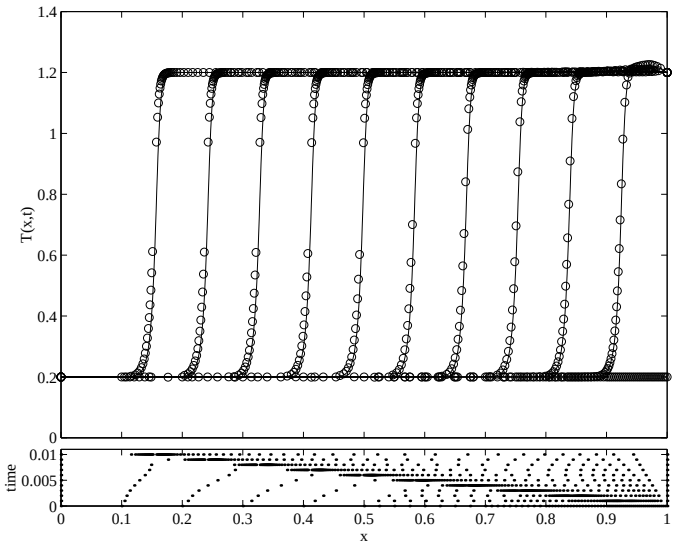

Figure 3: Example 2 - Trial 6: GWMFE with a nonuniform initial distribution of $N = 74$ nodes, $A^2 = 10^{-7}$ - temperature from $t = 0$ to 0.006 at time intervals of 0.0006

With GWMFE, the use of an initial nonuniform grid has less influence on the computational costs. However, the reduction in the number of nodes to $N = 21$ (Trial 7) requires:

(a) the use of a vertical rescaling factor $M = 20$, which deemphasizes the flame front. This is accomplished by replacing the dependent variables ρ and T by $M\rho$ and MT in the model PDEs and by reducing the initial values $\rho(x, 0)$ and $T(x, 0)$ by a factor of M. Accordingly, the predictor error

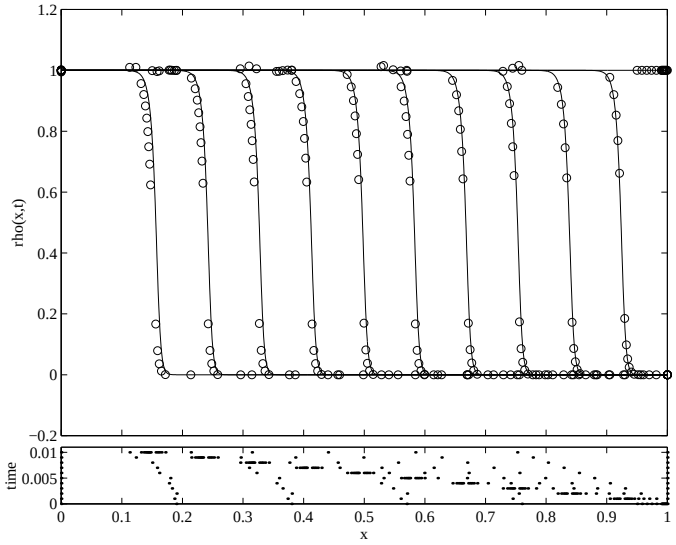

Figure 4: Example 2 - Trial 7: GWMFE with a nonuniform initial distribution of $N = 21$ nodes, $A^2 = 10^{-7}$, $M = 20$ - density from $t = 0$ to 0.006 at time intervals of 0.0006

tolerances for the solution and the grid are set to $ptol = 5 \times 10^{-5}, 5 \times 10^{-5}, 10^{-3}$.

(b) an adaptation of the initial grid distribution used in Trial 4 (which leads to a failure of the GWMFE method), i.e., 6 nodes in $[0, 0.95]$, 5 nodes in $(0.95, 0.99]$ and 10 nodes in $(0.99, 1]$ (note that, on the other hand, this initial node distribution gives satisfactory results with MFE).

Even with these refinements, the quality of the solution is not as good as in Trial 4. In particular, the estimated front velocity varies between 140 and 146. This experiment leads to the shortest CPU time ($0.22s$ with a Pentium II 233), which is used as a reference in Table 2.

For problems which immediately develop structure (front, shock, ...) out of nothingness at the initial time, the GWMFE method displays a higher sensitivity to the initial grid distribution than the MFE method. This situation, which contrasts with the first example in which the front gradually develops, will be further illustrated with the next example (Sod's shock tube problem).

Finally, attention is focused on the computation of the inner products of the basis functions α_i and β_i (or $w\alpha_i$ and $w\beta_i$) with the nonlinear source terms $N_{DA}\rho$ in (20). The level of accuracy of the numerical quadrature rule used to

evaluate these inner products has a considerable influence on the overall accuracy of the numerical scheme. When using a 3- or 4-point closed Newton-Cotes' rule, the flame front is virtually moved instantaneously to the left boundary in the first times, so that at least a 5-point closed Newton-Cotes' rule (or Milne's rule) is required to obtain satisfactory results corresponding to an average front velocity of 142.4. In all the numerical experiments presented in this section, a 6-point closed Newton-Cotes' rule (or Boole's rule) has been used.

Example 3. Conservation laws for gas dynamics. We consider Sod's shock tube problem [17] for the Euler equations of gas dynamics in conservation law form

$$\begin{aligned}
\partial u/\partial t &= -\partial v/\partial x + \nu \partial^2 u/\partial x^2 \,, \\
\partial v/\partial t &= -\partial (v^2/u + p)/\partial x + \nu \partial^2 v/\partial x^2 \,, \\
\partial w/\partial t &= -\partial (v(w+p)/u)/\partial x + \nu \partial^2 w/\partial x^2, \quad 0 < x < 1, \ t > 0 \,,
\end{aligned} \tag{24}$$

where u, v, w are the densities of mass, momentum and energy of a gas in a 1-D tube.

In the case of an ideal gas, the pressure is given by the equation of state $p = (\gamma - 1)(w - v^2/2u)$ with $\gamma = 1.4$.

Initially, a diaphragm located at $x = 0.5$ separates the gas at rest into two parts. The gas at higher density and pressure is to the left of the diaphragm. At $t = 0$, the diaphragm is burst yielding the piecewise constant initial conditions

$$\begin{aligned}
&u(x,0) = 1, \qquad v(x,0) = 0, \quad w(x,0) = 2.5, \quad 0 \le x \le 0.5 \,, \\
&u(x,0) = 0.125, \quad v(x,0) = 0, \quad w(x,0) = 0.25, \quad 0.5 < x \le 1 \,.
\end{aligned} \tag{25}$$

The initial discontinuity gives rise to a rarefaction wave travelling to the left and a shock followed by a contact discontinuity travelling to the right. At both extremities of the tube, reflection boundary conditions are imposed

$$\begin{aligned}
&\partial u/\partial x(0,t) = 0, \quad v(0,t) = 0, \quad \partial w/\partial x(0,t) = 0 \,, \\
&\partial u/\partial x(1,t) = 0, \quad v(1,t) = 0, \quad \partial w/\partial x(1,t) = 0, \qquad t \ge 0 \,.
\end{aligned} \tag{26}$$

This nontrivial test example is solved on the time interval $(0, 0.4)$ and with a small diffusion coefficient $\nu = 10^{-4}$. At $t \approx 0.288$, the shock reaches the right boundary, where it is reflected.

To solve this difficult problem with our implementation of the MFE method, several precautions had to be taken:

- an appropriate initial node distribution must be selected.

	grid 1	grid 2	grid 3	grid 4
$[0, 0.480]$	7	4	3	3
$(0.480, 0.497]$	10	5	5	4
$(0.497, 0.500]$	15	7	6	6
$(0.500, 0.503]$	25	13	16	16
$(0.503, 0.520]$	10	5	5	6
$(0.520, 1.00]$	6	3	2	2

Table 3: Several initial node distibutions

- a block-diagonal preconditioning of the equations must be performed.
- the inner products of the basis functions with the nonlinear right-hand side terms of the Euler equations must be computed using accurate numerical quadrature rules, e.g., Boole's rule (note that a straightforward analytical evaluation of these inner products does not yield satisfactory results since the resulting formulas involve divisions by the difference of nodal amplitudes $\Delta U_j = U_{j+1} - U_j$; these operations are inaccurate, when ΔU_j becomes very small and it is better to resort to a numerical quadrature rule).

In the following, we focus attention on the selection of an initial node distribution. We start with grid 1 in Table 3, which corresponds to the nonuniform distribution of $N = 73$ nodes recommended in [10]. This way, the initial discontinuity is replaced by a thin ramp of width 0.006. More nodes are placed in the right half of the ramp, where most of the solution structure develops. The problem is solved, on the one hand, using the MFE method with the regularization parameters $c_1 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 10^{-2}$, $\delta = 10^{-4}$ and the error tolerances $atol = rtol = 10^{-3}$, and on the other hand, with the GWMFE method with the regularization parameter $A^2 = 10^{-7}$ ($B = 0$), a vertical rescaling factor $M = 10$, and the predictor error tolerances for the solution and the grid $ptol = 10^{-4}, 10^{-4}, 10^{-4}, 10^{-3}$. Both methods yield a very satisfactory solution (Trials 1 and 5).

Then, the method robustness is tested by reducing the number of nodes up to a factor 2 ($N = 37$) while keeping the same tuning parameters. With the initial grid 2, MFE gives satisfactory results (Trial 2 and Figure 5), while GWMFE fails at the initial time (Trial 6). In grid 3, three more nodes are put in the right half of the initial ramp, which are taken from the first, third and sixth intervals defined in Table 3, so that GWMFE performs well (Trial 7 and Figure 6), except when the shock reaches the right boundary where it is

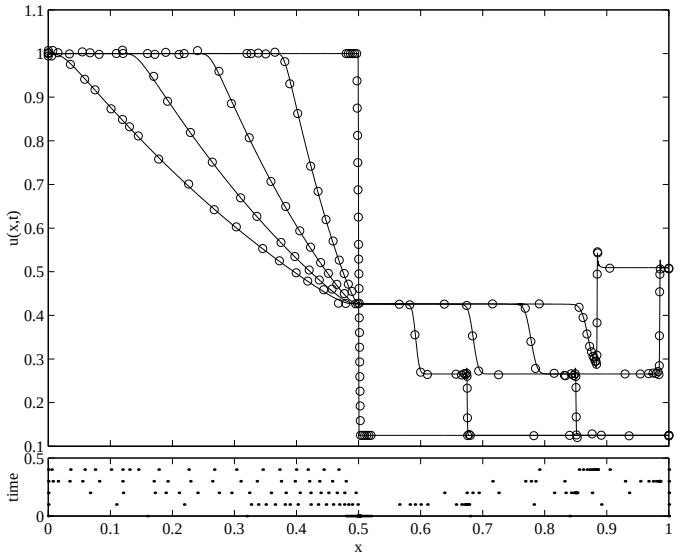

Figure 5: Example 3 - Trial 2: MFE with a nonuniform initial distribution of $N = 37$ nodes (grid 2), $c_1 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 10^{-2}$, $\delta = 10^{-4}$ - density from $t = 0$ to 0.4 at time intervals of 0.1

reflected. Again, MFE provides a very satisfactory solution (Trial 3). Finally, one node is moved from the second to the fifth interval to build grid 4. With this latter grid, both MFE and GWMFE provide very satisfactory results, and in this case, GWMFE leads to the fastest simulation run (Trial 8 and Figure 7 corresponding to a CPU time of $1.97s$ with a Pentium II 233, which is used as a reference in Table 4). These experiments demonstrate, as in the flame propagation model, the higher sensitivity of GWMFE to the initial distribution of nodes, which must be carefully located in order to detect the several solution structures.

Example 4. The cubic Schrödinger equation. The 1-D cubic Schrödinger equation (CSE) is given by

$$i\ \partial u/\partial t + \partial^2 u/\partial x^2 + q\left|u^2\right| u = 0, \qquad -\infty < x < \infty, t > 0 \tag{27}$$

where $u(x,t)$ is a complex-valued function of space and time and q is a real parameter. In particular, we consider the propagation of a single soliton, which corresponds to the initial condition:

$$u(x,0) = u_0(x) = \sqrt{2aq}\ e^{i0.5s(x-x_0)}\ \mathrm{sech}(\sqrt{a}(x-x_0))\,. \tag{28}$$

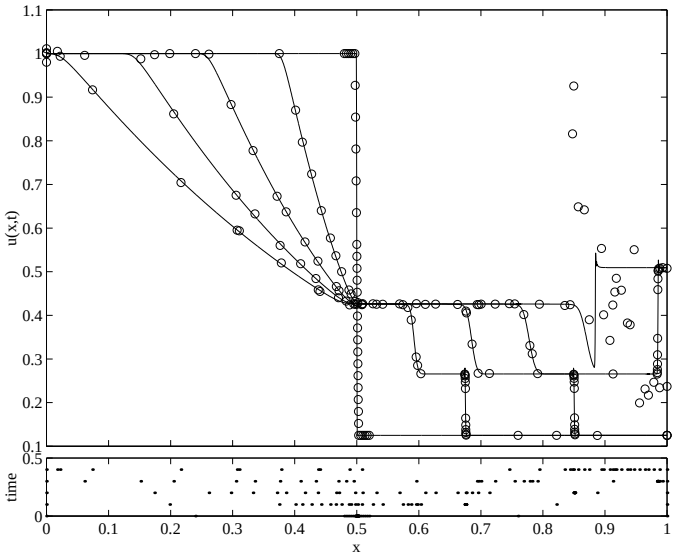

Figure 6: Example 3 - Trial 7: GWMFE with a nonuniform initial distribution of $N = 37$ nodes (grid 3), $A^2 = 10^{-7}$, $M = 10$ - density from $t = 0$ to 0.4 at time intervals of 0.1

In this case, the CSE possesses an exact solution:

$$u(x,t) = \sqrt{2aq}\; e^{i(0.5s(x-x_0)-(0.25s^2-a)t)}\; \mathrm{sech}(\sqrt{a}((x-x_0)-st)) \tag{29}$$

The modulus $\rho(x,t) = |u(x,t)|$ represents a wave initially centered at $x = x_0$, whose amplitude $\sqrt{2aq}$ is determined by the real parameter a, and which propagates with speed s in the positive direction of x. Note that the soliton travels without change of shape.

For the numerical treatment of the CSE, we assume that, for the time interval $0 \le t \le T$ under consideration, the solution vanishes outside some interval (x_L, x_R) so that artificial Dirichlet boundary conditions can be introduced, and we decompose $u(x,t)$ into its real and imaginary parts $v(x,t)$ and $w(x,t)$, which yields

$$\begin{aligned} &\partial v/\partial t + \partial^2 w/\partial x^2 + q\left(v^2+w^2\right)w = 0, \\ &\partial w/\partial t - \partial^2 v/\partial x^2 - q\left(v^2+w^2\right)v = 0, \qquad x_L < x < x_R,\ t > 0\,, \end{aligned} \tag{30}$$

supplemented by the initial conditions

$$v(x,0) = u_{0R}(x), \qquad w(x,0) = u_{0I}(x), \qquad x_L \le x \le x_R\,, \tag{31}$$

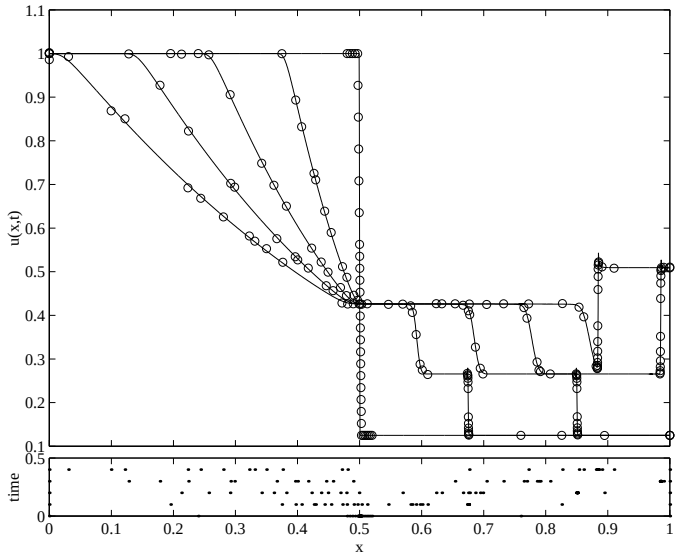

Figure 7: Example 3 - Trial 8: GWMFE with a nonuniform initial distribution of $N = 37$ nodes (grid 4), $A^2 = 10^{-7}$, $M = 10$ - density from $t = 0$ to 0.4 at time intervals of 0.1

where $u_0(x) = u_{0R}(x) + iu_{0I}(x)$, and by the boundary conditions

$$\begin{aligned} v(x_L,t) &= 0, \qquad v(x_R,t) = 0, \\ w(x_L,t) &= 0, \qquad w(x_R,t) = 0, \qquad 0 \le t \le T\,. \end{aligned} \tag{32}$$

Several attempts have been made to solve this problem (for the special case $q = 1$) with the MFE method, and to our surprise, we observed that the nodes, which are initially concentrated in the soliton, do not move, or even worse, move away from the soliton! These observations have been confirmed by K. Miller, who mentions this problem in [9].

Figures 8 and 9 show the graphs of the real and imaginary parts at the initial time ($x_0 = 0$) and, for each of the nodal values, the direction of the velocity vector $(\dot{x}_i, \dot{v}_i)$ and $(\dot{x}_i, \dot{w}_i)$, respectively. It is apparent that the velocity vectors either point in the vertical direction, which corresponds to no node motion, or away from the soliton, so that the soliton would then be deserted by the nodes!

The origin of these surprising observations is in the unusual combination $\partial v/\partial t$ - $\partial^2 w/\partial x^2$ and $\partial w/\partial t$ - $\partial^2 v/\partial x^2$ in Eq-s (30), in which the second order terms do not play a dispersive role as in the previous examples, but rather a "convective" role.

Trial #	Initial grid	FNS	JACS	STEPS	CPU
1	1	4267	222	339	11.5
2	2	3241	166	275	4.3
3	3	3090	159	267	4.1
4	4	3651	188	305	4.8
5	1	2644	152	708	2.0
6	2	–	–	–	–
7	3	9126	370	2745	3.1
8	4	2933	124	870	1.0

Table 4: Computational statistics for example 3

In order to alleviate this problem, we use a transformation from rectangular to polar coordinates and express $u(x,t) = \rho(x,t)e^{i\varphi(x,t)}$, so that (27) can be written as:

$$\begin{aligned} \partial\rho/\partial t + 2(\partial\rho/\partial x)(\partial\varphi/\partial x) + \rho\partial^2\varphi/\partial x^2 &= 0, \\ \partial\varphi/\partial t + (\partial\varphi/\partial x)^2 - (1/\rho)(\partial^2\rho/\partial x^2) - q\rho^2 &= 0, \end{aligned} \tag{33}$$

$x_L < x < x_R, t > 0$, supplemented by the initial conditions

$$\rho(x,0) = \rho_0(x), \qquad \varphi(x,0) = \varphi_0(x), \qquad x_L \le x \le x_R\,, \tag{34}$$

and by the boundary conditions

$$\begin{aligned} \rho(x_L,t) &= 0, \qquad \rho(x_R,t) = 0, \\ \varphi(x_L,t) &= 0, \qquad \varphi(x_R,t) = 0, \qquad 0 \le t \le T\,. \end{aligned} \tag{35}$$

Accordingly, the soliton (29) is given by

$$\begin{aligned} \rho(x,t) &= \sqrt{2aq}\frac{2}{e^{\sqrt{a}(x-st)} + e^{-\sqrt{a}(x-st)}}\,, \\ \varphi(x,t) &= 0.5sx - (0.25s^2 - a)t\,. \end{aligned} \tag{36}$$

Figures 10 and 11 show the graphs of the modulus $\rho(x,0)$ and argument $\varphi(x,0)$ at the initial time, respectively, and for each of the nodal values, the direction of velocity vectors $(\dot{x}_i, \dot{\rho}_i)$ and $(\dot{x}_i, \dot{\varphi}_i)$, which now have components pointing in the correct direction. We can therefore expect that the MFE method will achieve appropriate node movements. This is indeed the case, and in order to get some insight into the way the nodes will move, it is interesting to observe

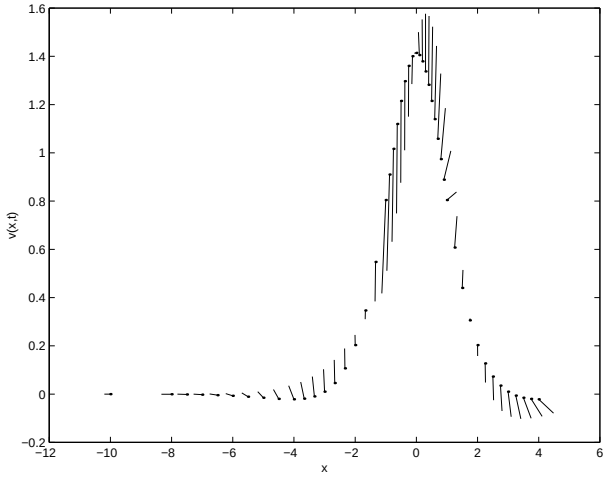

Figure 8: Example 4 - real part $v(x,0)$ and direction of the velocity vectors at the initial time

from (36) that $\partial\varphi/\partial x = 0.5s$ and $\partial^2\varphi/\partial x^2 = 0$, so that the equation for the modulus $\rho(x,t)$ reduces to

$$\partial\rho/\partial t = -s(\partial\rho/\partial x)\,. \tag{37}$$

This latter PDE is a simple advection equation, which can be efficiently solved with the MFE method, as illustrated in Figure 12. Again, the initial node distribution is critical, and in order to obtain a satisfactory solution, the nodes must be initially concentrated in the soliton. This problem is solved using an initial grid with $N = 71$ nonuniformly distributed nodes, and the MFE method with the regularization parameters $c_1 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 10^{-2}$, $\delta = 10^{-3}$. The error tolerances for the time integrator LSODI are $atol = rtol = 10^{-5}$.

In spite of satisfactory node movements, all our attempts to solve the transformed system (33) using the MFE method were unsuccessful. The problem now seems to lie in the computation of the inner products of the basis functions with the term $(1/\rho)(\partial^2\rho/\partial x^2) + q\rho^2$ in the equation for the phase. Indeed, if we assume that $(1/\rho)(\partial^2\rho/\partial x^2) + q\rho^2 = a$, as can be observed from (36), then the transformed system, which reduces to

$$\begin{aligned} &\partial\rho/\partial t + 2(\partial\rho/\partial x)(\partial\varphi/\partial x) + \rho\partial^2\varphi/\partial x^2 = 0, \\ &\partial\varphi/\partial t + (\partial\varphi/\partial x)^2 - a = 0, \qquad x_L < x < x_R, \quad t > 0 \end{aligned} \tag{38}$$

can be easily solved with the MFE method, leading to results similar to Figure 12. Inaccuracies in the computation of the inner products (recall that second-

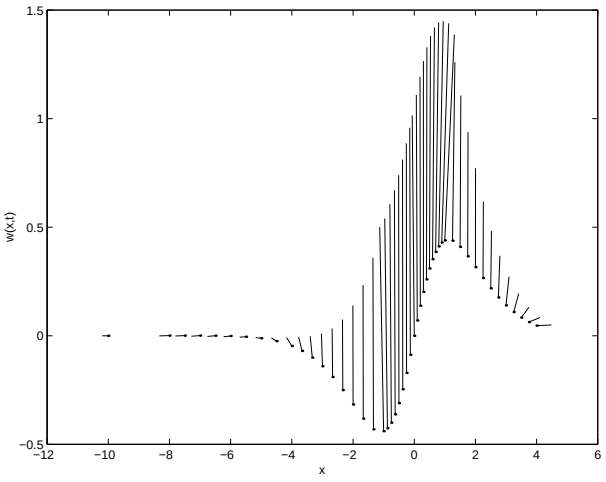

Figure 9: Example 4 - imaginary part $w(x,0)$ and direction of the velocity vectors at the initial time

order terms are computed using a mollification technique), leading to changes in the amplitude $\sqrt{2aq}$ of the soliton, result in failures of the simulation run.

4. Concluding Remarks

Over the past several years, the moving finite element method has proved particularly efficient in solving a wide range of difficult problems. However, there are potential pitfalls in the implementation of MFE codes and their subsequent use, a situation that has led to many improvements of the basic method developed in the late seventies.

This paper, which is directed to the non-specialist, aims at illustrating in a practical way the importance of several of the method refinements, including regularizations, gradient-weighting, vertical rescaling, matrix preconditioning. The test-examples have been selected so as to show, on the one hand, the impressive efficiency of the method, and on the other hand, the dramatic influence that some parameters or method variations can have on the solution quality. In this connection, some limiting cases are explored by reducing substantially the number of moving nodes and selecting carefully their initial distribution. The last example, the cubic Schrödinger equation, represents an extreme case for which the MFE method experiences severe difficulties and appears intrinsically inadequate.

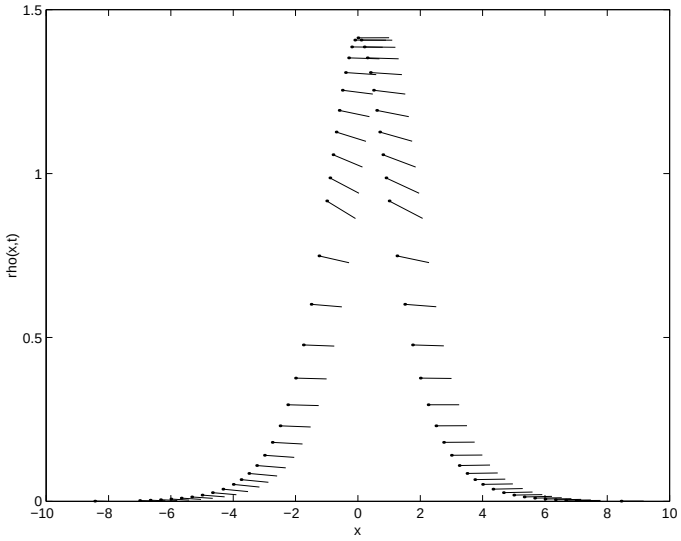

Figure 10: Example 4 - modulus $\rho(x,0)$ and direction of the velocity vectors at the initial time

In contrast to earlier criticisms of the method, particularly concerning regularizations and parameter tuning, we have found that even the basic MFE method can be very efficient and robust if some precautions are taken in formulating the MFE ODEs (such as a careful computation of inner products) and in solving them with a time integrator (such as matrix preconditioning of the residual equations and order selection of the BDF method). Under these conditions, tuning remains problem-dependent, but appears much easier and less sensitive to changes in node number and initial distribution. On the other hand, N. Carlson and K. Miller have recently reported on the design of state-of-the-art codes [10], [11], which achieve the combined objectives of efficiency, robustness and ease of use. The price to pay is a high degree of numerical sophistication, which is, however, fully transparent for the user.

References

[1] D.F. Hawken, J.J. Gottlieb, J.S. Hansen, Review of some adaptive node-movement techniques in finite-element and finite difference solutions of partial differential equations, *J. Comp. Phys.*, **95** (1991), 254-302.

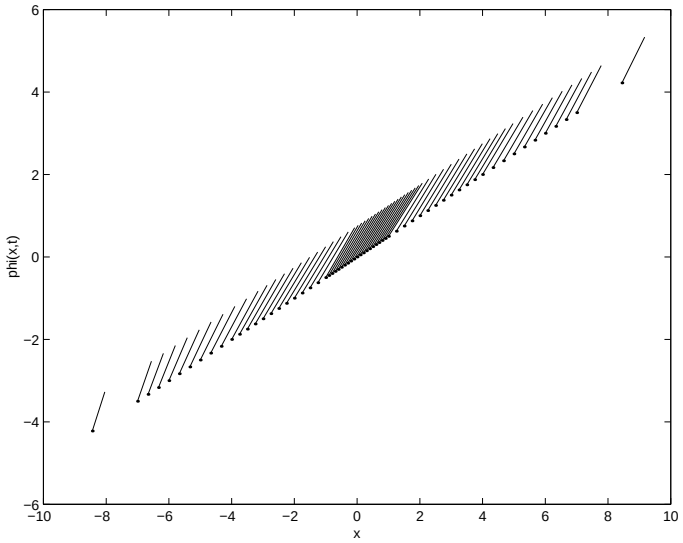

Figure 11: Example 4 - argument $\varphi(x, 0)$ and direction of the velocity vectors at the initial time

[2] K. Miller, R. Miller, Moving finite elements, Part I, *SIAM J. Numer.*, **18** (1981), 1019-1032.

[3] K. Miller, Moving finite elements, Part II, *SIAM J. Numer.*, **18** (1981), 1033-1057.

[4] R. Gelinas, S. Doss, K. Miller, The moving finite element method: applications to general equations with multiple large gradients, *J. Comp. Phys.*, **40** (1981), 202-249.

[5] K. Miller, Alternate modes to control the nodes in the moving finite element method, In: *Adaptive Computational Methods for PDEs* (Ed-s. I. Babuska, J. Chandra, J.E. Flaherty), SIAM (1983), 165-182.

[6] A.J. Wathen, M.J. Baines, On the structure of the moving finite-element equations, *IMA J. Numer. Anal.*, **5** (1985), 161-182.

[7] A.J. Wathen, Mesh-independent spectra in the moving finite element equations, *SIAM J. Numer.*, **23** (1986), 797-814.

[8] K. Miller, On the mass matrix spectrum bounds of Wathen and the local moving finite elements of Baines, *SIAM J. Numer.*, **29** (1992), 89-106.

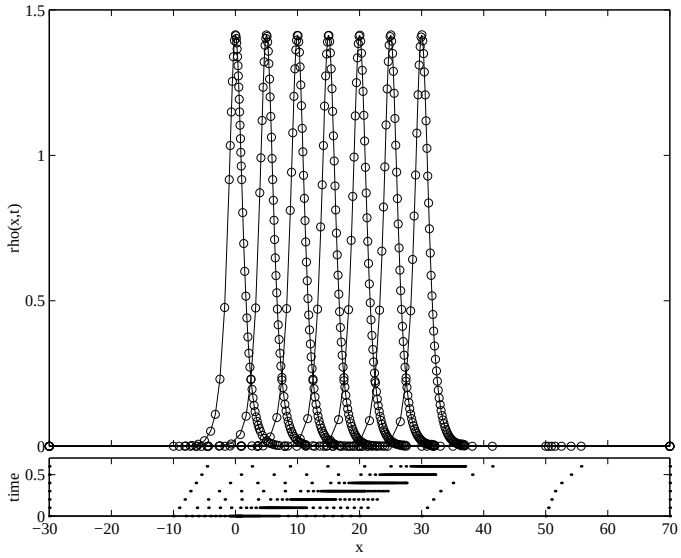

Figure 12: Example 4 - propagation of a single soliton from $t = 0$ to 30 at time intervals of 5 - MFE with a nonuniform initial distribution of $N = 71$ nodes, $c_1 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 10^{-2}$, $\delta = 10^{-3}$

[9] K. Miller, A geometrical-mechanical interpretation of gradient-weighted moving finite elements, *SIAM J. Numer.*, **34** (1997), 67-90.

[10] N. Carlson, K. Miller, Design and application of a gradient-weighted moving finite element code, Part I, in 1-D, *SIAM J. Sci. Comput.*, **19** (1998), 728-765.

[11] N. Carlson, K. Miller, Design and application of a gradient-weighted moving finite element code, Part II, in 2-D, *SIAM J. Sci. Comput.*, **19** (1998), 766-798.

[12] P. Zegeling, *Moving-grid Methods for Time-dependent Partial Differential Equations,* CWI Tract 94, Stichting Mathematisch Centrum, Amsterdam, The Netherlands (1993).

[13] K. Miller, Recent results on finite element methods with moving nodes, In: *Accuracy Estimates and Adaptive Refinements in Finite Element Computations* (Ed-s: I. Babuska, O.C. Zienkiewicz, J. Gago, E.R. DE A. Oliveira), John Wiley (1986), 325-338.

[14] A.C. Hindmarsh, ODEPACK: A systematized collection of ODE solvers, In: *Scientific Computing* (Ed. R.S. Stepleman), IMACS, North Holland (1983), 55-64.

[15] S. Adjerid, J.E. Flaherty, A moving finite element method with error estimation and refinement for one-dimensional time-dependent partial differential equations, *SIAM J. Numer.*, **23** (1986), 778-796.

[16] H.A. Dwyer, R.J. Kee, B.R. Sanders, Adaptive grid method for problems in fluid mechanics and heat transfer, *AIAA J.*, **18** (1980), 1205-1212.

[17] G.A. Sod, A survey of several finite difference methods for systems of hyperbolic conservation laws, *J. Comp. Phys.*, **27** (1978), 1-31.