Moving Mesh Methods with Upwinding Schemes for Time-Dependent PDEs

Shengtai Li*and Linda Petzold†

Abstract

It is well-known that moving mesh and upwinding schemes are two kinds of techniques for tracking the shock or steep wave front in the solution of PDEs. It is expected that their combination should produce more robust methods. Several upwinding schemes are considered for non-uniform meshes. A self-adaptive moving mesh method is also described. Numerical examples are given to illustrate that in some cases, especially for hyperbolic conservative laws with nonconvex flux, the upwinding schemes improve the results of the moving mesh methods. Comparing the results of several upwinding schemes, we find the local piecewise hyperbolic method (PHM) is very efficient and accurate when combined with a moving mesh strategy.

Keywords: Time-dependent PDEs, method of lines, moving mesh, ENO, upwinding, r-refinement.

*Dept. of Computer Science and Army High Performance Computing Research Center, University of Minnesota, MN 55455. The work of this author was partially supported by the Army High Performance Computing Research Center and by ARO contract number DAAL03-92-6-0247.

†Dept. of Computer Science and Army High Performance Computing Research Center, University of Minnesota, MN 55455. The work of this author was partially supported by NIST contract number 60 NANB2D1272, NSF contract number NSF CCR-95-27151, by the Minnesota Supercomputer Institute and by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory Cooperative agreement number DAAH04-95-2-0003/contract number DAAH04-95-C-0008, the content of which does not necessarily reflect the position or the policy of the government, and no official endorsement should be inferred.

1 Introduction

Moving mesh 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 mesh methods use nonuniform spatial grids, and move the grid continuously in the space-time domain. The discretization of the PDE and the grid selection procedure are intrinsically coupled.

In this paper we investigate moving mesh methods applied to PDEs of the following form

$$\frac{\partial u}{\partial t} = \frac{\partial f(u)}{\partial x} + \mu u_{xx} + h(u) \tag{1}$$

where $f(u)$, $h(u)$ are differentiable functions. This equation, which is often called a reaction-convection-diffusion equation, is representative of many problems which are solved by moving mesh methods in one dimension.

Much work has been devoted to the construction of shock capturing schemes on uniform grids, such as TVD, Godunov, MUSCL, PPM, FCT, ENO and PHM (see Salari and Steinberg[21] and its references, Shu and Osher[22] and Marquina[16] for more information). The conventional scheme on a uniform grid needs enough points to capture the shock or steep wave front, which often puts too many points in the area where the solution is very flat and smooth. Moving mesh methods, on the other hand, can redistribute the nodes to those areas where the solution varies rapidly at each time step to gain computational efficiency. Thus, many useful schemes have been applied on moving grids in recent years. Hyman and Harten [8] applied the Godunov method on a nonuniform grid where the grids move along the characteristic direction. Salari and Steinberg [21] developed a FCT method on a moving grid (FCTMG) based on adaptive grid generation algorithms. Zarnowski [26] introduced a moving grid method for scalar conservation laws based on nonlinear interpolation with a means of adding or removing grid points accordingly as characteristics either spread in a rarefaction fan or merge into a shock wave. All these methods use the full discretized PDE form and a fixed time step in time integration.

In this paper, we apply the method of lines (MOL) to discretize the PDEs. The advantages and disadvantages of MOL have been discussed by many authors. The moving grid method we choose is the class of methods based on the equidistributing principle (see [3,5,10,11,22,23] for detailed information). During discretization using MOL, the most troublesome term is the convection term $\frac{\partial f(u)}{\partial x}$. If the central difference scheme, which is adopted by most moving mesh software, is used to discretize this term, the odd-order derivatives predominate in the

truncation error (leading to numerical dispersion). The extra term can cause the solution to have an oscillatory behaviour (Gibbs oscillations) that the proper solution did not possess in the vicinity of the transition. Although the moving mesh method can place enough nodes in the wave front and works very well for a convex flux function and sufficiently smooth initial conditions (see the result of [5] or [11] for Burgers' equation), for nonconvex flux functions or piecewise initial conditions, there are still some oscillations appearing in the solution (see the result in section 4). High order upwinding schemes are needed in these cases. Almost all the upwinding schemes can be constructed on a nonuniform mesh. We will examine the ENO scheme, flux-limited scheme and PHM scheme. ENO schemes work very well with the constant time-step integration method but have great trouble when solved by adaptive time integration. The flux-limited scheme can be used for adaptive time integration; however it does not yield good results for some problems. The local third order PHM method[16], however, can be solved efficiently using a conventional ODE or DAE solver and can yield very good results.

When equation (1) is of hyperbolic form (i.e. $\mu = 0$), a very small artificial viscosity can be added to improve the computational stability. Numerical results show that this does not affect the accuracy of the solution very much. Under the moving mesh frame, an additional convective term, $u_x\dot{x}$, appears in the equation. The equation (1) becomes

$$\begin{cases} \dot{u} = \dfrac{\partial f(u)}{\partial x} + \mu u_{xx} + h(u) + u_x\dot{x} \\ 0 = g(u, x, \dot{x}) \end{cases}. \tag{2}$$

where mesh moving is governed by $g(u, x, \dot{x})$.

In section 2, we describe upwinding schemes which can be used on a nonuniform mesh following the construction methods of Shu and Osher [22] and other authors. In section 3, we discuss moving mesh methods based on equidistribution. In section 4, numerical experiments are given to illustrate the advantages of combining the moving mesh with an upwinding scheme and in particular the advantages of the PHM method.

2 Upwinding schemes on nonuniform mesh

In this section, we consider the discretization of the convective terms on a nonuniform mesh. Unless specified, we consider the general convective term $f(u)_x$ only. For simplicity, we consider the 1D scalar problem.

Generally, the convective term is discretized into conservation form, e.g.

$$\frac{\partial f(u(x))}{\partial x}\Big|_{x_j} = \frac{2(\widehat{f}_{j+\frac{1}{2}} - \widehat{f}_{j-\frac{1}{2}})}{x_{j+1} - x_{j-1}} \tag{3}$$

where $\widehat{f}_{j+\frac{1}{2}}, j = 1, 2, \cdots, n-1$, is called the numerical flux. It is a function of $2k$ variables

$$\widehat{f}_{j+\frac{1}{2}} = \widehat{f}(u^n_{j-k+1}, \cdots, u^n_{j+k}),$$

which satisfies the consistency condition $\widehat{f}(u, \cdots, u) = f(u)$. The most commonly used scheme in moving mesh systems, central difference, is also of the form (3), in which $\widehat{f}_{j+\frac{1}{2}} = \frac{1}{2}(f_{j+1}+f_j)$. For solutions with a rapid transition, the central difference is insufficient to resolve the solution in the transition area. Spurious oscillations often occur in those cases. Many useful schemes have been proposed to solve this problem. Among them, the upwinding schemes have been shown to be stable and to avoid spurious oscillations. ENO is one of the most commonly used upwinding schemes to gain high resolution and accuracy with a uniform mesh. It can be of high order and some of them can be used for non-uniform mesh without any modification. The most efficient implementation of ENO methods has been investigated by Shu and Osher in [22]. To be concise, we omit the detailed discussion of Shu and Osher on how to construct high order ENO schemes and give only the explicit form of the second and third order ENO schemes following their algorithm ENO-Roe, because to our knowledge, second order or third order is state of the art in moving mesh computation and higher order differencing in moving mesh has proven to be unstable by numerical experiments. In our experience, the first-order Roe scheme [20] can give a good result in many cases and it is the basis for other higher order methods. Thus, we first give the Roe scheme.

First order Roe scheme

The numerical flux of the Roe scheme is given by

$$\widehat{f}^{(1)}_{j+\frac{1}{2}} = \frac{1}{2}(f_j + f_{j+1} - |a_{j+\frac{1}{2}}|(u_{j+1} - u_j)) \tag{4}$$

where $a_{j+\frac{1}{2}}$ is called the "Roe" speed, which is defined by

$$a_{j+\frac{1}{2}} = \begin{cases} \dfrac{f_{j+1} - f_j}{u_{j+1} - u_j}, & u_{j+1} \neq u_j \\ \dfrac{\partial f}{\partial u}|_{u_{j+\frac{1}{2}}}, & u_j = u_{j+1}. \end{cases} \tag{5}$$

It is easy to see that for the Roe scheme $\widehat{f}^{(1)}_{j+\frac{1}{2}} = f_{k_1}$, where k_1 is defined by

$$k_1 = \begin{cases} j, & a_{j+\frac{1}{2}} \geq 0 \\ j+1, & a_{j+\frac{1}{2}} < 0, \end{cases} \tag{6}$$

which is in agreement with the principle of upwinding. In the moving mesh system, there is another convection term $\dot{x}\frac{\partial u}{\partial x}$. When we want to improve the stability of systems solved by moving mesh methods, this term should also be discretized via upwinding. We give here the first order upwinding scheme,

$$\dot{x} \cdot u_x = \begin{cases} \dot{x}_i \dfrac{u_{i+1} - u_i}{x_{i+1} - x_i}, & \dot{x}_i \geq 0 \\ \dot{x}_i \dfrac{u_i - u_{i-1}}{x_i - x_{i-1}}, & \dot{x}_i < 0. \end{cases} \tag{7}$$

The upwinding discretization (7) has been shown to improve the stability of the moving mesh system[15]. However, for some problems it does not yield a very accurate solution.

Second order ENO scheme

The second order ENO scheme is based on the first order Roe scheme. By choosing the smaller of the first order divided differences of f at x_{k_1}, i.e.,

$$k_2 = \begin{cases} k_1 - 1, & \text{if } \left|\dfrac{f_{k_1+1} - f_{k_1}}{x_{k_1+1} - x_{k_1}}\right| \geq \left|\dfrac{f_{k_1} - f_{k_1-1}}{x_{k_1} - x_{k_1-1}}\right| \\ k_1, & \text{otherwise.} \end{cases}$$

the estimate of the flux $\widehat{f}_{j+\frac{1}{2}}$ is then

$$\widehat{f}^{(2)}_{j+\frac{1}{2}} = f_{k_1} + \left(\frac{f_{k_2+1} - f_{k_2}}{x_{k_2+1} - x_{k_2}}\right) \cdot \frac{1}{2}(2x_{j+\frac{1}{2}} - x_{k_1+\frac{1}{2}} - x_{k_1-\frac{1}{2}}). \tag{8}$$

Eq.(8) is derived from interpolation at x_{k_1} (see Shu and Osher[22]). It is second order on a quasi-uniform mesh (see [23]). In practice, we find when replacing the interval $\frac{1}{2}(2x_{j+\frac{1}{2}} - x_{k_1+\frac{1}{2}} - x_{k_1-\frac{1}{2}})$ with

$$0.5(x_{j+1} - x_j), \quad \text{if } k_1 = j; \tag{9.1}$$

or

$$0.5(x_j - x_{j+1}), \quad \text{if } k_1 = j+1; \tag{9.2}$$

the method is still second order and can be solved more efficiently when combined with the moving mesh.

Third order ENO scheme

The idea of the third order ENO scheme is to compute the third order approximation to f at $x_{j+\frac{1}{2}}$ in a non-oscillatory way by choosing the smaller of the second order divided differences at x_{k_2}, i.e. to choose

$$k_3 = \begin{cases} k_2 - 1, & \text{if } |f[x_{k_2+2}, x_{k_2+1}, x_{k_2}]| \geq |f[x_{k_2+1}, x_{k_2}, x_{k_2-1}]| \\ k_2, & \text{otherwise.} \end{cases}$$

Then the numerical flux is

$$\begin{aligned} f^{(3)}_{j+\frac{1}{2}} = \; & f^{(2)}_{j+\frac{1}{2}} + c[(x_{j+\frac{1}{2}} - x_{k_2+\frac{1}{2}})(x_{j+\frac{1}{2}} - x_{k_2+\frac{3}{2}}) \\ & +(x_{j+\frac{1}{2}} - x_{k_2-\frac{1}{2}})(x_{j-\frac{1}{2}} - x_{k_2+\frac{3}{2}}) + (x_{j-\frac{1}{2}} - x_{k_2-\frac{1}{2}})(x_{j+\frac{1}{2}} - x_{k_2+\frac{1}{2}})], \end{aligned} \tag{10}$$

where

$$c = \frac{1}{3}\frac{f[x_{k_3+2}, x_{k_3+1}] - f[x_{k_3+1}, x_{k_3}]}{x_{k_3+2} - x_{k_3}}.$$

Eq. (10) is also derived from an interpolation formula and is given by Shu and Osher[22]. From Eq. (3) and Eq. (10), it is easily seen that third order ENO schemes are functions of four or five variables. Similarly, we can also localize the equation (10) and restrict the interval to $[x_{j-1}, x_{j+2}]$. Eq. (8) and (10) appear complicated, however j is between $[k_i - 1, k_i + 1]$ and some terms can be eliminated. Therefore the computational cost is relatively small.

Other high order upwinding schemes

There are many other kinds of upwinding schemes. Among them, the flux-limited scheme due to Van Leer[14] and local third order PHM (piecewise hyperbolic method) scheme due to A. Marquina[16] can be used in the moving mesh system with slight modification. The second order flux-limited scheme has been used in the new NAG library software for first-order partial differential equations (see [18]). The method is to construct the numerical flux as follows

$$f_{j+\frac{1}{2}} = \widehat{f}_{j+\frac{1}{2}}(u_L, u_R) = \frac{1}{2}(f(u_L) + f(u_R)). \tag{11}$$

Denote $h_i = x_i - x_{i-1}$. Then we have

$$u_L = u_j + \frac{h_{i+1}}{2}\frac{\partial u}{\partial x}\Big|_{x=x_j} = u_j + \frac{h_{i+1}}{2}\Big(\frac{u_j - u_{j-1}}{h_j}\Big) \cdot B(r_i)$$

where $B(r_i)$ is a flux limiter. It is defined by

$$B(r_i) = \frac{r_i + |r_i|}{1 + |r_i|}.$$

and

$$r_i = \frac{u^c_{i+\frac{1}{2}} - u_i}{u^u_{i+\frac{1}{2}} - u_i} = \frac{(u_{i+1} - u_i)/h_{i+1}}{(u_i - u_{i-1})/h_i}$$

where $u^c_{i+1} = \frac{1}{2}(u_i + u_{i+1})$ is the central difference term, and $u^u_{i+1} = u_i + \frac{h_{i+1}}{2}\frac{(u_{i+1}-u_i)}{h_i}$ is the left upwinding term. Similarly,

$$u_R = u_{i+1} - \frac{h_{i+1}}{2}\frac{(u_{i+2} - u_{i+1})}{h_{i+2}} B(\frac{1}{r_{i+1}}).$$

The construction of a local PHM scheme was introduced by A. Marquina[16] for scalar conservative laws. The idea of the PHM scheme is to construct the primitive function of flux via a piecewise hyperbolic equation. Because during our implementation of the moving mesh, we put as many nodes as possible in the transition area, we can simplify the algorithm PHM-REF of Marquina and consider only the nontransition cells. Define $d_j = f(u(x_j))$ and $d_{j+\frac{1}{2}} = \frac{d_{j+1}-d_j}{x_{j+1}-x_j}$ for all j. We then assign the central derivative d_j to be the harmonic mean of $d_{j-\frac{1}{2}}$ and $d_{j+\frac{1}{2}}$, which is

$$d_j = \frac{2d_{j+\frac{1}{2}}d_{j-\frac{1}{2}}}{d_{j-\frac{1}{2}} + d_{j+\frac{1}{2}}},$$

and α to be

$$\alpha = \begin{cases} 2\left(\sqrt{\frac{d_j}{d_{j-\frac{1}{2}}}} - 1\right), & \text{if } |d_{j-\frac{1}{2}}| \le |d_{j+\frac{1}{2}}| \\ 2\left(1 - \sqrt{\frac{d_j}{d_{j+\frac{1}{2}}}}\right), & \text{otherwise} \end{cases}.$$

Then the numerical flux can be taken to be

$$\widehat{f}_{j+\frac{1}{2}} = f_j + d_j \cdot h_j \cdot \eta(\alpha), \quad \text{if } a_{j+\frac{1}{2}} \ge 0; \tag{12.1}$$

and

$$\widehat{f}_{j+\frac{1}{2}} = f_{j+1} - d_j \cdot h_j \cdot \eta(-\alpha), \quad \text{if } a_{j+\frac{1}{2}} < 0, \tag{12.2}$$

where $\eta(x)$ is a differentiable function chosen to ensure the scheme is of order three for a quasi-uniform mesh. This scheme is of second order for a nonuniform mesh. For transition cells $[x_j, x_{j+1}]$, we use the Roe scheme.

Remark: We give here only the general formula. In practice, some special cases like $u_{j+1} = u_j$ can be easily settled separately.

3 The self-adaptive moving mesh strategy

As discussed by Furzeland et al.[5], in the classical Lagrange method for fluid-flow problems, the movement of the nodes is attached to a physically motivated, specific flow quantity. For example, for a problem like (1), it makes sense to attach the movement of the nodes to the convection term $\partial f(u)/\partial x$, i.e., to choose $\dot{x} = df(u)/du$ so as to obtain a parabolic equation of diffusion form. The rationale behind this choice is that parabolic problems without large first order terms usually possess smoother solutions and thus may be easier to solve numerically. However, the numerical realization of the prescription $\dot{x} = df(u)/du$ involves its own difficulties, such as mesh point crossings. In order to obtain a robust moving mesh method which could solve a wide variety of problems, we follow many authors in adopting the equidistribution principle mesh moving strategy with arclength monitor function. We give a brief discussion on how to regularize the mesh moving equation to get a robust and efficient mesh movement and physical solution.

The original idea of the equidistribution principle (EP) is to equidistribute a given mesh function in space, placing more nodes where the spatial error is large so as to gain high accuracy overall with a small number of nodes. It is natural to think to incorporate some error measurement into the monitor function. However, in the numerical computation of the moving mesh, high order derivatives cause much numerical noise and instability. Numerical experiments show that at least for finite difference methods, the mesh generated from the error measurement monitor is not very good (see Blom and Verwer[2]). Some further work needs to be done.

Regularization is very important for the mesh moving equation (see [15] for detailed information). It has been shown by numerical experiment that without regularization, the Newton iteration for implicit methods often fails to converge and the mesh appears very skew. For explicit methods, without regularization, the computation often exits due to too small time step. The regularization can also be seen as smoothing and diffusing (see [15]), which can generate smoother grids and more accurate solutions.

Given a monitor function $M(u, x)$, the equidistribution principle is

$$\int_{x_i}^{x_{i+1}} M(x,u)dx = \int_{x_{i-1}}^{x_i} M(x,u)dx = Const.$$

Discretizing it using the centered integral leads to

$$M(u,x)_{x_{i+\frac{1}{2}}}(x_{x+1} - x_i) = M(x,u)_{x_{i-\frac{1}{2}}}(x_i - x_{i-1}). \tag{13}$$

We use the arclength monitor here,

$$M(u,x)|_{x_{i+\frac{1}{2}}} = \sqrt{1+\Big(\frac{u_{i+1}-u_i}{x_{i+1}-x_i}\Big)^2}.$$

Also, the trapezoidal rule can be used in the discretization of the integral. After coupling Eq.(13) to the semi-discretized PDE and regularization, we obtain the moving mesh equation

$$\tau A\dot{x} = Mx + b, \tag{14}$$

where τ is a temporal regularization parameter, A can be viewed as a smoothing operator, and $Mx+b$ is the result of Eq. (13). A can be taken to be several forms besides the identity, each one corresponding to a discretized MMPDE (see [11]).

We now discuss the spatial smoothing technique. First, it should be noted that Equation (13) is not a strict constraint which should be satisfied during computation of the physical solution. It is only a strategy which pushes the nodes to the locations where we want them to be. The discretization of $M(x,u)$ does not affect the accuracy of the physical solution in an analytical sense. But it does affect the mesh distribution. From numerical experiments we know that the mesh quality affects directly the errors due to the spatial discretization, convergence of the Newton iteration for implicit methods, and time step size for explicit methods. Based on equation (13), we know there are two ways in which smoothing can be applied. One is to smooth the mesh x itself, the other is to smooth the monitor. They are equivalent in the sense of Eq.(13). In practice, we often smooth the monitor function. There are two smoothing approaches. One is global smoothing, i.e., smoothing the monitor function over all mesh nodes as

$$\widehat{M}_i = \sum_{j=1}^{N} \Big(\frac{k}{k+1}\Big)^{|i-j|} M_j. \tag{15}$$

This approach can restrict the adjacent intervals of mesh nodes to satisfy

$$\frac{k}{k+1} < \max\left(\frac{\Delta x_i}{\Delta x_{i+1}}, \frac{\Delta x_{i-1}}{\Delta x_i}\right) < \frac{k+1}{k};$$

for $k>0$. Using Eq.(13), the smoothing (15) can be implemented using the node concentration as

$$\tilde{n}_i = n_i - k(k+1)(n_{i+1} - 2n_i + n_{i-1}), \quad 1 \le i \le N;$$

where

$$n_i = (\Delta x_i)^{-1}, \quad \Delta x_i = x_{i+1} - x_i.$$

This method, called the Dorfi and Drury method, was first proposed in [3], analyzed by Verwer et al.[24] and reconsidered by Huang and Russell[12]. In our experience, Eq. (15) results in a

quasi-uniform mesh and can yield better results and efficiency if enough nodes are available (see [24] for the needed number of nodes); otherwise, the computation can fail. The other strategy is a local smoothing method, which is implemented by Huang et al.[11]. This method restricts the summation of Eq. (15) to its adjacent nodes, i.e.,

$$\widehat{M}_i = \sum_{j=i-p}^{i+p} \Big(\frac{k}{k+1}\Big)^{|i-j|} M_j, \tag{16}$$

where p is often chosen to be 1 or 2. This strategy does not restrict the mesh too much and can also smooth the nodes in the transition corners. However, it often puts too many nodes in the wave fronts or flat areas and is sensitive to the time scale τ. Generally, Eq. (16) is used when the number of nodes is small, and the Dorfi and Drury method is used when there are more nodes. Similarly we can impose the smoothing methods on the mesh nodes themselves, which yields

$$\Delta\widehat{x}_i = \sum_{j=i-p}^{i+p} \Big(\frac{k}{k+1}\Big)^{|i-j|} \Delta x_j.$$

This also can have good results.

All the methods above result in an implicit ODE, with stiffness depending on the time scale τ. When we use implicit methods, a nonlinear system must be solved at each time step. This often costs much time. Thus, the ODE form of the mesh moving equation is useful. The simplest ODE form is to choose $A = I$ in Eq.(14). However, we find that the result is not as good as that of other choices. Since there is no Jacobian evaluation in the explicit method, we can adopt the approach introduced by Hyman and Larrouturou[13], which considered the velocity of intervals

$$\frac{\Delta\dot{x}_{i+\frac{1}{2}}}{\Delta x_{i+\frac{1}{2}}} = \frac{\beta}{\tau}\frac{\overline{m} - m_{i+\frac{1}{2}}}{\overline{m} + m_{i+\frac{1}{2}}}, \tag{17}$$

where $m_{i+\frac{1}{2}}$ is the value of the mesh function at $x_{i+\frac{1}{2}} = \frac{1}{2}(x_i + x_{i+1})$, $\overline{m}$ is the average value of m, and β determines the relaxation time with respect to the time-scale τ. Eq.(17) limits the relative mesh velocity to $\pm\beta/\tau$. The relation between the mesh function and monitor function can be written in discretized form as

$$m(x_i, u_i) = M_{i+\frac{1}{2}}(x_{i+1} - x_i) = M_{i+\frac{1}{2}}\Delta x_{i+\frac{1}{2}}.$$

Eq. (17) has explicit physical meaning. When the value of the local mesh function is less than the average, the interval will increase. Otherwise, it will decrease. Because

$$\sum_{0}^{n} \Delta\dot{x}_{i+\frac{1}{2}} = \dot{x}_n - \dot{x}_0 = 0, \tag{18}$$

equation (17) requires normalization. We can append a small normalized constant to each equation of (17) as

$$\Delta\dot{x}_{i+\frac{1}{2}} = \frac{\beta}{\tau}\frac{\overline{m} - m_{i+\frac{1}{2}}}{\overline{m} + m_{i+\frac{1}{2}}}\Delta x_{i+\frac{1}{2}} + \varepsilon, \tag{19}$$

where

$$\varepsilon = \sum(\frac{\beta}{\tau}\frac{\overline{m} - m_{i+\frac{1}{2}}}{\overline{m} + m_{i+\frac{1}{2}}}\Delta x_{i+\frac{1}{2}})/N,$$

to ensure that Eq.(18) is satisfied.

Similarly, explicit methods corresponding to the Dorfi and Drury method can be easily derived. Using the node concentration, Eq. (17) can be written as

$$\frac{\dot{n}_i}{n_i} = \frac{\beta}{\tau}\frac{\overline{m} - m_{i+\frac{1}{2}}}{\overline{m} + m_{i+\frac{1}{2}}},$$

where $n_i = 1/\Delta x_{i+\frac{1}{2}}$, and $m_{i+\frac{1}{2}}$ in this case is

$$m_{i+\frac{1}{2}} = \frac{n_i}{M_{i+\frac{1}{2}}}.$$

We can see that it is just the inverse of the mesh function. If we choose $m_{i+\frac{1}{2}}$ as the mesh function, we can see this is exactly Eq.(17) before we apply smoothing to n_i.

Because the time step for the explicit methods is subject to the CFL restriction, we should control the minimal spatial length during the mesh distribution. One way is to add a penalty term to pull mesh points back when the mesh interval is less than the minimum. Winkler et al.[25] proposed a method to add the following penalty term to the mesh equation

$$\widetilde{\dot{x}_i} = \dot{x}_i + \left(\frac{\Delta x_{\min}}{\Delta x_{i+\frac{1}{2}}}\right)^4 - \left(\frac{\Delta x_{\min}}{\Delta x_{i-\frac{1}{2}}}\right)^4 \tag{20}$$

to achieve the minimum spacing. This method has a drawback that the value of $\dot{x}_i$ often dominates the value of the penalty term and sometimes we cannot actually achieve $\Delta\dot{x}_{i+\frac{1}{2}} > 0$. Another drawback is that $\Delta\dot{x}_{i+\frac{1}{2}}$ in Eq. (20) is also affected by $\Delta x_{i-\frac{1}{2}}$ and $\Delta x_{i+\frac{3}{2}}$, which we do not want to include. A modification of Eq. (20) acts on $\Delta\dot{x}_{i+\frac{1}{2}}$ instead of $\dot{x}_i$, which results in

$$\widetilde{\Delta\dot{x}_{i+\frac{1}{2}}} = \Delta\dot{x}_{i+\frac{1}{2}} + \mathrm{abs}(\Delta\dot{x}_{i+\frac{1}{2}})\left(\frac{\Delta x_{\min}}{\Delta x_{i+\frac{1}{2}}}\right)^4. \tag{21}$$

This ensures that no matter what the value of $\Delta\dot{x}_{i+\frac{1}{2}}$ is, the mesh interval will enlarge as soon as it is less than $\Delta x_{\min}$. The power 4 is chosen so that it has a force strong enough to react quickly once the spacings are less than the minimum spacing. Another problem is that

when $\mathrm{abs}(\Delta\dot{x}_{i+\frac{1}{2}})$ is very small, the force to enlarge the mesh interval will be very small. One method to cope with this problem is to replace $\mathrm{abs}(\Delta\dot{x}_{i+\frac{1}{2}})$ by

$$\max(\mathrm{abs}(\Delta\dot{x}_{i+\frac{1}{2}}), 1.0). \tag{22}$$

Numerical experiments show that Eq. (22) can offer better control on the minimum spacing than Eq. (21).

We now consider the CFL condition for the moving mesh system. We consider only the convection term in Eq.(2). Then we observe that a CFL time step criterion for Eq. (2) would be

$$\frac{|\dot{x} + c_u|\Delta t}{\Delta x} \leq 1,$$

which shows that the time step can be large even if Δx is very small in the area of a shock or wave front, because in that area, $|\dot{x} + c_u|$ is usually extremely small. From this we also can see a potential advantage of moving mesh methods over the adaptive methods. The allowable time step becomes very small because of small grid spacing near the shock for static adaptive gridding.

4 Numerical experiments

Here we consider the comparison of moving mesh methods with central difference and moving mesh methods with upwinding schemes. Throughout, we shall use the arclength monitor. For moving mesh equations of DAE form (e.g. Eq. (14) or the Dorfi and Drury method), we solve the systems using the double precision version of the stiff DAE solver DASSL[19] with RTOL=ATOL=10^{-6}, in which the time integration methods are the backward differentiation formula(BDF), where the approximate Jacobian is computed by DASSL internally using finite differences. For moving mesh equations of ODE form like Eq. (17), we solve the system using the DDRIVE package of ODE solver codes by Kahaner and Sutherland with error tolerance EPS=10^{-6}, in which the time integration methods are the Adams mutistep methods[6]. Unless specified otherwise, we select the stiffness parameter $k = 2$ and $\tau = 0.001$ for the Dorfi and Drury method; for other methods, we choose $\tau = 0.01$. All the computations are done on a Sun SPARC-5.

Example I: The first example we consider is the famous Burgers' equation

$$\partial u/\partial t = -\partial f(u)/\partial x + \varepsilon \partial^2 u/\partial x^2, \quad 0 < x < 1, \quad t > 0,$$

where

$$f(u) = u^2/2, \quad \varepsilon = 10^{-4}.$$

with piecewise continuous initial condition

$$u(x,0) = \begin{cases} 0.2, & x \le 0.1 \\ 8x - 0.6, & 0.1 \le x \le 0.2 \\ 1.0, & 0.2 \le x \le 0.5 \\ -10x + 6, & 0.5 \le x \le 0.6 \\ 0.0; & 0.6 \le x \le 1.0 \end{cases}$$

and homogeneous Dirichlet conditions at $x = 0$ and $x = 1$. This problem has served as a test example for moving mesh methods by Verwer et al.[24]. The solution is a wave that steepens and moves to the right until a layer is formed at the end point $x = 1$. This takes place for $t \approx 0.2$. Then the solution slowly decays to zero, near $x = 1$, because of the zero boundary condition. We solved this problem using the Dorfi and Drury method. First, with $N = 20$ nodes, the method generates spurious oscillations in Fig. 1(a) and then the Newton iteration fails to converge around $t = 0.9433$. When we use the 2-nd order ENO scheme instead of central difference, the spurious oscillations disappear. The result is shown in Fig. 1(b). Next,

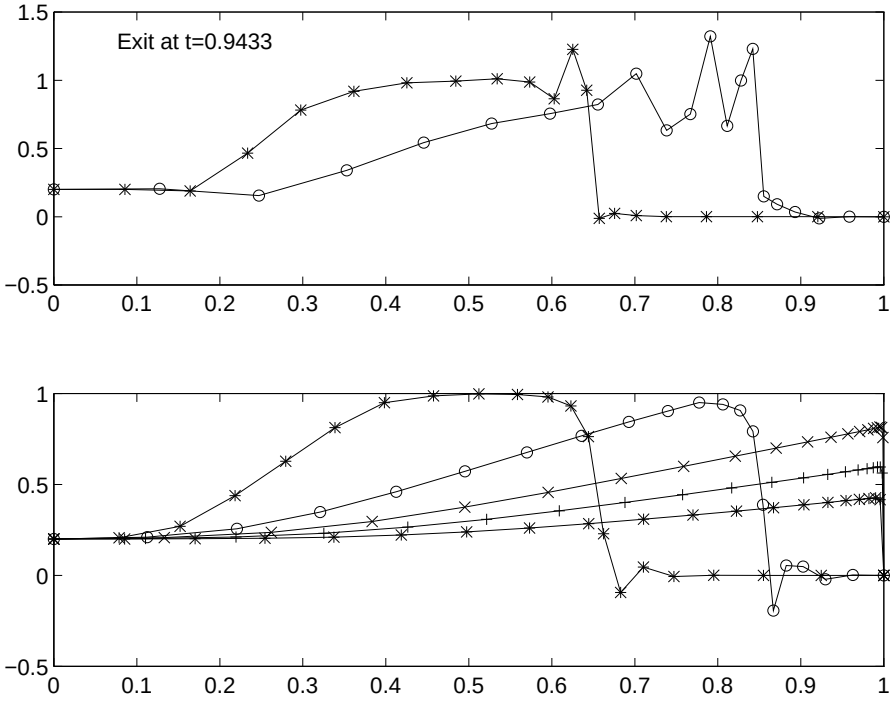

Figure 1: Result of the Dorfi and Drury method for Burgers' equation with 20 nodes and (a)(top) central difference, (b)(bottom) 2-nd order ENO scheme (9).

we use 40 nodes. There is still some oscillation for the central difference (the result is shown

in Fig. 2). However, the result for PHM (12) is very good as shown in Fig. 3. We also

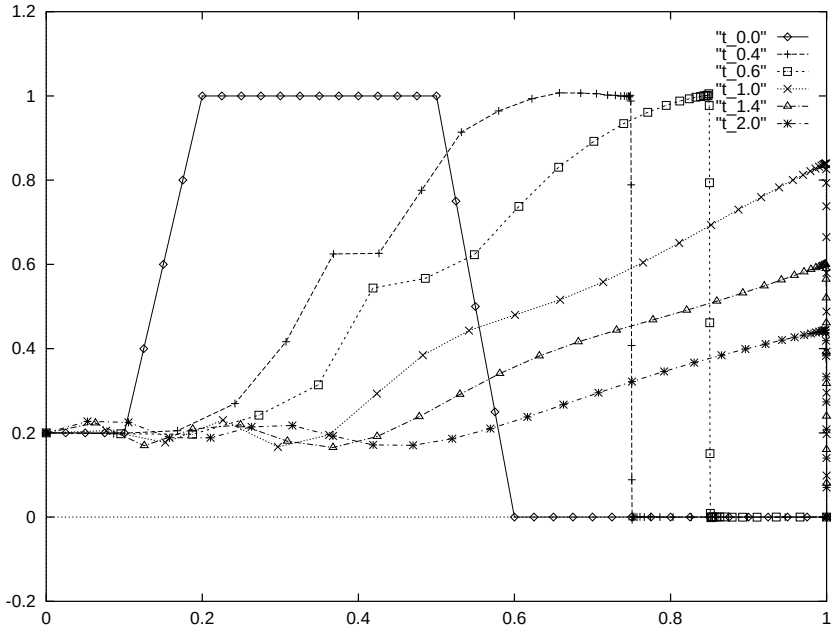

Figure 2: Result of the Dorfi and Drury method for Burgers' equation with 40 nodes and central difference.

have tested this problem using other schemes and find scheme (8) is prohibitively slow (it takes 4573 time steps and 6361 Jacobian evaluations to reach $t = 0.6$.); scheme (9) works well before the wave front reaches the left side but takes a much longer time to adjust itself when the shock hits the wall (it takes 203 time steps and 45 Jacobian evaluations before $t = 0.6$ but takes 1386 time steps and 1650 Jacobian evaluations to reach $t = 1.0$.); scheme (11) works faster than scheme (8) and (9) but yields a poor solution (even worse than central difference). The solution results for schemes (8) and (9) are very similar to PHM scheme (12). The computational efficiency (with 40 nodes) can be seen from the following table, where CD denotes the central difference.

Scheme	NSTP	NJAC	NRES	CPU
CD	327	77	650	6.43
PHM (12)	368	109	761	10.38
Eq. (9)	1409	2931	1653	82.85
Eq. (11)	556	132	1181	14.87

Example II: For convection-diffusion problems with convex flux (convection term), the moving mesh with central difference works well when the number of nodes is great enough or sufficient smoothing is used. To test the moving mesh methods on more difficult problems, we consider

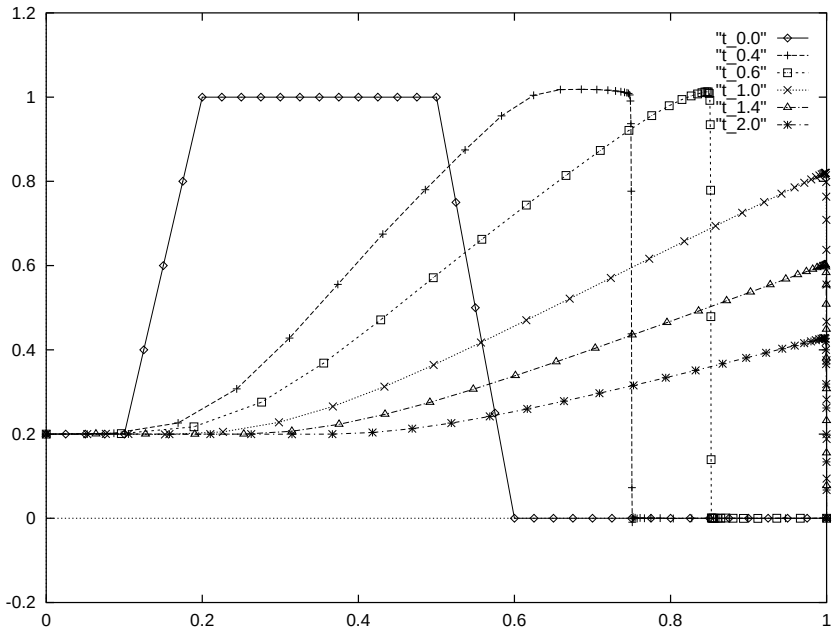

Figure 3: Result of the Dorfi and Drury method for Burgers' equation with 40 nodes and PHM (12).

the following Riemann problem with nonconvex flux.

$$u_t + (f(u))_x = \varepsilon \partial u^2/\partial^2 x, \quad -1 < x < 1, \quad t > 0, \ \varepsilon = 10^{-4},$$

and

$$f(u) = \frac{1}{4}(u^2 - 1)(u^2 - 4);$$

with the following initial data:

$$u(x, 0) = -3 \text{ if } x \leq 0, \text{ and } u = 3 \text{ if } x > 0. \tag{23}$$

This problem has been chosen as a test example for ENO schemes by many authors([22], [16]). For initial condition (23), instead of the expected sonic rarefaction fan that appears for convex fluxes, a nonconvex sonic stationary shock at $x = 0$ develops. Our scheme solves this problem well. In Fig. (4)–(7), we compare the numerical resolution of (23) solved with central difference, 2-nd order ENO scheme (9), 2-nd order flux limited scheme (11) and PHM scheme (12). We observe that for the central difference, there is a spurious oscillation. However, the results for the upwinding schemes are good. The PHM scheme is the best. The advantages of the PHM scheme can also be seen from the following table (with 40 nodes).

scheme	NSTP	NJAC	NRES	CPU
CD	7148	1224	12678	144.283
PHM	216	31	409	5.7833

Example III The last problem we consider is a reaction-diffusion problem. The example we choose is Fisher's equation with large parameter

$$\begin{cases} \frac{\partial y}{\partial t} = y_{xx} + \alpha y(1-y), \quad \alpha = 10^4 \\ y = \left(\frac{1}{1 + e^{\sqrt{\frac{\alpha}{6}} x - \frac{5}{6}\alpha t}} \right)^2 \end{cases} \tag{24}$$

where wave velocity $c = \sqrt{\frac{25\alpha}{6}}$. The initial condition and left boundary condition are determined by the exact solution. The right boundary condition is assigned to zero. The physical solution is a wave with steep front propagating from left to right. Because of its rapidly varying initial condition and unstable property at the bottom of the wave front (i.e. at the corner of $x = 0$), almost all the moving mesh methods fail to solve this problem. The wave front moves fast ahead of the true solution and the error becomes bigger and bigger. This is due somewhat to the instability of the resulting moving mesh system. After using the first order Roe scheme for the additional convection term, we can see from Fig. (8) and (9) that the result is relatively good because in some sense, the upwinding scheme can improve the stability of the solution. We solve this problem using MMPDE(6) (see [11]) and choose time scale $\tau = 1$. However, from Figure (9) we can see that the grid is not equidistributed because of the big τ. Actually, no moving mesh method can work well for this problem as pointed out in [15].

5 Conclusion

Several upwinding schemes have been described for the convection term of the moving mesh. Some strategies and techniques are also discussed. They are applied to different types of problems. The numerical results are very encouraging and clearly show that in many cases the moving mesh with upwinding scheme can track the shock or the wave front more accurately than the commonly used central difference. Of course, a wide range of experiments should be done. However, from the examples we have given, we can see that combining the moving mesh with an upwinding scheme can make those methods which failed because of spurious oscillations work well now, can track the rarefaction as well as the shock for nonconvex flux in reaction-diffusion problems and can improve the stability of the moving mesh system. According to the numerical experiments, we find the local PHM scheme can be solved efficiently by implicit or explicit methods compared with other schemes. We will investigate this method and generalize to multiple equations in later work.

References

[1] U. Ascher and L. Petzold, *Stability of computational methods for constrained dynamics systems*, SIAM J. Sci. Comput. 14 (1993), 95-120.

[2] J. G. Blom and J. G. Verwer, *On the use of arc length and curvature in a moving-grid method which is based on the method of lines*, Rept. NM-N8402, CWI Amsterdam, 1989.

[3] E. A. Dorfi and L. O'c. Drury, *Simple adaptive grids for 1-D initial value problems*, J. Comput. Phys. 69 (1987), 175-195.

[4] P.R. Eiseman, *Adaptive Grid Generation*, Computer Methods in Applied Mechanics and Engineering 64 (1987), 321-376.

[5] R. M. Furzeland, J. G. Verwer and P. A. Zegeling, *A numerical study of three moving grid methods for one-dimensional partial differential equations which are based on the method of lines*, J. Comput. Phys. 89 (1990), 349-388.

[6] C. W. Gear, Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, 1971.

[7] R. J. Gelinas, S. K. Doss and K. Miller, *The moving finite-element method: Applications to general partial differential equations with multiple large gradients*, J. Comput. Phys. 50 (1981), 202-268.

[8] A. Harten and J. M. Hyman, *A self-adjusting grid for the computation of weak solutions of hyperbolic conservation laws*, J. Comput. Phys. 50 (1983), 235-269.

[9] A. Harten and S. Osher, *Uniformly high order accurate nonoscillatory schemes I*, SIAM J. Numer. Anal., 24(1987), 279-309

[10] A. Harten, *High resolution schemes for hyperbolic conservation laws*, J. Comput. Phys. 49(1983), 357-393.

[11] W. Z. Huang, Y. Ren and R. D. Russell, *Moving mesh methods based on moving mesh partial differential equations*, J. Comput. Phys. 113(1994), 279-290.

[12] W. Huang and R.D. Russell, *Analysis of moving mesh partial differential equations with spatial smoothing*, Simon Fraser University Mathematics and Statistics Research Report 93–17 (Burnaby B.C. V5A 1S6, Canada, 1993).

[13] J. M. Hyman and B. Larrouturou, *Dynamic rezone methods for partial differential equations in one space dimension*, Appl. Numer. Math. 5 (1989), 435-450.

[14] B. Van Leer, *Towards the ultimate conservative difference scheme II, monotonicity and conservation combined in a second order scheme,* J. Comput. Phys. 14(1974), 361-470.

[15] S. Li, L. Petzold and Y. Ren, *Stability of moving mesh systems of partial differential equations*, preprint, 1996.

[16] A. Marquina, *Local piecewise hyperbolic resolution of numerical fluxes for nonlinear scalar conservation laws*, SIAM J. Sci. Comput. 15 (1994), 892-915.

[17] E.S. Oran and J.P.Boris, *Numerical simulation of reaction flows*, Elsevier, New York, 1987.

[18] S.V. Pennington and M. Berzins, *New NAG library software for first order partial differential equations,* ACM Trans. on Math. Software 20(1994), 63-100.

[19] L. R. Petzold, *A description of DASSL: A differential/algebraic system solver*, in Scientific Computing, eds. R.S. Stepleman et al., North- Holland, Amsterdam (1983), 65-68.

[20] P.L.Roe, *Approximate Riemman solvers, parameter vectors, and difference schemes*, J. Comput. Phys., 43(1981), 357-372

[21] K. Salari and S. Steinberg, *Flux-corrected transport in a moving grid*, J. of Comput. Phys., 111(1994), 24-32.

[22] C.W. Shu and S.J. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes II*, J. Comput. Phys., 83(1989), 32-78

[23] J.F. Thompson, *A survey of dynamically-adaptive grids in the numerical solution of partial differential equations*, Applied Numer. Math., 1(1985), 3-27.

[24] J. G. Verwer, J. G. Blom, R. M. Furzeland, and P. A. Zegeling, *A moving-grid method for one-dimensional PDEs based on the method of lines*, in Adaptive Methods for Partial Differential Equations, J. E. Flaherty, P.J. Paslow, M. S. Shephard, and J. D. Vasilakis, eds., Society for Industrial and Applied Mathematics, Philadelphia, PA, 1989.

[25] k. Winkler, , D. Mihalas and M. Norman, *Adaptive-grid methods with asymmetric time-filtering.*, preprint, 1984.

[26] R. Zarnowski *A moving grid method for conservation laws based on nonlinear interpolation and shock tracking*, preprint, 1993.

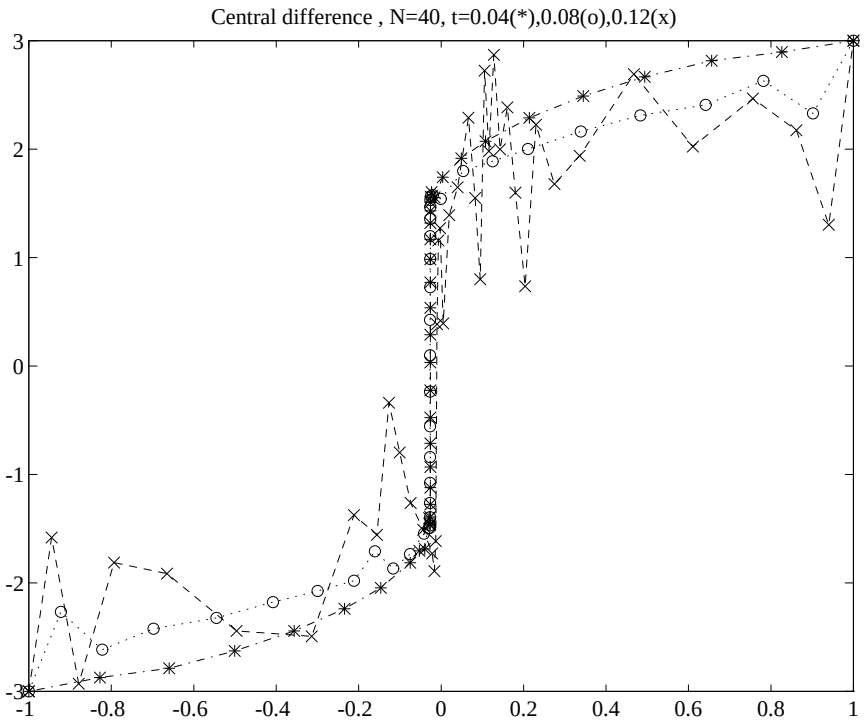

Figure 4: Result of moving mesh with central difference for Prob II.

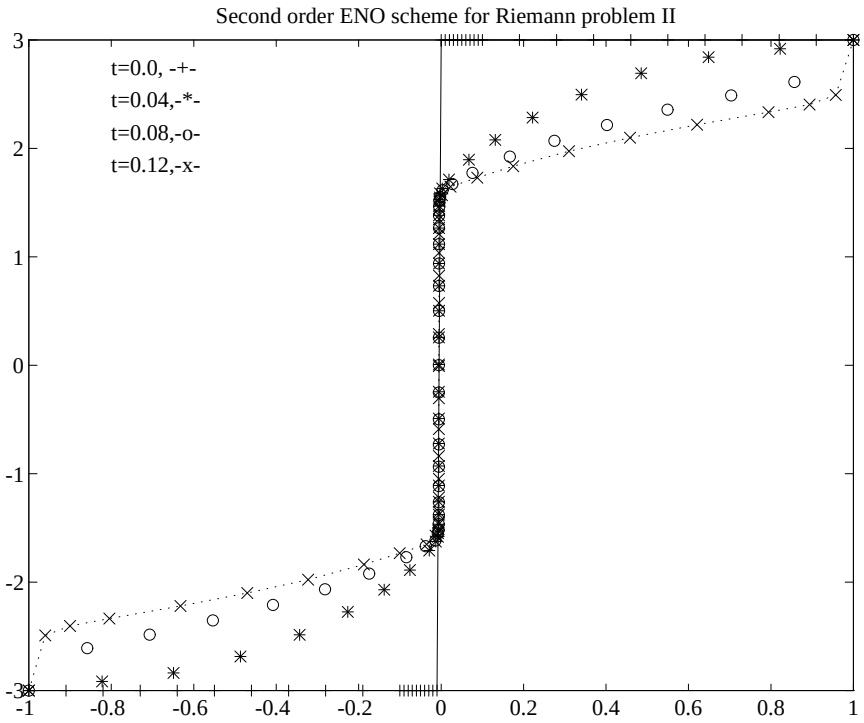

Figure 5: Result of moving mesh with 2-nd order ENO scheme (9) for Prob II.

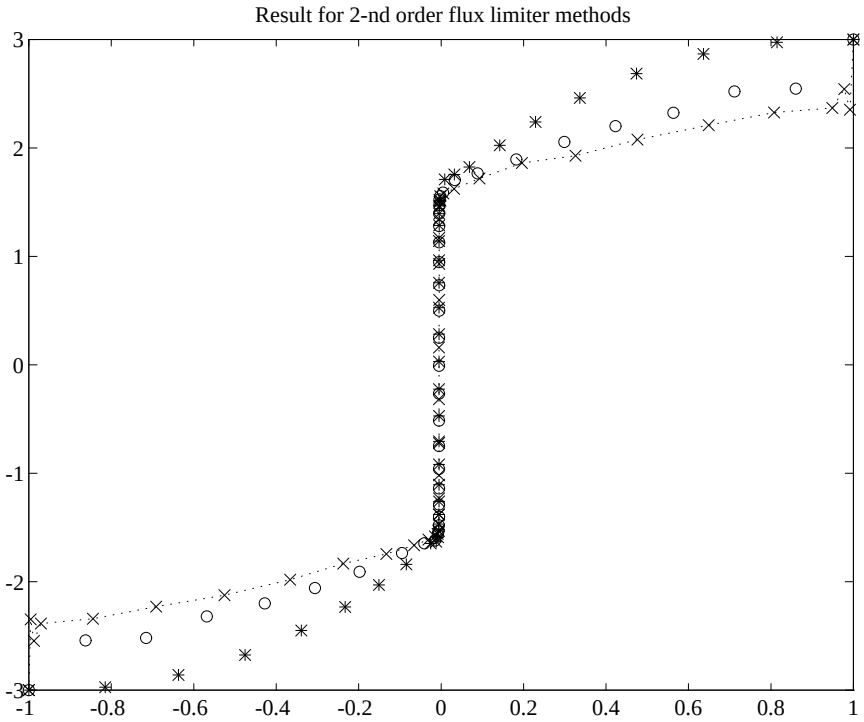

Figure 6: Result of moving mesh with flux limiter scheme (11) for Prob II.

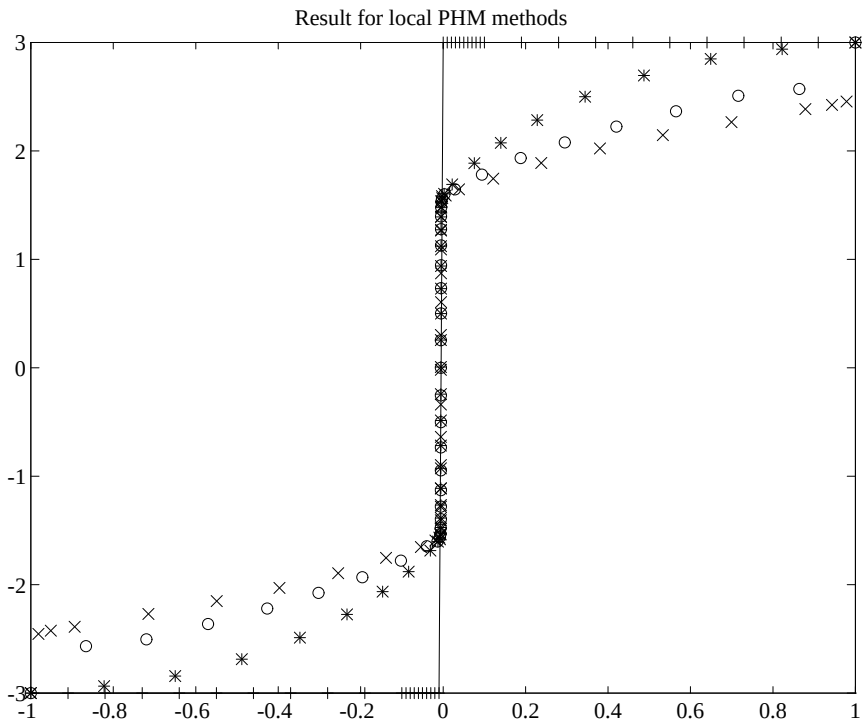

Figure 7: Result of moving mesh with local PHM scheme (12) for Prob II.

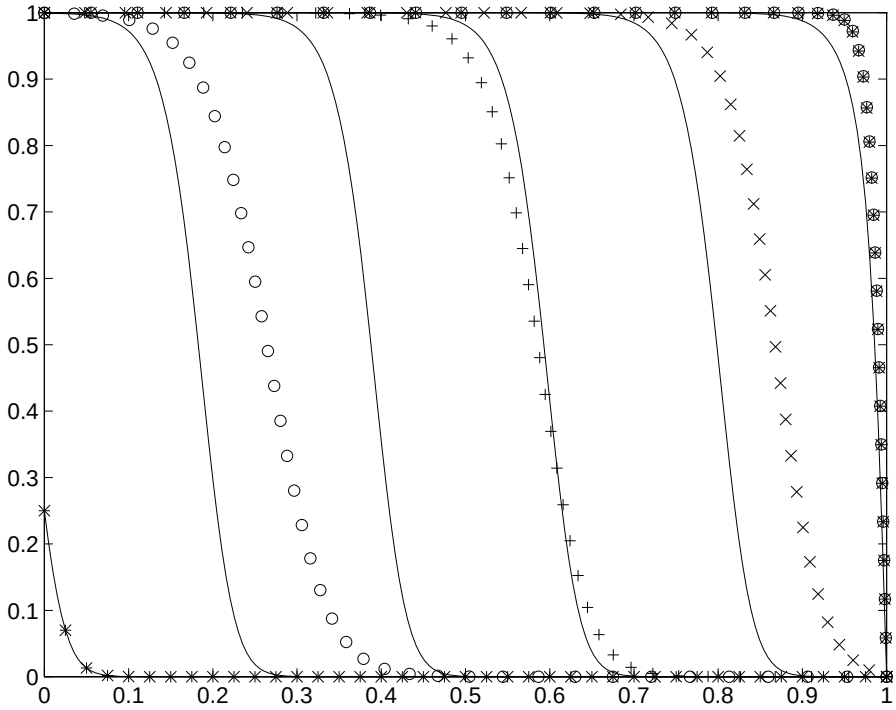

Figure 8: Result of moving mesh with central difference for Fisher equation.

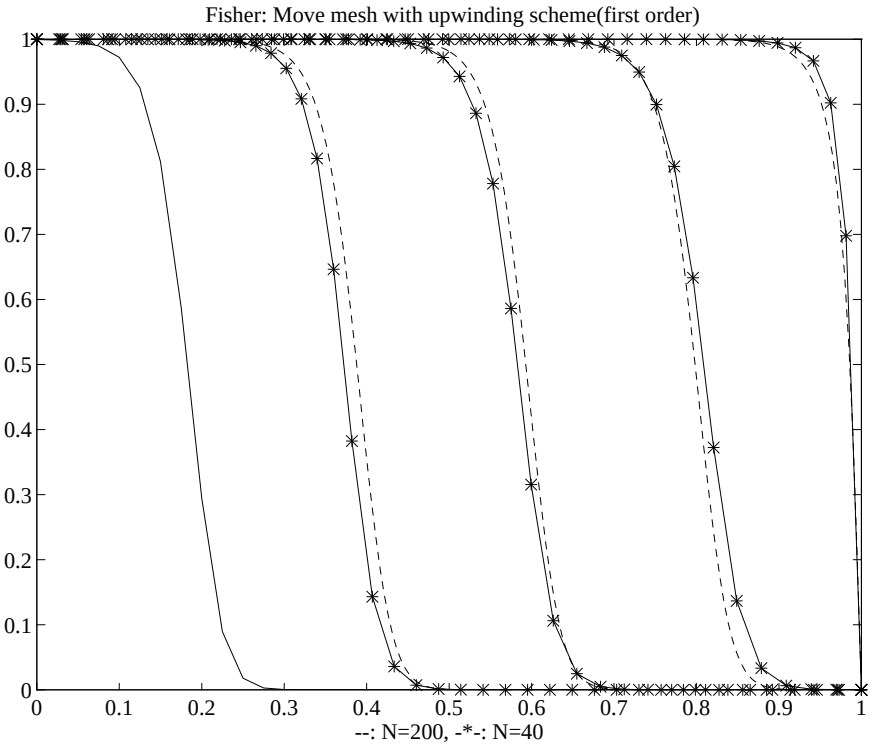

Figure 9: Result of moving mesh with Roe scheme for Fisher equation.