Unconditionally stable high-order time integration for moving mesh finite difference solution of linear convection-diffusion equations

Weizhang Huang *

Abstract

This paper is concerned with moving mesh finite difference solution of partial differential equations. It is known that mesh movement introduces an extra convection term and its numerical treatment has a significant impact on the stability of numerical schemes. Moreover, many implicit second and higher order schemes, such as the Crank-Nicolson scheme, will lose their unconditional stability. A strategy is presented for developing temporally high order, unconditionally stable finite difference schemes for solving linear convection-diffusion equations using moving meshes. Numerical results are given to demonstrate the theoretical findings.

AMS 2010 Mathematics Subject Classification. 65M06, 65M12, 65L20
Key Words. Moving mesh, high order method, time integration, stability, finite difference.
Abbreviated title. Unconditionally stable high order time integration on moving meshes.

1 Introduction

In the last two decades, moving mesh methods have attracted considerable attention from scientists and engineers and been successfully applied to a variety of problems; e.g., see [8, 21] and references therein. A common feature of those methods is that they employ a time varying mesh to follow the motion of the physical boundary and/or certain solution properties. Generally speaking, partial differential equations (PDEs) can be discretized directly on a moving mesh using finite element and finite volume methods. When a finite difference method is desired, it is common practice to transform PDEs to a reference domain and discretize them on a fixed mesh defined thereon. The interested reader is referred to [21] for more detailed discussion on the discretization of PDEs on moving meshes. It should be emphasized that, despite what discretization method is used, the movement of the mesh inevitably introduces an extra convection term into PDEs, whose numerical treatment often has a significant impact on the stability and convergence of moving mesh methods. The term also makes the analysis of stability and convergence much more difficult for moving mesh methods even when linear PDEs are involved.

While a number of convergence results have been developed for the numerical solution of two-point boundary value problems using equidistributing meshes (a type of moving meshes) (e.g., see [4, 5, 6, 9, 16, 20, 23, 25, 27, 28]), theoretical studies of moving mesh methods for time dependent PDEs are far from complete. For example, for linear convection-diffusion problems Dupont [11],
Bank and Santos [3], Dupont and Liu [12], and Liu et al. [24] establish symmetric error estimates for various finite element methods (FEM), including semi-discrete FEM, semi-discrete mixed FEM, FEM-implicit Euler, and space-time FEM. These results essentially require the conditions (or their discrete counterparts)

\[
|\dot{x}(x, t) - b(x, t)| \leq C_1, \quad \forall x \in \Omega, \ t > 0 \\
|\nabla \cdot \dot{x}(x, t)| \leq C_2, \quad \forall x \in \Omega, \ t > 0
\]

where \(C_1\) and \(C_2\) are positive constants, \(\Omega\) is the physical domain, \(\dot{x}\) is the mesh speed, and \(b\) is the coefficient of the convection term (see (5) below). Condition (1) requires that the mesh move not too fast with reference to \(b\). To see the geometric meaning of (2), we view the moving mesh as the image of a computational mesh under a coordinate transformation \(x = x(\xi, t): \Omega_c \rightarrow \Omega\), where \(\Omega_c\) is the computational domain. Denote the Jacobian of the coordinate transformation by \(J\), i.e., \(J = \det(\partial x/\partial \xi)\). It is known [21] that

\[
\frac{1}{J} \dot{J} = \nabla \cdot \dot{x}.
\]

Then (2) reduces to

\[
|\dot{J}| \leq C_2 |J|.
\]

Since \(J\) represents the volume of mesh elements, (3), or equivalently (2), requires that the relative change of the volume of mesh elements be in a constant order.

Formaggia and Nobile [14, 15] and Boffi and Gastaldi [7] study the relation between stability and satisfaction of the geometric conservation law (GCL) [29] in the Arbitrary-Langragian-Eulerian formulation [19] with finite element spatial discretization. They show that satisfying GCL is neither a necessary nor a sufficient condition for the stability of a scheme although it often helps improve accuracy and enhance stability. In particular, Boffi and Gastaldi show that the FEM-implicit Euler scheme is only conditionally stable. Formaggia and Nobile [14, 15] present several modifications of the implicit Euler, Crank-Nicolson, and BDF2 (two step backward differentiation formula [10, 17]) schemes, which can be made to satisfy GCL. They show that the FEM-modified implicit Euler scheme is unconditionally stable whereas the FEM-modified Crank-Nicolson and FEM-BDF(2) schemes are only conditionally stable with the maximum allowable time step depending on \(\nabla \cdot \dot{x}\).

Ferreira [13] studies a finite difference (FDM)-implicit Euler scheme applied to the nonconservative form of transformed linear convection-diffusion equations and shows that the scheme is stable and convergent under discrete analogs of conditions (1) and (3). Mackenzie and Mekwi [26] consider an FDM-\(\theta\) scheme for the conservative form of transformed linear convection-diffusion equations. They show that the FDM-implicit Euler scheme is unconditionally stable in an energy norm but the Crank-Nicolson scheme is only conditionally stable. They also show that the FDM-\(\theta\) scheme can be made to be unconditionally stable and of second order in time (and in space) when \(\theta\) is properly chosen depending on the mesh. Although this variable \(\theta\) scheme is unconventional, there are no other moving mesh methods that are known to be of second or higher order in time and unconditionally stable. More recently, Huang and Schaeffer [22] show that several FDM-\(\theta\) moving mesh schemes for one dimensional convection-diffusion problems are stable in \(L^\infty\) norm under the conventional CFL condition and a mesh speed related condition which can be roughly expressed as

\[
|\dot{x} - b| \leq \frac{C_3}{h},
\]

(4)
where $C_3$ is a constant and $h$ is the maximum spacing of the mesh. Notice that this condition is weaker than (1).

It is well known that high-order methods are practically important in enhancing the computational accuracy and efficiency. It is also important in theory to know if unconditionally stable high-order methods exist in the moving mesh context. The objective of this paper is to develop unconditionally stable, high-order (in time) moving finite difference schemes for the solution of the initial-boundary value problem (IBVP) of a general linear convection-diffusion equation,

\[
\begin{cases}
\frac{\partial u}{\partial t} + \nabla \cdot (u \mathbf{b}) + cu = \nabla \cdot (a \nabla u) + f, & \forall \mathbf{x} \in \Omega, \ t \in (0, T] \\
u(\mathbf{x}, t) = g(\mathbf{x}, t), & \forall \mathbf{x} \in \partial \Omega \\
u(\mathbf{x}, 0) = u^0(\mathbf{x}), & \forall \mathbf{x} \in \Omega
\end{cases}
\] (5)

where $\Omega \subset \mathbb{R}^d$ ($d = 1, 2$) is the physical domain with probably moving or deforming boundary and $a = a(\mathbf{x}, t)$, $\mathbf{b} = \mathbf{b}(\mathbf{x}, t)$, $c = c(\mathbf{x}, t)$, $f = f(\mathbf{x}, t)$, $g = g(\mathbf{x}, t)$, and $u^0(\mathbf{x})$ are given, sufficiently smooth functions. For the posedness of IBVP, we assume that the coefficients satisfy

\[
0 < a(\mathbf{x}, t) \leq \overline{a} < \infty, \quad \forall \mathbf{x} \in \Omega, \ t \in (0, T]
\] (6)

\[
c(\mathbf{x}, t) + \frac{1}{2} \nabla \cdot \mathbf{b}(\mathbf{x}, t) \geq 0, \quad \forall \mathbf{x} \in \Omega, \ t \in (0, T]
\] (7)

where $a$ and $\overline{a}$ are two positive constants. We note that for stability analysis, we only need to consider the homogeneous system, i.e., the IBVP with $f \equiv 0$ and $g \equiv 0$. In this situation, it is easy to show that the solution satisfies the stability inequality

\[
\frac{d}{dt} \int_{\Omega} u^2 dx \leq 0.
\] (8)

The key to our development is the preservation of stability inequality (8). We use the method of lines approach and first discretize the PDE in space in such a way that the resulting system of ordinary differential equations (ODEs) satisfies a semi-discrete stability inequality. Then high-order schemes satisfying a fully discrete version of (8) are constructed for integrating the ODE system. The resulting schemes are stable in the energy norm corresponding to the fully discrete stability inequality.

An outline of this paper is given as follows. In §2, high-order unconditionally stable schemes for ODE systems satisfying a stability inequality are developed. Strategies for spatially discretizing IBVP (5) into such an ODE system in one and two dimensions are explored in §3 and §4. Numerical examples obtained with the developed schemes are presented in §5. Finally, §6 contains conclusions and comments.

## 2 High-order unconditionally stable schemes for nonautonomous ODE systems

In this section we present an approach of constructing unconditionally stable schemes for the initial value problem of a nonautonomous ODE system

\[
\begin{cases}
M(t) \frac{du}{dt} = A(t)u + f(t), \\
u(0) = u^0
\end{cases}
\] (9)
where \( M(t) \) and \( A(t) \) are \( l \times l \) matrices (for some positive integer \( l \)) and \( f(t) \) is a given vector-valued function. We assume that \( M(t) \) is symmetric and positive definite and \( A(t) + \sqrt{M(t)} \frac{d\sqrt{M}}{dt}(t) \) is negative semi-definite, i.e.,

\[
z^T \left( A(t) + \sqrt{M(t)} \frac{d\sqrt{M}}{dt}(t) \right) z \leq 0, \quad \forall z \in \mathbb{R}^l, \quad \forall t > 0 \tag{10}
\]

where \( \sqrt{M} \) denotes the square root of \( M \). As will be seen in §3 and §4, such systems arise from finite difference discretization of IBVP (5) on moving meshes, with \( M(t) \) and \( A(t) \) being the mass and stiffness matrices, respectively.

The idea of the approach is simple. Indeed, we rewrite (9) into

\[
\frac{d}{dt} \left( \sqrt{M} u \right) = (\sqrt{M})^{-1} \left( A + \sqrt{M} \frac{d\sqrt{M}}{dt} \right) u + (\sqrt{M})^{-1} f, \tag{11}
\]

which can further be written as

\[
\frac{dv}{dt} = B(t)v + (\sqrt{M(t)})^{-1} f(t), \tag{12}
\]

where

\[
v(t) = \sqrt{M(t)} u(t), \tag{13}
\]

\[
B(t) = (\sqrt{M})^{-1} \left( A + \sqrt{M} \frac{d\sqrt{M}}{dt} \right) (\sqrt{M})^{-1}. \tag{14}
\]

Then, unconditionally stable schemes can be obtained by applying conventional implicit schemes to (12). The initial condition for the new variable is

\[
v(0) = v^0 \equiv \sqrt{M(0)} u^0. \tag{15}
\]

Moreover, assumption (10) implies that \( B(t) \) is negative semi-definite for any \( t > 0 \).

In the following, we illustrate the idea with three schemes. To this end, we assume that a partition is given for \([0, T] \),

\[
t_0 = 0 < t_1 < \cdots < t_N = T.
\]

We denote \( \Delta t_n = t_{n+1} - t_n \).

We first apply the backward Euler discretization to (12). It gives

\[
\frac{v^{n+1} - v^n}{\Delta t_n} = B(t_{n+1}) v^{n+1} + (\sqrt{M(t_{n+1})})^{-1} f(t_{n+1}). \tag{16}
\]

Since \( B \) is negative semi-definite, for the homogeneous situation (with \( f = 0 \)) we can readily show that (16) satisfies

\[
(v^{n+1})^T v^{n+1} \leq (v^n)^T v^n, \tag{17}
\]

which can be written in terms of the original variable \( u \) as

\[
(u^{n+1})^T \sqrt{M(t_{n+1})} u^{n+1} \leq (u^n)^T \sqrt{M(t_n)} u^n. \tag{18}
\]
Inequality (18) is a discrete analog to (8). It implies that the backward Euler scheme (16), which is of first order, is unconditionally stable.

A second order unconditionally stable scheme can be obtained by applying the midpoint discretization to (12). Indeed, we have

$$\frac{v^{n+1} - v^n}{\Delta t_n} = B \left( \frac{t_n + t_{n+1}}{2} \right) + \left( \sqrt{M} \left( \frac{t_n + t_{n+1}}{2} \right) \right)^{-1} f \left( \frac{t_n + t_{n+1}}{2} \right).$$

(19)

It is easy to show that the solution of (19) (with \(f = 0\)) satisfies stability inequality (17).

Higher order unconditionally stable schemes can be obtained by applying collocation schemes to (12). To explain this, we consider the \(m\) Gauss-Legendre points in \((0,1)\), \(\rho_1, \ldots, \rho_m\), and denote the collocation points by

$$t_{n,j} = t_n + \rho_j \Delta t_n, \quad j = 1, \ldots, m.$$  

(20)

Applying the \(m\)-point collocation scheme (e.g., see [1, 2]) to (12), we get, for \(n = 0, \ldots, N - 1\),

$$\frac{d \vartheta_h(t_{n,j})}{dt}(t_{n,j}) = B(t_{n,j}) \vartheta_h(t_{n,j}) + (\sqrt{M}(t_{n,j}))^{-1} f(t_{n,j}), \quad j = 1, \ldots, m$$

(21)

where \(\vartheta_h(t)\), an approximation to the exact solution \(\vartheta(t)\), is continuous on \([0,T]\) and a polynomial of degree \(m\) on each subinterval \([t_n, t_{n+1}]\).

We can rewrite (21) into a more explicit form. Let \(\tilde{\rho}_1, \ldots, \tilde{\rho}_m\) be the \((m-1)\) Gauss-Legendre points in \((0,1)\) and denote \(\tilde{\rho}_0 = 0\) and \(\tilde{\rho}_m = 1\). Define

$$\tilde{t}_{n,j} = t_n + \tilde{\rho}_j \Delta t_n, \quad j = 0, \ldots, m.$$  

(22)

(We may also take \(\tilde{\rho}_0, \ldots, \tilde{\rho}_m\) as the \((m+1)\) Gauss-Lobatto Legendre points on \([0,1]\).) Then, \(\vartheta_h\) can be expressed as

$$\vartheta_h(t) = \sum_{k=0}^{m} \tilde{v}_{n,k} \tilde{\ell}_k \left( \frac{t - t_n}{\Delta t_n} \right), \quad \forall t \in [t_n, t_{n+1}], \quad n = 0, \ldots, N - 1$$

(23)

where \(\tilde{v}_{n,k} \approx \vartheta(\tilde{t}_{n,k}) \quad (k = 0, \ldots, m)\) are the unknown variables and \(\tilde{\ell}_k\)’s are the Lagrange polynomials associated with nodes \(\tilde{\rho}_0, \ldots, \tilde{\rho}_m\), i.e.,

$$\tilde{\ell}_k(\rho) = \prod_{i=0 \atop i \neq k}^m \frac{\rho - \tilde{\rho}_i}{\tilde{\rho}_k - \tilde{\rho}_i}.$$  

Inserting (23) into (21), we have

$$\frac{1}{\Delta t_n} \sum_{k=0}^{m} \tilde{v}_{n,k} \tilde{\ell}_k(\rho_j) = B(t_{n,j}) \sum_{k=0}^{m} \tilde{v}_{n,k} \tilde{\ell}_k(\rho_j) + (\sqrt{M}(t_{n,j}))^{-1} f(t_{n,j}), \quad j = 1, \ldots, m$$

(24)

This equation, together with the initial condition

$$\tilde{v}_{n,0} = \tilde{v}_{n-1,m},$$  

(25)

forms a system of ODEs for unknowns \(\tilde{v}_{n,k}, \quad k = 0, \ldots, m.\)
Once \( \tilde{v}_{n,m} \) has been obtained, we can compute the original variable \( u \) at \( t = t_{n+1} \) by

\[
  u^{n+1} = (\sqrt{M})^{-1}(t_{n+1}) \tilde{v}_{n,m}, \quad n = 0, ..., N - 1.
\]  (26)

Moreover, it is known (e.g., see [1, 2]) that the convergence order of the \( m \)-point collocation scheme (21) is \( 2^m \). (It is easy to verify that scheme (21) reduces to the midpoint scheme (19) when \( m = 1 \).)

In the following we show that scheme (21) (with \( f = 0 \)) satisfies (17). To this end, we denote \( v^{n+1} = \tilde{v}_{n,m} \) for \( n = 0, ..., N - 1 \). Obviously, \( v^{n+1} = v_h(t_{n+1}) \). Let \( \omega_1, \cdots, \omega_m \) be the weights for the Gaussian quadrature rule associated with the points \( \rho_1, \cdots, \rho_m \), viz.,

\[
  \int_0^1 f(\rho) d\rho \approx m \sum_{j=1}^{m} \omega_j f(\rho_j).
\]

It is known that \( \omega_j \)'s are positive and the quadrature rule is exact for all polynomials of degree up to \((2m - 1)\). Multiplying (21) with \( \Delta t_n \omega_j v_h(t_{n,j})^T \) and summing the result over \( j \), we have

\[
  \Delta t_n \sum_{j=1}^{m} \omega_j v_h(t_{n,j})^T \frac{dv_h}{dt}(t_{n,j}) = \Delta t_n \sum_{j=1}^{m} \omega_j v_h(t_{n,j})^T B(t_{n,j}) v_h(t_{n,j}).
\]

The negative semi-definiteness of \( B \) implies that the right-hand side is nonpositive. Moreover, \( v_h^{T} \frac{dv_h}{dt} \) is a polynomial of degree at most \((2m - 1)\) and the sum on the left-hand side is equal to the integral of \( v_h^{T} \frac{dv_h}{dt} \) over \([t_n, t_{n+1}]\). Thus, we have

\[
  \int_{t_n}^{t_{n+1}} v_h^{T} \frac{dv_h}{dt} dt \leq 0.
\]

This, combined with the fact that

\[
  \int_{t_n}^{t_{n+1}} v_h^{T} \frac{dv_h}{dt} dt = \frac{1}{2} \int_{t_n}^{t_{n+1}} \frac{d}{dt} (v_h^{T} v_h) dt = \frac{1}{2} (v_h^{n+1} v_h^{n+1} - (v_h^n)^T v_h^n),
\]

leads to (17).

It is remarked that scheme (21) is expressed in the new variable \( v \). It can be reformulated in terms of the original variable \( u \) using the relation (13). Moreover, scheme (21) involves \( \sqrt{M} \) and its inverse whose computation can be expensive in general. In our current situation with finite difference discretization for PDEs, however, the mass matrix \( M \) is diagonal and its square root and the inverse can be computed easily (see the next two sections) and thus the involvement of \( \sqrt{M} \) and its inverse has very mild effects on the computational efficiency. The argument also goes for finite volume methods or finite element methods with lumped mass matrix.

### 3 1D convection-diffusion equations

In this and next sections, we study the finite difference discretization of IBVP (5). Our goal is to obtain schemes that satisfy the property (10) and thus unconditionally stable integration schemes can be developed. We consider one dimensional convection-diffusion equations in this section and two dimensional ones in the next section.
In one dimension, IBVP \((5)\) becomes

\[
\begin{aligned}
&u_t + (bu)_x + cu = (au_x)_x + f, \quad \forall x \in (x_l(t), x_r(t)), \quad 0 < t \leq T \\
u(x_l(t), t) = g(x_l(t), t), \quad u(x_r(t), t) = g(x_r(t), t), \\
u(x, t) = u^0(x).
\end{aligned}
\]

(27)

Notice that this setting permits moving domains.

For spatial discretization, we assume that a moving mesh is given at time instants \(t_0, \cdots, t_N\), i.e.,

\[x^n_0 = x_l(t_n) < x^n_1 < \cdots < x^n_{J_{\text{max}}} = x_r(t_n), \quad n = 0, \cdots, N.\]

The mesh points are considered to vary linearly on each time subinterval, viz.,

\[x_j(t) = \frac{t_{n+1} - t}{\Delta t_n} x^n_j + \frac{t - t_n}{\Delta t_n} x^n_{j+1}, \quad \forall t \in [t_n, t_{n+1}], \quad j = 0, \cdots, J_{\text{max}}.\]

Notice that the mesh speed on \([t_n, t_{n+1}]\) is constant, viz.,

\[\dot{x}_j(t) = \frac{x^n_{j+1} - x^n_j}{\Delta t_n}, \quad \forall t \in [t_n, t_{n+1}], \quad j = 0, \cdots, J_{\text{max}}.\]

We view this moving mesh as the image of a uniform computational mesh under a coordinate transformation. Denote the coordinate transformation by \(x = x(\xi, t): [0, J_{\text{max}}] \rightarrow [x_l(t), x_r(t)]\). Then the moving mesh can be expressed as

\[x_j(t) = x(\xi_j, t), \quad \xi_j = j, \quad j = 0, \cdots, J_{\text{max}}.\]

To discretize the PDE \((27)\) on the moving mesh, we first transform it from the physical domain into the computational domain using the coordinate transformation. It is easy to show (e.g., see [21]) that the transformed PDE can be written either in a conservative form as

\[x_\xi \dot{u} + x_\xi u + \frac{\partial}{\partial \xi}((b - \dot{x})u) + x_\xi cu = \frac{\partial}{\partial \xi} \left( \frac{a}{x_\xi} \frac{\partial u}{\partial \xi} \right) + x_\xi f,\]

(28)

or in a non-conservative form as

\[x_\xi \dot{u} - \dot{x} \frac{\partial u}{\partial \xi} + \frac{\partial}{\partial \xi} (bu) + x_\xi cu = \frac{\partial}{\partial \xi} \left( \frac{a}{x_\xi} \frac{\partial u}{\partial \xi} \right) + x_\xi f,\]

(29)

where \(\dot{x}\) (the mesh speed) and \(\dot{u}\) denote the time derivatives of \(x\) and \(u\) in the new variables \(t\) and \(\xi\), respectively.

We now consider a central finite difference scheme based on the conservative form \((28)\). It reads as

\[
\frac{h_{j+1} + h_j}{2} \dot{u}_j + \frac{h_{j+1} + h_j}{2} \dot{u}_j + (b_{j+\frac{1}{2}} - \dot{x}_{j+\frac{1}{2}}) \frac{u_{j+1} + u_j}{2} - \frac{u_{j-1} + u_{j-\frac{1}{2}}}{2} \frac{u_j - u_{j-1}}{h_j} + \frac{h_{j+1} + h_j}{2} f_j,
\]

(30)

where \(u_j = u(x_j(t), t) \approx u(x_j(t), t), \dot{x}_{j+\frac{1}{2}} = (\dot{x}_j + \dot{x}_{j+1})/2\), and \(h_j = h_j(t) = x_j(t) - x_{j-1}(t)\). The boundary condition has the discrete form as

\[u_0(t) = g(x_l(t), t), \quad u_{J_{\text{max}}}(t) = g(x_r(t), t).
\]

(31)
Notice that a special treatment has been used in (30) for the convection term, viz.,

\[ \frac{\partial}{\partial \xi}((b - \dot{x})u) \bigg|_{x_j} \approx (b - \dot{x})_{j+\frac{1}{2}} u_{j+\frac{1}{2}} - (b - \dot{x})_{j-\frac{1}{2}} u_{j-\frac{1}{2}} \]

\[ \approx (b - \dot{x})_{j+\frac{1}{2}} \frac{u_j + u_{j+1}}{2} - (b - \dot{x})_{j-\frac{1}{2}} \frac{u_j + u_{j-1}}{2}. \]

In words, the flux \((b - \dot{x})u\) is approximated at half mesh points \(x_{j+\frac{1}{2}}\). This treatment is crucial for scheme (30) to satisfy condition (10).

Scheme (30) can be cast into the matrix form (9), with the mass and stiffness matrices given by, for \(j = 1, ..., J_{\text{max}} - 1\),

\[ (Mu)_j = \frac{h_{j+1} + h_j}{2} u_j, \]

\[ (Au)_j = a_{j+\frac{1}{2}} \frac{u_{j+1} - u_j}{h_{j+1}} - a_{j-\frac{1}{2}} \frac{u_j - u_{j-1}}{h_j} - \frac{h_{j+1} + h_j}{2} u_j - \frac{h_{j+1} + h_j}{2} c_j u_j - (b_{j+\frac{1}{2}} - \dot{x}_{j+\frac{1}{2}}) \frac{u_{j+1} + u_j}{2} + (b_{j-\frac{1}{2}} - \dot{x}_{j-\frac{1}{2}}) \frac{u_j + u_{j-1}}{2}, \]

where \(u = [u_0, ..., u_{J_{\text{max}}}]^T\). Notice that the mass matrix \(M\) is diagonal and its square root is

\[ \sqrt{M} = \text{diag} \left( \sqrt{\frac{h_{j+1} + h_j}{2}} \right). \]

Moreover,

\[ \frac{d}{dt} \sqrt{h_{j+1} + h_j} = \frac{\sqrt{2}}{4} \frac{(h_{j+1} + h_j)}{\sqrt{h_{j+1} + h_j}}. \]

**Theorem 3.1.** Assume that there holds

\[ c_j + \frac{1}{2} \frac{(b_{j+\frac{1}{2}} - b_{j-\frac{1}{2}})}{(h_{j+1} + h_j)/2} \geq 0, \quad j = 1, ..., J_{\text{max}} - 1. \]

Then, the finite difference scheme (30) satisfies the condition (10). As a consequence, the fully discrete scheme resulting from the application of the time integration (21) to (4) with \(M\) and \(A\) defined in (32) and (33) is unconditionally stable and of order \((2m)\) in time and order \(2\) in space.

**Proof.** Recall that we only need to consider the homogeneous situation with \(f \equiv 0\) and \(g \equiv 0\). From (31), (33), (34), and (35), we have

\[ u^T (A + \sqrt{M} \frac{d}{dt} \sqrt{M}) u \]

\[ = \sum_{j=1}^{J_{\text{max}}-1} u_j \left[ a_{j+\frac{1}{2}} \frac{u_{j+1} - u_j}{h_{j+1}} - a_{j-\frac{1}{2}} \frac{u_j - u_{j-1}}{h_j} \right] \]

\[ + \sum_{j=1}^{J_{\text{max}}-1} u_j \left[ - (b_{j+\frac{1}{2}} - \dot{x}_{j+\frac{1}{2}}) \frac{u_{j+1} + u_j}{2} + (b_{j-\frac{1}{2}} - \dot{x}_{j-\frac{1}{2}}) \frac{u_j + u_{j-1}}{2} \right] \]

\[ - \sum_{j=1}^{J_{\text{max}}-1} \frac{h_{j+1} + h_j}{2} u_j^2 - \sum_{j=1}^{J_{\text{max}}-1} \frac{h_{j+1} + h_j}{2} c_j u_j^2 + \sum_{j=1}^{J_{\text{max}}-1} \frac{h_{j+1} + h_j}{4} u_j^2. \]
For the first term on the right-hand side, using the boundary condition (31) and summation by parts we get
\[
\sum_{j=1}^{J_{\text{max}}-1} u_j \left[ a_{j+\frac{1}{2}} \frac{u_{j+1} - u_j}{h_{j+1}} - a_{j-\frac{1}{2}} \frac{u_j - u_{j-1}}{h_j} \right] = \sum_{j=2}^{J_{\text{max}}} u_{j-1} a_{j-\frac{1}{2}} \frac{u_j - u_{j-1}}{h_j} - \sum_{j=1}^{J_{\text{max}}-1} u_j a_{j-\frac{1}{2}} \frac{u_j - u_{j-1}}{h_j} = -\sum_{j=1}^{J_{\text{max}}} a_{j-\frac{1}{2}} \frac{(u_j - u_{j-1})^2}{h_j}.
\]

(38)

For the second term, we have
\[
\sum_{j=1}^{J_{\text{max}}-1} u_j \left[ -b_{j+\frac{1}{2}} \frac{u_{j+1} + u_j}{2} + b_{j-\frac{1}{2}} \frac{u_j + u_{j-1}}{2} \right] = -\sum_{j=2}^{J_{\text{max}}} u_{j-1} b_{j-\frac{1}{2}} \frac{u_j + u_{j-1}}{2} + \sum_{j=1}^{J_{\text{max}}-1} u_j b_{j-\frac{1}{2}} \frac{u_j + u_{j-1}}{2} = \sum_{j=1}^{J_{\text{max}}-1} b_{j-\frac{1}{2}} \frac{u_j^2 - u_{j-1}^2}{2} - u_{J_{\text{max}}-1} b_{J_{\text{max}}-\frac{1}{2}} \frac{u_{J_{\text{max}}} + u_{J_{\text{max}}-1}}{2} + u_1 b_{1-\frac{1}{2}} \frac{u_1 + u_0}{2} = \frac{1}{2} \sum_{j=1}^{J_{\text{max}}} b_{j-\frac{1}{2}} u_j^2 - \frac{1}{2} \sum_{j=0}^{J_{\text{max}}-1} b_{j+\frac{1}{2}} u_j^2 = \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} (b_{j+\frac{1}{2}} - b_{j-\frac{1}{2}}) u_j^2.
\]

(39)

Inserting (38) and (39) into (37), we get
\[
\mathbf{u}^T (A + \sqrt{M} \frac{d}{dt} \sqrt{M}) \mathbf{u} = -\sum_{j=1}^{J_{\text{max}}} a_{j-\frac{1}{2}} \frac{(u_j - u_{j-1})^2}{h_j} - \sum_{j=1}^{J_{\text{max}}-1} h_{j+1} + h_j \frac{u_j^2}{4} - \sum_{j=1}^{J_{\text{max}}-1} h_{j+1} + h_j \frac{u_j^2}{2} c_j u_j^2 - \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} (b_{j+\frac{1}{2}} - b_{j-\frac{1}{2}}) u_j^2 + \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} (\dot{x}_{j+\frac{1}{2}} - \dot{x}_{j-\frac{1}{2}}) u_j^2 = -\sum_{j=1}^{J_{\text{max}}} a_{j-\frac{1}{2}} \frac{(u_j - u_{j-1})^2}{h_j} - \sum_{j=1}^{J_{\text{max}}-1} h_{j+1} + h_j \frac{u_j^2}{2} \left[ c_j + \frac{1}{2} \frac{(b_{j+\frac{1}{2}} - b_{j-\frac{1}{2}})}{(h_{j+1} + h_j) / 2} \right].
\]

The assumption (36) implies that the right-hand side is nonpositive. Thus, scheme (30) satisfies the condition (10).
Remark 3.1. The assumption (36) is a discrete analog to the continuous condition (7). It is satisfied when \( b \) is constant and \( c \) is nonnegative. For the general situation with variable \( b \), it is reasonable to expect (36) to hold too provided that the continuous condition (7) holds and the mesh is sufficiently fine.

It is instructive to spell out one of the full discrete schemes. We consider the case with \( m = 1 \) which results from the application of the midpoint discretization to the \( v \) equation (12). Let

\[
m_j(t) = \sqrt{h_{j+1}(t) + h_j(t)}.
\]

Then, the new and old variables are related by \( v_j = m_j u_j \) and (30) reads as

\[
m_j \ddot{v}_j + \frac{\dot{h}_{j+1} + \dot{h}_j}{4} \frac{v_j}{m_j} + \frac{1}{2} \left( b \frac{j+1}{2} - \dot{x}_{j+1} \right) \left( \frac{v_{j+1}}{m_{j+1}} + \frac{v_j}{m_j} \right) - \frac{1}{2} \left( b \frac{j-1}{2} - \dot{x}_j \right) \left( \frac{v_j}{m_j} + \frac{v_{j-1}}{m_{j-1}} \right) + m_j c_j v_j
\]

\[
= \frac{a_{j+\frac{1}{2}}}{h_{j+1}} \left( \frac{v_{j+1}}{m_{j+1}} - \frac{v_j}{m_j} \right) - \frac{a_{j-\frac{1}{2}}}{h_j} \left( \frac{v_j}{m_j} - \frac{v_{j-1}}{m_{j-1}} \right) + m_j f_j. \tag{40}
\]

Applying the midpoint discretization to the above equation, we get

\[
m_j^{n+\frac{1}{2}} \frac{v_j^{n+1} - v_j^n}{\Delta t} + \frac{\dot{h}_{j+1} + \dot{h}_j}{8} \frac{v_j^{n+\frac{1}{2}}}{m_j^{n+\frac{1}{2}}} + \frac{1}{4} \left( b \frac{j+1}{2} - \dot{x}_{j+\frac{1}{2}} \right) \left( \frac{v_{j+1}^{n+\frac{1}{2}} + v_{j+\frac{1}{2}}^{n\frac{1}{2}}}{m_{j+1}^{n+\frac{1}{2}}} + \frac{v_j^{n+\frac{1}{2}}}{m_j^{n+\frac{1}{2}}} \right) - \frac{1}{4} \left( b \frac{j-1}{2} - \dot{x}_{j-\frac{1}{2}} \right) \left( \frac{v_j^{n+\frac{1}{2}} + v_{j-1}^{n+\frac{1}{2}}}{m_j^{n+\frac{1}{2}}} + \frac{v_{j-1}^{n+\frac{1}{2}}}{m_{j-1}^{n+\frac{1}{2}}} \right) + \frac{1}{2} \frac{m_j^{n+\frac{1}{2}} c_j^{n+\frac{1}{2}}}{h_{j+1}} (v_j^n + v_j^{n+1})
\]

\[
= \frac{a_{j+\frac{1}{2}}^{n+\frac{1}{2}}}{h_{j+1}^{n+\frac{1}{2}}} \left( \frac{v_{j+1}^{n+\frac{1}{2}} + v_{j+\frac{1}{2}}^{n\frac{1}{2}}}{m_{j+1}^{n+\frac{1}{2}}} - \frac{v_j^{n+\frac{1}{2}}}{m_j^{n+\frac{1}{2}}} \right) - \frac{a_{j-\frac{1}{2}}^{n-\frac{1}{2}}}{h_j^{n-\frac{1}{2}}} \left( \frac{v_j^{n+\frac{1}{2}} + v_{j-1}^{n+\frac{1}{2}}}{m_j^{n+\frac{1}{2}}} - \frac{v_{j-1}^{n+\frac{1}{2}}}{m_{j-1}^{n+\frac{1}{2}}} \right) + \frac{m_j^{n+\frac{1}{2}} f_j^{n+\frac{1}{2}}}{h_{j+1}^{n+\frac{1}{2}}}. \tag{41}
\]

This scheme is unconditionally stable and of second order in both time and space.

Next, we consider finite difference schemes based on the non-conservative form (29). Approximating the mesh movement related convection term using the half point fluxes, i.e.,

\[
\frac{\partial u}{\partial \xi} \bigg|_{x_j} \approx \frac{1}{2} \frac{\partial u}{\partial \xi} \bigg|_{x_{j+\frac{1}{2}}} + \frac{1}{2} \frac{\partial u}{\partial \xi} \bigg|_{x_{j-\frac{1}{2}}}
\]

\[
\approx \frac{1}{2} \dot{x}_{j+\frac{1}{2}} (u_{j+1} - u_j) + \frac{1}{2} \dot{x}_{j-\frac{1}{2}} (u_j - u_{j-1}),
\]

we have

\[
\frac{h_{j+1} + h_j}{2} \dot{u}_j - \frac{1}{2} \dot{x}_{j+\frac{1}{2}} (u_{j+1} - u_j) - \frac{1}{2} \dot{x}_{j-\frac{1}{2}} (u_j - u_{j-1}) + b_j \frac{u_{j+1} + u_j}{2} - b_j \frac{u_j + u_{j-1}}{2} + \frac{h_{j+1} + h_j}{2} c_j u_j
\]
Interestingly, it can be verified that this semi-discrete scheme is equivalent to scheme (30). Thus, (42) can be cast in the form (9) with property (10).

When a two-cell central finite difference approximation is used for the mesh movement related convection term, the finite difference scheme becomes

\[
\frac{h_{j+1} + h_j}{2} u_j - \frac{1}{2} \hat{x}_j (u_{j+1} - u_{j-1}) + b_j + \frac{u_{j+1} + u_j}{2} - b_j - \frac{u_j + u_{j-1}}{2} \frac{h_{j+1} + h_j}{2} c_j u_j = a_j \frac{u_{j+1} - u_j}{h_{j+1}} - a_j - \frac{u_j - u_{j-1}}{h_j} + \frac{h_{j+1} + h_j}{2} f_j. \tag{43}
\]

It can be shown that

\[
\mathbf{u}^T (A + \sqrt{M} \frac{d}{dt} \sqrt{M}) \mathbf{u} = - \sum_{j=1}^{J_{\text{max}}} \left[ a_j - \frac{1}{2} \frac{(\hat{x}_j - \hat{x}_{j-1})}{h_j} \right]^2 (u_j - u_{j-1})^2 - \sum_{j=1}^{J_{\text{max}}-1} \frac{h_{j+1} + h_j}{2} u_j^2 \left[ c_j + \frac{1}{2} \frac{(b_j + \frac{1}{2} - b_j - \frac{1}{2})}{(h_{j+1} + h_j)/2} \right].
\]

Thus, the right-hand side is nonpositive and therefore scheme (43) satisfies (10) if there hold the conditions (36) and

\[
(\hat{x}_j - \hat{x}_{j-1}) \leq 4 \frac{a_j}{h_j}, \quad j = 1, ..., J_{\text{max}} - 1. \tag{45}
\]

Note that both (36) and (45) can be satisfied when the mesh speed is bounded and the mesh is sufficiently fine.

\section{2D convection-diffusion equations}

In this section we study finite difference discretization of IBVP (5) in two dimensions. The procedure is similar to that in the previous section for one dimension but the derivation is more complicated for the current situation.

We assume that a curvilinear moving mesh, \{(x_{j,k}^n, y_{j,k}^n), j = 0, ..., J_{\text{max}}, k = 0, ..., K_{\text{max}}\}, is given for the physical domain \(\Omega\) at time instants \(t_0, ..., t_N\). As for the 1D case, we consider the mesh to vary linearly on each time interval, i.e.,

\[
x_{j,k}(t) = \frac{t_{n+1} - t}{\Delta t_n} x_{j,k}^n + \frac{t - t_n}{\Delta t_n} x_{j,k}^{n+1}, \quad y_{j,k}(t) = \frac{t_{n+1} - t}{\Delta t_n} y_{j,k}^n + \frac{t - t_n}{\Delta t_n} y_{j,k}^{n+1}, \quad \forall t \in [t_n, t_{n+1}].
\]

Note that the mesh speed \((\hat{x}_{j,k}, \hat{y}_{j,k})\) is constant on \([t_n, t_{n+1}]\). We view the the moving mesh as the image of a Cartesian computational mesh under an invertible coordinate transformation \(x = x(\xi, \eta, t), \quad y = y(\xi, \eta, t)\) : \([0, J_{\text{max}}] \times [0, K_{\text{max}}] \rightarrow \Omega\), i.e.,

\[
x_{j,k}(t) = x(\xi_j, \eta_k, t), \quad y_{j,k}(t) = y(\xi_j, \eta_k, t), \quad j = 0, ..., J_{\text{max}} \quad k = 0, ..., K_{\text{max}}
\]

where \(\xi_j = j, \quad j = 0, ..., J_{\text{max}}\) and \(\eta_k = k, \quad k = 0, ..., K_{\text{max}}\). Through the coordinate transformation, the PDE in (5) can be transformed (e.g., see [21]) into a conservative form

\[
J \frac{d}{dt} + \frac{\partial q_1}{\partial \xi} + \frac{\partial q_2}{\partial \eta} + cuJ = \frac{\partial p_1}{\partial \xi} + \frac{\partial p_2}{\partial \eta} + J f,
\]

(48)
where

\[
\begin{align*}
q_1 &= u \left[ (J\xi_x)(b_1 - \hat{x}) + (J\xi_y)(b_2 - \hat{y}) \right], \\
q_2 &= u \left[ (J\eta_x)(b_1 - \hat{x}) + (J\eta_y)(b_2 - \hat{y}) \right], \\
p_1 &= \frac{a}{2} \left[ (J\xi_x)^2 + (J\xi_y)^2 \right] \frac{\partial q_1}{\partial y} + \frac{a}{2} \left[ (J\xi_x)(J\eta_x) + (J\xi_y)(J\eta_y) \right] \frac{\partial q_1}{\partial x}, \\
p_2 &= \frac{a}{2} \left[ (J\xi_x)(J\eta_x) + (J\xi_y)(J\eta_y) \right] \frac{\partial p_1}{\partial y} + \frac{a}{2} \left[ (J\eta_x)^2 + (J\eta_y)^2 \right] \frac{\partial p_1}{\partial x}.
\end{align*}
\]

(49)

Here, \( b = (b_1, b_2)^T \), \( J \) is the Jacobian of the coordinate transformation, and \( \mathbf{q} = (q_1, q_2)^T \) and \( \mathbf{p} = (p_1, p_2)^T \) are the convection and diffusion fluxes, respectively. Moreover, we have the transformation identities

\[
\begin{align*}
J &= x_\xi y_\eta - x_\eta y_\xi = (J\xi_x)(J\eta_y) - (J\xi_y)(J\eta_x), \\
(J\xi_x) &= y_\eta, \quad (J\xi_y) = -x_\eta, \quad (J\eta_x) = -y_\xi, \quad (J\eta_y) = x_\xi.
\end{align*}
\]

(50)

We consider a central finite difference discretization for (48). The scheme reads as

\[
J_{j,k} \hat{u}_{j,k} + \hat{J}_{j,k} u_{j,k} + \left( q_{1,j+\frac{1}{2},k} - q_{1,j-\frac{1}{2},k} \right) + \left( p_{2,j,k+\frac{1}{2}} - p_{2,j,k-\frac{1}{2}} \right) = \frac{1}{2} \left( p_{1,j+\frac{1}{2},k-\frac{1}{2}} - p_{1,j-\frac{1}{2},k+\frac{1}{2}} \right) + \frac{1}{2} \left( p_{2,j+\frac{1}{2},k-\frac{1}{2}} - p_{2,j-\frac{1}{2},k+\frac{1}{2}} \right) + f_{j,k} J_{j,k},
\]

(51)

where the convection fluxes \( q_1 \) and \( q_2 \) are approximated at integer-half and half-integer points and the diffusion fluxes are approximated at half-half points; i.e.,

\[
\begin{align*}
q_{1,j-\frac{1}{2},k} &= \frac{u_{j,k} + u_{j-1,k}}{2} \left[ (J\xi_x)_{j-\frac{1}{2},k} (b_{1,j-\frac{1}{2},k} - \hat{x}_{j-\frac{1}{2},k}) \right. \\
&\left. + (J\xi_y)_{j-\frac{1}{2},k} (b_{2,j-\frac{1}{2},k} - \hat{y}_{j-\frac{1}{2},k}) \right], \\
q_{2,j,k-\frac{1}{2}} &= \frac{u_{j,k} + u_{j,k-1}}{2} \left[ (J\eta_x)_{j,k-\frac{1}{2}} (b_{1,j,k-\frac{1}{2}} - \hat{x}_{j,k-\frac{1}{2}}) \right. \\
&\left. + (J\eta_y)_{j,k-\frac{1}{2}} (b_{2,j,k-\frac{1}{2}} - \hat{y}_{j,k-\frac{1}{2}}) \right], \\
p_{1,j-\frac{1}{2},k-\frac{1}{2}} &= \frac{a_{j-\frac{1}{2},k-\frac{1}{2}}}{2J_{j-\frac{1}{2},k-\frac{1}{2}}} \left[ (J\xi_x)^2 + (J\xi_y)^2 \right]_{j-\frac{1}{2},k-\frac{1}{2}} \left( u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1} \right), \\
&\quad + \frac{a_{j-\frac{1}{2},k-\frac{1}{2}}}{2J_{j-\frac{1}{2},k-\frac{1}{2}}} \left[ (J\xi_x)(J\eta_x) + (J\xi_y)(J\eta_y) \right]_{j-\frac{1}{2},k-\frac{1}{2}} \times \left( u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1} \right), \\
p_{2,j,k-\frac{1}{2}} &= \frac{a_{j-\frac{1}{2},k-\frac{1}{2}}}{2J_{j-\frac{1}{2},k-\frac{1}{2}}} \left[ (J\xi_x)^2 + (J\xi_y)^2 \right]_{j-\frac{1}{2},k-\frac{1}{2}} \left( u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1} \right), \\
&\quad + \frac{a_{j-\frac{1}{2},k-\frac{1}{2}}}{2J_{j-\frac{1}{2},k-\frac{1}{2}}} \left[ (J\eta_x)^2 + (J\eta_y)^2 \right]_{j-\frac{1}{2},k-\frac{1}{2}} \left( u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1} \right).
\end{align*}
\]

(52)

(53)

(54)

(55)

Here, the transformation quantities (used in the approximations of the convection fluxes)

\[
(J\xi_x)_{j-\frac{1}{2},k}, \quad (J\xi_y)_{j-\frac{1}{2},k}, \quad (J\eta_x)_{j,k-\frac{1}{2}}, \quad (J\eta_y)_{j,k-\frac{1}{2}}, \quad \hat{x}_{j-\frac{1}{2},k}, \quad \hat{x}_{j,k-\frac{1}{2}}, \quad \hat{y}_{j-\frac{1}{2},k}, \quad \hat{y}_{j,k-\frac{1}{2}}.
\]

(56)
are to be defined so that condition (10) is satisfied while the other quantities (used in the approximations of the diffusion fluxes)

\[ (J_{\xi_x})_{j-\frac{1}{2},k-\frac{1}{2}}, \quad (J_{\xi_y})_{j-\frac{1}{2},k-\frac{1}{2}}, \quad (J_{\eta_x})_{j-\frac{1}{2},k-\frac{1}{2}}, \quad (J_{\eta_y})_{j-\frac{1}{2},k-\frac{1}{2}}, \quad J_{j-\frac{1}{2},k-\frac{1}{2}}, \quad (57) \]

are approximated using central finite differences based on relation (50). For example,

\[ (J_{\xi_x})_{j-\frac{1}{2},k-\frac{1}{2}} = (y_{\eta})_{j-\frac{1}{2},k-\frac{1}{2}} \approx \frac{1}{2} (y_{j,k} - y_{j,k-1} + y_{j-1,k} - y_{j-1,k-1}). \]

The discretization of the boundary condition is

\[ u_{j,k} = g(x_{j,k}(t), y_{j,k}(t), t), \quad \forall (j,k) \text{ with } j = 0, \quad j = J_{\text{max}}, \quad k = 0, \text{ or } k = K_{\text{max}}. \quad (58) \]

Let \( u = \{u_{j,k}\} \). The above scheme can then be cast in the form (9) with

\[ (M u)_{j,k} = J_{j,k} \dot{u}_{j,k}, \quad (A u)_{j,k} = -J_{j,k} u_{j,k} \left( q_{1,j+\frac{1}{2},k} - q_{1,j-\frac{1}{2},k} \right) - \left( q_{2,j,k+\frac{1}{2}} - q_{2,j,k-\frac{1}{2}} \right) - c_{j,k} u_{j,k} I_{j,k} \]

\[ + \frac{1}{2} \left( f_{1,j+\frac{1}{2},k+\frac{1}{2}} - f_{1,j-\frac{1}{2},k+\frac{1}{2}} + f_{1,j+\frac{1}{2},k-\frac{1}{2}} - f_{1,j-\frac{1}{2},k-\frac{1}{2}} \right) \]

\[ + \frac{1}{2} \left( f_{2,j+\frac{1}{2},k+\frac{1}{2}} - f_{2,j+\frac{1}{2},k-\frac{1}{2}} + f_{2,j-\frac{1}{2},k+\frac{1}{2}} - f_{2,j-\frac{1}{2},k-\frac{1}{2}} \right). \quad (60) \]

Note that the mass matrix is diagonal, with the diagonal entries being \( J_{j,k} \).

**Theorem 4.1.** Assume that for \( j = 1, \ldots, J_{\text{max}} - 1 \) and \( k = 1, \ldots, K_{\text{max}} - 1 \), there hold

\[ J_{j,k} c_{j,k} + \frac{1}{2} \left[ (J_{\xi_x})_{j+\frac{1}{2},k} b_{1,j+\frac{1}{2},k} - (J_{\xi_x})_{j-\frac{1}{2},k} b_{1,j-\frac{1}{2},k} + (J_{\xi_y})_{j+\frac{1}{2},k} b_{2,j+\frac{1}{2},k} - (J_{\xi_y})_{j-\frac{1}{2},k} b_{2,j-\frac{1}{2},k} \right] \]

\[ + \frac{1}{2} \left[ (J_{\eta_x})_{j,k+\frac{1}{2}} b_{1,j,k+\frac{1}{2}} - (J_{\eta_x})_{j,k-\frac{1}{2}} b_{1,j,k-\frac{1}{2}} + (J_{\eta_y})_{j,k+\frac{1}{2}} b_{2,j,k+\frac{1}{2}} - (J_{\eta_y})_{j,k-\frac{1}{2}} b_{2,j,k-\frac{1}{2}} \right] \geq 0, \]

\[ \dot{J}_{j,k} = (J_{\xi_x})_{j+\frac{1}{2},k} \dot{x}_{j+\frac{1}{2},k} - (J_{\xi_x})_{j-\frac{1}{2},k} \dot{x}_{j-\frac{1}{2},k} + (J_{\eta_x})_{j+\frac{1}{2},k} \dot{y}_{j+\frac{1}{2},k} - (J_{\eta_x})_{j-\frac{1}{2},k} \dot{y}_{j-\frac{1}{2},k} \]

\[ + (J_{\eta_x})_{j,k+\frac{1}{2}} \dot{x}_{j,k+\frac{1}{2}} - (J_{\eta_x})_{j,k-\frac{1}{2}} \dot{x}_{j,k-\frac{1}{2}} + (J_{\eta_y})_{j,k+\frac{1}{2}} \dot{y}_{j,k+\frac{1}{2}} - (J_{\eta_y})_{j,k-\frac{1}{2}} \dot{y}_{j,k-\frac{1}{2}}. \quad (62) \]

Then, the finite difference scheme (57) satisfies the condition (11). As a consequence, the fully discrete scheme resulting from the application of the time integration method (27) to (9) with \( M \) and \( A \) given in (59) and (60) is unconditionally stable and of order \((2m)\) in time and order 2 in space.

**Proof.** Once again, we take \( f \equiv 0 \) and \( g \equiv 0 \) for stability analysis. From (58), (59), and (60), we have

\[ u^T (A + \sqrt{M} \frac{d}{dt} \sqrt{M}) u = -\frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} \dot{J}_{j,k} u_{j,k}^2 - \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} c_{j,k} J_{j,k} u_{j,k}^2 \]

\[ - \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} u_{j,k} \left[ (q_{1,j+\frac{1}{2},k} - q_{1,j-\frac{1}{2},k}) + (q_{2,j,k+\frac{1}{2}} - q_{2,j,k-\frac{1}{2}}) \right] \]
Moreover, from (54) and (55) the diffusion terms in (63) can be written as

\[ + \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} u_{j,k} \left( f_{1,j-\frac{1}{2},k+\frac{1}{2}} - f_{1,j-\frac{1}{2},k-\frac{1}{2}} + f_{1,j+\frac{1}{2},k-\frac{1}{2}} - f_{1,j+\frac{1}{2},k+\frac{1}{2}} \right) \]

\[ + \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} u_{j,k} \left( f_{2,j+\frac{1}{2},k+\frac{1}{2}} - f_{2,j+\frac{1}{2},k-\frac{1}{2}} + f_{2,j-\frac{1}{2},k+\frac{1}{2}} - f_{2,j-\frac{1}{2},k-\frac{1}{2}} \right) \]

\[ = -\frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} J_{j,k} u_{j,k}^2 - \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} c_{j,k} J_{j,k} u_{j,k}^2 - \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} q_{1,j-\frac{1}{2},k} (u_{j-1,k} - u_{j,k}) - \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} q_{2,j,k-\frac{1}{2}} (u_{j,k-1} - u_{j,k}) - \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} f_{1,j-\frac{1}{2},k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \]

\[ - \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} f_{2,j-\frac{1}{2},k} (u_{j,k} - u_{j,k-1} + u_{j-1,k} - u_{j-1,k-1}). \] (63)

From (52) and (53), the convection-related terms (the third and fourth terms on the right-hand side of the above equality) become

\[ \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}-1} q_{1,j-\frac{1}{2},k} (u_{j-1,k} - u_{j,k}) \]

\[ = \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} u_{j,k}^2 \left[ (J \xi_x)_{j, j+\frac{1}{2}, k} (b_{1,j+\frac{1}{2}, k} - \hat{x}_{j+\frac{1}{2}, k}) \right. \]

\[ + (J \xi_y)_{j, j+\frac{1}{2}, k} (b_{2,j+\frac{1}{2}, k} - \hat{y}_{j+\frac{1}{2}, k}) - (J \xi_x)_{j, j-\frac{1}{2}, k} (b_{1,j-\frac{1}{2}, k} - \hat{x}_{j-\frac{1}{2}, k}) \]

\[ - (J \xi_y)_{j, j-\frac{1}{2}, k} (b_{2,j-\frac{1}{2}, k} - \hat{y}_{j-\frac{1}{2}, k}) \] \] . (64)

and

\[ \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} q_{2,j,k-\frac{1}{2}} (u_{j,k-1} - u_{j,k}) \]

\[ = \frac{1}{2} \sum_{j=1}^{J_{\text{max}}-1} \sum_{k=1}^{K_{\text{max}}-1} u_{j,k}^2 \left[ (J \eta_x)_{j, j+\frac{1}{2}, k} (b_{1,j+\frac{1}{2}, k} - \hat{x}_{j,k, +\frac{1}{2}}) \right. \]

\[ + (J \eta_y)_{j, j+\frac{1}{2}, k} (b_{2,j+\frac{1}{2}, k} + \hat{y}_{j,k+\frac{1}{2}} - \hat{y}_{j,k, +\frac{1}{2}}) - (J \eta_x)_{j, j-\frac{1}{2}, k} (b_{1,j,k-\frac{1}{2}} - \hat{x}_{j,k, -\frac{1}{2}}) \]

\[ - (J \eta_y)_{j, j-\frac{1}{2}, k} (b_{2,j,k-\frac{1}{2}} - \hat{y}_{j,k, -\frac{1}{2}}) \] \] . (65)

Moreover, from (54) and (55) the diffusion terms in (63) can be written as

\[ -\frac{1}{2} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} f_{1,j-\frac{1}{2},k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \]

\[ -\frac{1}{2} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} f_{2,j-\frac{1}{2},k} (u_{j,k} - u_{j,k-1} + u_{j-1,k} - u_{j-1,k-1}) \]
Combining (64)–(66) with (63), we obtain

$$\begin{aligned}
\mathbf{u}^T (A + \sqrt{M} \frac{d}{dt} \sqrt{M}) \mathbf{u} & = - \frac{1}{4} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} a_{j-k} \left[ (J_{\xi})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \\
+ (J_{\eta})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \right]^2 \\
- \frac{1}{4} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} a_{j-k} \left[ (J_{\xi})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \\
+ (J_{\eta})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \right]^2 \\
- \frac{1}{2} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} \frac{1}{2} \mathbf{u}_{j,k}^2 \left[ \dot{J}_{j,k} - (J_{\xi})_{j-k} \dot{x}_{j,k} + (J_{\xi})_{j-k} \dot{x}_{j,k} - \right] \\
- (J_{\xi})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \\
+ (J_{\eta})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \\
- \frac{1}{2} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} \frac{1}{2} \mathbf{u}_{j,k}^2 \left[ 2c_{j,k} J_{j,k} + (J_{\xi})_{j-k} \dot{x}_{j,k} - (J_{\xi})_{j-k} \dot{x}_{j,k} \right] \\
+ (J_{\eta})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \\
- \frac{1}{2} \sum_{j=1}^{J_{\text{max}}} \sum_{k=1}^{K_{\text{max}}} \frac{1}{2} \mathbf{u}_{j,k}^2 \left[ 2c_{j,k} J_{j,k} + (J_{\xi})_{j-k} \dot{x}_{j,k} - (J_{\xi})_{j-k} \dot{x}_{j,k} \right] \\
- (J_{\eta})_{j-k} (u_{j,k} - u_{j-1,k} + u_{j,k-1} - u_{j-1,k-1}) \right].
\end{aligned}$$

The right-hand side of the above inequality is nonpositive and therefore scheme (61) satisfies the condition (10) if (61) and (62) hold.

**Remark 4.1.** The condition (61) is a central finite difference approximation to the condition (7) which takes the form in the new coordinates $\xi$ and $\eta$ as

$$J_c + \frac{1}{2} \frac{\partial}{\partial \xi} \left[ (J_{\xi}) b_1 - (J_{\xi}) b_2 \right] + \frac{1}{2} \frac{\partial}{\partial \eta} \left[ (J_{\xi}) b_1 + (J_{\eta}) b_2 \right] \geq 0.$$ 

As mentioned in Remark 3.1, (61) holds when $b = (b_1, b_2)^T$ is constant and $c$ is nonnegative. For the general situation with variable $b$, it is reasonable to expect the condition to hold provided that (7) holds and the mesh is sufficiently fine.
Remark 4.2. The condition (62) is new in multidimensions and often referred to as a geometric conservation law (GCL) in the literature. As a matter of fact, it is a central finite difference approximation of the continuous identity

\[ \hat{J} = (\dot{J}_x)(J_{\eta y}) + (\dot{J}_y)(J_{\eta x}) - (\dot{J}_y)(J_{\xi x}) - (\dot{J}_x)(J_{\xi y}), \]  

(68)

where the symbol \( \dot{\cdot} \) denotes the time derivative of a product. The importance of satisfying GCLs by numerical algorithms has been extensively studied; e.g., see [7] [14] [15] [18] [29]. Generally speaking, there are two approaches to enforce GCL (62). The first one, proposed by Thomas and Lombard [29], is to consider \( J_{j,k} \) as an unknown variable and update it by integrating (62) using the same scheme as for the underlying PDE. The main advantage of this approach is that the transformation quantities in (62) (also see (66)) can be approximated using arbitrary finite differences.

The other approach is to choose proper approximations for those transformation quantities in (66) such that (62) is satisfied automatically. An example set of such approximations is given by

\[
\begin{align*}
\dot{x}_{j-\frac{1}{2},k} & \equiv \frac{1}{8} [\dot{x}_{j,k-1} + \dot{x}_{j-1,k-1} + 2\dot{x}_{j,k} + 2\dot{x}_{j-1,k} + \dot{x}_{j,k+1} + \dot{x}_{j-1,k+1}], \\
\dot{x}_{j,k-\frac{1}{2}} & \equiv \frac{1}{8} [\dot{x}_{j-1,k} + \dot{x}_{j-1,k-1} + 0\dot{x}_{j,k-1} + 0\dot{x}_{j,k-1} + \dot{x}_{j,k+1} + \dot{x}_{j,k+1}], \\
\dot{y}_{j-\frac{1}{2},k} & \equiv \frac{1}{8} [\dot{y}_{j,k-1} + \dot{y}_{j-1,k-1} + 2\dot{y}_{j,k} + 2\dot{y}_{j-1,k} + \dot{y}_{j,k+1} + \dot{y}_{j-1,k+1}], \\
\dot{y}_{j,k-\frac{1}{2}} & \equiv \frac{1}{8} [\dot{y}_{j-1,k} + \dot{y}_{j-1,k-1} + 0\dot{y}_{j,k-1} + 0\dot{y}_{j,k-1} + \dot{y}_{j,k+1} + \dot{y}_{j,k+1}],
\end{align*}
\]

(69)

\[
\begin{align*}
(J_{\xi x})_{j-\frac{1}{2},k} & = (y_{\eta})_{j-\frac{1}{2},k} \equiv \frac{1}{4} [y_{j,k+1} - y_{j,k-1} + y_{j-1,k+1} - y_{j-1,k-1}], \\
(J_{\xi y})_{j-\frac{1}{2},k} & = -(x_{\eta})_{j-\frac{1}{2},k} \equiv -\frac{1}{4} [x_{j,k+1} - x_{j-1,k+1} + x_{j,k-1} - x_{j-1,k-1}], \\
(J_{\eta x})_{j,k-\frac{1}{2}} & = -(y_{\xi})_{j,k-\frac{1}{2}} \equiv -\frac{1}{4} [y_{j+1,k} - y_{j+1,k-1} + y_{j,k+1} - y_{j,k-1}], \\
(J_{\eta y})_{j,k-\frac{1}{2}} & = (x_{\xi})_{j,k-\frac{1}{2}} \equiv \frac{1}{4} [x_{j+1,k} - x_{j,k+1} + x_{j+1,k-1} - x_{j,k-1}],
\end{align*}
\]

(70)

\[ J_{j,k} = [(J_{\xi x})_{j+\frac{1}{2},k} + (J_{\xi y})_{j+\frac{1}{2},k} + (J_{\eta y})_{j,k+\frac{1}{2}} + (J_{\eta x})_{j,k+\frac{1}{2}}] \]

\[ \equiv \frac{1}{4} [(J_{\xi x})_{j+\frac{1}{2},k} + (J_{\xi y})_{j+\frac{1}{2},k} + (J_{\eta y})_{j,k+\frac{1}{2}} + (J_{\eta x})_{j,k+\frac{1}{2}}] - \frac{1}{4} [(J_{\xi y})_{j+\frac{1}{2},k} + (J_{\xi y})_{j+\frac{1}{2},k}] \]

(71)

To show that (62) is satisfied, using the identity

\[ ac - bd = \frac{1}{2} (a-b)(c+d) + \frac{1}{2} (a+b)(c-d), \]

we can rewrite (62) into

\[
\begin{align*}
\dot{J}_{j,k} & = \frac{1}{2} [(J_{\xi x})_{j+\frac{1}{2},k} - (J_{\xi x})_{j-\frac{1}{2},k}] (\dot{x}_{j+\frac{1}{2},k} + \dot{x}_{j-\frac{1}{2},k}) \\
& + \frac{1}{2} [(J_{\xi y})_{j+\frac{1}{2},k} - (J_{\xi y})_{j-\frac{1}{2},k}] (\dot{y}_{j+\frac{1}{2},k} - \dot{y}_{j-\frac{1}{2},k}) \\
& + \frac{1}{2} [(J_{\eta y})_{j+\frac{1}{2},k} - (J_{\eta y})_{j-\frac{1}{2},k}] (\dot{y}_{j+\frac{1}{2},k} + \dot{y}_{j-\frac{1}{2},k}) \\
& + \frac{1}{2} [(J_{\eta y})_{j+\frac{1}{2},k} - (J_{\eta y})_{j+\frac{1}{2},k}] (\dot{y}_{j+\frac{1}{2},k} - \dot{y}_{j-\frac{1}{2},k})
\end{align*}
\]
\[ + \frac{1}{2} [ (J_{\eta x})_{j,k+\frac{1}{2}} - (J_{\eta x})_{j,k-\frac{1}{2}} ] (\dot{x}_{j,k+\frac{1}{2}} + \dot{x}_{j,k-\frac{1}{2}}) \]
\[ + \frac{1}{2} [(J_{\eta y})_{j,k+\frac{1}{2}} + (J_{\eta y})_{j,k-\frac{1}{2}}] (\dot{y}_{j,k+\frac{1}{2}} - \dot{y}_{j,k-\frac{1}{2}}) \]
\[ + \frac{1}{2} [(J_{\xi y})_{j,k+\frac{1}{2}} + (J_{\xi y})_{j,k-\frac{1}{2}}] (\dot{y}_{j,k+\frac{1}{2}} - \dot{y}_{j,k-\frac{1}{2}}). \] (72)

Notice that the approximations defined in (69) and (70) satisfy
\[ \dot{x}_{j+\frac{1}{2},k} + \dot{x}_{j-\frac{1}{2},k} = \dot{x}_{j,k+\frac{1}{2}} + \dot{x}_{j,k-\frac{1}{2}}, \]
\[ \dot{y}_{j+\frac{1}{2},k} + \dot{y}_{j-\frac{1}{2},k} = \dot{y}_{j,k+\frac{1}{2}} + \dot{y}_{j,k-\frac{1}{2}}, \]
\[ (J_{\xi x})_{j+\frac{1}{2},k} - (J_{\xi x})_{j-\frac{1}{2},k} + (J_{\eta x})_{j,k+\frac{1}{2}} - (J_{\eta x})_{j,k-\frac{1}{2}} = 0, \]
\[ (J_{\xi y})_{j+\frac{1}{2},k} - (J_{\xi y})_{j-\frac{1}{2},k} + (J_{\eta y})_{j,k+\frac{1}{2}} - (J_{\eta y})_{j,k-\frac{1}{2}} = 0. \]

Inserting these identities into (72), we get
\[ \dot{j}_{j,k} = \frac{1}{2} [(J_{\xi x})_{j+\frac{1}{2},k} + (J_{\xi x})_{j-\frac{1}{2},k}] (\dot{x}_{j+\frac{1}{2},k} - \dot{x}_{j-\frac{1}{2},k}) \]
\[ + \frac{1}{2} [(J_{\xi y})_{j+\frac{1}{2},k} + (J_{\xi y})_{j-\frac{1}{2},k}] (\dot{y}_{j+\frac{1}{2},k} - \dot{y}_{j-\frac{1}{2},k}) \]
\[ + \frac{1}{2} [(J_{\eta x})_{j,k+\frac{1}{2}} + (J_{\eta x})_{j,k-\frac{1}{2}}] (\dot{x}_{j,k+\frac{1}{2}} - \dot{x}_{j,k-\frac{1}{2}}) \]
\[ + \frac{1}{2} [(J_{\eta y})_{j,k+\frac{1}{2}} + (J_{\eta y})_{j,k-\frac{1}{2}}] (\dot{y}_{j,k+\frac{1}{2}} - \dot{y}_{j,k-\frac{1}{2}}). \] (73)

Moreover, it can be shown that
\[ \frac{d}{dt} \frac{1}{2} [(J_{\eta y})_{j,k+\frac{1}{2}} + (J_{\eta y})_{j,k-\frac{1}{2}}] = (\dot{x}_{j+\frac{1}{2},k} - \dot{x}_{j-\frac{1}{2},k}), \]
\[ \frac{d}{dt} \frac{1}{2} [(J_{\xi x})_{j+\frac{1}{2},k} + (J_{\xi x})_{j-\frac{1}{2},k}] = (\dot{y}_{j,k+\frac{1}{2}} - \dot{y}_{j,k-\frac{1}{2}}), \]
\[ \frac{d}{dt} \frac{1}{2} [(J_{\xi y})_{j,k+\frac{1}{2}} + (J_{\xi y})_{j,k-\frac{1}{2}}] = -(\dot{y}_{j,k+\frac{1}{2}} - \dot{y}_{j,k-\frac{1}{2}}), \]
\[ \frac{d}{dt} \frac{1}{2} [(J_{\xi y})_{j+\frac{1}{2},k} + (J_{\xi y})_{j-\frac{1}{2},k}] = -(\dot{x}_{j,k+\frac{1}{2}} - \dot{x}_{j,k-\frac{1}{2}}). \]

Using this and (71) we can see that (73), and therefore (62), hold.

5 Numerical examples

In this section we present numerical results obtained with the numerical schemes developed in the previous sections for two one dimensional and one two dimensional examples. Our objective is to verify the stability and accuracy of those schemes.

Example 5.1. We first consider a one dimensional example in the form (27). The coefficients are given by
\[ \begin{cases} a(x,t) = 1, & b(x,t) = 0, & c(x,t) = 0, \\ x_l = 0, & x_r = \pi. \end{cases} \] (74)
The source term \( f(x,t) \), the initial solution, and the Dirichlet boundary conditions are chosen such that the exact solution of the IBVP is given by

\[
 u_{\text{exact}}(x,t) = (2 + \sin(\pi t)) \sin(x) \tag{75} 
\]

Notice that we have \( g(x,t) \equiv 0 \) for this exact solution. The mesh is chosen as

\[
 x_j(t) = \frac{j\pi}{J_{\text{max}}} + \frac{1}{4} \sin \left( \frac{2j\pi}{J_{\text{max}}} \right) \sin(\omega t), \quad j = 0, \ldots, J_{\text{max}} \tag{76} 
\]

where the parameter \( \omega \) is used to control the speed of mesh movement. The trajectories of two meshes with \( \omega = 2\pi \) and \( 20\pi \), respectively, are shown in Fig. 1.

Fig. 2(a) shows the maximum error as a function of \( \Delta t \) obtained with scheme (21) + (40) \((m = 1)\) for mesh (76) \((\omega = 2\pi)\). In the computation, \( J_{\text{max}} \) is taken sufficiently large \((J_{\text{max}} = 1000)\). In this case, the spatial discretization error is ignorable and the total error is dominated by the temporal discretization error. The results show the \( O(\Delta t^2) \) behavior of the error, consistent with the theoretical prediction. They also demonstrate that the scheme is stable for all used values of \( \Delta t \).

Fig. 2(b) shows the maximum error as a function of \( J_{\text{max}} \). The results are obtained with \( m = 1, 2, 3 \) and the time step size \( \Delta t = (\pi/J_{\text{max}})^{1/m} \), respectively. The reason \( \Delta t \) is chosen this way is that the error for scheme (11) is of order \( O(\Delta t^{2m}) + O(\Delta x^2) \), which reduces to \( O(J_{\text{max}}^{-2}) \) with this choice of \( \Delta t \). The figure confirms that the maximum error converges quadratically for all three cases.

The numerical results obtained with scheme (21) + (40) \((m = 1)\) for a faster moving mesh \( \omega = 20\pi \) are shown in Fig. 3. Once again, the theoretically predicted order of convergence (second order in both time and space for \( m = 1 \)) is observed. Interestingly, one may observe that there is a bump in the curve in Fig. 3(a). This is because for large \( \Delta t \), the fast movement of the mesh is not “felt” by the integration and this results in smaller error.

To conclude this example, we study the approximation of the boundary condition (31) with a nonhomogeneous term. Applying the collocation scheme (21) to (31) (for simplicity we keep it in the old variable), we get

\[
 u_0(t_{n,j}) = g(x_l(t_{n,j}), t_{n,j}), \quad u_{J_{\text{max}}}(t_{n,j}) = g(x_r(t_{n,j}), t_{n,j}), \quad j = 1, \ldots, m \tag{77} 
\]

where \( u_0(t) \approx u(x_0(t), t) \), \( u_{J_{\text{max}}}(t) \approx u(x_{J_{\text{max}}}(t), t) \), and \( t_{n,1}, \ldots, t_{n,m} \) are the collocation points (cf. (20)). This has the advantage that the boundary condition is collocated at the same points as the PDE and thus can be implemented conveniently within the framework of the method of lines. On the other hand, since the boundary condition is not imposed at \( t = t_{n+1} \), error may occur there and accumulate and eventually result in inaccurate computational solutions. This does not seem to be an issue for a homogeneous boundary condition. However, the situation is different when a non-homogeneous boundary condition is involved.

To see this, we consider a case where the same PDE and mesh (with \( \omega = 2\pi \)) are used but the source term, initial condition, and (non-homogeneous) boundary condition are chosen such that the exact solution is given by

\[
 u_{\text{exact}}(x,t) = (2 + \sin(\pi t)) \cos(x) \tag{78} 
\]
Figure 1: Example 5.1. The trajectories of two meshes of 41 points for \( \omega = 2\pi \) and \( \omega = 20\pi \) are shown in (a) and (b), respectively.

Fig. 4(a) shows the numerical results obtained with (77). It can be seen that the maximum error converges at a rate of \( O(J_{max}^{-1}) \) for both cases \( m = 2 \) and \( m = 3 \), much worse than the predicted order \( O(J_{max}^{-2}) \).

To avoid this difficulty, we consider collocating the boundary condition at the approximation points (cf. (22)), i.e.,

\[
    u_0(\tilde{t}_{n,j}) = g(x_l(\tilde{t}_{n,j}), \tilde{t}_{n,j}), \quad u_{\max}(\tilde{t}_{n,j}) = g(x_r(\tilde{t}_{n,j}), \tilde{t}_{n,j}), \quad j = 1, ..., m.
\]

(79)

The main advantage of this approximation is that the boundary condition is now enforced at \( t = \tilde{t}_{n,m} = t_{n+1} \). It is not difficult to show that (77) and (79) are equivalent for homogeneous Dirichlet boundary conditions but different in general. Fig. 4(b) shows the results obtained with (79). It can be seen that the second order convergence rate is recovered for all three cases.

(a) \( m = 1 \)  
(b) \( m = 1, 2, 3 \)

Figure 2: The maximum solution error for scheme (21) + (40) applied to Example 5.1 (\( \omega = 2\pi \)). The error is plotted in (a) as a function of \( \Delta t \) for \( J_{max} = 1000 \) and in (b) as a function of \( J_{max} \) for \( m = 1 \) (\( \Delta t = \pi/J_{max} \)), \( m = 2 \) (\( \Delta t = \sqrt{\pi}/J_{max} \)), and \( m = 3 \) (\( \Delta t = \sqrt[3]{\pi}/J_{max} \)).
The trajectories of two meshes with \( \omega = 2\pi \) and \( 20\pi \) are shown in Fig. 5.
Recall that the mesh is treated linearly on each time interval \([t_n, t_{n+1}]\). As a consequence, when the boundary of the domain is moving, the first and last mesh points, \(x_0(t)\) and \(x_{J_{\max}}(t)\), generally do not coincide with the boundary points \(x_l(t)\) and \(x_r(t)\) for \(t \in (t_n, t_{n+1})\); see the illustration in Fig. 6. In this situation, a more accurate approximation of the boundary conditions than (77) or (79) is needed. By expanding \(u(x_0(t), t)\) and \(u(x_1(t), t)\) about \(x_l(t)\) and \(u(x_{J_{\max}}(t), t)\) and \(u(x_{J_{\max}-1}(t), t)\) about \(x_r(t)\), we get

\[
\begin{align*}
&(x_1 - x_l)(u_0 - g(x_l, t)) - (x_0 - x_l)(u_1 - g(x_l, t)) = 0, \\
&(x_{J_{\max}} - x_r)(u_{J_{\max}} - g(x_r, t)) - (x_{J_{\max}} - x_r)(u_{J_{\max}-1} - g(x_r, t)) = 0.
\end{align*}
\]

(83)

As discussed in Example 5.1, the above conditions are imposed at the approximation points, \(\tilde{t}_{n,1}, ..., \tilde{t}_{n,m}\) (cf. (22)).

Numerical results obtained with scheme (21) + (40) \((m = 1, 2, 3)\) for \(\omega = 2\pi\) and \(20\pi\) are shown in Figs. 7 and 8, respectively. It can be seen that the scheme is stable (the solution is bounded) and the error is of order \(O(\Delta t^2 m) + O(\Delta x^2)\) \((m = 1, 2, 3)\), consistent with the theoretical prediction.

\[\square\]

Figure 5: Mesh trajectories of two meshes of 41 points for \(\omega = 2\pi\) and \(20\pi\), respectively.

**Example 5.3.** The last example is a two dimensional problem in the form of IBVP (5) with

\[
\begin{align*}
\begin{cases}
a(x,y,t) = 1, & b_1(x,y,t) = b_2(x,y,t) = 0, & c(x,t) = 0, \\
\Omega = (0, \pi) \times (0, \pi).
\end{cases}
\end{align*}
\]

(84)

The source term \(f(x, y, t)\), initial solution, and Dirichlet boundary condition are chosen such that the exact solution of the IBVP is given by

\[u_{\text{exact}}(x,y,t) = (2 + \sin(\pi t)) \sin(x) \sin(y).\]

(85)

The moving mesh is generated using the coordinate transformation

\[x = \xi + 0.2 \sin(2\xi) \sin(2\eta) \sin(\omega t), \quad y = \eta + 0.2 \sin(2\xi) \sin(2\eta) \sin(\omega t), \quad (\xi, \eta) \in \Omega\]

(86)
Figure 6: A sketch of the boundary \( x = x_l(t) \) and the first mesh point \( x = x_0(t) \). Generally speaking, \( x_0(t) \) does not coincide with \( x_l(t) \) for \( t \in (t_n, t_{n+1}) \).

Figure 7: The maximum error for scheme (21) + (40) \((m = 1, 2, 3)\) applied to Example 5.2 \((\omega = 2\pi)\). The error is plotted in (a) as a function of \( \Delta t \) for \( J_{\text{max}} = 1000 \) and in (b) as a function of \( J_{\text{max}} \) for \( \Delta t = (\pi/J_{\text{max}})^{1/m} \), \( m = 1, 2, 3 \).

where the parameter \( \omega \) is used to control the speed of mesh movement and a rectangular mesh of \((J+1)(K+1)\) points is used for the \( \xi-\eta \) domain. In our computation \( \omega = 2\pi \) is used. A 41 \( \times \) 41 moving mesh is shown in Fig. 9 for \( t = 0.25 \) and \( t = 0.75 \). The numerical results are shown in Fig. 10, which are consistent with the observations made for the 1D examples and with the theoretical prediction.

6 Conclusions and comments

In the previous sections we have developed a family of finite difference schemes for linear convection-diffusion equations on moving meshes. Those schemes can be of second and higher order in time, preserve a stability inequality, and are unconditionally stable in the sense that they impose no constraint on time step size for stability purpose. More specifically, scheme (21) has been developed for ODE system of the form (9) with property (10) by first transforming (9) into (12) and then applying the \( m \)-point collocation scheme to the transformed system. Several finite difference discretizations
Figure 8: The maximum error for scheme (21) + (40) (with \( m = 1 \)) applied to Example 5.2 (\( \omega = 20\pi \)). The error is plotted in (a) as a function of \( \Delta t \) for \( J_{\text{max}} = 1000 \) and in (b) as a function of \( J_{\text{max}} \) for \( \Delta t = 0.1\pi / J_{\text{max}} \).

Figure 9: A 41 \times 41 moving mesh for Example 5.3 is shown for \( t = 0.25 \) and \( t = 0.75 \).

for one dimensional and two dimensional convection-diffusion equations on moving meshes have been constructed and shown to satisfy property (10) (cf. Theorems 3.1 and 4.1). Numerical results presented in Section 5 verify the theoretically predicted stability and convergence order of those schemes.

Several generalizations of the current work are under investigation. It is interesting to know how the current strategy can be used for nonlinear differential equations and wave equations and how it can be combined with the method of lines for general differential equations. Moreover, it is noted that scheme (21) may not work efficiently when mass matrix \( M(t) \) is not diagonal (as in the case with finite element approximation). Development of an efficient implementation in the finite element case certainly deserves further investigations.
Figure 10: The maximum error for scheme (21) applied to (9) with (59) and (60) for Example 5.3 ($\omega = 2\pi$). The error is plotted in (a) as a function of $\Delta t$ for $J_{\text{max}} = K_{\text{max}} = 160$ ($J_{\text{max}} = K_{\text{max}} = 320$ was used for $\Delta t = 0.01$) and in (b) as a function of $J_{\text{max}}$ for $\Delta t = (\pi/J_{\text{max}})^{1/m}$, $m = 1, 2, 3$.

References


