Journal of Computational Mathematics
Vol.27, No.6, 2009, 713–728.
http://www.global-sci.org/jcm
doi:10.4208/jcm.2009.09-m2852

ON A MOVING MESH METHOD FOR SOLVING PARTIAL INTEGRO-DIFFERENTIAL EQUATIONS*

Jingtang Ma
School of Economic Mathematics, Southwestern University of Finance and Economics, Chengdu 610074, China
Email: mjt@swufe.edu.cn
Yingjun Jiang[1)]
Department of Mathematics and Scientific Computing, Changsha University of Science and Technology, Changsha 410076, China
Email: jiangyingjun@csust.edu.cn
Kaili Xiang
School of Economic Mathematics, Southwestern University of Finance and Economics, Chengdu 610074, China
Email: xiangkl@swufe.edu.cn

Abstract

This paper develops and analyzes a moving mesh finite difference method for solving partial integro-differential equations. First, the time-dependent mapping of the coordinate transformation is approximated by a piecewise quadratic polynomial in space and a piecewise linear function in time. Then, an efficient method to discretize the memory term of the equation is designed using the moving mesh approach. In each time slice, a simple piecewise constant approximation of the integrand is used, and thus a quadrature is constructed for the memory term. The central finite difference scheme for space and the backward Euler scheme for time are used. The paper proves that the accumulation of the quadrature error is uniformly bounded and that the convergence of the method is second order in space and first order in time. Numerical experiments are carried out to confirm the theoretical predictions.

Mathematics subject classification: 65R20, 65N50, 65M50, 65M20, 65M15, 65M10.
Key words: Partial integro-differential equations, Moving mesh methods, Stability and convergence.

1. Introduction

Moving mesh methods for solving partial differential equations(PDEs) work in the way of moving the mesh points towards the region of large gradient while keeping the number of mesh points fixed during the process. Over the past three decades focuses have been on the development of algorithms for both mesh generation and discretizing the physical PDEs on variable meshes. Blom, Sanz-Serna and Verwer [2] first classify the moving mesh algorithms — BJCN scheme (it can be regarded as a special case of Godunov methods), IEL scheme (implicit-Euler Lagrangian scheme), and RFDM (rezoning finite difference method). After then Tang [16] classifies the moving mesh methods into two general classes: interpolation-free algorithms and interpolation-based algorithms. In interpolation-free algorithms, the mesh equations and

* Received May 4, 2008 / Revised version received December 6, 2008 / Accepted December 30, 2008 /
1) corresponding author.

the original PDEs are solved simultaneously for the physical solution and the mesh, see e.g., [5,8,17]. In the interpolation-based algorithms, the solutions of the mesh equations and physical equations are separate. The algorithms summarized in [2, 10, 15] (see also [17] and references therein) belong to this group. Recently, the analysis on stability and convergence has been brought up to attention. However, little achievement has been made in this direction. Jamet [9] gives a convergence proof of BJCN scheme for the heat equation with moving boundaries, where the mesh evolves uniformly with the moving boundaries. Their proof highly relies on the uniformity of the spatial mesh in each time level, while the technique cannot be used for the variable mesh in each time level. Recently, Ma [13] gives a convergence proof of BJCN scheme on general variable mesh — equidistributing mesh. Mackenzie and Mekwi [14] prove an asymptotic second-order convergence for a conservative IEL scheme which can be regarded as a variant of BJCN scheme. Lipnikov and Shashkov [11] give a rigorous analysis on a rezoning method where the rezoning mesh is generated by minimizing a posteriori error in L_2 norm instead of by an equidistribution principle.

In this paper, we design a stable and second-order scheme for partial integro-differential equations which arise in many applications (e.g., [3, 19] and the references). Although the scheme is more or less motivated by Mackenzie and Mekwi [14], it improves the accuracy to second order using quadratic approximation to the mesh trajectory. In addition, the analysis is given for integro-differential equations. This type of equations can generate singular solutions whose locations are not known a priori. Thus moving mesh methods deserve to solving this type of equations. To discretize the memory term with moving mesh, a piecewise constant polynomial is used to approximate the integrand in each time slice. The accumulation of the quadrature error is proven to be uniformly bounded and thus the stability is derived under several mild assumptions.

In particular, we consider a partial integro-differential equation of the form

$$u_t + au_x - \kappa u_{xx} + \int_0^t k(x,t,s)u(x,s)\,ds = 0, \quad x \in I \equiv [x_L, x_R],\ t \in J \equiv [0,T], \tag{1.1}$$

$$u(x,0) = u_0(x), \qquad x \in I, \tag{1.2}$$

$$u(x_L,t) = b_L(t), \quad u(x_R,t) = b_R(t), \qquad t \in J. \tag{1.3}$$

where a is a constant advection velocity and κ a constant diffusivity, the integral is called memory term, $k(x,t,s)$ is the kernel function satisfying

$$\max_{x\in I} |k(x,t,s)| \le C|A(t,s)K_\alpha(t-s)|,$$

where A is sufficiently smooth in t and s, and the Hammerstein kernel

$$K_\alpha(t-s) = \begin{cases} (t-s)^{-\alpha}, & 0<\alpha<1, \\ K(t-s), & \text{otherwise,} \end{cases}$$

K is smooth function, $K_\alpha(t-s) = (t-s)^{-\alpha}$ is said to be weakly singular kernel.

The mesh movement is based on the time-dependent mapping

$$x(\cdot,t):\ I_c \equiv [0,1] \to I \equiv [x_L, x_R].$$

Then a function $u(x,t)$ in physical variables is transformed into the function in computational variables

$$u(x,t) = u(x(\xi,t),t).$$

Thus by chain rule the material derivative is given by

$$\dot{u} = u_t + \dot{x}u_x.$$

Moreover,

$$u_t = \dot{u} - \dot{x}u_x, \quad u_x = \frac{u_\xi}{x_\xi}, \quad u_{xx} = \frac{(u_x)_\xi}{x_\xi} = \left(\frac{u_\xi}{x_\xi}\right)_\xi / x_\xi. \tag{1.4}$$

Put the arguments (1.4) into the original equation (1.1). Then equation (1.1) is recast into

$$\dot{(x_\xi u)} - \dot{x}_\xi u - (\dot{x} - a)u_\xi - \left(\kappa \frac{u_\xi}{x_\xi}\right)_\xi + x_\xi \int_0^t k(x,t,s)u(x,s)\, ds = 0. \tag{1.5}$$

The moving mesh method studied in this paper is essentially the finite difference method for the transformed equation (1.5) in computational variable ξ and temporal variable t (see (2.11)). Furthermore, the mapping $x(\xi, t)$ is approximated by a piecewise quadratic polynomial in space and linear polynomial in time to obtain second order convergence which is one of the major achievements of this paper.

2. Discretization of Moving Mesh Equation and Numerical Scheme

Define a uniform mesh in the computational variables

$$\xi_j = \frac{j}{N}, \quad j = 0, 1, \ldots, N.$$

Then a moving mesh is given by

$$x_j(t) \equiv x(\xi_j, t), \quad j = 0, 1, \ldots, N,$$

where $x_0 \equiv x_L$ and $x_N \equiv x_R$. The measure of each physical cell will be denoted by

$$h_j(t) = x_j(t) - x_{j-1}(t), \quad j = 1, 2, \ldots, N.$$

Define a temporal mesh $0 = t_0 < t_1 < \cdots < t_L = T$, and write $\Delta t_n = t_n - t_{n-1}$. Given the mapping $x(\xi, t)$, the location of the physical mesh points at time level $t = t_n$ and $t = t_{n+1}$, is well determined. Define

$$x_{j-1/2}(t_n) = \frac{x_{j-1}(t_n) + x_j(t_n)}{2}, \quad j = 1, \ldots, N, \quad n = 0, 1, \ldots, L.$$

Define $x^h(\xi, t)$ that is piecewise quadratic in space and linear in time (see figure 2.1) as an approximation of function $x(\xi, t)$. More precisely define

$$x^h(\xi, t_n) = x_{j-1/2}(t_n)\ell_0(\xi) + x_j(t_n)\ell_1(\xi) + x_{j+1/2}(t_n)\ell_2(\xi),$$

for $\xi \in [\xi_{j-1/2}, \xi_{j+1/2}]$, where $\xi_{j-1/2}$ is the midpoint of ξ_{j-1} and ξ_j, ℓ_0, ℓ_1, ℓ_2 are quadratic Lagrange polynomials on points $\xi_{j-1/2}, \xi_j, \xi_{j+1/2}$. Then for $t \in [t_n, t_{n+1}]$ and $\xi \in [\xi_{j-1/2}, \xi_{j+1/2}]$,

$$x^h(\xi, t) = x^h(\xi, t_n) + (t - t_n)\frac{x^h(\xi, t_{n+1}) - x^h(\xi, t_n)}{t_{n+1} - t_n}.$$

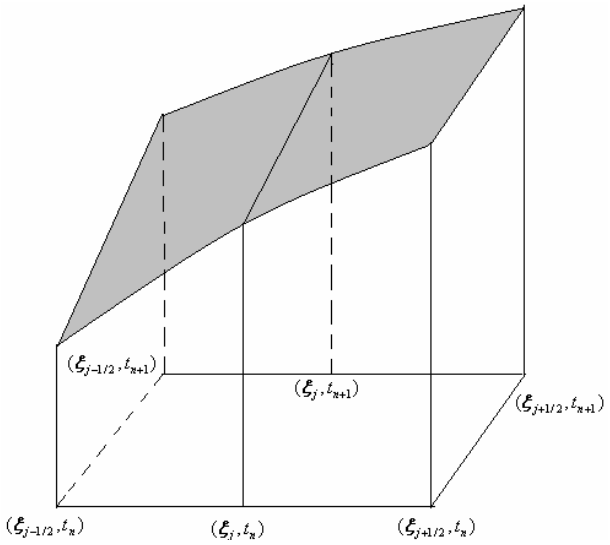

Fig. 2.1. $x^h(\xi,t)$, $(\xi,t)\in[\xi_{j-1/2},\xi_{j+1/2}]\times[t_n,t_{n+1}]$

It is easy to see that $x^h(\xi,t)$ satisfies

$$(\dot{x}^h_\xi)^{n+1}_j = \frac{(x^h_\xi)^{n+1}_j - (x^h_\xi)^n_j}{t_{n+1}-t_n} = \frac{\dot{x}^h(\xi_{j+1/2},t_{n+1}) - \dot{x}^h(\xi_{j-1/2},t_{n+1})}{\triangle\xi}, \tag{2.1}$$

where the second equality states a discrete geometric conservation law (DGCL) (or discrete version of (4) in [4]).

We make three assumptions for the moving mesh:

$$(A1) \qquad |x^{n+1}_j - x^n_j| \le C\Delta t_{n+1} \tag{2.2}$$

$$(A2) \qquad |h^n_{j+1} - h^n_j| \le C\min((h^n_{j+1})^2,(h^n_j)^2). \tag{2.3}$$

$$(A3) \qquad \frac{|h^{n+1}_{j+1} - h^n_{j+1}|}{\min(h^{n+1}_{j+1},h^n_{j+1})} \le C\Delta t_{n+1}. \tag{2.4}$$

In this paper, C denotes a generic positive constant.

Remark 2.1. The limit of (A1) as $t_n\to 0$ means that the speed of mesh movement is uniformly bounded. Assumption (A3) is also a condition on the mesh speed, which is equivalent to the one used in Bank and Santos [1] for analysis of moving mesh finite element methods with time being continuous. Also, we can see that using (A3) can yield (A1). In this sense, assumption (A3) is more restrictive than (A1). However both of these assumptions hold true for the meshes obtained by the equidistribution of a suitably smooth monitor function (cf. [6]). (A2) makes an assumption on the smoothness of the time-dependent coordinate transformation (see also [14]).

Lemma 2.1. *With assumption (A3), we have the estimations*

$$\left|(\dot{x}^h)^{n+1}_{j+1/2} - (\dot{x}^h)^{n+1}_{j-1/2}\right| \le C(h^{n+1}_{j+1} + h^{n+1}_j), \tag{2.5}$$

and

$$\left|(\dot{x}^h)^{n+1}_{j+1} - (\dot{x}^h)^{n+1}_j\right| \le C\min\left(h^{n+1}_{j+1}, \frac{h^{n+1}_{j+1}+h^{n+1}_{j+2}}{2}, \frac{h^{n+1}_{j+1}+h^{n+1}_j}{2}\right). \tag{2.6}$$

Proof. By (2.1)

$$\left|(\dot{x}^h)^{n+1}_{j+1/2} - (\dot{x}^h)^{n+1}_{j-1/2}\right| = \left|\Delta\xi \frac{(\dot{x}^h)^{n+1}_{j+1/2} - (\dot{x}^h)^{n+1}_{j-1/2}}{\Delta\xi}\right| = \left|\Delta\xi \frac{(x^h_\xi)^{n+1}_j - (x^h_\xi)^n_j}{\Delta t_{n+1}}\right|.$$

Using

$$(x^h_\xi)^n_j \equiv x^h_\xi(\xi_j, t_n) = \frac{x_{j+1/2}(t_n) - x_{j-1/2}(t_n)}{\Delta\xi} = \frac{1}{\Delta\xi}\left(\frac{h^n_{j+1} + h^n_j}{2}\right), \tag{2.7}$$

gives that

$$\left|(\dot{x}^h)^{n+1}_{j+1/2} - (\dot{x}^h)^{n+1}_{j-1/2}\right| = \left|\frac{(h^{n+1}_{j+1} + h^{n+1}_j) - (h^n_{j+1} + h^n_j)}{2\Delta t_{n+1}}\right|.$$

Then using assumption (A3) leads to (2.5).
Since

$$\dot{x}^h(\xi_j, t) = \frac{x_j(t_{n+1}) - x_j(t_n)}{\Delta t_n} \equiv \frac{x^{n+1}_j - x^n_j}{\Delta t_n}, \quad t \in (t_n, t_{n+1}], \tag{2.8}$$

we have

$$\left|(\dot{x}^h)^{n+1}_{j+1} - (\dot{x}^h)^{n+1}_j\right| = \left|\frac{x^{n+1}_{j+1} - x^n_{j+1}}{\Delta t_n} - \frac{x^{n+1}_j - x^n_j}{\Delta t_n}\right| = \left|\frac{h^{n+1}_{j+1} - h^n_{j+1}}{\Delta t_n}\right|.$$

Finally assumption (A3) gives the inequality (2.6). □

Now we derive the numerical scheme for (1.5) (equivalently for (1.1)). Denote the approximation of solution u by U and $U(x^n_j, t_n)$ by U^n_j. The forward and backward divided differences are given by

$$(D_+U)_j = \frac{U_{j+1} - U_j}{h_{j+1}}, \quad (D_-U)_j = \frac{U_j - U_{j-1}}{h_j},$$

and the average operator by

$$(\delta U)_{j+1/2} = \frac{1}{2}(U_j + U_{j+1}).$$

Using central finite difference method for space and Backward Euler (BE) for time to discretize the non-integral parts of (1.5) gives that

$$\begin{aligned} &(\dot{\overline{x_\xi u}}) - \dot{x}_\xi u - (\dot{x} - a)u_\xi - \left(\kappa \frac{u_\xi}{x_\xi}\right)_\xi \\ \approx\ & \frac{(x^h_\xi U)^{n+1}_j - (x^h_\xi U)^n_j}{\Delta t_{n+1}} - (\dot{x}^h_\xi)^{n+1}_j U^{n+1}_j \\ &- \frac{1}{\Delta\xi}\left[(\kappa(D_+ - D_-)U)^{n+1}_j + ((\dot{x}^h)^{n+1}_j - a)\left((\delta U)^{n+1}_{j+1/2} - (\delta U)^{n+1}_{j-1/2}\right)\right]. \end{aligned}$$

Denote by ϕ^n_j the nodal basis function (piecewise linear) at point x^n_j, $j = 1, 2, \ldots, N-1$. Then define temporal piecewise constant and spatial piecewise linear polynomial

$$\widetilde{U}(x, t) \equiv \sum_{j=0}^{N-1} U^k_j \phi^k_j, \quad t \in [t_k, t_{k+1}), \quad \text{for all } k \le n. \tag{2.9}$$

The integral term in (1.5) is approximated by

$$\begin{aligned}
&(x_\xi^h)_j^{n+1}\int_0^{t_{n+1}} k(x_j^{n+1},t_{n+1},s)u(x_j^{n+1},s)\,ds\\
=&\ \frac{1}{\Delta\xi}\frac{h_j+h_{j+1}}{2}\int_0^{t_{n+1}} k(x_j^{n+1},t_{n+1},s)u(x_j^{n+1},s)\,ds\\
\approx&\ \frac{1}{\Delta\xi}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}\left(\int_0^{t_{n+1}} k(x_j^{n+1},t_{n+1},s)u(x,s)ds\right)dx\\
\approx&\ \frac{1}{\Delta\xi}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}\left(\int_0^{t_{n+1}} k(x_j^{n+1},t_{n+1},s)\widetilde{U}(x,s)ds\right)dx. \qquad (2.10)
\end{aligned}$$

Hence we obtain a numerical scheme for (1.5) (i.e., for (1.1)) as follows:

$$\begin{aligned}
(x_\xi^h U)_j^{n+1} =&(x_\xi^h U)_j^n+\Delta t_{n+1}(\dot{x}_\xi^h)_j^{n+1}U_j^{n+1}\\
&+\frac{\Delta t_{n+1}}{\Delta\xi}\Big[(\kappa(D_+-D_-)U)_j^{n+1}+((\dot{x}^h)_j^{n+1}-a)\left((\delta U)_{j+1/2}^{n+1}-(\delta U)_{j-1/2}^{n+1}\right)\Big]\\
&-\frac{\Delta t_{n+1}}{\Delta\xi}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}\left(\int_0^{t_{n+1}} k(x_j^{n+1},t_{n+1},s)\widetilde{U}(x,s)ds\right)dx. \qquad (2.11)
\end{aligned}$$

3. Stability and Convergence

Define a mesh-dependent L_2 norm (noting the homogeneous boundary conditions)

$$\|U\|_n=\left(\sum_{j=1}^{N-1}\left(\frac{h_j^n+h_{j+1}^n}{2}\right)(U_j)^2\right)^{\frac{1}{2}}=\left\|\sum_{j=1}^{N-1}U_j\chi_j^n\right\|_{L^2}, \qquad (3.1)$$

where functions χ_j^n, $j=1,\dots,N-1$ are given by

$$\chi_j^n(x)=\begin{cases}1 & \text{if } x\in(x_j^n-\frac{h_j^n}{2},x_j^n+\frac{h_{j+1}^n}{2});\\ 0, & \text{otherwise.}\end{cases} \qquad (3.2)$$

Approximations of the derivatives will be measured in the cell-based norm

$$|||v|||_n=\left(\sum_{j=1}^{N-1}h_j^n(v_j)^2\right)^{\frac{1}{2}}. \qquad (3.3)$$

Now we present the stability result.

Theorem 3.1. *When the scheme* (2.11) *is applied to solve problem* (1.1)-(1.3)*, for sufficiently small temporal mesh size, under assumption* ($A3$)*, the following a priori bound holds*

$$\|U^{n+1}\|_{n+1}\le C\|U^0\|_0, \qquad (3.4)$$

where C is a generic positive constant.

Proof. Multiplying U_j^{n+1} on both sides of (2.11) and summing over all interior nodes (since $U_0 = U_N = 0$), we obtain

$$\sum_{j=1}^{N-1} (x_\xi^h)_j^{n+1} (U_j^{n+1})^2 = \mathbf{I} + \mathbf{II} + \mathbf{III} + \mathbf{IV} + \mathbf{V}, \tag{3.5}$$

where

$$\begin{aligned}
\mathbf{I} &= \sum_{j=1}^{N-1} (x_\xi^h)_j^n U_j^n U_j^{n+1}, \\
\mathbf{II} &= \Delta t_{n+1} \sum_{j=1}^{N-1} (\dot{x}_\xi^h)_j^{n+1} (U_j^{n+1})^2, \\
\mathbf{III} &= \frac{\kappa \Delta t_{n+1}}{\Delta \xi} \sum_{j=1}^{N-1} [((D_+ - D_-)U)_j^{n+1}] U_j^{n+1}, \\
\mathbf{IV} &= \frac{\Delta t_{n+1}}{\Delta \xi} \sum_{j=1}^{N-1} \left[((\dot{x}^h)_j^{n+1} - a)((\delta U)_{j+1/2}^{n+1} - (\delta U)_{j-1/2}^{n+1}) \right] U_j^{n+1}, \\
\mathbf{V} &= -\frac{\Delta t_{n+1}}{\Delta \xi} \sum_{j=1}^{N-1} \int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}} \left(\int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s) \widetilde{U}(x,s) ds \right) dx U_j^{n+1}.
\end{aligned}$$

Applying the identity

$$ab = \frac{1}{2}a^2 + \frac{1}{2}b^2 - \frac{1}{2}(a-b)^2$$

to product $U_j^n U_j^{n+1}$ in term $\mathbf{I}$, we establish

$$\begin{aligned}
\mathbf{I} &= \frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^n [(U_j^n)^2 + (U_j^{n+1})^2 - (U_j^n - U_j^{n+1})^2] \\
&= \frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^n (U_j^n)^2 + \frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^n (U_j^{n+1})^2 - \frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^n (U_j^n - U_j^{n+1})^2.
\end{aligned} \tag{3.6}$$

The second term in (3.6) is estimated, with the use of (2.1), (2.7) and then the definition of the mesh-dependent L_2 norm, to be

$$\begin{aligned}
&\frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^n (U_j^{n+1})^2 \\
=&\frac{1}{2} \sum_{j=1}^{N-1} (x_\xi^h)_j^{n+1} (U_j^{n+1})^2 - \frac{\Delta t_{n+1}}{2\Delta \xi} \sum_{j=1}^{N-1} \left((\dot{x}^h)_{j+1/2}^{n+1} - (\dot{x}^h)_{j-1/2}^{n+1} \right) (U_j^{n+1})^2 \\
=&\frac{1}{2\Delta \xi} ||U^{n+1}||_{n+1}^2 - \frac{\Delta t_{n+1}}{2\Delta \xi} \sum_{j=1}^{N-1} \left((\dot{x}^h)_{j+1/2}^{n+1} - (\dot{x}^h)_{j-1/2}^{n+1} \right) (U_j^{n+1})^2.
\end{aligned}$$

Replacing the second term of (3.6) with the above estimation, and with the assistance of (2.5), we obtain that

$$\mathbf{I} \le \frac{1}{2\Delta \xi} \left(\|U^n\|_n^2 + \|U^{n+1}\|_{n+1}^2 - \|U^{n+1} - U^n\|_n^2 \right) + C \frac{\Delta t_{n+1}}{\Delta \xi} \|U^{n+1}\|_{n+1}^2. \tag{3.7}$$

Using (2.1) and (2.5),

$$\mathbf{II} = \frac{\Delta t_{n+1}}{\Delta\xi}\sum_{j=1}^{N-1}\left(\dot{x}_{j-1/2}^{n+1} - \dot{x}_{j+1/2}^{n+1}\right)(U_j^{n+1})^2 \le C\frac{\Delta t_{n+1}}{\Delta\xi}||U^{n+1}||_{n+1}^2. \tag{3.8}$$

Using the same arguments as [14, (3.12)], we have

$$\mathbf{III} = -\frac{\kappa\Delta t_{n+1}}{\Delta\xi}|||D_+U^{n+1}|||_{n+1}^2. \tag{3.9}$$

Since $U_0^{n+1} = U_N^{n+1} \equiv 0$, we derive that

$$\begin{aligned}
\mathbf{IV} &= \frac{\Delta t_{n+1}}{2\Delta\xi}\sum_{j=1}^{N-1}\left[((\dot{x}^h)_j^{n+1} - a)U_{j+1}^{n+1}U_j^{n+1} - ((\dot{x}^h)_j^{n+1} - a)U_j^{n+1}U_{j-1}^{n+1}\right] \\
&= \frac{\Delta t_{n+1}}{2\Delta\xi}\left(\sum_{j=1}^{N-1}\left[((\dot{x}^h)_j^{n+1} - a)U_{j+1}^{n+1}U_j^{n+1}\right] - \sum_{j=1}^{N-2}\left[((\dot{x}^h)_{j+1}^{n+1} - a)U_{j+1}^{n+1}U_j^{n+1}\right]\right) \\
&= -\frac{\Delta t_{n+1}}{2\Delta\xi}\sum_{j=1}^{N-2}\left[(\dot{x}^h)_{j+1}^{n+1} - (\dot{x}^h)_j^{n+1}\right]U_{j+1}^{n+1}U_j^{n+1} \\
&\le \frac{\Delta t_{n+1}}{4\Delta\xi}\sum_{j=1}^{N-2}\left|(\dot{x}^h)_{j+1}^{n+1} - (\dot{x}^h)_j^{n+1}\right|((U_{j+1}^{n+1})^2 + (U_j^{n+1})^2) \\
&\le C\frac{\Delta t_{n+1}}{\Delta\xi}\|U^{n+1}\|_{n+1}^2,
\end{aligned} \tag{3.10}$$

where the last inequality is obtained by (2.6). Now we estimate term $\mathbf{V}$, mainly using the geometric inequality, as follows:

$$\begin{aligned}
\mathbf{V} \le&\frac{\Delta t_{n+1}}{\Delta\xi}\sum_{j=1}^{N-1}|U_j^{n+1}|\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}\left(\int_0^{t_{n+1}}|K(t_{n+1},s)|\,|\widetilde{U}(x,s)|ds\right)dx \\
=&\frac{\Delta t_{n+1}}{\Delta\xi}\sum_{j=1}^{N-1}\int_0^{t_{n+1}}|K(t_{n+1},s)|\left(\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}|U_j^{n+1}|\,|\widetilde{U}(x,s)|dx\right)ds \\
\le&\frac{\Delta t_{n+1}}{2\Delta\xi}\sum_{j=1}^{N-1}\int_0^{t_{n+1}}|K(t_{n+1},s)|\left(\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}(U_j^{n+1})^2dx + \int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}(\widetilde{U}(x,s))^2dx\right)ds \\
=&\frac{\Delta t_{n+1}}{2\Delta\xi}\int_0^{t_{n+1}}|K(t_{n+1},s)|\left(\sum_{j=1}^{N-1}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}(U_j^{n+1})^2dx + \sum_{j=1}^{N-1}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}}(\widetilde{U}(x,s))^2dx\right)ds \\
\le&\frac{\Delta t_{n+1}}{2\Delta\xi}\int_0^{t_{n+1}}|K(t_{n+1},s)|\,\|U^{n+1}\|_{n+1}^2ds \\
&+\frac{\Delta t_{n+1}}{2\Delta\xi}\sum_{k=0}^{n}\int_{t_k}^{t_{k+1}}|K(t_{n+1},s)|\left(\int_{x_L}^{x_R}(\widetilde{U}(x,s))^2dx\right)ds.
\end{aligned} \tag{3.11}$$

Since $\phi_i^k \phi_j^k = 0$ for $|i-j| \geq 2$, we derive that

$$(\widetilde{U}(x,s))^2 = \left(\sum_{j=1}^{N-1} U_j^k \phi_j^k\right)^2 = \sum_{j=1}^{N-1} (U_j^k \phi_j^k)^2 + 2\sum_{j=1}^{N-2} (U_j^k \phi_j^k)(U_{j+1}^k \phi_{j+1}^k)$$
$$\leq 3\sum_{j=1}^{N-1} (U_j^k \phi_j^k)^2 \leq 3\sum_{j=1}^{N-1} (U_j^k)^2 |\phi_j^k|.$$

Therefore,

$$\mathbf{V} \leq \frac{\Delta t_{n+1}}{2\Delta\xi} \int_0^{t_{n+1}} |K(t_{n+1},s)| ds \, \|U^{n+1}\|_{n+1}^2$$
$$+ \frac{3\Delta t_{n+1}}{2\Delta\xi} \sum_{k=0}^{n} \int_{t_k}^{t_{k+1}} |K(t_{n+1},s)| \left(\sum_{j=1}^{N-1} (U_j^k)^2 \int_{x_L}^{x_R} |\phi_j^k| dx\right) ds.$$

Finally by equality

$$\int_{x_L}^{x_R} |\phi_j^k| \, dx = \frac{h_j^k + h_{j+1}^k}{2},$$

we obtain

$$\mathbf{V} \leq \frac{\Delta t_{n+1}}{2\Delta\xi} \int_0^{t_{n+1}} |K(t_{n+1},s)| ds ||U^{n+1}||_{n+1}^2$$
$$+ \frac{3\Delta t_{n+1}}{2\Delta\xi} \sum_{k=0}^{n} \int_{t_k}^{t_{k+1}} |K(t_{n+1},s)| ds ||U^k||_k^2. \tag{3.12}$$

Combining the estimations for terms $\mathbf{I}, \mathbf{II}, \mathbf{III}, \mathbf{IV}, \mathbf{V}$, i.e., (3.7), (3.8), (3.9), (3.10), (3.12) into (3.5) gives that

$$\|U^{n+1}\|_{n+1}^2 \leq \left(\|U^n\|_n^2 - \|U^{n+1} - U^n\|_n^2\right) - 2\kappa \Delta t_{n+1} |||D_+ U^{n+1}|||_{n+1}^2$$
$$+ \; C\Delta t_{n+1} \|U^{n+1}\|_{n+1}^2 + \Delta t_{n+1} \int_0^{t_{n+1}} |K(t_{n+1},s)| ds \|U^{n+1}\|_{n+1}^2$$
$$+ \; 3\Delta t_{n+1} \sum_{k=0}^{n} \int_{t_k}^{t_{k+1}} |K(t_{n+1},s)| ds \|U^k\|_k^2. \tag{3.13}$$

Hence,

$$\|U^{\ell+1}\|_{\ell+1}^2 \leq \|U^\ell\|_\ell^2 + C\Delta t_{\ell+1} \Big[1 + \int_0^{t_{\ell+1}} |K(t_{\ell+1},s)| ds\Big] \|U^{\ell+1}\|_{\ell+1}^2$$
$$+ 3\Delta t_{\ell+1} \sum_{k=0}^{\ell} \int_{t_k}^{t_{k+1}} |K(t_{\ell+1},s)| ds \, \|U^k\|_k^2. \tag{3.14}$$

Summing (3.14) for $\ell = 0, 1, \ldots, n$ leads to

$$\|U^{n+1}\|_{n+1}^2 \leq \|U^0\|_0^2 + C\sum_{\ell=0}^{n} \Delta t_{\ell+1} \left[1 + \int_0^{t_{\ell+1}} |K(t_{\ell+1},s)| \, ds\right] \|U^{\ell+1}\|_{\ell+1}^2$$
$$+ 3\sum_{\ell=0}^{n} \Delta t_{\ell+1} \sum_{k=0}^{\ell} \int_{t_k}^{t_{k+1}} |K(t_{\ell+1},s) \, ds \, \|U^k\|_k^2.$$

Then

$$\|U^{n+1}\|_{n+1}^2 \leq \frac{\|U^0\|_0^2}{1-C\Delta t_{n+1}} + \frac{C}{1-C\Delta t_{n+1}} \left[\sum_{\ell=1}^{n} \Delta t_\ell \|U^\ell\|_\ell^2 + \sum_{\ell=0}^{n} \Delta t_{\ell+1} \sum_{k=0}^{\ell} \int_{t_k}^{t_{k+1}} |K(t_{\ell+1},s)|\, ds \|U^k\|_k^2 \right].$$

Applying standard discrete Gronwall inequality (or the variant version [18, Lemma 6.4] for weakly singular kernel $K = t^{-\alpha},\ 0 < \alpha < 1$), for sufficiently small temporal mesh size, we obtain the desired result. □

Now we establish a convergence result for the fully discrete scheme.

Theorem 3.2. *When the scheme* (2.11) *is applied to solve problem* (1.1)-(1.3), *under assumptions* (A1), (A2) *and* (A3), *the error has a bound*

$$\|e^n\|_n \leq C \max(\tau, N^{-2}), \tag{3.15}$$

where $\tau \equiv \max_{(n)} \Delta t_n$.

Proof. Use $x = x^h(\xi,t)$ ($(\xi,t) \in [\xi_{j-1/2}, \xi_{j+1/2}] \times (t_n, t_{n+1}]$) to recast the original equation (1.1) into

$$\dot{\left(x_\xi^h u\right)} - \dot{x}_\xi^h u - (\dot{x}^h - a)u_\xi - \left(\kappa \frac{u_\xi}{x_\xi^h}\right)_\xi + x_\xi^h \int_0^t k(x^h,t,s)u(x^h,s)\, ds = 0. \tag{3.16}$$

Define truncation error T_j^{n+1} to satisfy

$$\begin{aligned}(x_\xi^h u)_j^{n+1} =& (x_\xi^h u)_j^n + \Delta t_{n+1} (\dot{x}_\xi^h)_j^{n+1} u_j^{n+1} \\ &+ \frac{\Delta t_{n+1}}{\Delta \xi} \left((\kappa(D_+ - D_-)u)_j^{n+1} + ((\dot{x}^h)_j^{n+1} - a) \left((\delta u)_{j+1/2}^{n+1} - (\delta u)_{j-1/2}^{n+1} \right) \right) \\ &- \frac{\Delta t_{n+1}}{\Delta \xi} \int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}} \left(\int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s) \widetilde{u}(x,s) ds \right) dx + \Delta t_{n+1} T_j^{n+1}, \end{aligned} \tag{3.17}$$

where $u_j^n = u(x_j^n, t_n)$ and

$$\widetilde{u}(x,t) \equiv \sum_{j=0}^{N-1} u_j^k \phi_j^k, \quad t \in [t_k, t_{k+1}), \quad \text{for all } k \leq n. \tag{3.18}$$

Since $(x^h)_j^{n+1} \equiv x^h(\xi_j, t_{n+1}) = x(\xi_j, t_{n+1}) \equiv x_j^{n+1}$, we know that $u_j^n = u(x_j^{n+1}, t_{n+1}) = u((x^h)_j^{n+1}, t_{n+1})$. Hence we have that

$$T_j^{n+1} = \textbf{(i)} + \textbf{(ii)} + \textbf{(iii)} + \textbf{(iv)}, \tag{3.19}$$

where

$$
\begin{aligned}
(\mathbf{i}) &= \frac{(x_\xi^h)_j^{n+1}u_j^{n+1} - (x_\xi^h)_j^n u_j^n}{\Delta t_{n+1}} - \dot{(x_\xi^h u)}_j^{n+1},\\
(\mathbf{ii}) &= \left(\kappa \frac{u_\xi}{x_\xi^h}\right)_\xi - \kappa \frac{D_+ u_j^{n+1} - D_- u_j^{n+1}}{\Delta\xi},\\
(\mathbf{iii}) &= \frac{1}{\Delta\xi}\Big(((\dot{x}^h)_j^{n+1} - a)((\delta u)_{j+1/2}^{n+1} - (\delta u)_{j-1/2}^{n+1})\Big) - ((\dot{x}^h - a)u_\xi),\\
(\mathbf{iv}) &= \frac{1}{\Delta\xi}\int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}} \left(\int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s)\widetilde{u}(x,s)ds\right)dx\\
&\quad - (x_\xi^h)_j^{n+1}\int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s)u(x_j^{n+1}, s)ds.
\end{aligned}
$$

Here and in the following, all functions are evaluated at (x_j^{n+1}, t_{n+1}). We estimate (i)-(iv) term by term. Let $r_{j,n+1} = x_j^{n+1} - x_j^n$. Then from (2.8), we have that $r_{j,n+1} = (\dot{x}^h)_j^{n+1}\Delta t_{n+1}$. Using Taylor theorem, we derive that

$$
\begin{aligned}
u_j^n =& u(x_j^n, t_n) = u(x_j^{n+1}, t_{n+1}) - u_x(x_j^{n+1}, t_{n+1})(x_j^{n+1} - x_j^n)\\
& - u_t(x_j^{n+1}, t_{n+1})(t_{n+1} - t_n) + \frac{1}{2}u_{xx}(x_j^{n+1}, t_{n+1})(x_j^{n+1} - x_j^n)^2\\
& + u_{xt}(x_j^{n+1}, t_{n+1})(x_j^{n+1} - x_j^n)(t_{n+1} - t_n) + \frac{1}{2}u_{tt}(x_j^{n+1}, t_{n+1})(t_{n+1} - t_n)^2\\
& + o\left(r_{j,n+1}^2 + r_{j,n+1}\Delta t_n + (\Delta t_n)^2\right).
\end{aligned}
$$

Apply fundamental theorem of calculus to obtain

$$
\begin{aligned}
&\dot{(x_\xi^h u)}|_{(x_j^{n+1}, t_{n+1})}\\
=&(\dot{x}_\xi^h)_j^{n+1}u(x_j^{n+1}, t_{n+1}) + (x_\xi^h)_j^{n+1}u_x(x_j^{n+1}, t_{n+1})\dot{x}^h(\xi_j, t_{n+1}) + (x_\xi^h)_j^{n+1}u_t(x_j^{n+1}, t_{n+1})
\end{aligned}
$$

Inserting these estimates into the expression of $(\mathbf{i})$ and using (2.7), we derive that

$$
\begin{aligned}
(\mathbf{i}) =& -\left((x_\xi^h)_j^{n+1} - (x_\xi^h)_j^n\right)u_x(x_j^{n+1}, t_{n+1})\frac{r_{j,n+1}}{\Delta t_{n+1}} - \left((x_\xi^h)_j^{n+1} - (x_\xi^h)_j^n)\right)u_t(x_j^{n+1}, t_{n+1})\\
& - \frac{(x_\xi^h)_j^n}{2}\left(u_{xx}(x_j^{n+1}, t_{n+1})\frac{r_{j,n+1}^2}{\Delta t_{n+1}} + 2u_{xt}(x_j^{n+1}, t_{n+1})r_{j,n+1} + u_{tt}(x_j^{n+1}, t_{n+1})\Delta t_{n+1}\right)\\
& + \frac{(x_\xi^h)_j^n}{\Delta t_{n+1}}\, o\left(r_{j,n+1}^2 + r_{j,n+1}\Delta t_n + (\Delta t_n)^2\right)\\
=& -\frac{(h_{j+1}^{n+1} + h_j^{n+1}) - (h_j^n + h_{j+1}^n)}{2\Delta\xi}\left(u_x\frac{r_{j,n+1}}{\Delta t_{n+1}} + u_t\right)\\
& - \frac{h_{j+1}^n + h_j^n}{4\Delta\xi}\left(u_{xx}\frac{r_{j,n+1}^2}{\Delta t_{n+1}} + 2u_{xt}r_{j,n+1} + u_{tt}\Delta t_{n+1}\right)\\
& - \frac{h_{j+1}^n + h_j^n}{2\Delta\xi}\left(\frac{o(r_{j,n+1}^2)}{\Delta t_{n+1}} + o(r_{j,n+1}) + o(\Delta t_{n+1})\right).
\end{aligned} \tag{3.20}
$$

It follows from Taylor theorem that

$$(\mathbf{ii}) = \frac{\kappa}{6} u_{xxx} \frac{(h_{j+1}^{n+1} + h_j^{n+1})(h_{j+1}^{n+1} - h_j^{n+1})}{\Delta\xi} + \frac{\mathcal{O}((h_{j+1}^{n+1})^3) + \mathcal{O}((h_j^{n+1})^3)}{\Delta\xi}, \tag{3.21}$$

and also from (2.8) that

$$(\mathbf{iii}) = \frac{h_{j+1}^{n+1} + h_j^{n+1}}{4\Delta\xi} \Big(u_{xx}((\dot{x}^h)_j^{n+1} - a)(h_{j+1}^{n+1} - h_j^{n+1}) \Big) + \frac{\mathcal{O}((h_{j+1}^{n+1})^3 + (h_j^{n+1})^3)}{\Delta\xi}. \tag{3.22}$$

By (2.7), we have

$$\begin{aligned}(\mathbf{iv}) = &\frac{h_{j+1}^{n+1} + h_j^{n+1}}{2\Delta\xi} \int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s) \\ &\times \left(\frac{2}{h_{j+1}^{n+1} + h_j^{n+1}} \int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}} (\widetilde{u}(x,s) - u(x_j^{n+1}, s)) dx \right) ds. \end{aligned} \tag{3.23}$$

Let $e_j^n \equiv u(x_j^n, t_n) - U_j^n, \quad j = 1, \ldots, N-1; \ n = 0, 1, \ldots, L$. Then subtract (2.11) from (3.17) to give

$$\begin{aligned} &(x_\xi^h)_j^{n+1} e_j^{n+1} \\ =&(x_\xi^h)_j^n e_j^n + \frac{\Delta t_{n+1}}{\Delta\xi} \Big((\kappa(D_+ - D_-)e)_j^{n+1} + ((\dot{x}^h)_{j+1/2}^{n+1} - a)((\delta e)_{j+1/2}^{n+1} - (\delta e)_{j-1/2}^{n+1}) \Big) \\ &- \frac{\Delta t_{n+1}}{\Delta\xi} \int_{x_{j-1/2}^{n+1}}^{x_{j+1/2}^{n+1}} \left(\int_0^{t_{n+1}} k(x_j^{n+1}, t_{n+1}, s) E(x,s) ds \right) dx + \Delta t_{n+1} T_j^{n+1}, \end{aligned} \tag{3.24}$$

where

$$E(x,t) = \sum_{j=0}^{N-1} e_j^k \phi_j^k, \quad t \in [t_k, t_{k+1}), \quad k = 0, 1, \ldots, n. \tag{3.25}$$

Multiplying e_j^{n+1} on both sides of (3.24), summing over all interior nodes, and using the same technique as in Theorem 3.1 and geometric inequality to treat the last term, we obtain

$$\begin{aligned} ||e^{n+1}||_{n+1}^2 \le &||e^n||_n^2 + \Delta t_{n+1} \int_0^{t_{n+1}} |K(t_{n+1}, s)| ds ||e^{n+1}||_{n+1}^2 \\ &+ \ 3\Delta t_{n+1} \sum_{k=0}^{n} \int_{t_k}^{t_{k+1}} |K(t_{n+1}, s)| ds ||e^k||_k^2 + \Delta t_{n+1} ||e^{n+1}||_{n+1}^2 \\ &+ \ \Delta t_{n+1} \sum_{j=1}^{N-1} \frac{(\Delta\xi)^2}{2(h_{j+1}^{n+1} + h_j^{n+1})} \left(T_j^{n+1} \right)^2. \end{aligned} \tag{3.26}$$

This estimation is similar to (3.13). Thus similarly using Gronwall inequality we arrive at

$$\|e^{n+1}\|_{n+1}^2 \le C \left[\|e^0\|_0^2 + \sum_{\ell=0}^{n} \Delta t_{\ell+1} \sum_{j=1}^{N-1} (h_{j+1}^{\ell+1} + h_j^{\ell+1}) \left(\frac{\Delta\xi}{h_{j+1}^{\ell+1} + h_j^{\ell+1}} T_j^{\ell+1} \right)^2 \right]. \tag{3.27}$$

Recall the equi-distribution principle, for $j = 0, 1, \ldots, N$ and $n = 0, 1, \ldots, L$:

$$\int_{x_j^n}^{x_{j+1}^n} M(x,t)\, dx = \frac{1}{N} \int_{x_L}^{x_R} M(x,t)\, dx.$$

Since monitor functions $M > 1$, for suitable smooth monitor function such that

$$\int_{x_L}^{x_R} M(x,t)\,dx \le C,$$

we have

$$\begin{aligned} h_j^n \equiv x_{j+1}^n - x_j^n \le& \int_{x_j^n}^{x_{j+1}^n} M(x,t)\,dx \\ =& \frac{1}{N}\int_{x_L}^{x_R} M(x,t)\,dx \le CN^{-1}. \end{aligned} \tag{3.28}$$

Using the estimations (3.20)-(3.23) in (3.27), and applying assumptions (A1)-(A3), and estimation (3.28), we obtain that

$$\begin{aligned} &\sum_{\ell=0}^{n} \Delta t_{\ell+1} \sum_{j=1}^{N-1} (h_{j+1}^{\ell+1} + h_j^{\ell+1}) \left(\frac{\Delta\xi}{h_{j+1}^{\ell+1} + h_j^{\ell+1}} T_j^{\ell+1} \right)^2 \\ \le& C(\tau + N^{-2})^2 \le C\left[\max(\tau, N^{-2})\right]^2. \end{aligned}$$

Incorporating this estimation into (3.27) and using the fact that the initial error can be made as accurate as

$$\|e^0\|_0^2 \le C\left[\max(\tau, N^{-2})\right]^2,$$

we complete the proof of this theorem. □

4. Numerical Experiments

We use the following example to test the analysis in the previous sections:

$$u_t + au_x - \kappa u_{xx} + \int_0^t k(x,t,s)u(x,s)\,ds = f(x,t), \quad (x,t) \in [0,1]\times[0,2], \tag{4.1}$$

where $a = 0.5$, and

$$\begin{aligned} k(x,t,s) =& t\left[1 - \tanh^2(-\frac{x-0.5s}{\sqrt{\kappa}})\right], \\ f(x,t) =& 2\left[1 - \tanh^2(-\frac{x-0.5t}{\sqrt{\kappa}})\right]\tanh(-\frac{x-0.5t}{\sqrt{\kappa}}) \\ &+ \sqrt{\kappa}t\tanh^2(-\frac{x-0.5t}{\sqrt{\kappa}}) - \sqrt{\kappa}t\tanh^2\frac{x}{\sqrt{\kappa}}, \end{aligned}$$

with boundary conditions

$$\begin{aligned} u(0,t) =& b_L(t) = \tanh\left(\frac{0.5t}{\sqrt{\kappa}}\right), \\ u(1,t) =& b_R(t) = \tanh\left(\frac{-1+0.5t}{\sqrt{\kappa}}\right) \end{aligned}$$

and initial condition $u(x,0) = \tanh(\frac{-x}{\sqrt{\kappa}})$. The exact solution is given by $u = \tanh(\frac{-x+0.5t}{\sqrt{\kappa}})$.

Table 4.1: Rate for space using exact solution for adaptation of mesh

N	L	$\max_n \|e^n\|_\infty$	RS	$\max_n \|e^n\|_n$	RS
5	10240	5.8337×10^{-1}	N/A	2.5875×10^{-1}	N/A
10	10240	1.7247×10^{-1}	1.76	6.6467×10^{-2}	1.96
20	10240	4.1414×10^{-2}	2.06	1.6400×10^{-2}	2.01
40	10240	1.0250×10^{-2}	2.01	4.0544×10^{-3}	2.01
80	10240	2.5546×10^{-3}	2.00	9.9464×10^{-4}	2.02

Table 4.2: Rate for time using exact solution for adaptation of mesh

N	L	$\max_n \|e^n\|_\infty$	RT	$\max_n \|e^n\|_n$	RT
5	10	3.3067×10^{-1}	N/A	1.7117×10^{-1}	N/A
10	40	1.2805×10^{-1}	0.68	6.0434×10^{-2}	0.75
20	160	3.3246×10^{-2}	0.97	1.6131×10^{-2}	0.95
40	640	8.4293×10^{-3}	0.99	4.0684×10^{-3}	0.99
80	2560	2.1204×10^{-3}	1.00	1.0189×10^{-3}	0.99

We first consider $\kappa = 0.01$. The exact solution is used in the monitor function to adapt the mesh. The temporal mesh is taken to be uniform. The spatial mesh at each time level t_n is obtained by equidistributing a smoothed monitor function $\widetilde{M}(x, t_n)$ which is given by

$$\begin{aligned}\widetilde{M}(x,t_n) =& 0.1M(x-0.1,t_n) + 0.2M(x-0.05,t_n)\\ &+ 0.4M(x,t_n) + 0.2M(x+0.05,t_n) + 0.1M(x+0.1,t_n),\end{aligned}$$

where $M(x, t_n) = 1 + |u_x(x, t_n)|$.

Let N denote the number of the spacial mesh intervals and L the number of the temporal mesh intervals. Define the rate for space by $\text{RS} = \log_2 (\text{Error}(N_j)/\text{Error}(N_{j-1}))/\log_2 (N_{j-1}/N_j)$, and the rate for time by $\text{RT} = \log_2 (\text{Error}(L_j)/\text{Error}(L_{j-1})) / \log_2 (L_{j-1}/L_j)$. The numerics in Table 4.1 and Table 4.2 confirm that the convergence rate for space is $\mathcal{O}(N^{-2})$ and the rate for time is $\mathcal{O}(\tau)$ where $\tau = 1/L$. Also the error versus time is plotted in Fig. 4.1.

Now we consider $\kappa = 0.0005$ and use computational solution to adapt the mesh. The node-based smoothing technique, see e.g., [7], is adopted in the computation. The convergence rate

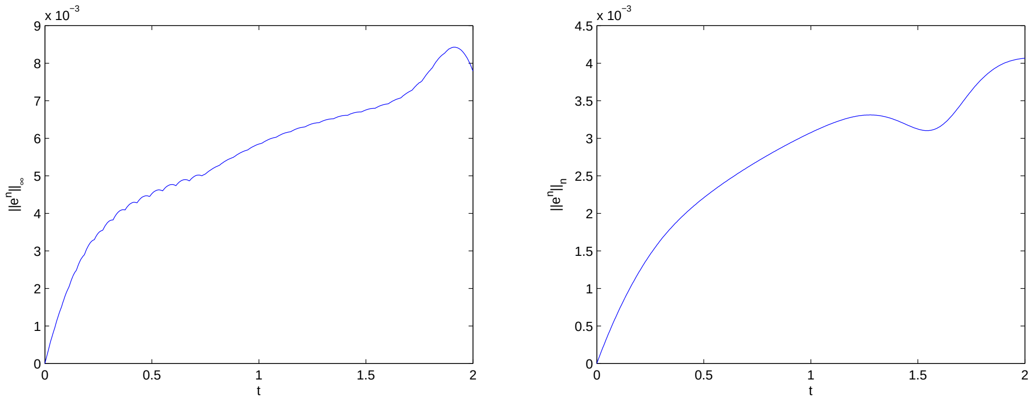

Fig. 4.1. Error v.s. time with $N = 40, L = 640$

Table 4.3: Rate for space using computational solution for adaptation of mesh

N	L	$\max_n \|\|e^n\|\|_\infty$	RS	RT	$\max_n \|\|e^n\|\|_n$	RS	RT
15	200	1.9794×10^{-1}	N/A	N/A	2.5781×10^{-2}	N/A	N/A
30	800	3.6415×10^{-2}	2.44	1.22	8.6567×10^{-3}	1.57	0.79
60	3200	7.9080×10^{-3}	2.20	1.10	2.4782×10^{-3}	1.80	0.90
120	12800	1.9447×10^{-3}	2.02	1.01	6.4929×10^{-4}	1.93	0.97

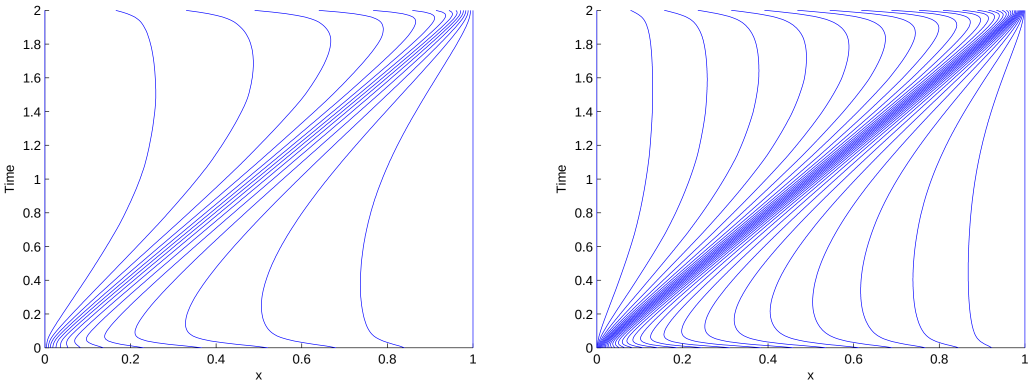

Fig. 4.2. Mesh trajectories: $N = 15, L = 200$ (left), $N = 30, L = 800$ (right)

for space as shown in Table 4.3 is pretty close to $\mathcal{O}(N^{-2})$, and the rate for time is close to $\mathcal{O}(\tau)$. The mesh trajectories are plotted in Fig. 4.2.

5. Concluding Remarks

In this paper, we construct a stable moving finite difference scheme for a class of linear partial integro-differential equations. We prove that the convergence is first order in time and second order in space. Although our proof is for linear equations, there is no essential difficulty to extend the analysis to a more general case with smooth variable coefficients or nonlinear terms with nonlinear function satisfying the Lipschitz conditions. Note that when $k \equiv 0$, the equation (1.1) is reduced to a partial differential equation. Hence the result is also true for partial differential equations. Mackenzie and Mekwi [14] use a piecewise linear approximation of the time-dependent mapping $x(\xi, t)$ for ξ in each time slice $[t_n, t_{n+1}]$ and only get asymptotically second-order convergence. While we use a piecewise quadratic approximation of the mapping to obtain a second order convergence. An application of the moving mesh method proposed in this paper to the simulation of blowup in reaction-diffusion equations with nonlocal terms is given in [12].

Acknowledgements. The authors are grateful to the referees for the very important comments and suggestions. The work was partly supported by SRF for ROCS, SEM. This research was also supported by a grant from the "project 211 (phase III)" of the Southwestern University of Finance and Economics.

References

[1] R.E. Bank and R.F. Santos, Analysis of some moving space-time finite element methods, *SIAM J. Numer. Anal.*, **30** (1993), 1-18.

[2] J.G. Blom, J.M. Sanz-Serna and J.G. Verwer, On simple moving grid methods for one-dimensional evolutionary partial differential equations, *J. Comput. Phys.*, **74** (1988), 191-213.

[3] H. Brunner, Collocation Methods for Volterra Integral and Related Functional Differential Equations, Cambridge University Press, 337 Cambridge, 2004.

[4] W. Cao, W.-Z. Huang and R. D. Russell, A moving mesh method based on the geometric conservation law, *SIAM J. Sci. Comput.*, **24** (2002), 118-142.

[5] E.A. Dorfi and L. Drury, simple adaptive grids for 1-d initial value problems, *J. Comput. Phys.*, **69** (1987), 175-195.

[6] J.A. Ferreira, On the convergence on nonrectangular grids, *J. Comput. Appl. Math.*, **85** (1997), 333-344.

[7] W.-Z. Huang and R.D. Russell, Analysis of moving mesh partial differential equations with spatial smoothing, *SIAM J. Numer. Anal.*, **34** (1997), 1106-1126.

[8] W.-Z. Huang and R.D. Russell, Adaptive mesh movement - the MMPDE approach and its applications, *J. Comput. Appl. Math.*, **128** (2001), 383-398.

[9] P. Jamet, Stability and convergence of a generalized Crank-Nicolson scheme on a variable mesh for the heat equation, *SIAM J. Numer. Anal.*, **17** (1980), 530-539.

[10] R. Li, T. Tang, and P.-W. Zhang, Moving mesh methods in multiple dimensions based on harmonic maps, *J. Comput. Phys.*, **170** (2001), 562-588.

[11] K. Lipnikov and M. Shashkov, The error-minimization-based strategy for moving mesh methods, *Commun. Comput. Phys.*, **1** (2006), 53-80.

[12] J. Ma, Y. Jiang and K. Xiang, Numerical simulation of blowup in nonlocal reaction-diffusion equations using a moving mesh method, *J. Comput. Appl. Math.*, to appear.

[13] J. Ma, Convergence analysis of moving Godunov schemes for dynamical boundary layers, *J. Comput. Appl. Math.*, to appear.

[14] J.A. Mackenzie and W. R. Mekwi, An analysis of stability and convergence of a finite-difference discretization of a model parabolic PDE in 1D using a moving mesh, *IMA J. Numer. Anal.*, **27** (2007), 507-528.

[15] H.-Z. Tang and T. Tang, Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws, *SIAM J. Numer. Anal.*, **41** (2003), 487-515.

[16] T. Tang, Moving mesh methods for computational fluid dynamics, Recent Advances in Adaptive Computation (Z. Shi, Z. Chen, T. Tang, and D. Yu, eds.), Contemp. Math., vol. 383, pp. 141-173, American Mathematical Society, 2005.

[17] T. Tang and J. Xu, Adaptive Computations: Theory and Algorithms, Science Press, Beijing, 2007.

[18] S. Larsson, V. Thomée and L. Wahlbin, Numerical solution of parabolic integro-differential equations by the discontinuous Galerkin method, *Math. Comput.*, **67** (1998), 45-71.

[19] E. G. Yanik and G. Fairweather, Finite element methods for parabolic partial integro-differential equations, *Nonlinear Anal.*, **12** (1988), 785-809.