Journal of Computational Physics 229 (2010) 7147–7161

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Variational multiscale element-free Galerkin method for 2D Burgers' equation

Lin Zhang, Jie Ouyang *, Xiaoxia Wang, Xiaohua Zhang

Department of Applied Mathematics, Northwestern Polytechnical University, Xi'an 710072, China

ARTICLE INFO

Article history:
Received 29 March 2009
Received in revised form 2 June 2010
Accepted 4 June 2010
Available online 10 June 2010

Keywords:
Burgers' equation
Variational multiscale
Meshfree method

ABSTRACT

A new numerical method, which is based on the coupling between variational multiscale method and meshfree methods, is developed for 2D Burgers' equation with various values of *Re*. The proposed method takes full advantage of meshfree methods, therefore, no mesh generation and mesh recreation are involved. Meanwhile, compared with the variational multiscale finite element method, a strong assumption, that is, the fine scale vanishes identically over the element boundaries although non-zero within the elements, is not needed. Subsequently two problems which have an available analytical solution of their own are solved to analyze the convergence behaviour of the proposed method. Finally a 2D Burgers' equation having large *Re* is solved and the results have also been compared with the ones computed by two other methods. The numerical results show that the proposed method can indeed obtain accurate numerical results for 2D Burgers' equation having large *Re*, which does not refer to the choice of a proper stabilization parameter.

1. Introduction

The Burgers' equation is a useful model for many interesting physical problems, such as shock wave, acoustic transmission, traffic and aerofoil flow theory, turbulence and supersonic flow as well as a prerequisite to the Navier–Stokes equations. This equation can be considered as an evolutionary process in which a convective phenomenon is in contrast with a diffusive phenomenon. When diffusive phenomenon is dominative, the Burgers' equation is parabolic, or else it is hyperbolic. That is to say, this equation usually exhibits a "mixed" property. Although there are some analytic solutions available in the literatures [1,2], the exact solutions for the practical applications are very limited due to the complex geometry and complicated initial and boundary conditions. Therefore, various kinds of numerical methods are proposed for the Burgers' equation. In general, these methods fall into the following classes: finite difference method (FDM) [3,4], finite volume method (FVM) [5–7], finite element method [8,9], boundary element method (BEM) [10] and so on. However, among the methods mentioned above, the large amount of efforts should be paid during the numerical implementation, which makes these conventional numerical methods very difficult to efficiently deal with the Burgers' equation, especially for treating the nonlinear, multidimensional flows and irregular domain problems.

Recently, the developments of the so-called meshfree or meshless methods have caught the researchers' attentions. This kind of methods only use a set of nodes scattered within the problem domain as well as a set of nodes scattered on the boundary, therefore, they have many advantages over the conventional numerical methods. At present, there are many meshfree methods such as element-free Galerkin (EFG) method [11–13], meshless local Petrov–Galerkin (MLPG) method [13], reproducing kernel particle method (RKPM) [11] and so on. Applying meshfree methods to solve the Burgers' equation,

* Corresponding author. Tel.: +86 29 88495234; fax: +86 29 88491000.
E-mail address: jieouyang@nwpu.edu.cn (J. Ouyang).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.06.004

there are the following papers which can be referred to. Ouyang et al. [14] developed a series of non-standard element-free Galerkin methods to solve 1D and 2D Burgers' equation. But among these methods, the choice of a proper stabilization parameter is difficult. Young et al. [15] applied the Eulerian–Lagrangian method of fundamental solutions (ELMFS) to solve 2D Burgers' equation. Zhang et al. [16] combined the characteristic Galerkin (CG) method with EFG to solve 1D and 2D Burgers' equation.

In the mid-90s Hughes revisited the origins of the stabilization schemes from a variational multiscale view point and presented the variational multiscale (VM) method [17,18]. In this method the different stabilization techniques come together as special cases of the underlying subgrid scale modeling concept. As to the aspect of the research in which VM was combined with FEM, Masud et al. developed multiscale/stabilized formulations for the linearized incompressible Navier–Stokes equations [19], the advection–diffusion equation [20], the convective–diffusive heat transfer [21], the Darcy flow equation [22], the Fokker–Planck equation [23], and Franca et al. proposed a two-level finite element method for the convection–diffusion problem [24] as well as the incompressible Navier–Stokes equations [25] and so on. However, among these studies, there is a main assumption that the fine scale vanishes identically over the element boundaries although non-zero within the elements. Hughes [17] regarded that this was a rather strong assumption, and there would be many cases of practical interest in which this assumption would be invalid. In addition, as to the aspect of the research in which VM was combined with meshfree methods, Zhang et al. [26–28] followed the idea of the variational multiscale finite element method (VMFEM) and proposed the variational multiscale EFG (VMEFG) method and the two-level EFG method respectively for some benchmark problems in the field of fluid as well as magnetic field. However, they still assumed that the fine scale vanished identically over the element boundaries although non-zero within the background integral elements, which should not be suitable. Besides the research of Zhang, Yeon et al. [29,30] also combined VM and meshfree methods for the analysis of softening elastoplastic solids.

In order to take full advantage of meshfree methods and avoid the strong assumption mentioned above, the paper proposes a new coupling between VM and meshfree methods for 2D Burgers' equation. An outline of the paper is as follows: Section 2 presents the moving least squares (MLS) approximation. Emphasis in the paper is the description of the variational multiscale element-free Galerkin method, which is presented in Section 3. Section 4 presents the numerical results, and conclusions are drawn in Section 5.

2. The moving least squares approximation

According to the moving least squares interpolant [11,12], a local approximation $u^h(\boldsymbol{x})$ to the function $u(\boldsymbol{x})$ in the domain $\Omega_{\boldsymbol{x}}$ of influence of node $\boldsymbol{x}$ can be defined by

$$u^h(\boldsymbol{x}) = \sum_{i=1}^{m} p_i(\boldsymbol{x})\mathrm{a}_i(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x})\mathbf{a}(\boldsymbol{x}), \tag{1}$$

where $\mathbf{p}(\boldsymbol{x})$ is a complete polynomial basis of arbitrary order and $\mathbf{a}(\boldsymbol{x})$ is a coefficient needed to be determined, which as indicated, is the function of the space coordinate $\boldsymbol{x}$. For the sake of simplicity, linear basis is chosen.

Assume that we have known the nodal value $u_I = u(\boldsymbol{x}_I)$ for the function $u(\boldsymbol{x})$ at N nodes $\boldsymbol{x}_I$ ($I = 1,2,\ldots,N$) in the domain Ω. Then the unknown coefficient $\mathbf{a}(\boldsymbol{x})$ in Eq. (1) is obtained at any point $\boldsymbol{x}$ by minimizing the following weighted, discrete error norm

$$J = \sum_{I=1}^{N} w(\boldsymbol{x} - \boldsymbol{x}_I)[\boldsymbol{p}^T(\boldsymbol{x}_I)\mathbf{a}(\boldsymbol{x}) - u_I]^2, \tag{2}$$

where $w(\boldsymbol{x} - \boldsymbol{x}_I)$ is a weight function of compact support (often called the domain of influence of node I). The choice of the weight function is more or less arbitrary, and the spline function is often used in practice together with the exponential function. In the paper the following cubic spline function is chosen

$$w(z) = \begin{cases} \frac{2}{3} - 4z^2 + 4z^3 & 0 \leqslant z \leqslant 0.5, \\ \frac{4}{3} - 4z + 4z^2 - \frac{4}{3}z^3 & 0.5 < z \leqslant 1, \\ 0 & \text{otherwise}, \end{cases}$$

in which $z = |\boldsymbol{x} - \boldsymbol{x}_I|/r$ and r denotes the influence radius of node $\boldsymbol{x}_I$.

Minimization of Eq. (2) with respect to $\mathbf{a}(\boldsymbol{x})$ then yields the following system of linear equations for the coefficient $\mathbf{a}(\boldsymbol{x})$:

$$\boldsymbol{A}(\boldsymbol{x})\mathbf{a}(\boldsymbol{x}) = \boldsymbol{B}(\boldsymbol{x})\mathbf{u},$$

where $\boldsymbol{A}(\boldsymbol{x})$ and $\boldsymbol{B}(\boldsymbol{x})$ can be easily obtained from Eq. (2), and $\boldsymbol{u}$ is the vector of nodal unknowns. If $\boldsymbol{A}$ is invertible, the coefficient $\mathbf{a}(\boldsymbol{x})$ can be expressed as

$$\mathbf{a}(\boldsymbol{x}) = \boldsymbol{A}^{-1}(\boldsymbol{x})\boldsymbol{B}(\boldsymbol{x})\mathbf{u}.$$

Substituting the above equation back into Eq. (1) leads to

$$u^h(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x})\boldsymbol{A}^{-1}(\boldsymbol{x})\boldsymbol{B}(\boldsymbol{x})\mathbf{u} = \boldsymbol{\varphi}^T(\boldsymbol{x})\mathbf{u},$$

where the shape function of the EFG method is given by

$$\boldsymbol{\varphi}^T(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x})\boldsymbol{A}^{-1}(\boldsymbol{x})\boldsymbol{B}(\boldsymbol{x}). \tag{3}$$

The MLS approximation is obtained by a special least squares method, thus the function obtained by the MLS approximation is smooth curve and it does not pass through the nodal values. Therefore, the MLS shape function does not, in general, satisfy the Kronecker delta condition at each node, i.e.

$$\varphi_i(\boldsymbol{x}_j) \neq \delta_{ij},$$

where $\varphi_i(\boldsymbol{x})$ is the i-component of $\boldsymbol{\varphi}^T(\boldsymbol{x})$. This brings great difficulties in exerting Dirichlet boundary condition. Today there have existed many techniques to resolve this problem, such as Lagrange multiplier approaches [11], penalty methods [11], modified variational principles [31], perturbed Lagrangian [32], coupling to finite elements [33] and so on. Among them penalty methods are adopted in the paper.

3. The variational multiscale element-free Galerkin method

3.1. The 2D Burgers' equation

The unsteady 2D Burgers' equation considered in the paper is defined over the square domain $\Omega = [0,1] \times [0,1]$, and described by the coupled equations as follows:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \tag{4.a}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right), \tag{4.b}$$

where u and v are the velocity along x-axis and y-axis respectively, Re is the Reynolds number. Here the following initial and boundary conditions are assumed:

$$u(x,y,0) = \sin(\pi x)\cos(\pi y), \quad v(x,y,0) = \cos(\pi x)\sin(\pi y), \tag{5}$$

$$u(0,y,t) = u(1,y,t) = 0, \quad v(x,0,t) = v(x,1,t) = 0, \tag{6}$$

$$\frac{\partial u}{\partial n}(x,0,t) = \frac{\partial u}{\partial n}(x,1,t) = 0, \quad \frac{\partial v}{\partial n}(0,y,t) = \frac{\partial v}{\partial n}(1,y,t) = 0. \tag{7}$$

For the 2D Burgers' equation mentioned above, there are various symmetries:

$$u(x,y,t) = v(y,x,t), \quad u(x,y,t) = -u(1-x,1-y,t).$$

Due to $u(x,y,t) = v(y,x,t)$, we think that the unknown variables contained in Eq. (4.a) are just about u, and then Eq. (4.a) can be solved independently. The same thing also happens to Eq. (4.b).

3.2. The standard weak form

Let $V \subset H^1(\Omega) \cap C^0(\Omega)$ denote the space of trial solutions and weighting functions for the unknown variables. The weak form is obtained by multiplying the governing equation with an admissible weighting function and integrating it over the domain. For Eq. (4.a), the standard weak form is as follows:

$$\left(w, \frac{\partial u}{\partial t}\right) + (w, \boldsymbol{a}\cdot\nabla u) + \left(\nabla w, \frac{1}{Re}\nabla u\right) = 0, \tag{8}$$

where $\boldsymbol{a} = (u, v)$, w is the weighting function for u, and $(\cdot,\cdot) = \int_\Omega (\cdot) d\Omega$ is the $L_2(\Omega)$ inner product.

3.3. Multiscale decomposition and the multiscale variational problem

We assume that the scalar field can be decomposed into the coarse scale and the fine scale, namely,

$$u = \bar{u} + \hat{u}, \tag{9}$$

where $\bar{u}$ and $\hat{u}$ are the coarse and fine scale parts, respectively. Subsequently, the trial function spaces of each scale are defined as

$$\bar{U} = \left\{u | u \in H^1(\Omega), u = g \text{ at } \Gamma_u\right\}, \quad \bar{u} \in \bar{U},$$
$$\hat{U} = \left\{u | u \in H^1(\Omega), u = 0 \text{ at } \Gamma_u\right\}, \quad \hat{u} \in \hat{U},$$
$$U = \bar{U} \oplus \hat{U},$$

where the function g is displacement boundary condition which is prescribed on smooth boundary Γ_u and equals to zero (see Eq. (6)) in the paper.

Likewise, test function can be also written in the decomposed form as follows:

$$w = \bar{w} + \hat{w}, \tag{10}$$

where $\bar{w}$ and $\hat{w}$ are the coarse and fine scale parts, respectively. We can define test function spaces of each scale as follows:

$$\overline{V} = \left\{w | w \in H^1(\Omega), w = 0 \text{ at } \Gamma_u\right\}, \quad \bar{w} \in \overline{V},$$
$$\widehat{V} = \left\{w | w \in H^1(\Omega), w = 0 \text{ at } \Gamma_u\right\}, \quad \hat{w} \in \widehat{V},$$
$$W = \overline{W} \oplus \widehat{W}.$$

After taking backward Euler as time discretization for Eq. (8) and linearization, we substitute the trial solutions Eq. (9) and the weighting functions Eq. (10) into the standard variational form to get

$$\left(\bar{w} + \hat{w}, \frac{\bar{u}^{n+1} + \hat{u}^{n+1}}{\Delta t}\right) + \left(\bar{w} + \hat{w}, \boldsymbol{a}^n \cdot \nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) + \left(\nabla(\bar{w} + \hat{w}), \frac{1}{Re}\nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) = \left(\bar{w} + \hat{w}, \frac{u^n}{\Delta t}\right), \tag{11}$$

where n and $n+1$ are used to denote two adjacent time points with time-step $\Delta t = t^{n+1} - t^n$, $\boldsymbol{a}^n = (u^n, v^n)$, $u^n = \bar{u}^n + \hat{u}^n$.

Employing the linearity of the weighting function slot, we can split Eq. (11) into the coarse and the fine scale parts, indicated as $\overline{W}$ and $\widehat{W}$ respectively.

$$\overline{W}: \left(\bar{w}, \frac{\bar{u}^{n+1} + \hat{u}^{n+1}}{\Delta t}\right) + \left(\bar{w}, \boldsymbol{a}^n \cdot \nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) + \left(\nabla\bar{w}, \frac{1}{Re}\nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) = \left(\bar{w}, \frac{u^n}{\Delta t}\right), \tag{12}$$
$$\widehat{W}: \left(\hat{w}, \frac{\bar{u}^{n+1} + \hat{u}^{n+1}}{\Delta t}\right) + \left(\hat{w}, \boldsymbol{a}^n \cdot \nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) + \left(\nabla\hat{w}, \frac{1}{Re}\nabla(\bar{u}^{n+1} + \hat{u}^{n+1})\right) = \left(\hat{w}, \frac{u^n}{\Delta t}\right). \tag{13}$$

Rearranging Eqs. (12) and (13) respectively, we have

$$\overline{W}: \left(\bar{w}, \frac{\bar{u}^{n+1}}{\Delta t}\right) + \left(\bar{w}, \boldsymbol{a}^n \cdot \nabla\bar{u}^{n+1}\right) + \left(\nabla\bar{w}, \frac{1}{Re}\nabla\bar{u}^{n+1}\right) = \left(\bar{w}, \frac{u^n}{\Delta t}\right) - \left(\bar{w}, \frac{\hat{u}^{n+1}}{\Delta t}\right) - \left(\bar{w}, \boldsymbol{a}^n \cdot \nabla\hat{u}^{n+1}\right) - \left(\nabla\bar{w}, \frac{1}{Re}\nabla\hat{u}^{n+1}\right), \tag{14}$$

$$\widehat{W}: \left(\hat{w}, \frac{\hat{u}^{n+1}}{\Delta t}\right) + \left(\hat{w}, \boldsymbol{a}^n \cdot \nabla\hat{u}^{n+1}\right) + \left(\nabla\hat{w}, \frac{1}{Re}\nabla\hat{u}^{n+1}\right) = \left(\hat{w}, \frac{u^n}{\Delta t}\right) - \left(\hat{w}, \frac{\bar{u}^{n+1}}{\Delta t}\right) - \left(\hat{w}, \boldsymbol{a}^n \cdot \nabla\bar{u}^{n+1}\right) - \left(\nabla\hat{w}, \frac{1}{Re}\nabla\bar{u}^{n+1}\right). \tag{15}$$

Remark. In VMFEM, the bounded domain Ω is discretized into non-overlapping regions Ω^e (element domains) with boundaries Γ^e, $e = 1, 2, \ldots, N$. During the decomposition of the scalar field into the coarse scale and the fine scale, people generally make an assumption that the fine scale vanishes identically over the element boundaries although non-zero within the elements, namely,

$$\hat{u} = \hat{w} = 0 \text{ on } \Gamma',$$

where $\Gamma' = \bigcup_{e=1}^{N} \Gamma^e$ (elem.boundaries). However, Hughes [17] regarded that this was a rather strong assumption, and there will be many cases of practical interest in which this assumption will be invalid. As far as the method presented in the paper is concerned, we can see that this assumption is not needed at all.

3.4. The fine scale approximation

In order to obtain the fine scale approximation, the knowledge about the partition of unity (PU) has been used [34–37]. The basic idea consists in the use of partition of unity functions (PU functions), a set of functions whose sum equals to the unity on the whole domain. Consider a set of functions $\{\phi_i\}$ and a domain Ω covered with a set of open domains $\{\Omega_i\}$ such that

$$\text{supp}(\phi_i) = \Omega_i,$$
$$\forall x \in \Omega, \quad \sum_i \phi_i = 1,$$

where supp(ϕ_i) denotes the support of definition of the function ϕ_i. This set of functions $\{\phi_i\}$ defines the partition of unity attached to the patch $\{\Omega_i\}$. We now consider the space of local approximation function V_i^j (local enrichment basis) defined on Ω_i :

$$v_i(\Omega_i) = \text{span}\left\{V_i^j\right\}.$$

The space of functions used for the approximation is obtained by the product of the PU functions and the local approximation functions

$$v(\Omega) = \text{span}\{\phi_i V_i^j\},$$

where span$\{\phi_i V_i^j\}$ denotes the space of functions generated by the set of function $\phi_i V_i^j$. Each node has several degrees of freedom (one per function V_i^j) and the approximation of the displacement field at point $\boldsymbol{x}$ is given by

$$u^h(\boldsymbol{x}) = \sum_i \sum_{V_i^j \in v_i} \phi_i V_i^j(\boldsymbol{x}) u_{i,j}. \tag{16}$$

An existing theorem (Babuška and Melenk, 1997) shows that if local approximation spaces have a convergent approximation property of a given function on Ω_i, then PU method space also has a convergent approximation property on Ω.

For a typical open domain Ω_i, local enrichment basis V_i^j may adopt the polynomial basis functions or any other analytical basis functions. Very often, the polynomial basis functions are used. In the following, some polynomial basis functions are presented [34].

First order ($p = 1$):

$$\{V_i^j\} = \{V_i^1\} = \{1\} \quad \text{in } 1D,$$
$$\{V_i^j\} = \{V_i^1\} = \{1\} \quad \text{in } 2D.$$

Second order ($p = 2$):

$$\left\{V_i^j\right\} = \left\{V_i^1, V_i^2\right\} = \left\{1, (x - x_i)^2\right\} \quad \text{in } 1D,$$
$$\left\{V_i^j\right\} = \left\{V_i^1, V_i^2, V_i^3\right\} = \left\{1, (x - x_i)^2, (y - y_i)^2\right\} \quad \text{in } 2D.$$

Third order ($p = 3$):

$$\{V_i^j\} = \{V_i^1, V_i^2, V_i^3\} = \left\{1, (x - x_i)^2, (x - x_i)^3\right\} \quad \text{in } 1D,$$

$$\begin{aligned}\left\{V_i^j\right\} &= \left\{V_i^1, V_i^2, V_i^3, V_i^4, V_i^5, V_i^6, V_i^7\right\} \\ &= \left\{1, (x - x_i)^2, (y - y_i)^2, (x - x_i)^3, (x - x_i)^2(y - y_i), (x - x_i)(y - y_i)^2, (y - y_i)^3\right\} \quad \text{in } 2D.\end{aligned}$$

Fourth order ($p = 4$):

$$\left\{V_i^j\right\} = \left\{V_i^1, V_i^2, V_i^3, V_i^4\right\} = \left\{1, (x - x_i)^2, (x - x_i)^3, (x - x_i)^4\right\} \quad \text{in } 1D,$$
$$\begin{aligned}\left\{V_i^j\right\} = \Big\{&V_i^1, V_i^2, V_i^3, V_i^4, V_i^5, V_i^6, V_i^7, V_i^8, V_i^9, V_i^{10}, V_i^{11}, V_i^{12}\Big\} = \Big\{1, (x - x_i)^2, (y - y_i)^2 \\ &(x - x_i)^3, (x - x_i)^2(y - y_i), (x - x_i)(y - y_i)^2, (y - y_i)^3, (x - x_i)^4, \\ &(x - x_i)^3(y - y_i), (x - x_i)^2(y - y_i)^2, (x - x_i)(y - y_i)^3, (y - y_i)^4\Big\} \quad \text{in } 2D.\end{aligned}$$

It is worth noting that the linear terms x, y and xy are excluded from the above lists. Inclusion of those linear terms in the polynomial often leads to some computational problems. Detailed discussion could be found in Ref. [38,39].

It should be pointed out that the set of EFG shape functions obtained by Eq. (3) belongs to PU functions, and if the polynomial basis functions are adopted as local enrichment basis V_i^j, then we can obtain one kind of expression form for Eq. (16). In order to state the following process simply, here the second order polynomial basis functions in 2D are used; other polynomial basis functions follow the same way. Therefore, we have

$$u^h(\boldsymbol{x}) = \sum_i \phi_i \Big(u_{i,0} + (x - x_i)^2 u_{i,1} + (y - y_i)^2 u_{i,2}\Big), \tag{17}$$

where ϕ_i is the EFG shape function. Rearranging Eq. (17), we obtain

$$u^h(\boldsymbol{x}) = \sum_i \phi_i u_{i,0} + \sum_i \phi_i (x - x_i)^2 u_{i,1} + \sum_i \phi_i (y - y_i)^2 u_{i,2}. \tag{18}$$

Compared with the approximation of classical EFG method, Eq. (18) has two more terms. Here we regard that the approximation of classical EFG method is coarse scale approximation and the added terms as a whole is fine scale approximation. Therefore, the coarse scale and the fine scale parts are descried respectively as follows:

$$\bar{u} = \sum_i \phi_i u_{i,0}, \tag{19}$$

$$\hat{u} = \sum_i \phi_i (x - x_i)^2 u_{i,1} + \sum_i \phi_i (y - y_i)^2 u_{i,2}. \tag{20}$$

Substituting Eqs. (19), (20) into Eqs. (14), (15), we will finally get the matrix form for $\overline{W}$ and $\widehat{W}$, respectively.

$$\overline{W}: \left[K_1^{n+1}\right]\left\{\bar{u}^{n+1}\right\} = \left\{F_1^n\right\} + \left\{F_2\left(\hat{u}^{n+1}\right)\right\}, \tag{21}$$

$$\widehat{W}: \left[K_2^{n+1}\right]\left\{\hat{u}^{n+1}\right\} = \left\{F_3^n\right\} + \left\{F_4\left(\bar{u}^{n+1}\right)\right\}. \tag{22}$$

Here, matrix $\left[K_1^{n+1}\right]$ is obtained from left-hand side of Eq. (14), vector $\left\{F_2\left(\hat{u}^{n+1}\right)\right\}$ from the right-hand side terms that contain $\hat{u}^{n+1}$, and $\left\{F_1^n\right\}$ from the rest. Similarly, matrix $\left[K_2^{n+1}\right]$ is obtained from left-hand side of Eq. (15), vector $\left\{F_4\left(\bar{u}^{n+1}\right)\right\}$ from the right-hand side terms that contain $\bar{u}^{n+1}$, and $\left\{F_3^n\right\}$ from the rest.

Introducing $(w, v)^* = \left(w, \frac{v}{\Delta t}\right) + \left(w, \boldsymbol{a}^n \cdot \nabla v\right) + \left(\nabla w, \frac{1}{Re}\nabla v\right)$, we have

$$\left[K_1^{n+1}\right] = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & (\phi_i, \phi_j)^* & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}_{N\times N},$$

$$\left[K_2^{n+1}\right] = \begin{bmatrix} \vdots & & \vdots & \vdots & & \vdots \\ \cdots & \left(\phi_i(x-x_i)^2, \phi_j(x-x_j)^2\right)^* & \cdots & \cdots & \left(\phi_i(x-x_i)^2, \phi_j(y-y_j)^2\right)^* & \cdots \\ \vdots & & \vdots & \vdots & & \vdots \\ \vdots & & \vdots & \vdots & & \vdots \\ \cdots & \left(\phi_i(y-y_i)^2, \phi_j(x-x_j)^2\right)^* & \cdots & \cdots & \left(\phi_i(y-y_i)^2, \phi_j(y-y_j)^2\right)^* & \cdots \\ \vdots & & \vdots & \vdots & & \vdots \end{bmatrix}_{2N\times 2N},$$

$$\left\{\bar{u}^{n+1}\right\} = \begin{bmatrix} \vdots \\ u_{j,0}^{n+1} \\ \vdots \end{bmatrix}_{N\times 1}, \quad \left\{\hat{u}^{n+1}\right\} = \begin{bmatrix} \vdots \\ u_{j,1}^{n+1} \\ \vdots \\ \vdots \\ u_{j,2}^{n+1} \\ \vdots \end{bmatrix}_{2N\times 1}$$

$$\left\{F_1^n\right\} = \begin{bmatrix} \vdots \\ \left(\phi_i, \frac{u^n}{\Delta t}\right) \\ \vdots \end{bmatrix}_{N\times 1}, \quad \left\{F_2\left(\hat{u}^{n+1}\right)\right\} = \begin{bmatrix} \vdots \\ -\left(\phi_i, \hat{u}^{n+1}\right)^* \\ \vdots \end{bmatrix}_{N\times 1},$$

$$\left\{F_3^n\right\} = \begin{bmatrix} \vdots \\ \left(\phi_i(x-x_i)^2, \frac{u^n}{\Delta t}\right) \\ \vdots \\ \vdots \\ \left(\phi_i(y-y_i)^2, \frac{u^n}{\Delta t}\right) \\ \vdots \end{bmatrix}_{2N\times 1}, \quad \left\{F_4\left(\bar{u}^{n+1}\right)\right\} \begin{bmatrix} \vdots \\ -\left(\phi_i(x-x_i)^2, \bar{u}^{n+1}\right)^* \\ \vdots \\ \vdots \\ -\left(\phi_i(y-y_i)^2, \bar{u}^{n+1}\right)^* \\ \vdots \end{bmatrix}_{2N\times 1},$$

where N is the number of nodes, $i = 1, 2, \ldots, N,\ j = 1, 2, \ldots, N$.

In order to obtain the numerical results, the coarse scale problem Eq. (21) and the fine scale problem Eq. (22) must be solved iteratively. The solution procedures are summarized as follows:

(1) $\bar{u}^{n+1,0} = u^n$;
(2) Solve the fine scale problem by replacing the coarse variable $\bar{u}^{n+1}$ in the right-hand side with $\bar{u}^{n+1,i}$ to determine $\hat{u}^{n+1,i+1}$;

$$\widehat{W}: \left[K_2^{n+1,i+1}\right]\left\{\hat{u}^{n+1,i+1}\right\} = \left\{F_3^n\right\} + \left\{F_4\left(\bar{u}^{n+1,i}\right)\right\}. \tag{23}$$

(3) Solve the coarse scale problem to determine $\bar{u}^{n+1,i+1}$;

$$\overline{W}: \left[K_1^{n+1,i+1}\right]\left\{\bar{u}^{n+1,i+1}\right\} = \left\{F_1^n\right\} + \left\{F_2\left(\hat{u}^{n+1,i+1}\right)\right\}. \tag{24}$$

(4) Employ Eq. (18) to get $u_{\text{real}}^{n+1,i}$ and $u_{\text{real}}^{n+1,i+1}$, then compute error $= \max\left|u_{\text{real}}^{n+1,i+1} - u_{\text{real}}^{n+1,i}\right|$;
(5) If error $< 10^{-5}$, then let $u^{n+1,i+1} \to u^n$, $n+1 \to n$ and go to (1); else go to (2)

4. Numerical results

4.1. The analysis of the method

In order to analyze the convergence behaviour of the proposed method, we first apply the method to solve two problems which have an available analytical solution of their own respectively. Although these two problems are the ones with constant coefficients, the obtained conclusions can be used to instruct us how to apply the proposed method to solve the nonlinear problems, such as 2D Burgers' equation.

4.1.1. Convection–diffusion problem

This problem simply consists of solving the following 1D boundary value problem

$$\begin{cases} au_x - vu_{xx} = s(x) \quad \text{in } [0, L], \\ u = 0 \text{ at } x = 0 \quad \text{and} \quad x = L, \end{cases}$$

with a constant source term $s = 1$ in a dimensionless domain $L = 1$. The exact solution to the above model problem is given by

$$u(x) = \frac{1}{a}\left(x - \frac{1 - \exp(\gamma x)}{1 - \exp(\gamma)}\right),$$

where $\gamma = \frac{a}{v}$. Additionally, in order to analyze the convergence rate of the proposed method, the following L^2 error norm is used as error measures:

$$\|u - u^h\|_{L^2} = \left(\int_\Omega \left(u - u^h\right)^2 d\Omega\right)^{1/2}$$

and the corresponding L^2 relative error norm $\|u - u^h\|_{L^2}^{rel} = \|u - u^h\|_{L^2} / \|u\|_{L^2}$ is also introduced.

In the paper, we adopt $a = 1$ and $v = 0.01$. The numerical results are obtained on a uniform distribution of nodes. We distribute 31, 61 and 91 nodes in the domain $[0, L]$ respectively, and different order of polynomial basis functions ($p = 2, 3$ and 4 in 1D) are also adopted respectively. Additionally, 3 Gaussian quadrature points in each integral cell are used and the influence radius of node is set as $1.15\Delta x$.

Due to that the numerical results with different order of polynomial basis functions ($p = 2, 3$ and 4 in 1D) are almost the same for the corresponding nodes, we only present the results with the polynomial basis functions $p = 4$ for different nodes. Fig. 1 shows the comparison of the numerical results with exact solution for 31, 61 and 91 nodes respectively, and we can observe that the numerical results agree well with the exact solution and the agreement becomes better with the increment of nodes.

In order to study the convergence of the method, in Fig. 2 the L^2 relative error norms are plotted with respect to the number of nodes and the order of polynomial basis functions respectively. From Fig. 2(a) we can know that with the increment of nodes the L^2 relative error norms become smaller, which is consistent with the error estimate theory. From Fig. 2(b) we can observe that with the enrichment of polynomial basis functions the L^2 relative error norms become smaller likewise. For the less nodes (e.g. 31 nodes), when we enrich the polynomial basis functions from $p = 2$ to $p = 3$, the effect to improve the convergence is evident. However, when we enrich the polynomial basis functions from $p = 3$ to $p = 4$, the effect to improve the convergence is a little. The reason is that most of fine scale information has been captured by the additional terms of the polynomial basis functions $p = 3$ compared with the polynomial basis functions $p = 2$, thus if we continue to enrich the polynomial basis functions, less fine scale information will be captured. For the enough nodes (e.g. 61 nodes, 91 nodes), due to that most of information has already been captured, the fine scale information left is little, so the effect to improve the convergence is not evident by the enrichment of polynomial basis functions.

4.1.2. Convection–diffusion across a source term

The proposed method is also studied by introducing a source term to the previous example:

$$\begin{cases} au_x - vu_{xx} = 10e^{-5x} - 4e^{-x} & \text{for } x \in [0,1], \\ u(0) = 0 \quad \text{and} \quad u(1) = 1. \end{cases}$$

Introducing the following parameters:

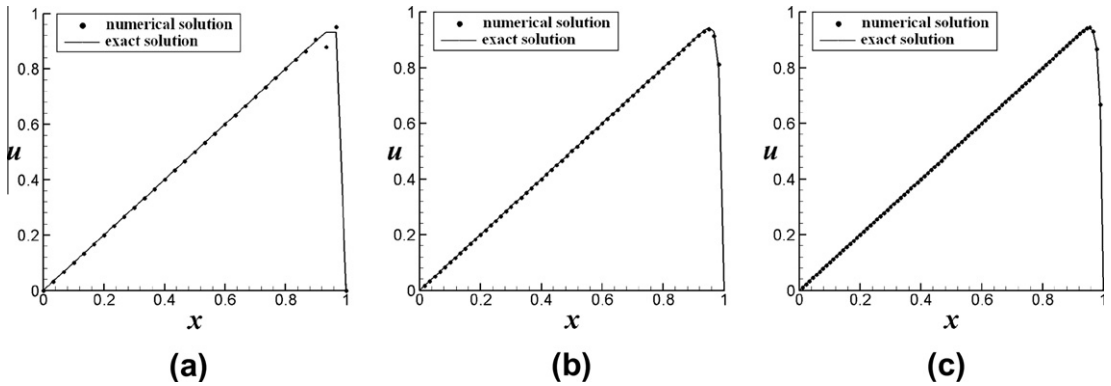

Fig. 1. The numerical results of the proposed method with the polynomial basis functions $p = 4$ and exact solution for different nodes respectively: (a) 31 nodes (b) 61 nodes (c) 91 nodes.

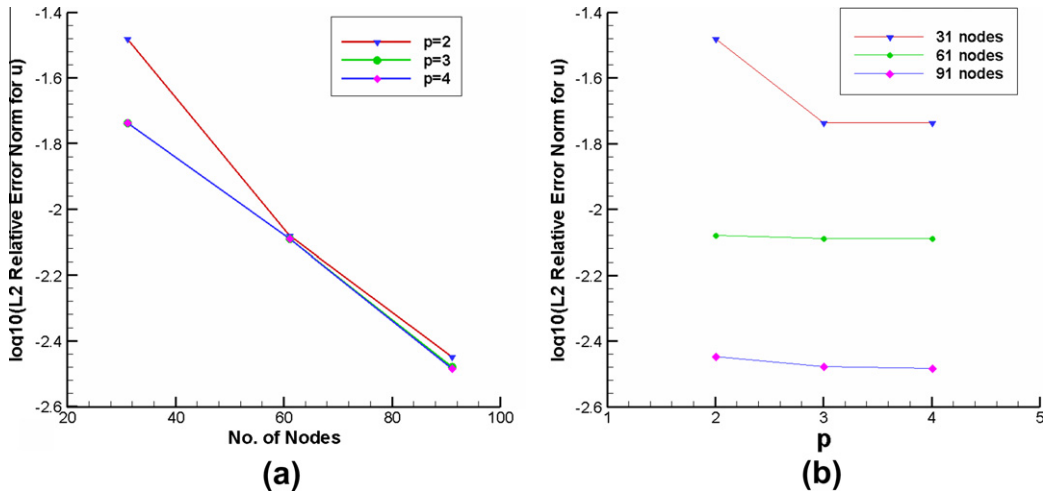

Fig. 2. The rate of convergence about: (a) the number of nodes (b) the order of polynomial basis functions.

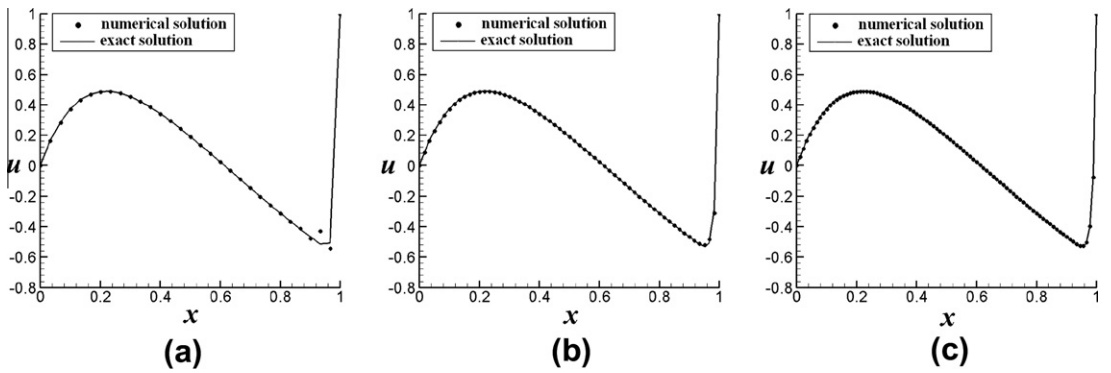

Fig. 3. The numerical results of the proposed method with the polynomial basis functions $p = 4$ and exact solution for different nodes respectively: (a) 31 nodes (b) 61 nodes (c) 91 nodes.

$$\gamma = \frac{a}{v}, \quad C_1 = \frac{2}{5+\gamma}, \quad C_2 = \frac{4}{1+\gamma}, \quad C_3 = \frac{1}{1-e^{\gamma}}\left(C_1 - C_2 - v - C_1 e^{-5} + C_2 e^{-1}\right),$$

the exact solution can be given by

$$u(x) = \frac{1}{v}\left(-C_1 e^{-5x} + C_2 e^{-x} + C_3 e^{\gamma x} + C_1 - C_2 - C_3\right).$$

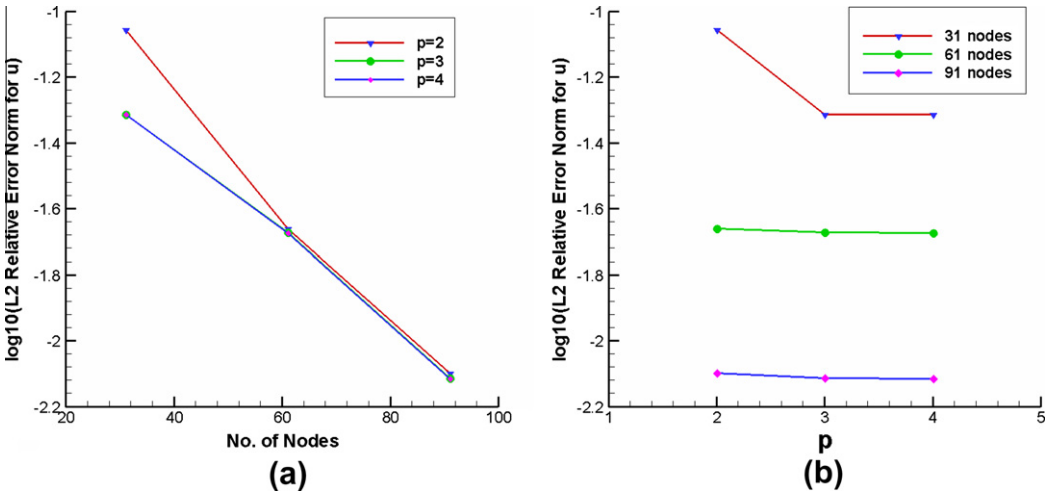

Fig. 4. The rate of convergence about: (a) the number of nodes (b) the order of polynomial basis functions.

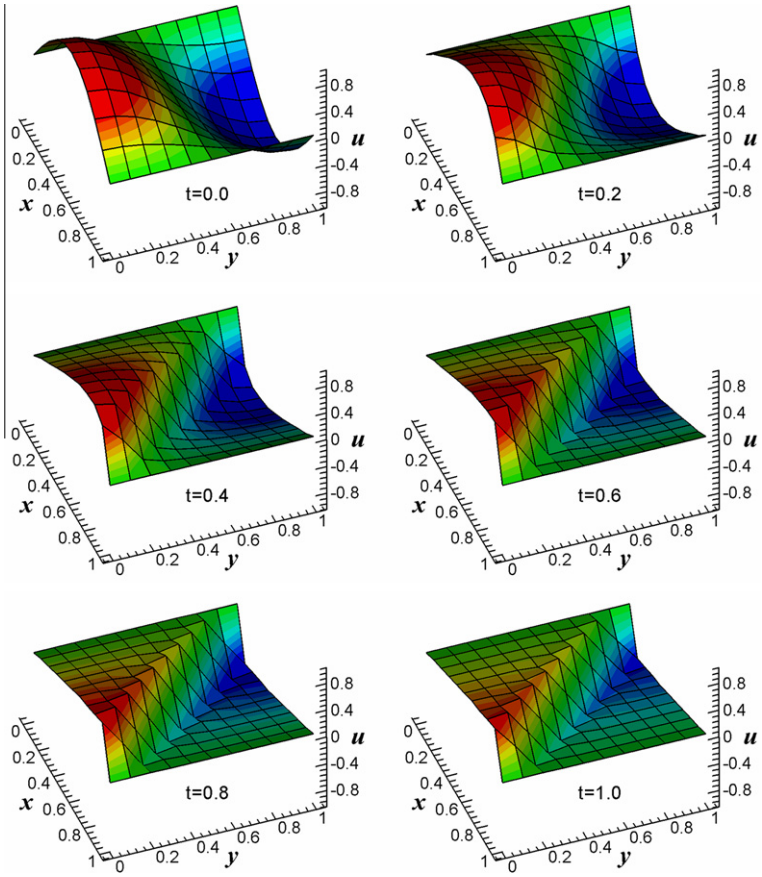

Fig. 5. The profiles of u velocity at different time for $Re = 100$ on 11×11 nodes.

Here, all the parameters are kept the same as those for the previous example. Following the same reason mentioned above, we also only present the results with the polynomial basis functions $p = 4$ for different nodes. Fig. 3 shows the comparison of the numerical results with exact solution for 31, 61 and 91 nodes respectively. Fig. 4 presents the L^2 relative error norms with respect to the number of nodes and the order of polynomial basis functions respectively. It is evident that the same conclusions such as those in Section 4.1.1 can be drawn from Figs. 3 and 4.

4.2. The 2D Burgers' equation

For 2D Burgers' equation, when convection dominates the nonlinear transport, the solution includes the formation of a discontinuity along the diagonal of the domain passing through the points $(0,1)$ and $(1,0)$. Here, we apply the proposed

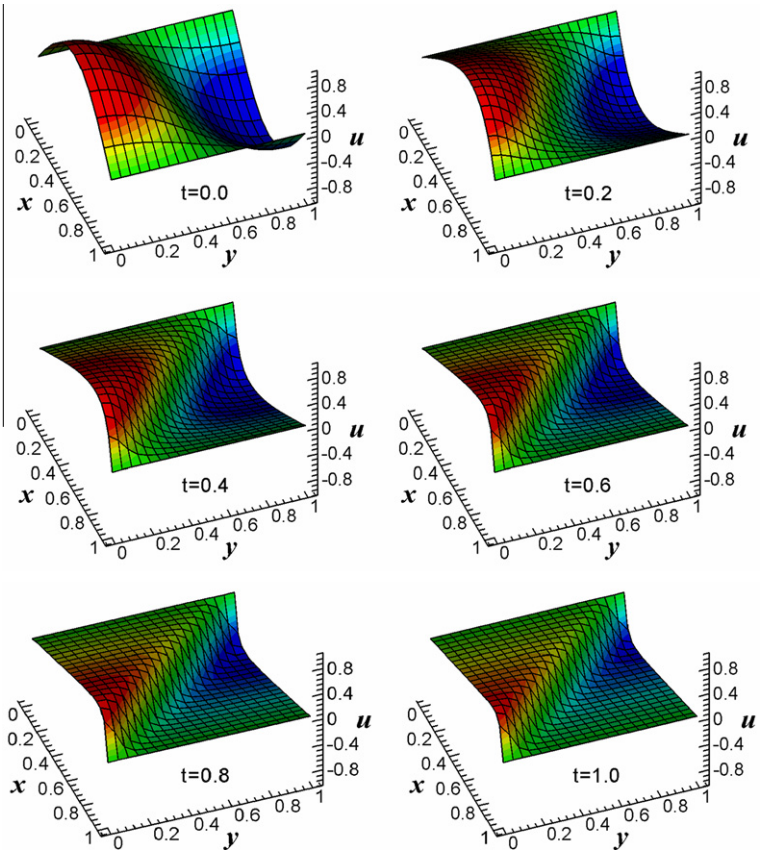

Fig. 6. The profiles of u velocity at different time for $Re = 100$ on 21×21 nodes.

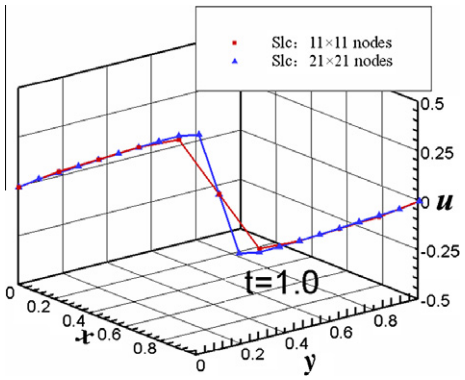

Fig. 7. The slice of u velocity passing through the points $(0,0)$ and $(1,1)$ for $t = 1.0$ when $Re = 100$.

method with the polynomial basis functions$p = 2$ to solve 2D Burgers' equation for different nodes, due to that the additional terms appeared in the higher order polynomial basis functions ($p > 2$) captures fine scale information little.

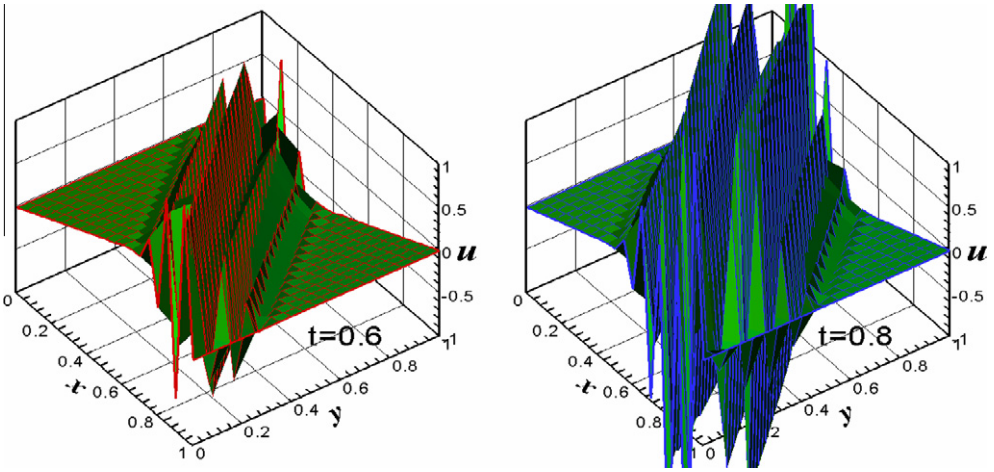

Fig. 8. The EFG's results on 21×21 nodes for $t = 0.6$ and $t = 0.8$ when $Re = 100$.

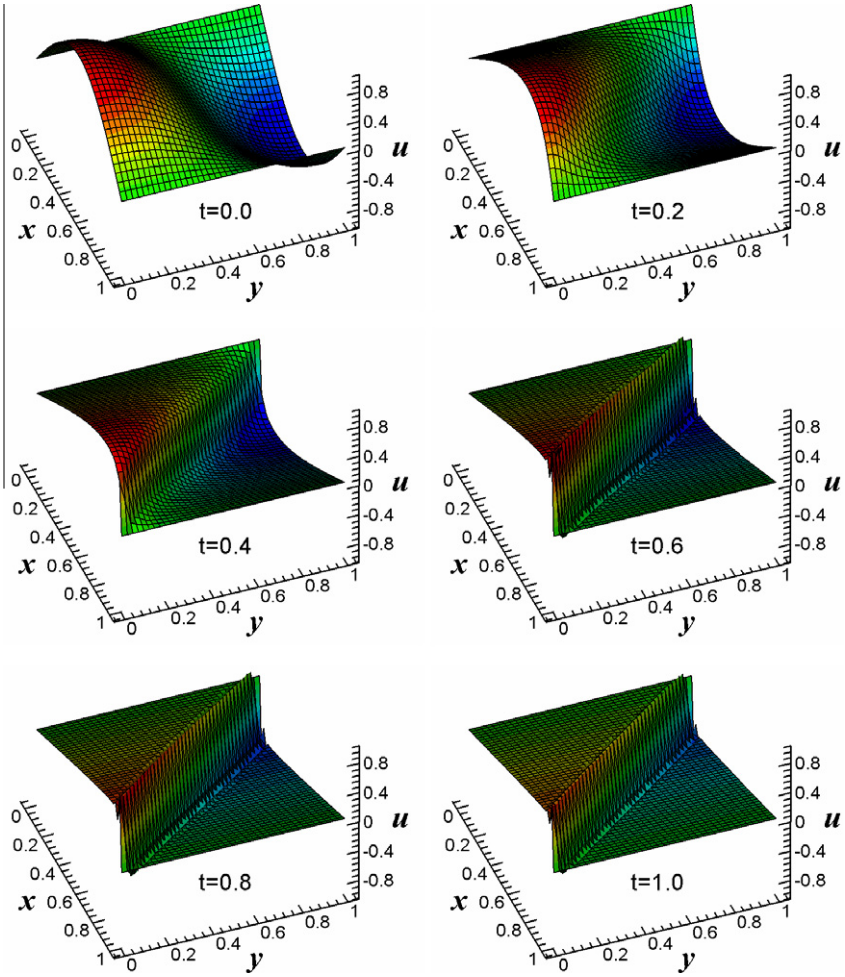

Fig. 9. The profiles of u velocity at different time for $Re = 500$ on 41×41 nodes.

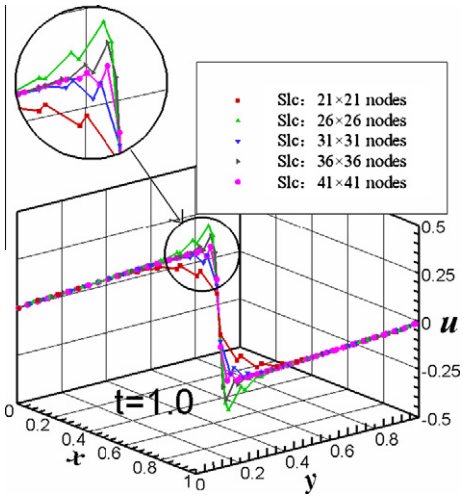

Fig. 10. The slice of u velocity passing through the points (0,0) and (1,1) for $t = 1.0$ when $Re = 500$.

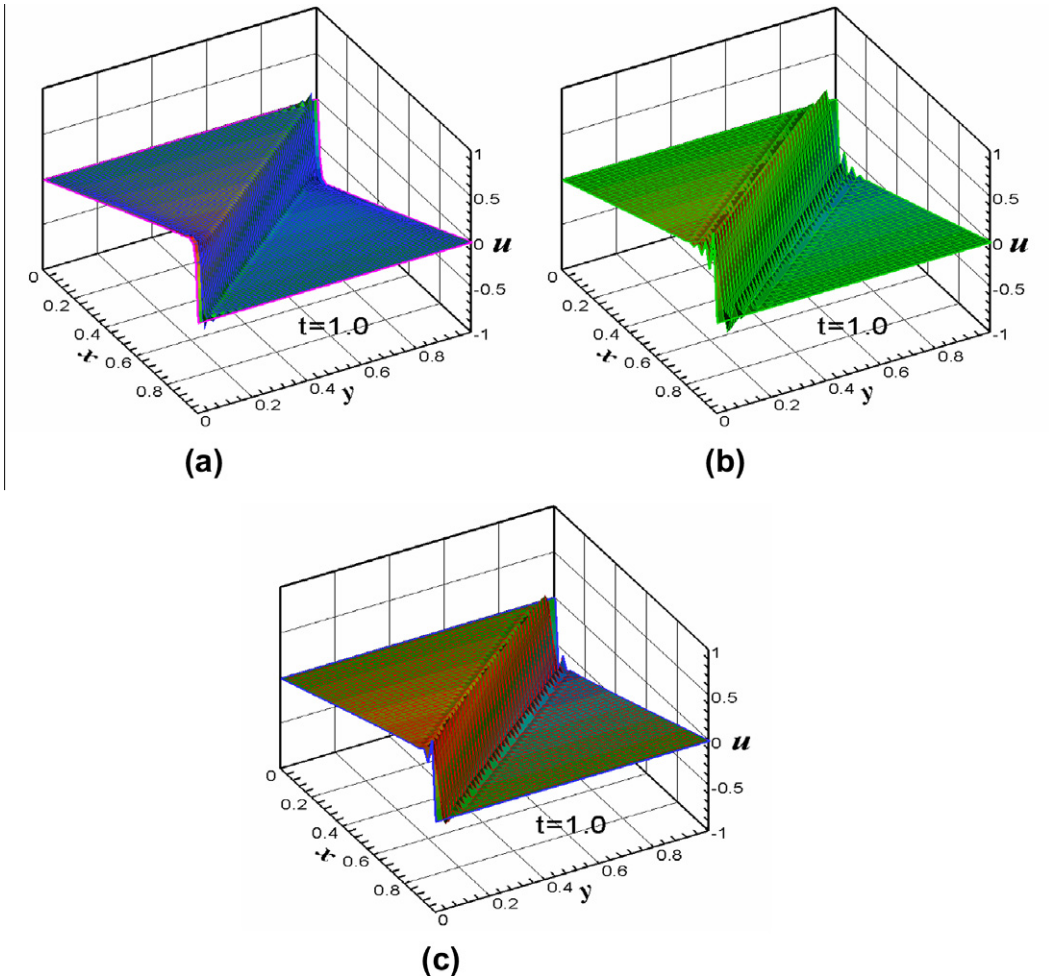

Fig. 11. The profiles of u velocity at $t = 1.0$ for $Re = 500$: (a) WCM on 65×65 nodes (b) EFCG on 41×41 nodes (c) VMEFG on 41×41 nodes.

The numerical results are obtained on a uniform distribution of nodes. We distribute $11 \times 11, 21 \times 21$ nodes respectively for $Re = 100$, and 41×41 nodes for $Re = 500$. 3×3 Gaussian quadrature points in each integral cell are used and the influence radius of node is set as $1.2\Delta x$ or $1.2\Delta y$ ($\Delta x = \Delta y$ in the paper).

Figs. 5 and 6 show the profiles of u velocity at different time for $Re = 100$ on $11 \times 11, 21 \times 21$ nodes, respectively. Fig. 7 presents the slice of Figs. 5 and 6 passing through the points $(0,0)$ and $(1,1)$ for $t = 1.0$. From Fig. 7 we can find that with the increment of nodes, the profiles along the diagonal of the domain passing through the points $(0,1)$ and $(1,0)$ are captured better. Fig. 8 presents the EFG's results on 21×21 nodes for $t = 0.6$ and $t = 0.8$ when $Re = 100$, in which the results are polluted extremely by non-physical oscillations. This illustrates that our method can eliminate the non-physical oscillations effectively.

Fig. 9 shows the profiles of u velocity at different time for $Re = 500$ on 41×41 nodes, from which we can find that though Re is much bigger, the results are polluted little by the non-physical oscillations. When $Re = 500$, Fig. 10 presents the slice of u velocity passing through the points $(0,0)$ and $(1,1)$ for $t = 1.0$ on $21 \times 21, 26 \times 26, 31 \times 31, 36 \times 36$ and 41×41 nodes respectively. We can observe that the discontinuity can be captured better with the increment of nodes.

In order to decide whether our results are reliable and accurate, we compare them with the counterparts computed by two other methods for $t = 1.0$ when $Re = 500$. First, the wavelet is characterized by multiresolution, thus we use the wavelet collocation method (WCM) [40] to compute 2D Burgers' equation of $Re = 500$ for supplying the reference results. Secondly, the element-free characteristic Galerkin (EFCG) method [16] is based on characteristic Galerkin (CG) which is an effective method to deal with the advection-dominated, thus we adopt EFCG as anther method to supply the reference results.

Fig. 11 shows the profiles of u velocity at $t = 1.0$ for $Re = 500$, which are computed by different methods: (a) WCM on 65×65 nodes (b) EFCG on 41×41nodes (c) VMEFG on 41×41 nodes. It should be noticed that in WCM the wavelet selected is the Shanno wavelet and scale parameter $J = 6$, thus the number of nodes is $(2^J + 1) \times (2^J + 1)$. Fig. 12 presents the comparison among the results of these three methods, from which we can find that: (1) The WCM method captures the discontinuity best in the interior of the domain, there is no oscillations appeared along the diagonal of the domain passing through the points $(0,1)$ and $(1,0)$. However, the results on the Neumann boundary are worse, because it is difficult to exert

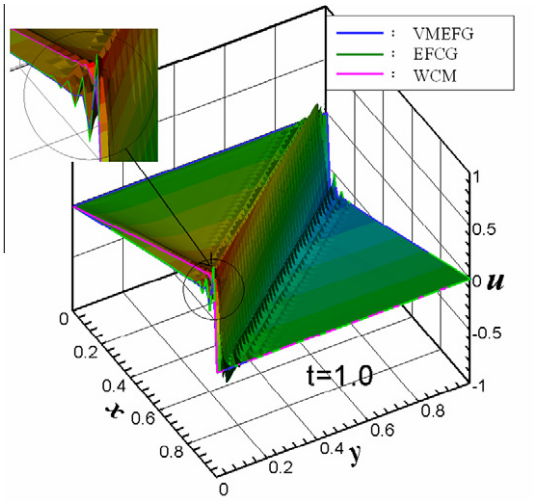

Fig. 12. The comparison of the results at $t = 1.0$ for $Re = 500$ among the methods: (a) WCM on 65×65 nodes (b) EFCG on 41×41 nodes (c) VMEFG on 41×41 nodes.

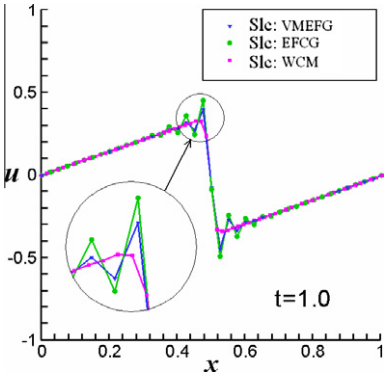

Fig. 13. The slice of u velocity at the section $y = 0.5$ for $t = 1.0$ when $Re = 500$ by different methods: (a) WCM on 65×65 nodes (b) EFCG on 41×41 nodes (c) VMEFG on 41×41 nodes.

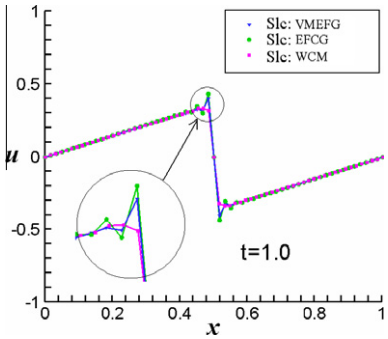

Fig. 14. The slice of u velocity at the section $y = 0.5$ for $t = 1.0$ when $Re = 500$ by different methods: (a) WCM on 65×65 nodes (b) EFCG on 61×61 nodes (c) VMEFG on 61×61 nodes.

this kind of boundary condition exactly. (2) When the same 41×41 nodes are adopted, the oscillations appeared in our results grow weaker than those appeared in EFCG's results, which illustrates that our method is better than EFCG. Meanwhile, unlike WCM, it is easy to exert the Neumann boundary condition exactly in our method.

Fig. 13 presents the slice of u velocity in Fig. 12 at the section $y = 0.5$, from which we can observe clearly that our method is better than EFCG, though a little worse than WCM. Due to that 41×41nodes are much less than 65×65 nodes, thus we compute the problem on 61×61nodes by our method and EFCG respectively. Fig. 14 shows the slice of u velocity at the section $y = 0.5$, from which we can find that when the nodes nearly keep the same, our results almost overlap the WCM's results except a little oscillations.

Based on the comparison mentioned above, it should be seen that our proposed method is reliable and accurate for solving 2D Burgers' equations.

5. Conclusions

A new numerical method, which is based on the coupling between VM and meshfree methods, is developed for the Burgers' equation in the paper. The proposed method avoids a strong assumption compared with VMFEM, that is, the fine scale vanishes identically over the element boundaries although non-zero within the elements. Then two problems which have an available analytical solution of their own are solved to analyze the convergence behaviour of the proposed method and finally a 2D Burgers' equation having large Re is also solved. Numerical results show that: (1) the proposed method has the ability to deal with the problem in which a convective phenomenon is dominative and at the same time it does not refer to the choice of a proper stabilization parameter; (2) with the increment of nodes or the enrichment of polynomial basis functions, the accuracy of results is improved; (3) most of fine scale information can be captured by the additional terms of the polynomial basis functions $p = 3$ compared with the polynomial basis functions $p = 2$, thus if we continue to enrich the polynomial basis functions, less fine scale information will be captured; (4) no mesh generation and mesh recreation are involved, therefore, one can easily add or delete nodes in the desired regions. In a word, the proposed method is an attractive approach to deal with nonlinear and high gradient problems. It is believed that this method will also be useful for solving more general problems in fluid dynamics, and in the future we will apply this approach to solve more complex fluid flow problems, such as Navier–Stokes problem and so on.

Acknowledgments

The support from the Natural Sciences Foundation of China (Nos. 10590353 and 10871159), the National Basic Research Program of China (No. 2005CB321704) are fully acknowledged.

References

[1] J.D. Cole, On a quasi-linear parabolic equation occurring in aerodynamic, Quart. Appl. Math. 19 (1951) 225–236.
[2] E. Hopf, The partial differential equation $u_t + uu_x = \upsilon u_{xx}$, Commun. Pure Appl. Math. 3 (1950) 201–230.
[3] M. Gulsu, A finite difference approach for solution of Burgers' equation, Appl. Math. Comput. 175 (2006) 1245–1255.
[4] I.A. Hassanien, A.A. Salama, H.A. Hosham, Fourth-order finite difference method for solving Burgers' equation, Appl. Math. Comput. 170 (2005) 781–800.
[5] B. Bihari, A. Harten, Multiresolution schemes for the numerical solution of 2D conservation laws, SIAM J. Sci. Comput. 18 (1997) 315–354.
[6] M.O. Domingues, O. Roussel, K. Schneider, An adaptive multiresolution method for parabolic PDEs with time-step control, Int. J. Numer. Meth. Eng. 78 (2009) 652–670.
[7] O. Roussel, K. Schneider, A. Tsigulin, H. Bockhorn, A conservative fully adaptive multiresolution algorithm for parabolic PDEs, J. Comput. Phys. 188 (2003) 493–523.
[8] E.N. Aksan, A numerical solution of Burgers' equation by finite element method constructed on the method of discretization in time, Appl. Math. Comput. 170 (2005) 895–904.

[9] A. Dogan, A Galerkin finite element approach to Burgers' equation, Appl. Math. Comput. 157 (2004) 331–346.
[10] E. Chino, N. Tosaka, Dual reciprocity boundary element analysis of time-independent Burgers' equation, Eng. Anal. Boundary Elem. 21 (1998) 261–270.
[11] T. Belytschko, Y. Krongauz, D. Organ, Meshless method: an overview and recent developments, Comput. Methods Appl. Mech. Eng. 139 (1996) 3–47.
[12] T. Belytschko, Y.Y. Lu, L. Gu, Element-free Galerkin methods, Int. J. Numer. Methods Eng. 37 (1994) 229–256.
[13] G.R. Liu, Y.T. Gu, An Introduction to Meshfree Methods and Their Programming, Springer, Berlin and New York, 2005.
[14] J. Ouyang, L. Zhang, X.H. Zhang, Nonstandard element-free Galerkin method for solving unsteady convection dominated problems, Acta. Aerodyn. Sin. 25 (2007) 287–293.
[15] D.L. Young, C.M. Fan, S.P. Hu, S.N. Atluri, The Eulerian–Lagrangian method of fundamental solutions for two-dimensional unsteady Burgers' equations, Eng. Anal. Boundary Elem. 32 (2008) 395–412.
[16] X.H. Zhang, J. Ouyang, L. Zhang, Element-free Characteristic Galerkin method for Burgers' equation, Eng. Anal. Boundary Elem. 33 (2009) 356–362.
[17] T.J.R. Hughes, Multiscale phenomena: Green's function, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Eng. 127 (1995) 387–401.
[18] T.J.R. Hughes, G.R. Feijoo, M. Luca, Q. Jean-Baptiste, The variational multiscale method - a paradigm for computational mechanics, Comput. Methods Appl. Mech. Eng. 166 (1998) 3–24.
[19] A. Masud, On a stabilized finite element formulation for incompressible Navier–Stokes equations, in: Proceedings of the Fourth US–Japan Conference on Computational Fluid Dynamics, Tokyo Japan, May 2002, pp. 28–30.
[20] A. Masud, R.A. Khurram, A multiscale/stabilized finite element method for the advection–diffusion equation, Comput. Methods Appl. Mech. Eng. 193 (2004) 1997–2018.
[21] M. Ayub, A. Masud, A new stabilized formulation for convective–diffusive heat transfer, Numer. Heat Transfer 43 (2003) 601–625.
[22] A. Masud, T.J.R. Hughes, A stabilized mixed finite element method for Darcy flow, Comput. Methods Appl. Mech. Eng. 1991 (2002) 4341–4370.
[23] A. Masud, L.A. Bergman, Application of multiscale finite element methods to the solution of the Fokker–Planck equation, Comput. Methods Appl. Mech. Eng. 194 (2005) 1513–1526.
[24] L.P. Franca, A. Nesliturk, M. Stynes, On the stability of residual-free bubbles for convection–diffusion problems and their approximation by a two-level finite element method, Comput. Methods Appl. Mech. Eng. 166 (1998) 35–49.
[25] L.P. Franca, A. Nesliturk, On a two-level finite element method for the incompressible Navier–Stokes equations, Int. J. Numer. Meth. Eng. 52 (2001) 433–453.
[26] L. Zhang, J. Ouyang, X.H. Zhang, The two-level element free Galerkin method for MHD flow at high Hartmann numbers, Phys. Lett. A 372 (2008) 5625–5638.
[27] L. Zhang, J. Ouyang, X.H. Zhang, On a two-level element free Galerkin method for incompressible fluid flow, Appl. Numer. Math. 59 (2009) 1894–1904.
[28] L. Zhang, J. Ouyang, X.H. Zhang, W.B. Zhang, On a multiscale element free Galerkin method for the Stokes problem, Appl. Math. Comput. 203 (2008) 745–753.
[29] J.H. Yeon, S.K. Youn, Meshfree analysis of softening elastoplastic solids using variational multiscale method, Int. J. Solids Struct. 42 (2005) 4030–4057.
[30] J.H. Yeon, S.K. Youn, Variational multiscale analysis of elastoplastic deformation using meshfree approximation, Int. J. Solids Struct. 45 (2008) 4709–4724.
[31] Y.Y. Lu, T. Belytschko, L. Gu, A new implementation of the element-free Galerkin method, Comput. Methods Appl. Mech. Eng. 113 (1994) 397–414.
[32] Y.A. Chu, B. Moran, A computational model for nucleation of the solid-solid phase transformations, Model. Simul. Mater. Sci. Eng. 3 (1995) 455–471.
[33] T. Belytschko, D. Organ, Y. Krongauz, A coupled finite element-element-free Galerkin method, Comput. Mech. 17 (1995) 186–195.
[34] S.C. Fan, X. Liu, C.K. Lee, Enriched partition of unity finite element method for stress intensity factors at crack tips, Comput. Struct. 82 (2004) 445–461.
[35] L. Hazard, P. Bouillard, Structural dynamics of viscoelastic sandwich plates by the partition of unity finite element method, Comput. Methods Appl. Eng. 196 (2007) 4101–4116.
[36] S.C. Li, S.C. Li, Y.M. Cheng, Enriched meshless manifold method for two-dimensional crack modeling, Theor. Appl. Fract. Mec. 44 (2005) 234–248.
[37] J.M. Melenk, I. Babuška, The partition of unity finite element method: Basic theory and applications, Comput. Methods Appl. Eng. 139 (1996) 289–314.
[38] J.T. Oden, C.A. Duarte, O.C. Zienkiewicz, A new cloud-based hp finite element method, Int. J. Numer. Methods Eng. 50 (1998) 160–170.
[39] R. Tian, G. Yagawa, H. Terasaka, Linear dependence problems of partition of unity-based generalized FEMs, Comput. Methods Appl. Mech. Eng. 195 (2006) 4768–4782.
[40] S. Bertoluzza, G. Naldi, A wavelet collocation method for the numerical solution of partial differential equations, Appl. Comput. Harmon. A. 3 (1996) 1–9.