Research Article

Finite Element Method of BBM-Burgers Equation with Dissipative Term Based on Adaptive Moving Mesh

Changna Lu,[1] Qianqian Gao,[1] Chen Fu,[1] and Hongwei Yang[2,3]

[1]*School of Mathematics and Statistics, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[2]*College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao 266590, China*
[3]*Key Laboratory of Meteorological Disaster, Ministry of Education, Nanjing University of Information Science and Technology, Nanjing 210044, China*

Correspondence should be addressed to Hongwei Yang; hwyang1979@163.com

A finite element model is proposed for the Benjamin-Bona-Mahony-Burgers (BBM-Burgers) equation with a high-order dissipative term; the scheme is based on adaptive moving meshes. The model can be applied to the equations with spatial-time mixed derivatives and high-order derivative terms. In this scheme, new variables are needed to make the equation become a coupled system, and then the linear finite element method is used to discretize the spatial derivative and the fifth-order Radau IIA method is used to discretize the time derivative. The simulations of 1D and 2D BBM-Burgers equations with high-order dissipative terms are presented in numerical examples. The numerical results show that the method keeps a second-order convergence in space and provides a smaller error than that based on the fixed mesh, which demonstrates the effectiveness and feasibility of the finite element method based on the moving mesh. We also study the effect of the dissipative terms with different coefficients in the equation; by numerical simulations, we find that the dissipative term u_{xx} plays a more important role than u_{xxxx} in dissipation.

1. Introduction

In mathematics, physics, and engineering, nonlinear partial differential equations play an important role due to their rich mathematical structures and features [1–4] as well as important applications in fluid dynamics, plasma physics, fiber optics, condensed matter physics, and chemistry [5–11]. It is important to search for solutions. These nonlinear partially differential equations can be solved exactly or numerically to understand the properties of solutions.

In this paper, we consider the nonlinear multidimensional Benjamin-Bona-Mahony-Burgers (BBM-Burgers) equation with a dissipative term and use the finite element method based on the moving mesh to solve its numerical solution. The equation has the following form:

$$
u_t + \sum_{i=1}^{d} \alpha_i u_{x_i} + \sum_{i=1}^{d} \beta_i u u_{x_i} - \gamma \Delta u - \mu \Delta u_t
$$
$$
+ \sum_{i=1}^{d} \sum_{n=1}^{p} \delta_n \frac{\partial^{2n+2} u}{\partial x_i^{2n+2}} = F, \quad \mathbf{X} \in \Omega, \ t \in (0, T]
$$

$$
u(\mathbf{X}, t) = g(\mathbf{X}, t), \quad \mathbf{X} \in \partial\Omega, \ t \in (0, T]
$$
$$
u(\mathbf{X}, 0) = u_0(\mathbf{X}), \quad \mathbf{X} \in \Omega,
$$

$$(1)$$

where $\mathbf{X} = (x_1, x_2, \ldots, x_d)$, d stands for the dimension, x_i stands for the corresponding spatial vector, the coefficients $\alpha_i, \beta_i \in R$, $\gamma, \mu, \delta_n > 0$, $\Omega \subseteq R^n$, Δ is a Laplace operator, and p is one constant to define the order of the dissipative term. In the equation, F is a sufficiently smooth function and stands for the source term, and $g(\mathbf{X})$ and $u_0(\mathbf{X})$ are given functions.

Equation (1) is related to the well-known Benjamin-Bona-Mahony (BBM) equation, which was advocated by Benjamin et al. [12] in 1972 as a refinement of the Korteweg-de-Vries (KdV) equation. There are several different transformations of this equation; when the coefficients $\mu = \delta_n = 0$, $F = 0$, the equation becomes $u_t + \sum_{i=1}^{d} \alpha_i u_{x_i} + \sum_{i=1}^{d} \beta_i u u_{x_i} = 0$, which is the Burgers equation with a nonlinear term. When parameters $\gamma, \delta_n = 0$, (1) becomes $u_t + \sum_{i=1}^{d} \alpha_i u_{x_i} + \sum_{i=1}^{d} \beta_i u u_{x_i} - \mu \Delta u_t = F$, which is called Regularized Long Wave (RLW)

equation, also called BBM equation. The RLW equation is usually used to analyze small amplitude long waves on the surface of water in fluid dynamics. When $\delta_n = 0$, (1) becomes a generalized BBM-Burgers equation, which is a mathematical model of propagation of small-amplitude long wave in nonlinear dispersive media and better than Korteweg-de-Vries-Burgers (KdV-B) equation in fluid mechanics; when $F = 0$, $d = 1$, and $g(x) = 0$, the BBM-Burgers equation satisfies two conservation laws:

$$\text{Mass function } E_1(t) = \int_\Omega u \, dx,$$

$$\text{Energy function } E_2(t) \tag{2}$$

$$= \int_\Omega \left(\frac{1}{2} u^2 + \frac{\mu}{2} (u_x)^2 \right) dx + \alpha \int_0^t \int_\Omega u_x{}^2 (\varsigma) \, dx \, d\varsigma.$$

If $d = 2$, $p = 1$, (1) becomes

$$u_t + \alpha_1 u_x + \alpha_2 u_y + \beta_1 u u_x + \beta_2 u u_y - \gamma \Delta u - \mu \Delta u_t$$

$$+ \delta \left(u_{xxxx} + u_{yyyy} \right) = F, \quad (x, y) \in \Omega, \ t \in (0, T]$$

$$u(x, y, t) = g(x, y, t), \quad (x, y) \in \partial\Omega, \ t \in (0, T] \tag{3}$$

$$u(x, y, 0) = u_0(x, y), \quad (x, y) \in \Omega.$$

In physical phenomena, the effect of dispersion in the BBM-Burgers equation with a high-order dissipative term is similar to the RLW equation; the effect of dissipation is similar to the Burgers equation.

To get the approximate solution of the BBM-Burgers equation with a high-order dissipative term, some numerical and analytical methods have been proposed in recent years. In the aspect of mathematical theory, Zhao et al. consider the existence of the generalized BBM-Burgers equation with a dissipative term and verify the corresponding convergence in H-measures [13, 14]. Kondo and Webler apply kinetic decomposition for the generalized BBM-Burgers equations with a dissipative term to obtain the approximate transport equation, and then take advantage of the averaging lemma to get the convergence [15]. Seok states the existence and conservation laws of the generalized BBM-Burgers equation with a dissipative term [16]. For more theoretical analyses, refer to the references in [13–16]. In the aspect of numerical solutions, many methods have been used, like Painlev'e test, Darboux Transformation, bilinear method, symmetry method [17], and so on [18–21]. Hong et al. apply the homotopy perturbation method and He's variational iteration method to solve the BBM-Burgers equation. Ganji et al. put forward the notion that the method they used in [22] is better than the homotopy analysis method proposed in [23]. Finite element Galerkin methods have been discussed by Kadri et al. [24], Lee [25], Yin and Piao [26], and Dehghan et al. [27] to solve the BBM-Burgers equations with a dissipative term. Finite difference methods have also been widely used to solve the equations in [28–30]; as for other numerical methods, refer to [31–34].

The references introducing the numerical simulations above are mostly based on fixed meshes; in this paper, we consider the finite element method based on the adaptive moving mesh to solve the BBM-Burgers equation with a high-order dissipative term. The moving mesh method is a kind of adaptive mesh method which adds elements near the place where the numerical solution changes rapidly and decreases elements in the solution changing slowly; the total element number still remains unchanged. When we compare the finite element method based on adaptive moving mesh with methods based on fixed mesh or proposed in [26], we find that the moving mesh method takes fewer elements to get the same distinguishability and simulates steep waves and the transition of gap distinctly. In recent years, the moving mesh method has witnessed further development and extensive application [35]. This method was applied to approximate the anisotropic diffusion-convection-reaction problems to satisfy a discrete maximum principle, and it was adopted to study the porous medium equation and approximate the solution [36]. For more information about the moving mesh method, please refer to [37–41].

The remainder of this paper is divided into three sections. In Section 2, we introduce the method we used, the discretization of the BBM-Burgers equation with a fourth-order dissipative term in one dimension, and the implementation of the moving mesh method. Section 3 presents 1D and 2D numerical examples of BBM-Burgers equations with a dissipative term. Finally, Section 4 shows the conclusion we draw from the numerical examples.

2. Finite Element Method Based on Moving Mesh

In this section, we mainly introduce the linear finite element method based on moving mesh and make use of the method to solve the BBM-Burgers equation with a high-order dissipative term. For convenience, in Section 2.2, we introduce the finite element method of the one-dimensional equation with a fourth-order dissipative term u_{xxxx}; the other parts of this section will focus on two dimensions.

2.1. Adaptive Moving Mesh Method. In this subsection, we mainly want to show the generation of physical mesh $\mathscr{T}_h{}^{n+1}$ at a new time level by an adaptive moving mesh strategy based on the Moving Mesh Partial Differential Equation (MMPDE) [42]; at the current time level, the physical mesh $\mathscr{T}_h{}^n = \{\mathbf{X}_j{}^n\}$ and the numerical solution $(u^h)^n$ are known (the mesh and value will be introduced clearly in the following parts). During the strategy, one of the crucially important rules is to view the nonuniform mesh (i.e., the physical mesh in this paper) as a uniform mesh under the metric tensor $\mathbb{M}$; the tensor $\mathbb{M}$ is a symmetric and positive definite matrix. Under the metric tensor $\mathbb{M}$, the physical mesh $\mathscr{T}_h$ and the computational mesh $\mathscr{T}_c$ can be transformed into each other if and only if it satisfies (the following notations and concepts about the mesh are the same as those in [36])

$$|K| \det \left(\mathbb{M}_K \right)^{1/2} = \frac{|K_c| \, \sigma_h}{|\Omega_c|}, \quad \forall K \in \mathscr{T}_h, \tag{4}$$

$$\frac{1}{2}\mathrm{tr}\left(\left(F_K'\right)^{-1}\mathbb{M}_K^{-1}\left(F_K'\right)^{-T}\right)$$
$$= \det\left(\left(F_K'\right)^{-1}\mathbb{M}_K^{-1}\left(F_K'\right)^{-T}\right)^{1/2}, \quad \forall K \in \mathscr{T}_h, \tag{5}$$

where K is an element of the physical mesh $\mathscr{T}_h$, K_c is an element of the computational mesh $\mathscr{T}_c$ corresponding to K, and $|K|$ and $|K_c|$, respectively, show the volume of the corresponding element. Meanwhile, $|\Omega_c| = \sum_{K_c \in \mathscr{T}_c} |K_c|$, $\det(\cdot)$ denotes the determinant of a matrix, and $\mathrm{tr}(\cdot)$ stands for the trace of a matrix. $\sigma_h = \sum_{K \in \mathscr{T}_h} |K| \det(\mathbb{M}_K)^{1/2}$ represents the total error of the numerical solution, F_K' is the Jacobian matrix of the affine mapping $F_K\colon K_C \to K$, and M_K is the average of the metric tensor over the elements K. In this paper, we take $\mathbb{M} = \det(\alpha_h + |H((u^h)^n)|)^{-1/6}(\alpha_h I + |H((u^h)^n)|)$, where I is an identity matrix and $H(\cdot)$ stands for a Hessian matrix. Equations (4) and (5) also satisfy the equidistribution principle and alignment principle on moving mesh, which means that, under the metric tensor $\mathbb{M}$, each element has the same numerical error and is similar to a reference element $\widehat{K}$; the reference element $\widehat{K}$ is uniform. From (4) and (5), we find that the metric tensor $\mathbb{M}$ plays an important role in the adaptive moving mesh and controls the adaptation.

If we want to get the mesh which satisfies the two principles, we need to minimize the energy function that contains the metric tensor $\mathbb{M}$:

$$I_h\left(\mathscr{T}_h, \mathscr{T}_c\right) = \theta \sum_K |K| \det\left(\mathbb{M}_K\right)^{1/2}$$
$$\cdot \left(\mathrm{tr}\left(\left(F_K'\right)^{-1}\mathbb{M}_K^{-1}\left(F_K'\right)^{-T}\right)\right)^{dp/2} + (1 \tag{6}$$
$$- 2\theta)\, d^{dp/2} \sum_K |K| \det\left(\mathbb{M}_K\right)^{1/2}\left(\det\left(F_K'\right)^{-1}\det\left(\mathbb{M}_K\right)^{-1/2}\right)^p,$$

where θ and p are dimensionless parameters. When we consider the equation to be just one-dimensional, (6) will have a relationship with (4) and $\theta = 0$; when the equation is two-dimensional, parameters θ, p are usually defined as $1/3$ and 2, and d stands for the dimension in space. The energy function I_h is also a function about the physical mesh coordinate $\mathbf{X}_j = (x_j, y_j)$ and the computational mesh coordinate $\boldsymbol{\xi}_j = (\xi_j, \eta_j)$ $(j = 0, \ldots, J)$, where $J + 1$ stands for the number of mesh vertices. The coordinate $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{X}, t)$ is an inverse transformation of $\mathbf{X} = \mathbf{X}(\boldsymbol{\xi}, t)$; when $j = 0$ or $j = J$, the coordinate $\boldsymbol{\xi}_j$ is the same as $\mathbf{X}_j$. The energy function is too long to write; we write it in a simple form:

$$I_h = \sum_K |K|\, G\left(\left(F_K'\right)^{-1}, \det\left(F_K'\right)^{-1}, \mathbb{M}\left(\mathbf{X_K}\right), \mathbf{X}_K\right). \tag{7}$$

Assume that the current physical mesh $\mathscr{T}_h{}^n$ and $(u^h)^n$ are known; by minimizing the energy function, we can get the corresponding computational mesh of the next time level, which can be expressed by

$$\mathscr{T}_c{}^{n+1} = \arg\min_{\mathscr{T}_c} I_h\left(\mathscr{T}_h{}^n; \mathscr{T}_c\right). \tag{8}$$

In order to solve the minimum of energy function in (6), we need to take the derivative of the coordinate with respect to the energy function; that is,

$$\frac{\partial I_h}{\partial \mathbf{X}_j} = \frac{\partial I_h}{\partial \boldsymbol{\xi}_j}, \quad j = 0, \ldots, J. \tag{9}$$

It is not difficult to find that the equations are nonlinear and difficult to solve directly; in order to solve this problem, many scholars put forward a large number of methods to improve it. In this paper, we use the MMPDE [42] method to get the minimum value of the energy function (6). For convenience and efficiency, ξ-formulation MMPDE is made use of rather than x-formulation MMPDE; during the calculation, ξ-formulation only involves the differentiation of G with respect to its first two arguments in the energy function, and there is no need for the metric tensor $\mathbb{M}$ to update. The minimization process can be replaced as the derivative of the computational mesh coordinate with respect to t, which is an ordinary differential equation and has the following form:

$$\frac{d\xi_j}{dt} = \frac{P_j}{\tau}\left[\frac{\partial I_h}{\partial \xi_j}\right]^T, \tag{10}$$

where the derivative $\partial I_h/\partial \xi_j$ is seen as a row vector, $\tau > 0$ is a relaxation factor which is used for adjusting the time scale on moving mesh, and $P = \{P_0, \ldots, P_J\}$ is a positive function used to make (10) have some desired properties above; in this paper, $\tau = 10^{-4}$, $P_j = -\det(\mathbb{M}(\mathbf{X}_j))^{1/2}$.

At the current physical mesh $\mathscr{T}_h{}^n$ $(n = 0, \ldots, K)$ and computational solution $(u^h)^n$, we define a reference computational mesh $\widehat{\mathscr{T}}_c = \{\widehat{\xi}_0, \ldots, \widehat{\xi}_J\}$, which has uniform elements; the mesh $\widehat{\mathscr{T}}_c$ also has the same relationship, grid connection, and number of mesh vertices as the physical mesh $\mathscr{T}_h$ and computational mesh $\mathscr{T}_c$. The three kinds of meshes just differ in the location of the mesh coordinate. During the mesh generation, by integrating (10) from t_n to t_{n+1}, we get the new computational mesh $\mathscr{T}_c{}^{n+1} = \{\xi_j{}^{n+1}\}$; there is no need for the metric tensor $\mathbb{M}$ to update. The relationship between the physical mesh and the computational mesh can be expressed by an affine mapping: $\mathscr{T}_h{}^n = \Phi_h(\mathscr{T}_c{}^{n+1})$; the physical mesh in the next time level can take the reference computational mesh by using a form of linear interpolation to get the approximate result: $\mathscr{T}_h{}^{n+1} = \Phi_h(\widehat{\mathscr{T}}_c)$.

2.2. Linear Finite Element Discretization of the 1D BBM-Burgers Equation with a Dissipative Term.
In this subsection, we mainly introduce the linear finite element discretization of the 1D BBM-Burgers equation with a dissipative term. When $(u^h)^n$, x^n, and the discretization of the equation are known, we will get the value $(u^h)^{n+1}$ at the new time level. The 1D BBM-Burgers equation with a fourth-order dissipative term has the formulation

$$u_t + \alpha u_x + \beta u u_x - \gamma u_{xx} - \mu u_{xxt} + \delta u_{xxxx} = F,$$
$$x \in \Omega, \ t \in (0, T]. \tag{11}$$

Before solving (11) by the finite element method based on moving mesh, we have to introduce a new variable $w = u_{xx}$ rather than solving the equation directly; because a mixed derivative of the finite element approximation based on the moving mesh method is unable to be defined, then (11) becomes a coupled system. Before the discretization of the 2D equation with a fourth-order dissipative term, we need to introduce two new different variables to replace u_{xx} and u_{yy}; the equation then becomes three coupled equations; the other steps in the 2D case are similar to the 1D case. The new formulation of (11) is as follows:

$$u_t - \mu w_t + \alpha u_x + \beta u u_x - \gamma w + \delta w_{xx} = F,$$

$$x \in \Omega, \ t \in (0, T],$$

$$w = u_{xx},$$

$$x \in \Omega, \ t \in (0, T], \tag{12}$$

$$u(x, t) = 0,$$

$$x \in \partial\Omega, \ t \in (0, T].$$

We divide the domain Ω into N uniform elements and $J + 1$ mesh vertices; the initial physical mesh is $\mathcal{T}_h^0 = \{x_j^0\}$ $(j = 0, 1, \ldots, J)$. We define the linear basis function of the finite element method as φ. On a fixed mesh, the basis function φ is just related to x; on a moving mesh, it is also related to x and t. The basis function spans the space $V_0^h = \{\varphi_1(x, t), \ldots, \varphi_{J-1}(x, t)\}$, $\forall \varphi \in V_0^h$. We need to find whether $u^h(x, t) \in V_0^h$ and $w^h(x, t) \in V_0^h$ with the Dirichlet boundary condition equal to zero can satisfy the following system:

$$\int_\Omega (u_t - \mu w_t + \alpha u_x + \beta u u_x - \gamma w - F) \varphi \, dx$$

$$- \int_\Omega \delta w_x \varphi_x dx = 0, \quad \forall \varphi \in V_0^h(\Omega), \ t \in (0, T], \tag{13}$$

$$\int_\Omega (w\varphi + \mu u_x \varphi_x) \, dx = 0, \quad \forall \varphi \in V_0^h(\Omega), \ t \in (0, T].$$

Take $u^h(x, t) = \sum_{j=1}^{J-1} u_j(t) \varphi_j(x, t)$, $w^h(x, t) = \sum_{j=1}^{J-1} w_j(t) \varphi_j(x, t)$, where the derivative of the basis function [43] φ_j with respect to t is $\partial \varphi_j / \partial t = -(\partial \varphi_j / \partial x) \dot{X}(x, t)$, and $\dot{X}$ has the following formulation: $\dot{X}(x, t) = \sum_{j=1}^{J-1} \dot{x}_j \varphi_j(x, t)$ (in the two-dimensional equation, $\dot{X}(x, y, t) = \sum_{j=1}^{J-1} \begin{bmatrix} \dot{x}_j \\ \dot{y}_j \end{bmatrix} \varphi_j(x, y, t)$), so $(w^h)_t$ can be expressed as

$$\left(w^h\right)_t = \sum_{j=1}^{J-1} \frac{dw_j}{dt} \varphi_j - \sum_{j=1}^{J-1} w_j \frac{\partial \varphi_j}{\partial x} \dot{X}$$

$$= \sum_{j=1}^{J-1} \frac{dw_j}{dt} \varphi_j - \frac{\partial w^h}{\partial x} \dot{X}. \tag{14}$$

Take the basis function $\varphi = \varphi_k$ $(k = 1, 2, \ldots, J - 1)$ and substitute (14) into (13); we can get

$$\sum_{j=1}^{J-1} \int_\Omega \left(\left(\frac{du_j}{dt} - \mu \frac{dw_j}{dt} \right) \varphi_j \varphi_k - \delta w_j \frac{\partial \varphi_j}{\partial x} \frac{\partial \varphi_k}{\partial x} \right) dx$$

$$+ \int_\Omega \left(\alpha \frac{\partial u^h}{\partial x} + \beta u^h \frac{\partial u^h}{\partial x} - \gamma w^h - F \right.$$

$$\left. - \left(\frac{\partial u^h}{\partial x} \sum_{j=1}^{J-1} \dot{x}_j \varphi_j - \mu \frac{\partial w^h}{\partial x} \sum_{j=1}^{J-1} \dot{x}_j \varphi_j \right) \right) \varphi_k dx = 0, \tag{15}$$

$$\sum_{j=1}^{J-1} \int_\Omega \left(w_j \varphi_j \varphi_k + u_j \frac{\partial \varphi_j}{\partial x} \frac{\partial \varphi_k}{\partial x} \right) dx = 0.$$

We rewrite (15) into a matrix form:

$$M \frac{d}{dt} (u - \mu w) + f - \delta A w = 0,$$

$$Mw + Au = 0, \tag{16}$$

where M and A stand for $M_{kj} = \int_\Omega \varphi_j \varphi_k dx$ and $A_{kj} = \int_\Omega (\partial \varphi_j / \partial x)(\partial \varphi_k / \partial x) dx$ and the vector f can be expressed as $f_k = \int_\Omega (\alpha(\partial u^h / \partial x) + \beta u^h (\partial u^h / \partial x) - \gamma w^h - F - ((\partial u^h / \partial x) \sum_{j=1}^{J-1} \dot{x}_j \varphi_j - \mu(\partial w^h / \partial x) \sum_{j=1}^{J-1} \dot{x}_j \varphi_j)) \varphi_k dx$. On a moving mesh, with the boundary condition, the solution of (11) can be obtained by solving an ordinary differential equation firstly:

$$M \frac{d}{dt} \left(A^{-1} Mw \right) + \mu M \frac{dw}{dt} - f \left(-A^{-1} Mw, w, F \right)$$

$$+ \delta A w = 0. \tag{17}$$

Only if the ODE (17) is solved can the solution u of the BBM-Burgers equation with a dissipative term be expressed, where $u = -A^{-1} Mw$.

All the process of discretization is based on the homogeneous boundary condition $u_0 = g$; when the boundary condition is nonhomogeneous, we just need to subtract the integration at the boundary.

2.3. The Implementation of the Method Combined with Finite Element and Moving Mesh.

In this subsection, we mainly introduce the implementation of the two-dimensional method. The ways to solve the equation usually can be divided into two kinds: a simultaneous solution procedure and an alternate solution procedure. The simultaneous solution procedure combines the finite element discretization with the mesh equation (10) into a large system and solves them together; the alternate solution procedure solves one by one. In this paper, we introduce a formulation of the second method which is called MP (the method solves the moving equation firstly and solves the physical equation later), which is more flexible and efficient. The method contains four important parts: initialize the mesh, move the mesh, discretize the equation, and integrate time from t_0 to t_K.

TABLE 1: Example 1. The L_2 and L_∞ error and convergence order on moving and fixed meshes for the single solitary wave.

N	Moving mesh				Fixed mesh			
	L_2 error	Order	L_∞ error	Order	L_2 error	Order	L_∞ error	Order
20	$3.65E-2$		$1.70E-1$		$8.35E-1$		$3.20E-0$	
40	$9.12E-2$	2.00	$3.99E-2$	2.09	$1.54E-1$	2.44	$1.12E-0$	1.51
80	$2.49E-3$	1.87	$1.05E-2$	1.93	$4.09E-2$	1.91	$3.74E-1$	1.58
160	$6.53E-4$	1.93	$2.74E-3$	1.93	$1.04E-2$	1.98	$1.00E-1$	1.90
320	$1.84E-4$	1.83	$7.65E-4$	1.84	$2.60E-3$	1.99	$2.55E-2$	1.97
640	$5.01E-4$	1.88	$2.06E-4$	1.89	$6.51E-4$	2.00	$6.41E-3$	1.99

Step 1 (initialize the mesh). In the whole computational domain, we first divide the domain into N uniform elements; the corresponding initial physical mesh is $\mathcal{T}_h^0$, and the initial coordinate, initial value of the BBM-Burgers equation with a dissipative term, initial time step, and initial time are known; they are $\mathbf{X}_j^0 = (x_j^0, y_j^0)$, $(u^h)^0$, dt_0, and t_0.

Step 2 (move the mesh). (a) Suppose the current time $t = t_n$ $(n = 0, 1, \ldots, K)$, the numerical solution $(u^h)^n$ of the equation, the coordinate $\mathbf{X}_j^n$, $\mathcal{T}_h^n$ of the mesh, and the time step dt_n are known; (b) we make use of the energy function (9) and ODE function (10) to obtain the coordinate $\boldsymbol{\xi}_j^{n+1} = (\xi_j^{n+1}, \eta_j^{n+1})$, and then we derive the relationship $\Phi_h(\cdot)$ between the new computational mesh $\mathcal{T}_c^{n+1} = \{\boldsymbol{\xi}_j^{n+1}\}$ and the physical mesh $\mathcal{T}_h^n$, where $\mathcal{T}_h^n$ can be expressed by $\Phi_h(\mathcal{T}_c^{n+1})$. When the reference computational mesh $\widetilde{\mathcal{T}}_c$ and $\Phi_h(\cdot)$ are known, we get the new physical mesh $\mathcal{T}_h^{n+1} = \Phi_h(\widetilde{\mathcal{T}}_c)$ approximately; the new coordinate is $\widetilde{\mathbf{X}}_j^{n+1}$; (c) when $t \in [t_n, t_{n+1}]$, using linear interpolation $\mathbf{X}_j(t) = ((t_{n+1} - t)/dt_n)\mathbf{X}_j^n + ((t - t_n)/dt_n)\widetilde{\mathbf{X}}_j^{n+1}$, the velocity of the corresponding grid node is $\dot{\mathbf{X}}_j(t) = (\widetilde{\mathbf{X}}_j^{n+1} - \mathbf{X}_j^n)/dt_n$. The fifth-order Radau IIA method is used in time discretization; if you are interested in more details, please refer to [44]. From the method, we can determine the time step $\widetilde{dt_n}$; if the real time step $\widetilde{dt_n} < dt_n$, the next time level, the mesh coordinate, and the time step of next time level update: $t_{n+1} \leftarrow t_n + \widetilde{dt_n}$, $\mathbf{X}_j^{n+1} \leftarrow \mathbf{X}_j^n + \widetilde{dt_n}\dot{\mathbf{X}}_j$, and $dt_{n+1} \leftarrow \widetilde{dt_n}$.

Step 3. On the new physical mesh $\mathcal{T}_h^{n+1}$, use the discrete version (16) based on the linear finite element method to solve (12) from time t_n to t_{n+1}; we get the new numerical solution at the new time level; the solution is $(u^h)^{n+1}$.

Step 4. When $t_{n+1} < T$, $n \leftarrow n + 1$, go to Step 2; else, stop the computation and save the value of solution.

3. Numerical Examples

Example 1 (a single solitary wave in 1D BBM-Burgers equation). In the one-dimensional model,

$$u_t + \alpha u_x + \beta u u_x - \gamma u_{xx} - \mu u_{xxt} = F,$$

$$x \in (-20, 30), \ t \in (0, T],$$

$$u(-20, t) = u(30, t) = 0,$$

$$t \in (0, T],$$

(18)

where the coefficients $\alpha = \gamma = \mu = 1$, $\beta = -1$, the final time $T = 10$, the exact solution of the equation is $u_{ex}(x, t) = \mathrm{sech}(x - t)$, and the source term $F = F(x, t) = (1 - 6\tanh^3(x - t) - 2\tanh^2(x - t) + (\mathrm{sech}(x - t) + 5)\tanh(x - t))\mathrm{sech}(x - t)$.

We solve the numerical solution with the method we introduced; with the different element numbers, the corresponding error and the order of convergence for both moving mesh and fixed mesh are reported in Table 1. When the element number is the same, the error based on moving mesh is always smaller than that on fixed mesh, and the convergence order of the two kinds of meshes can lead to the second order in the spatial direction. Comparison between the exact solution and the numerical solution based on moving mesh at the final time $T = 10$ is shown in Figure 1. From the figure, we can see that the grid points mainly focus on the domain, where the solution fluctuates greatly, and match the exact solution well. With time going on, the moving mesh trajectory is plotted in Figure 1; the feature of the moving process indicates that most of the points focus on nearby solitary waves and move with the propagation of the solitary wave, which also demonstrates the superiority of the moving mesh method.

Example 2 (the elastic collision of three solitary waves in 1D BBM-Burgers equation). In this example, we study the elastic collision of three solitary waves and simulate numerical solutions based on moving mesh and fixed mesh; the BBM-Burgers equation has the following form:

$$u_t + u_x + u u_x - 0.01 u_{xx} - u_{xxt} = 0,$$

$$x \in (-50, 300), \ t \in (0, 50],$$

$$u(-50, t) = u(300, t) = 0,$$

$$t \in (0, 50],$$

(19)

$$u(x, 0) = u_0(x),$$

$$x \in (-50, 300),$$

(a) Computed solution and exact solution
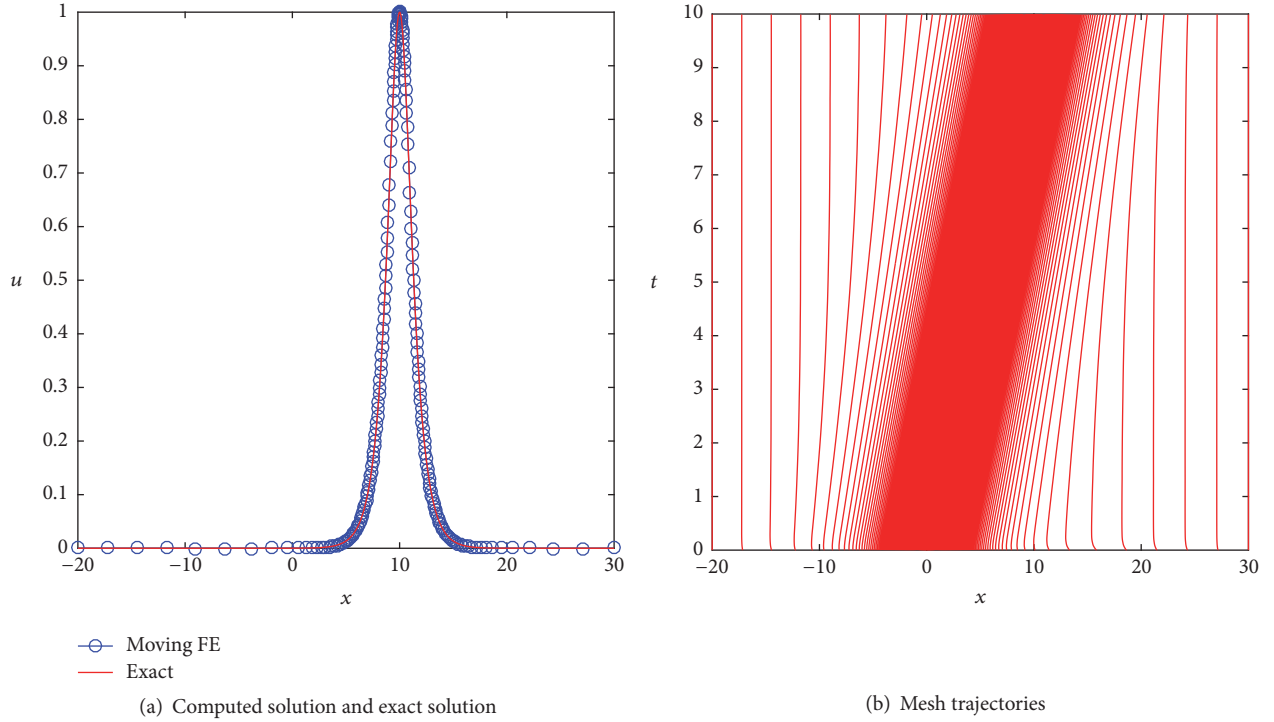

(b) Mesh trajectories

FIGURE 1: Example 1. (a) The exact solution and the numerical solution based on adaptive moving mesh with element number $N = 200$ at $t = 10$; (b) the corresponding mesh trajectories.

with the initial condition $u_0(x) = \sum_{j=1}^{3} 3d_j \operatorname{sech}^2(k_j(x-x_j))^2$, where $k_1 = 0.39, k_2 = 0.3, k_3 = 0.3, x_1 = 10, x_2 = 28, x_3 = 52$, $d_j = 4k_j^2/(1 - 4k_j^2)$, and the final time $T = 50$.

In this example, the exact solution of the equation is unknown; we mainly want to show the superiority of the moving mesh. In Figure 2, the conservation laws (2) are plotted; we can see that $\Delta E_1(T)$ is close to zero and $\Delta E_2(T)$ is a little bigger than zero. The two conservation laws are not conserved by finite element approximation on the moving mesh, because, in this case, $\dot{X} \equiv 0$ and both matrices A and M are time-dependent (if you want to know more, read [45]). Since the exact solution is unknown, we take the element number $N = 8000$ on the fixed mesh as a reference solution and plot the numerical solutions of moving mesh and fixed mesh with element number $N = 400$ at final time $T = 50$ in Figure 3. It can be seen that the numerical solution of fixed mesh is on the left of the reference solution and the wave peaks are shorter than those in the reference solution; there are several small waves in the left of the first wave on the fixed mesh. The result also shows that the numerical solution of moving mesh can match the reference better than that based on the fixed mesh. If we consider the computer memory and computational efficiency, the moving mesh is better for that it moves its mesh points quickly to the place where the numerical solution changes rapidly to get higher distinguishability with fewer elements.
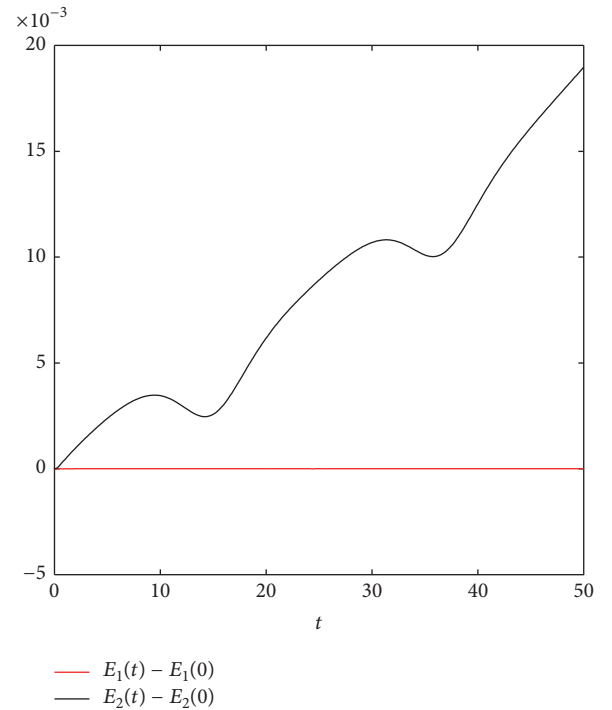

FIGURE 2: Example 2. The variation of mass function (E_1) and energy function (E_2) for the elastic collision of three solitary waves in the BBM-Burgers equation.
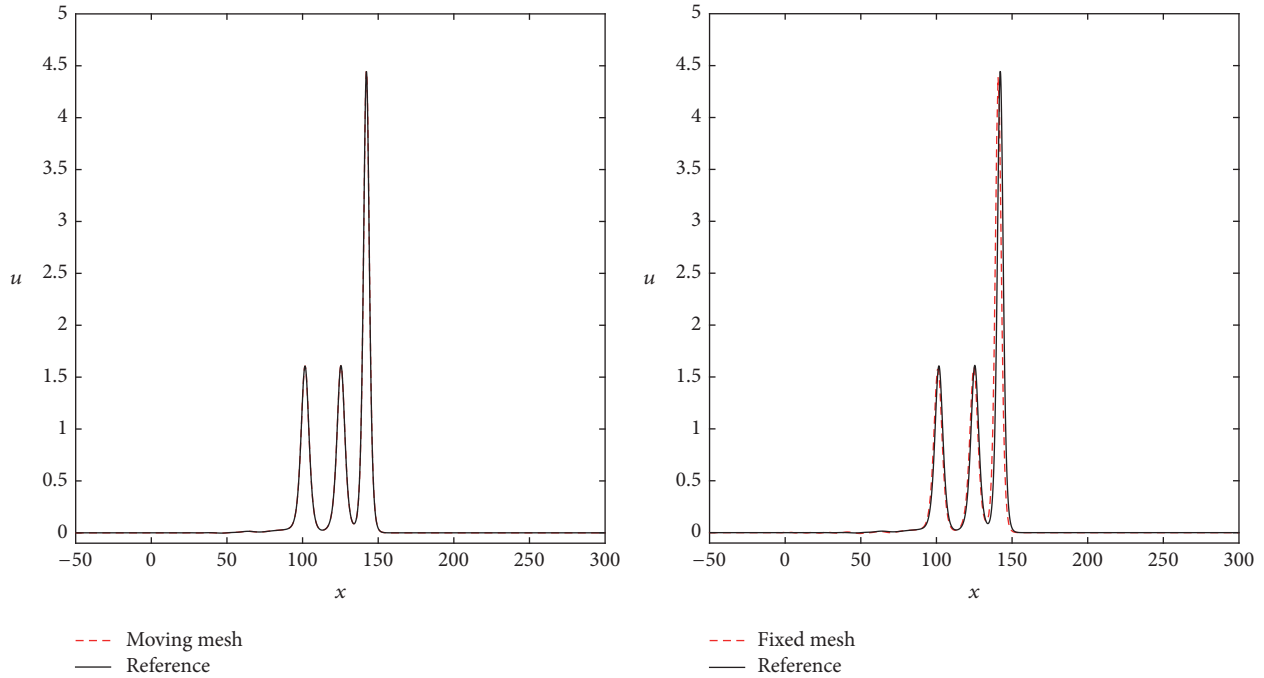
FIGURE 3: Example 2. The comparison between the numerical solution and the reference solution on moving and fixed meshes for the interaction of three solitary waves with $N = 400$.
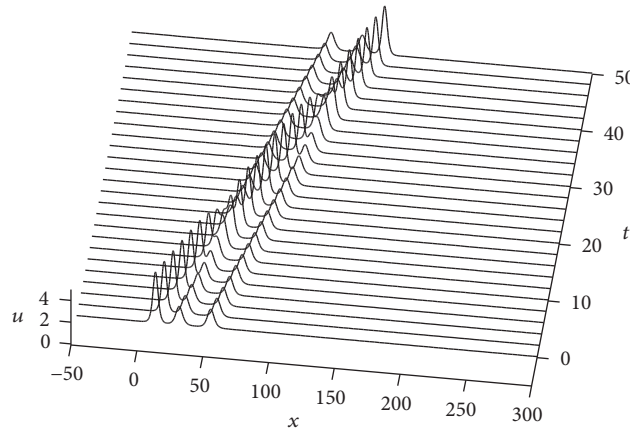

FIGURE 4: Example 2. The propagation process for the elastic collision of three solitary waves at different times with $N = 400$.

The propagation process of the three solitary waves is plotted in Figure 4.

Example 3 (different coefficients in 1D BBM-Burgers equation with a high-order dissipative term). In this case, we mainly consider the following BBM-Burgers equation with a fourth-order dissipative term; the equation is

$$u_t + u_x + 2uu_x - u_{xxt} - \gamma u_{xx} + \delta u_{xxxx} = 0,$$

$$x \in (-100, 150), \ t \in (0, 20],$$

$$u(-100, t) = u(150, t) = 0, \quad t \in (0, 20],$$

$$u(x, 0) = u_0(x), \quad x \in (-100, 150).$$

(20)

The initial condition $u_0(x) = 0.15 \operatorname{sech}(k(x - 40))^2$, where $k = 0.5 \times \sqrt{1/11}$; when the coefficients γ, δ change, the maximum value $U_{\max}$ of the numerical solution changes as well.

In this example, we mainly want to talk about the dissipation effect of the dissipative terms u_{xx} and u_{xxxx} in the BBM equation. Numerical solutions obtained with different coefficients and one of the coefficients fixed at $T = 20$ are shown in Figure 5. When two BBM equations with one dissipative term u_{xx} or u_{xxxx} have the same dissipation effect or similar wave peak at the final time, the corresponding coefficient of dissipative term is plotted in Figure 6. From these figures, we find that when the coefficient γ of the second-order dissipative term increases and the coefficient δ

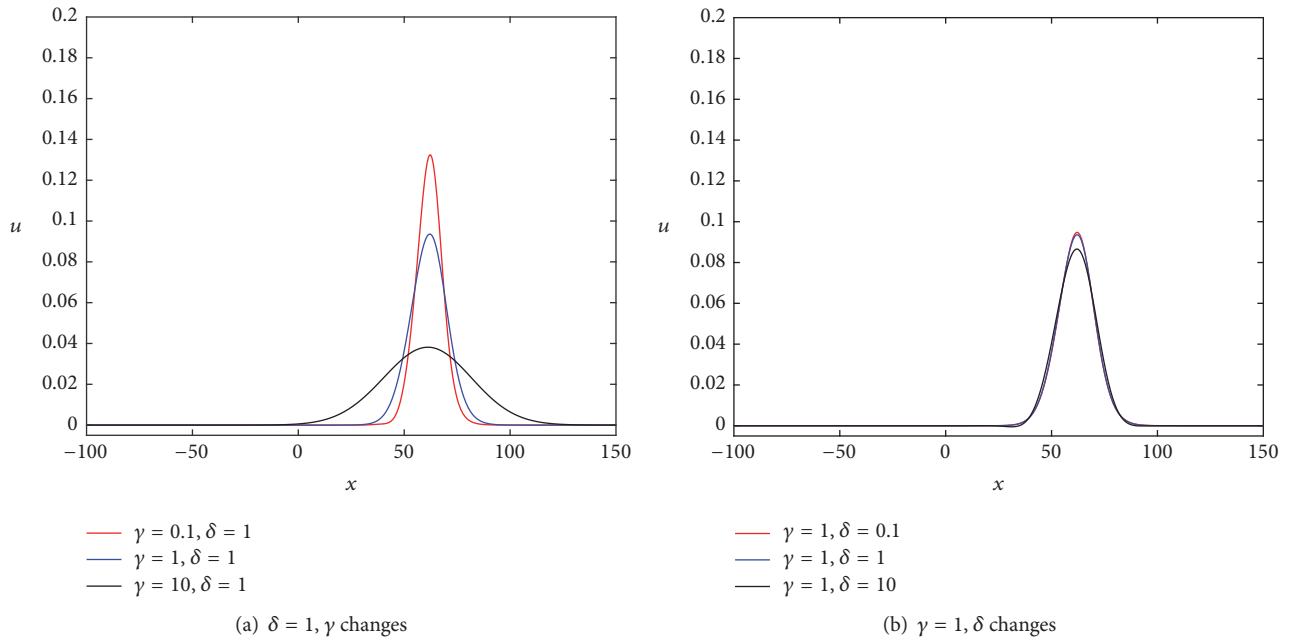(a) $\delta = 1$, γ changes

(b) $\gamma = 1$, δ changes

FIGURE 5: Example 3. The numerical solutions at the final time for different coefficients. (a) The numerical solutions of the equation with $\delta = 1$, $\gamma = 0.1, 1, 10$; (b) the numerical solutions of the equation with $\gamma = 1$, $\delta = 0.1, 1, 10$.

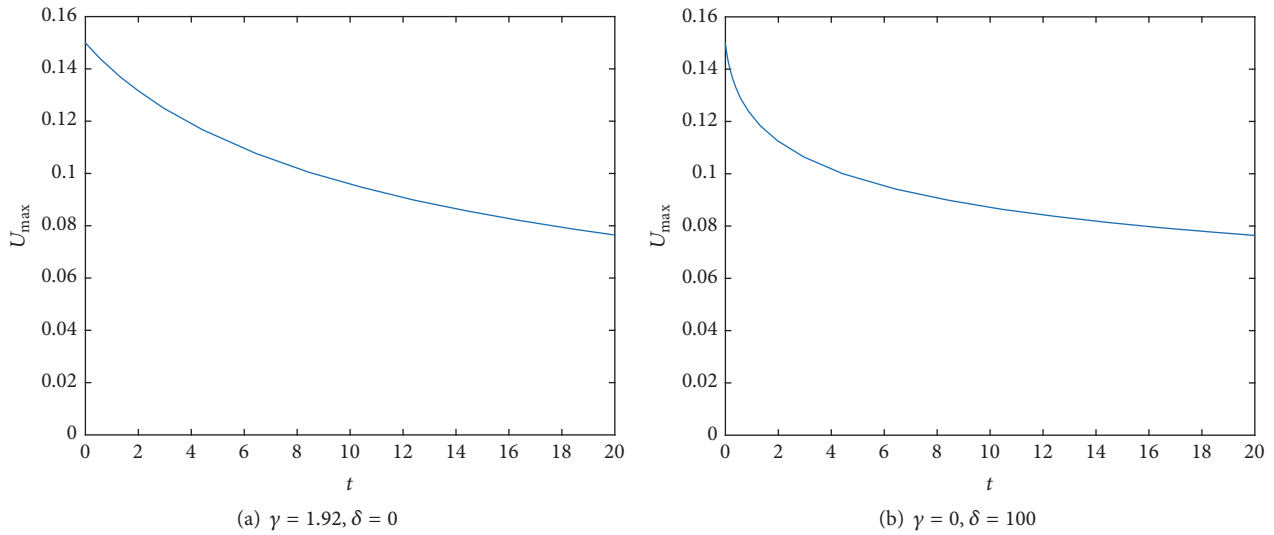(a) $\gamma = 1.92$, $\delta = 0$

(b) $\gamma = 0$, $\delta = 100$

FIGURE 6: Example 3. The corresponding dissipative term coefficient when U_{max} has the same value at the beginning and final time; (a) U_{max} of the equation with just one dissipative u_{xx} and the coefficient $\gamma = 1.92$; (b) U_{max} of the equation with just one dissipative u_{xxxx} and the coefficient $\delta = 100$.

of the fourth-order dissipative term is fixed, the numerical solution decreases a lot; however, when the coefficient δ increases and the coefficient γ is fixed, the numerical solution changes a little. The numerical simulation provides three sets of data to show the result in Figure 5: $\delta = 1$; when $\gamma = 0.1, 1, 10$, the corresponding maximum values of the numerical solutions are $U_{max} = 0.139, 0.1324, 0.1091$, $\gamma = 1$; when $\delta = 0.1, 1, 10$, the corresponding values are

$U_{max} = 0.0947, 0.0936, 0.0866$. A lot of numerical tests have been carried out to adjust and confirm the corresponding coefficient of different dissipative terms to keep the equation with dissipation effect or similar wave peak at $T = 20$. The value of the wave peak is $U_{max} = 0.0764$ at $T = 20$, and the corresponding data are shown in Figure 6. When $\delta = 0$, $\gamma = 1.92$, with time going on, U_{max} drops steadily; however, when $\gamma = 0$, $\delta = 100$, the value drops quickly before time $t = 2$,

(a) Moving mesh

(b) The corresponding numerical solution

(c) Fixed mesh

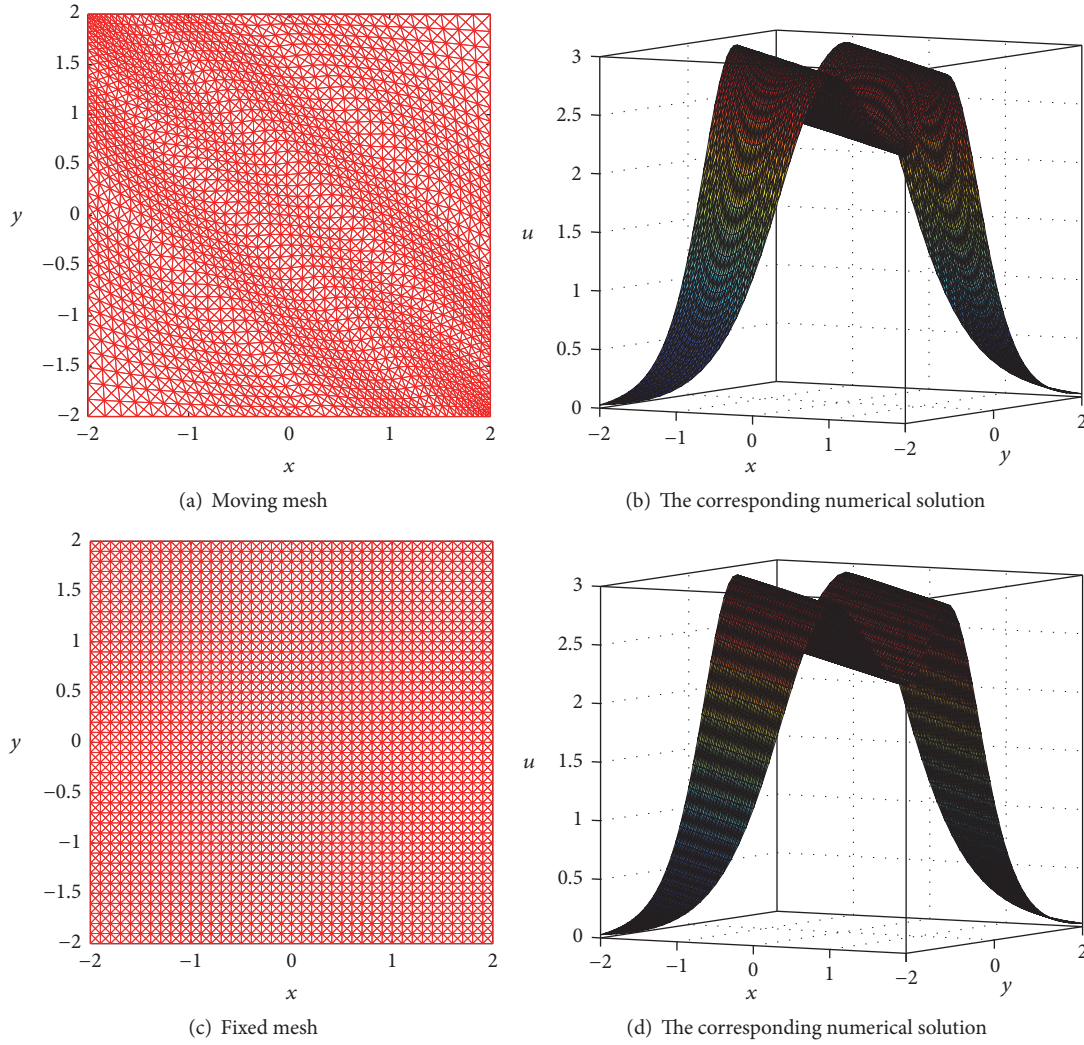(d) The corresponding numerical solution

Figure 7: Example 4. Meshes (a, c) and the corresponding numerical solutions (b, d) with $N = 6400$.

and then the value drops slowly and steadily. We simulate a series of numerical examples with different coefficients; all of these data show us that the dissipative term u_{xx} plays a more important role in dissipation than the dissipative term u_{xxxx}.

Example 4 (two solitary waves in 2D BBM-Burgers equation). This example has two solitary waves and we want to verify the convergence order for a 2D BBM-Burgers equation; the equation has the following formulation:

$$u_t + (\alpha + \beta \cos(u)) (u_x + u_y) - \gamma (u_{xx} + u_{yy})$$
$$- \mu (u_{xxt} + u_{yyt}) = F, \quad (x, y) \in \Omega, \ t \in (0, T], \tag{21}$$

where $\alpha, \gamma, \mu = 1$, $\beta = -1$, and the exact solution of the equation is $u_{ex}(x, y, t) = e^t (\text{sech}^2(x_1) + \text{sech}^2(x_2))$, where $x_1 = x + y - 1$, $x_2 = x + y + 1$, $\Omega = (-2, 2) \times (-2, 2)$, $T = 1$. To maintain the balance of the equation, $F = F(x, y, t) = e^t ((9(\text{sech}^2(x_1) +$

$\text{sech}^2(x_2)) - 4(\text{sech}^2(x_1) \tanh(x_1) + \text{sech}^2(x_2) \tanh(x_2)) - 24(\text{sech}^2(x_1) \tanh^2(x_1) + \text{sech}^2(x_2) \tanh^2(x_2)) + 4 \cos(e^t (\text{sech}^2(x_1) + \text{sech}^2(x_2))))(\text{sech}^2(x_1) \tanh(x_1) + \text{sech}^2(x_2) \tanh(x_2)))$.

Numerical solutions are shown in Table 2 and Figure 7. From Table 2, we can find that the finite element method leads to the same second order of convergence for both fixed and moving meshes, and the convergence result is similar to Example 1. The error of the numerical solution on the moving mesh is smaller than that on the fixed mesh with the same element number. What is more, with fewer mesh elements, the moving mesh reaches second-order convergence more quickly than the fixed mesh. The trajectory and the numerical solution of both the moving mesh and the fixed mesh at final time can be seen in Figure 7. From the figure, we can see that most points focus on the place where the numerical solution changes a lot.

TABLE 2: Example 4. The numerical error and convergence order for the 2D BBM-Burgers equation.

N	Moving mesh				Fixed mesh			
	L_2 error	Order	L_∞ error	Order	L_2 error	Order	L_∞ error	Order
100	$3.08E-2$		$1.41E-1$		$4.10E-2$		$2.28E-1$	
400	$8.04E-3$	1.94	$4.44E-2$	1.67	$1.20E-2$	1.77	$7.52E-2$	1.60
1600	$2.09E-3$	1.94	$1.12E-2$	1.99	$3.11E-3$	1.95	$2.02E-2$	1.90
6400	$5.48E-4$	1.93	$2.87E-3$	1.96	$7.83E-4$	1.99	$5.19E-3$	1.96
25600	$1.42E-4$	1.95	$7.41E-4$	1.95	$1.96E-4$	2.00	$1.30E-3$	1.99

4. Conclusion

In this paper, we study the numerical model based on adaptive moving mesh finite element method to simulate 1D and 2D BBM-Burgers equations with a high-order dissipative term. The method can be used in equations with mixed derivatives and high-order derivative terms; on the moving mesh, new variables need to be introduced to transform the equation into coupled equations. A series of numerical examples demonstrate that the finite element method based on moving mesh keeps a second-order convergence in space as that based on fixed mesh. With the same element number, the finite element moving mesh method improves the distinguishability and provides a smaller error than the fixed mesh. Meanwhile, numerical tests indicate that the dissipative term u_{xx} plays a more important role than u_{xxxx} in dissipation. We compare two BBM equations with different dissipative terms u_{xx} and u_{xxxx}; to keep the maximum value U_{max} of the numerical solutions the same at the final time, the coefficient of u_{xxxx} is around 50 times that of u_{xx} in the same BBM equation with different dissipative terms in the dissipation aspect.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] X. Xu X, "An integrable coupling hierarchy of the Mkdvintegrable systems, its Hamiltonian structure and corresponding nonisospectral integrable hierarchy," *Applied Mathematics and Computation*, vol. 216, no. 1, pp. 344–353, 2010.

[2] X. Li Y, Q. Zhao L, Y. Li X, and etal., "Binary Bargmann symmetry constraint associated with 3×3 discrete matrix spectral problem," *Journal of Nonlinear Sciences and Applications*, vol. 8, no. 5, pp. 496–506, 2015.

[3] Y. Zhang, H. Dong, X. Zhang, and H. Yang, "Rational solutions and lump solutions to the generalized (3+1)-dimensional Shallow Water-like equation," *Computers & Mathematics with Applications*, vol. 73, no. 2, pp. 246–252, 2017.

[4] M. P. Barnett, J. F. Capitani, J. von zur Gathen, and J. Gerhard, "Symbolic calculation in chemistry: selected examples," *International Journal of Quantum Chemistry*, vol. 100, no. 2, pp. 80–104, 2004.

[5] M. F. Saleh, W. Chang, J. C. Travers, P. S. J. Russell, and F. Biancalana, "Plasma-induced asymmetric self-phase modulation and modulational instability in gas-filled hollow-core photonic crystal fibers," *Physical Review Letters*, vol. 109, no. 11, Article ID 113902, pp. 1–5, 2012.

[6] H. W. Yang, D. Z. Yang, Y. L. Shi, S. S. Jin, and B. S. Yin, "Interaction of algebraic Rossby solitary waves with topography and atmospheric blocking," *Dynamics of Atmospheres and Oceans*, vol. 71, pp. 21–34, 2015.

[7] H. W. Yang, Z. H. Xu, D. Z. Yang, X. R. Feng, B. S. Yin, and H. H. Dong, "ZK-Burgers equation for three-dimensional Rossby solitary waves and its solutions as well as chirp effect," *Advances in Difference Equations*, vol. 2016, article no. 167, 2016.

[8] S.-P. Gorza, B. Deconinck, P. Emplit, T. Trogdon, and M. Haelterman, "Experimental demonstration of the oscillatory snake instability of the bright soliton of the (2+1)D hyperbolic nonlinear Schrödinger equation," *Physical Review Letters*, vol. 106, no. 9, Article ID 094101, 2011.

[9] Y. Shi, B. Yin, H. Yang, D. Yang, and Z. Xu, "Dissipative nonlinear Schrödinger equation for envelope solitary Rossby waves with dissipation effect in stratified fluids and its solution," *Abstract and Applied Analysis*, vol. 2014, Article ID 643652, 9 pages, 2014.

[10] H. Yang, Q. Zhao, B. Yin, and H. Dong, "A new integro-differential equation for rossby solitary waves with topography effect in deep rotational fluids," *Abstract and Applied Analysis*, vol. 2013, Article ID 597807, 8 pages, 2013.

[11] H. Yang, B. Yin, Y. Shi, and Q. Wang, "Forced ILW-Burgers equation as a model for Rossby solitary waves generated by topography in finite depth fluids," *Journal of Applied Mathematics*, vol. 2012, Article ID 491343, 17 pages, 2012.

[12] T. B. Benjamin, J. L. Bona, and J. J. Mahony, "Model equations for long waves in nonlinear dispersive systems," *Philosophical Transactions of the Royal Society of London*, vol. 272, no. 1220, pp. 47–78, 1972.

[13] H. Zhao and B. Xuan, "Existence and convergence of solutions for the generalized BBM-Burgers equations with dissipative term," *Nonlinear Analysis Theory Methods and Applications*, vol. 75, no. 1-2, pp. 107–135, 2000.

[14] C. I. Kondo and C. M. Webler, "The generalized BBM-Burgers equations: convergence results for conservation law with discontinuous flux function," *Applicable Analysis: An International Journal*, vol. 95, no. 3, pp. 1–21, 2016.

[15] C. I. Kondo and C. M. Webler, "Higher-order for the multi-dimensional generalized bbm-burgers equation: existence and convergence results," *Acta Applicandae Mathematicae*, vol. 88, no. 7, pp. 977–995, 2010.

[16] H. Seok, "Kinetic decomposition for the generalized BBM-Burgers equations with dissipative term," in *Proceedings of the Royal Society of Edinburgh*, vol. 134, pp. 1149–1162, 2004.

[17] H. Dong, T. Chen, L. Chen, and Y. Zhang, "A new integrable symplectic map and the Lie point symmetry associated with nonlinear lattice equations," *Journal of Nonlinear Sciences and Applications*, vol. 9, no. 7, pp. 5107–5118, 2016.

[18] A. Bekir, "Painlevé test for some (2+1)-dimensional nonlinear equations," *Chaos Solitons and Fractals*, vol. 32, no. 2, pp. 449–455, 2007.

[19] X.-X. Xu, "A deformed reduced semi-discrete Kaup-Newell equation, the related integrable family and Darboux transformation," *Applied Mathematics and Computation*, vol. 251, pp. 275–283, 2015.

[20] X. Guo, "On bilinear representations and infinite conservation laws of a nonlinear variable-coefficient equation," *Applied Mathematics and Computation*, vol. 248, pp. 531–535, 2014.

[21] Q.-L. Zhao, X.-Y. Li, and F.-S. Liu, "Two integrable lattice hierarchies and their respective Darboux transformations," *Applied Mathematics and Computation*, vol. 219, no. 10, pp. 5693–5705, 2013.

[22] D. D. Ganji, H. Babazadeh, M. H. Jalaei, and H. Tashakkorian, "Application of He's variational iteration method for solving nonlinear BBMB equations and free vibration of systems," *Acta Applicandae Mathematicae*, vol. 106, no. 3, pp. 359–367, 2009.

[23] B. Hong and D. Lu, "Homotopic approximate solutions for the general perturbed burgers-BBM equation," *Journal of Information and Computational Science*, vol. 11, no. 11, pp. 4003–4011, 2014.

[24] T. Kadri, N. Khiari, F. Abidi, and K. Omrani, "Methods for the numerical solution of the Benjamin-BONa-Mahony-BURgers equation," *Numerical Methods for Partial Differential Equations*, vol. 24, no. 6, pp. 1501–1516, 2008.

[25] H. C. Lee, "Distributed feedback control of the Benjamin-Bona-Mahony-Burgers equation by a reduced-order model," *East Asian Journal on Applied Mathematics*, vol. 5, no. 1, pp. 61–74, 2015.

[26] Y. Yin and G. R. Piao, "Quadratic B-spline finite element method for the generalized Benjamin-Bona-Mahony-Burgers equation," *Journal of the Korean Mathematical Society*, vol. 29, no. 5, 2013.

[27] M. Dehghan, M. Abbaszadeh, and A. Mohebbi, "The use of interpolating element-free Galerkin technique for solving 2D generalized Benjamin-BONa-Mahony-BURgers and regularized long-wave equations on non-rectangular domains with error estimate," *Journal of Computational and Applied Mathematics*, vol. 286, Article ID 10067, pp. 211–231, 2015.

[28] K. Omrani and M. Ayadi, "Finite difference discretization of the Benjamin-BONa-Mahony-BURgers equation," *Numerical Methods for Partial Differential Equations*, vol. 24, no. 1, pp. 239–248, 2008.

[29] A. Mohebbi and Z. Faraz, "Solitary wave solution of nonlinear Benjamin-BONa-Mahony-BURgers equation using a high-order difference scheme," *Computational & Applied Mathematics*, pp. 1–13, 2015.

[30] H.-T. Che, X.-T. Pan, L.-M. Zhang, and Y.-J. Wang, "Numerical analysis of a linear-implicit average scheme for generalized Benjamin-Bona-Mahony-Burgers equation," *Journal of Applied Mathematics*, vol. 2012, Article ID 308410, pp. 295–305, 2012.

[31] G. Arora, R. C. Mittal, and B. K. Singh, "Numerical solution of BBM-Burger equation with quartic B-spline collocation method," *Journal of Engineering Science and Technology*, vol. 9, pp. 104–116, 2014.

[32] M. Alquran and K. Al-Khaled, "Sinc and solitary wave solutions to the generalized Benjamin-Bona-Mahony- Burgers equations," *Physica Scripta*, vol. 83, no. 6, Article ID 065010, 2011.

[33] Ö. Oruç, F. Bulut, and A. Esen, "Numerical solutions of regularized long wave equation by haar wavelet method," *Mediterranean Journal of Mathematics*, vol. 13, no. 5, pp. 1–19, 2016.

[34] Y. Fang, H. Dong, Y. Hou, and Y. Kong, "Frobenius integrable decompositions of nonlinear evolution equations with modified term," *Applied Mathematics and Computation*, vol. 226, no. 1, pp. 435–440, 2014.

[35] C. Lu, W. Huang, and J. Qiu, "Maximum principle in linear finite element approximations of anisotropic diffusion-convection-reaction problems," *Numerische Mathematik*, vol. 127, no. 3, pp. 515–537, 2014.

[36] C. Ngo and W. Huang, "A study on moving mesh finite element solution of the porous medium equation," *Journal of Computational Physics*, vol. 331, pp. 357–380, 2017.

[37] W. Huang and R. D. Russell, *Adaptive Moving Mesh Methods*, vol. 174 of *Applied Mathematical Sciences*, Springer Science & Business Media, New York, NY, USA, 2011.

[38] R. Li, T. Tang, and P. Zhang, "Moving mesh methods in multiple dimensions based on harmonic maps," *Journal of Computational Physics*, vol. 170, no. 2, pp. 562–588, 2001.

[39] C. J. Budd, W. Huang, and R. D. Russell, "Adaptivity with moving grids," *Acta Numerica*, vol. 18, pp. 111–241, 2009.

[40] T. Tang, "Moving mesh methods for computational fluid dynamics," in *Recent Advances in Adaptive Computation*, vol. 383 of *Contemporary Mathematics*, pp. 141–174, 2005.

[41] W. Huang and L. Kamenski, "A geometric discretization and a simple implementation for variational mesh generation and adaptation," *Journal of Computational Physics*, vol. 301, pp. 322–337, 2015.

[42] W. Huang, Y. Ren, and R. D. Russell, "Moving mesh partial differential equations (MMPDES) based on the equidistribution principle," *SIAM Journal on Numerical Analysis*, vol. 31, no. 3, pp. 709–730, 1994.

[43] P. K. Jimack and A. J. Wathen, "Temporal derivatives in the finite-element method on continuously deforming grids," *SIAM Journal on Numerical Analysis*, vol. 28, no. 4, pp. 990–1003, 1991.

[44] S. Gonzalez-Pinto, J. Montijano I, and S. Perez-Rodriguez, "Two-step error estimators for implicit Runge-Kutta methods applied to stiff systems," *Acm Transactions on Mathematical Software*, vol. 30, pp. 1–18, 2004.

[45] C. Lu, W. Huang, and J. Qiu, "An adaptive moving mesh finite element solution of the regularized long wave equation," *Journal of Scientific Computing*, pp. 1–23, 2017.