Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 206 (2005) 182–207
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

The least-squares meshfree method for the steady incompressible viscous flow

Xiang Kun Zhang *, Kie-Chan Kwon, Sung-Kie Youn

Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 373-1, Gusung-dong, Yusung-ku, Taejon 305-701, Republic of Korea

Received 19 July 2004; accepted 25 November 2004
Available online 28 January 2005

Abstract

A least-squares meshfree method (LSMFM) based on the first-order velocity–pressure–vorticity formulation for two-dimensional steady incompressible viscous flow is presented. The discretization of all governing equations is implemented by the least-squares method. The equal-order moving least-squares (MLS) approximation is employed. Gauss quadrature is used in the background cells constructed by the quadtree algorithm and the boundary conditions are enforced by the penalty method. The matrix-free element-by-element Jacobi preconditioned conjugate method is applied to solve the discretized linear systems. A numerical example with analytical solution for the Stokes problem is devised to analyze the error estimates of the LSMFM. Also, cavity-driven flow for the Stokes problem and the flow past a circular cylinder at low Reynolds numbers for the steady incompressible viscous flow are solved. Through the comparisons of the LSMFM's results with other experimental and numerical results, the numerical features of the presented LSMFM are investigated and discussed.

Keywords: Least-squares; Meshfree method; Moving least-squares approximation; Incompressible Navier–Stokes equations; Stokes equation; Error estimates; Equal-order interpolations

1. Introduction

In computational fluid dynamics (CFD), various finite elements methods, finite difference methods or finite volume methods for incompressible fluid flow have been developed. Typically four approaches are

* Corresponding author. Present address: No. 502-1-102, Xiangronglii Building, Lubei District, Tangshan City, Hebei Province 063000, PR China, Tel.: +86 315 204 7284.
E-mail addresses: xkzhang97@hotmail.com (X.K. Zhang), kkc@skylab.kaist.ac.kr (K.-C. Kwon), skyoun@sorak.kaist.ac.kr (S.-K. Youn).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2004.11.033

commonly taken to implement the discretization process in finite element methods based on the velocity–pressure formulation. They are the classical Galerkin mixed method [1] (including the projection methods [2]), penalty method [3], streamline upwind Petrov–Galerkin (SUPG) method [4] and least-squares method [5], respectively. The classical Galerkin mixed method is restricted by Ladyzhenskaya–Babuška–Brezzi (LBB) condition. The resulting algebraic matrix is non-symmetric and some oscillations on the result of pressure are observed. In the penalty method, the penalty parameter affects the accuracy and convergence of the solution. In the SUPG method, resulting algebraic system is non-symmetric. Thus the leading three methods are not always satisfactory methods for large-scale problems in CFD. Compared with the former three methods, least-squares method is robust, which is based on the minimization of the squared residuals. The least-squares method can overcome the above difficulties. It can reduce oscillations and instability of the solutions from the methods based on Galerkin formulation, and its resulting system matrix is symmetric and positive definite; it is easier to use equal-order approximations on all variables which are computed in the fully coupled manner and can be efficiently solved by iterative methods for the large-scale computation; no special treatments, such as upwinding or adjustable parameters are required.

In attempts to reduce the meshing-related difficulties, many meshfree methods have been developed. Among them are the smooth particle hydrodynamics (SPH) [6,7], generalized finite difference method (GFDM) [8], element-free Galerkin method (EFGM) [9,10], reproducing kernel particle method (RKPM) [11,12], partition of unity finite element method (PUM) [13], *hp*-cloud method [14,15], meshless local Petrov–Galerkin approach (MLPG) [16], diffuse element method (DEM) [17], etc. Meshfree method does not involve remeshing process and easy to realize adaptivity strategy. Besides these, the first derivatives of all variables are continuous in the whole computation domain even if the linear basis function is used, which is impossible for C^0 element in FEM. For CFD problems, the meshfree methods have recently been employed. Liu et al. [18] used the reproducing kernel particle method (RKPM) with SUPG formulation to solve 2D advection–diffusion equation. Sadat and Couturier [19] employed the diffuse element method (DEM) with the project method to study the laminar natural convection problem. Yagawa and Shirazaki [20] applied the free mesh method (FMM) with the weighted residual-Galerkin method to unsteady two- dimensional incompressible viscous flow. Cheng and Liu [21] adopted the finite point method (FPM) with the discretization defined by the positions of points to analyze two-dimensional driven cavity flow. Kim and Kim [22] presented some analyses of fluids by meshfree point collocation method (MPCM).

Recently, the least-squares meshfree method (LSMFM) was proposed by Park and Youn [23]. It combines the advantages of least-square method and meshfree approximation to devise a so-called truly meshfree method. Its effectiveness comes from the robustness of the least-squares method to integration errors. This implies that simple schemes for generating integration points can be used without degrading the solution accuracy. The convergences under inaccurate integration and an adaptive scheme with a posteriori error estimates for LSMFM have been studied [24,25]. However, the works with LSMFM has been limited to linear problems such as Poisson equation, and thus its application to various engineering fields should be investigated. In this paper, the LSMFM based on the first-order velocity–pressure–vorticity formulation for two- dimensional steady incompressible viscous flow is presented and through numerical examples its validity and performance are studied.

2. Velocity–pressure–vorticity formulation

Introducing the vorticity as an independent variable, the second-order velocity–pressure formula for the steady incompressible Navier–Stokes equations can be reduced to the first-order velocity–pressure–vorticity expression:

$$\begin{aligned}
&\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0,\\
&u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{\partial p}{\partial x}+\frac{1}{Re}\frac{\partial \omega}{\partial y}=f_x,\\
&u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{\partial p}{\partial y}-\frac{1}{Re}\frac{\partial \omega}{\partial x}=f_y,\\
&\omega+\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}=0, \qquad \text{in } \Omega,
\end{aligned} \tag{1}$$

Here all variables are non-dimensionalized, u and v are the component of the velocity, p the pressure, ω the z-component of vorticity, (f_x, f_y) the body force vector, and the Reynolds number Re is defined as:

$$Re = V_0 D/v, \tag{2}$$

where V_0 is the characteristic velocity, D is the characteristic length of the domain Ω and v is the kinematic viscosity.

The convective term can be linearized by the successive substitution or Newton's method, and then Eq. (2) can be expressed into another form:

$$\begin{aligned}
&\frac{\partial u_{k+1}}{\partial x}+\frac{\partial v_{k+1}}{\partial y}=0,\\
&u_{k+1}\frac{\partial u_{k+1}}{\partial x}+v_{k+1}\frac{\partial u_{k+1}}{\partial y}+\frac{\partial p_{k+1}}{\partial x}+\frac{1}{Re}\frac{\partial \omega_{k+1}}{\partial y}=f_x,\\
&u_{k+1}\frac{\partial v_{k+1}}{\partial x}+v_{k+1}\frac{\partial v_{k+1}}{\partial y}+\frac{\partial p_{k+1}}{\partial y}-\frac{1}{Re}\frac{\partial \omega_{k+1}}{\partial x}=f_y,\\
&\omega_{k+1}+\frac{\partial u_{k+1}}{\partial y}-\frac{\partial v_{k+1}}{\partial x}=0.
\end{aligned} \tag{3}$$

For example, the successive substitution for the convective term is given as

$$v_{k+1}\frac{\partial u_{k+1}}{\partial y}\approx v_k\frac{\partial u_{k+1}}{\partial y}. \tag{4}$$

The convective term is linearized by the Newton's method:

$$v_{k+1}\frac{\partial u_{k+1}}{\partial y}\approx v_k\frac{\partial u_{k+1}}{\partial y}+v_{k+1}\frac{\partial u_k}{\partial y}-v_k\frac{\partial u_k}{\partial y}, \tag{5}$$

where the subscripts k and $k+1$ denote the kth and the $(k+1)$th linearization step, respectively.

After linearization processes, Eq. (3) can be written in a general form of a first-order system:

$$\mathbf{Lu}=\mathbf{f}, \tag{6}$$

where $\mathbf{L}$ is the first-order differential operator and can be written as:

$$\mathbf{L}=\mathbf{A}_1\frac{\partial}{\partial x}+\mathbf{A}_2\frac{\partial}{\partial y}+\mathbf{A}_0. \tag{7}$$

The coefficient matrix $\mathbf{A}_i$, the force vector $\mathbf{f}$ and $\mathbf{u}$ (including velocity, pressure and vorticity) are given as follows for the successive substitution:

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ u_k & 0 & 1 & 0 \\ 0 & u_k & 0 & -1/Re \\ 0 & -1 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ v_k & 0 & 0 & 1/Re \\ 0 & v_k & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$
$$\mathbf{A}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{f} = \begin{Bmatrix} 0 \\ f_x \\ f_y \\ 0 \end{Bmatrix}, \quad \mathbf{u} = \begin{Bmatrix} u \\ v \\ p \\ \omega \end{Bmatrix}_{k+1}. \tag{8}$$

For the Newton's linearization, the coefficient matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ are same as those from successive substitution, but matrix $\mathbf{A}_0$ and force vector $\mathbf{f}$ are different and given as follows:

$$\mathbf{A}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{\partial u_k}{\partial x} & \frac{\partial u_k}{\partial y} & 0 & 0 \\ \frac{\partial v_k}{\partial x} & \frac{\partial v_k}{\partial y} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{f} = \begin{Bmatrix} 0 \\ f_x + u_k \frac{\partial u_k}{\partial x} + v_k \frac{\partial u_k}{\partial y} \\ f_y + u_k \frac{\partial v_k}{\partial x} + v_k \frac{\partial v_k}{\partial y} \\ 0 \end{Bmatrix}. \tag{9}$$

For the Stokes problem, the convective item will be neglected and the Reynolds number is set one. Therefore, the coefficient matrix and force vector will be described as

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{f} = \begin{Bmatrix} 0 \\ f_x \\ f_y \\ 0 \end{Bmatrix}, \quad \mathbf{u} = \begin{Bmatrix} u \\ v \\ p \\ \omega \end{Bmatrix}. \tag{10}$$

3. Moving least-squares approximation

3.1. The MLS approximation scheme

The moving least-squares (MLS) approximation scheme is widely used in recent meshfree methods. Also in the present LSMFM the MLS approximation is employed. In the sequel, the construction of the MLS shape functions and their derivatives are briefly reviewed. More details can be found in [26].

The MLS approximation $u^h(\mathbf{x})$ of a continuous function u in the domain Ω is defined as

$$u^h(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x}) a_i(\mathbf{x}) \equiv \mathbf{p}^{\mathrm{T}}(\mathbf{x}) \mathbf{a}(\mathbf{x}), \tag{11}$$

where $\mathbf{p}^{\mathrm{T}}(x) = [p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x})]$ is a complete monomial basis of m dimensions; $\mathbf{a}(\mathbf{x})$ is a vector containing coefficients $a_i(\mathbf{x})$, $i = 1, 2, \ldots, m$, which are the functions of space coordinates $\mathbf{x} = [x, y, z]^{\mathrm{T}}$.

Coefficients $a_i(\mathbf{x})$ are obtained by minimizing the error in the weighted least-squares sense. This yields the following quadratic form:

$$J = \sum_{I}^{n} w_I(\mathbf{x}) \left[\mathbf{p}^{\mathrm{T}}(\mathbf{x}_I) \mathbf{a}(\mathbf{x}) - \hat{u}_I \right]^2, \tag{12}$$

where $w_I(\mathbf{x})$, $I = 1,2,\ldots,n$, are the weight function associated with node I which should be compactly supported, n the number of points in the neighborhood of $\mathbf{x}$ for which the weight function $w_I(\mathbf{x}) > 0$, $\hat{u}_I$ the approximation to the value $u(\mathbf{x}_I)$ at the node $\mathbf{x}_I$.

To obtain the coefficients $\mathbf{a}(\mathbf{x})$, the quadratic form J in Eq. (12) is minimized:

$$\frac{\partial J}{\partial \mathbf{a}(\mathbf{x})} = 2\sum_{I}^{n} w_I(\mathbf{x})\mathbf{p}(\mathbf{x}_I)\left[\mathbf{p}^{\mathrm{T}}(\mathbf{x}_I)\mathbf{a}(\mathbf{x}) - \hat{u}_I\right]. \tag{13}$$

Thus the coefficients $\mathbf{a}(\mathbf{x})$ can be determined by

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\hat{\mathbf{u}}, \tag{14}$$

where

$$\mathbf{A}(\mathbf{x}) = \sum_{I=1}^{n} w_I(\mathbf{x})\mathbf{p}(\mathbf{x}_I)\mathbf{p}(\mathbf{x}_I)^{\mathrm{T}}, \tag{15a}$$

$$\mathbf{B}(\mathbf{x}) = [w_1(\mathbf{x})\mathbf{p}(\mathbf{x}_1), w_2(\mathbf{x})\mathbf{p}_2(\mathbf{x}_2), \ldots, w_n(\mathbf{x})\mathbf{p}_n(\mathbf{x}_n)], \tag{15b}$$

$$\hat{\mathbf{u}}^{\mathrm{T}} = [\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_n]. \tag{15c}$$

From Eq. (14), substituting $a(\mathbf{x})$ in Eq. (11), the approximation $u^h(\mathbf{x})$ can be expressed as

$$u^h(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x})a_j(\mathbf{x}) \equiv \mathbf{p}(\mathbf{x})^{\mathrm{T}}\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\hat{\mathbf{u}} = \mathbf{N}(\mathbf{x})\hat{\mathbf{u}}, \tag{16}$$

where $\mathbf{N}(\mathbf{x})$ is the shape function that is defined by

$$\mathbf{N}(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}). \tag{17}$$

The first-order derivative of the MLS shape function can be calculated as follows:

$$\mathbf{N}_{,i} = \mathbf{p}_{,i}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B} + \mathbf{p}^{\mathrm{T}}(\mathbf{A}^{-1}\mathbf{B}_{,i} + \mathbf{A}_{,i}^{-1}\mathbf{B}), \tag{18}$$

where

$$\mathbf{A}_{,i}^{-1} = -\mathbf{A}^{-1}\mathbf{A}_{,i}\mathbf{A}^{-1}. \tag{19}$$

It should be noted that the number of neighbors at each integration point is at least equal to the basis dimension (nm) for the regularity of $\mathbf{A}$.

In this paper, linear complete basis in two-dimensional problem is used:

$$\mathbf{p}(\mathbf{x}) = [1, x, y]^{\mathrm{T}}. \tag{20}$$

The following weight function in [14] is employed.

$$w_I(\mathbf{x}) = \begin{cases} \sqrt{4/\pi}\left(1 - \frac{\|\mathbf{x}-\mathbf{x}_I\|^2}{r_I^2}\right)^4, & \|\mathbf{x} - \mathbf{x}_I\| < r_I, \\ 0, & \|\mathbf{x} - \mathbf{x}_I\| \geqslant r_I, \end{cases} \tag{21}$$

where r_I is the size of nodal support selected by $r_I = \alpha d$ with d denoting the nodal spacing, and constant α a chosen proportional coefficient.

3.2. Fast computation on shape functions and derivatives

In terms of consistency conditions for shape functions in the MLS approximation, the fast computation on shape functions and derivatives can be obtained [27].

The shape functions can be described in the following form:

$$N_I(\mathbf{x}) = \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{p}(\mathbf{x}_I)w_I(\mathbf{x}), \tag{22}$$

where $\boldsymbol{\alpha}$ is the coefficient vector.

The linear consistency equation can be expressed as follows:

$$\sum_I N_I(\mathbf{x})\mathbf{p}(\mathbf{x}_I) = \mathbf{p}(\mathbf{x}) = [\,1 \quad x \quad y\,]^{\mathrm{T}}. \tag{23}$$

Substituting Eq. (22) into Eq. (23) results

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{p}(\mathbf{x}), \tag{24}$$

where matrix $\mathbf{A}$ is exactly the same matrix with Eq. (15a).

The derivatives of $\boldsymbol{\alpha}$ can be obtained by differentiation of Eq. (24),

$$\mathbf{A}\boldsymbol{\alpha}_{,i} = \mathbf{p}_{,i}(\mathbf{x}) - \mathbf{A}_{,i}\mathbf{A}\boldsymbol{\alpha}. \tag{25}$$

Then, the derivatives of the shape function are given by

$$N_{I,i} = \mathbf{p}(\mathbf{x}_I)^{\mathrm{T}}\alpha_{,i}(\mathbf{x})w_I(\mathbf{x}) + \mathbf{p}(\mathbf{x}_I)^{\mathrm{T}}\alpha(\mathbf{x})w_{I,i}(\mathbf{x}). \tag{26}$$

As indicated in [27], compared to standard EFG, this method can save about half computation time for shape functions and derivatives in two-dimensional problems and even more in 3D case.

4. The least-squares formulation

In meshfree method, Galerkin method or collocation point method are usually used to implement the discretization process. In the present work, the least-squares method is taken for the discretization in meshfree framework to employ its advantages in dealing with CFD problems and its robustness to integration errors, which are mentioned in Section 1.

For the first-order system written as the standard form Eq. (6), the quadratic functional is defined as follows:

$$I(\mathbf{u}) = \int_\Omega (\mathbf{L}\mathbf{u} - \mathbf{f})^2 \,\mathrm{d}\Omega = \int_\Omega (\mathbf{L}\mathbf{u} - \mathbf{f})^{\mathrm{T}}(\mathbf{L}\mathbf{u} - \mathbf{f}) \,\mathrm{d}\Omega. \tag{27}$$

Using Gauss quadrature for integration in background cells, Eq. (27) can be written in the discrete form:

$$I(\mathbf{u}) = \sum_{i_{\mathrm{cell}}=1}^{N_{\mathrm{cell}}} \sum_{j_{\mathrm{gauss}}=1}^{N_{\mathrm{gauss}}} w_{j_{\mathrm{gauss}}}(\mathbf{L}\mathbf{u} - \mathbf{f})^{\mathrm{T}}(\mathbf{L}\mathbf{u} - \mathbf{f}), \tag{28}$$

where $w_{j_{\mathrm{gauss}}}$ is the weight factor at the integration point (including area information), N_{gauss} the number of Gauss integration points in each background cell, and N_{cell} the total number of the background cells.

For all unknown variables, equal-order local approximation is employed. It should be noted that the following necessary condition for the existence of the solution must be satisfied:

$$N_{\mathrm{cell}} \times N_{\mathrm{Gauss}} \times N_{\mathrm{eq}} \geqslant N_{\mathrm{point}} \times N_{\mathrm{dof}} - N_{\mathrm{bc}}, \tag{29}$$

where N_{eq} is the number of governing equations in the first-order system, N_{point} the total number of nodes, N_{dof} the number of degrees of freedom at each node and N_{bc} the total number of specified nodal degrees of freedom at boundaries.

The approximation of $\mathbf{u}$ at each integration point can be written as Eq. (16). Then the stiffness matrix and force vector for jth Gaussian integration point are:

$$\mathbf{k}^{j_{gauss}} = w_{j_{gauss}}(\mathbf{LN})^{T}(\mathbf{LN}), \tag{30a}$$

$$\mathbf{f}^{j_{gauss}} = w_{j_{gauss}}(\mathbf{LN})^{T}\mathbf{f}. \tag{30b}$$

For Eq. (16), if there are N_{dof} variables at each node, the format of the shape function $\mathbf{N}$ is described as follows:

$$\mathbf{N} = [N_1\mathbf{I} \quad N_2\mathbf{I} \quad \cdots \quad N_n\mathbf{I}], \tag{31}$$

where $\mathbf{I}$ is $N_{dof} \times N_{dof}$ identity matrix.

Assembling the stiffness matrix and force vector at the Gauss integration points in all background cells, the linear algebraic equation is obtained:

$$\mathbf{K}_G\hat{\mathbf{U}} = \mathbf{F}_G, \tag{32}$$

where $\mathbf{K}_G$ is the global stiffness matrix, $\mathbf{F}_G$ the global force vector, and $\hat{\mathbf{U}}$ the approximation to the global vector of nodal degrees of freedom

$$\mathbf{K}_G = \sum_{i_{cell}=1}^{N_{cell}} \sum_{j_{gauss}=1}^{N_{gauss}} \mathbf{k}^{j_{gauss}}, \tag{33a}$$

$$\mathbf{F}_G = \sum_{i_{cell}=1}^{N_{cell}} \sum_{j_{gauss}=1}^{N_{gauss}} \mathbf{f}^{j_{gauss}}. \tag{33b}$$

It should be noted that the global system matrix $\mathbf{K}_G$ is symmetric and positive definite. Therefore, the resulting linear system of Eq. (32) can be solved by the matrix-free element-by-element Jacobi preconditioned conjugate gradient (MFEBEJCG) method. It is simple to get the Jacobi preconditioned matrix. It needs not to form global stiffness matrix and the stiffness matrix at each evaluation point, so that computation cost can be reduced greatly [28]. It should be mentioned that the element-by-element technique is used in FEM because assemblage process is performed on the element level. In meshfree methods, assemblage process is carried out at all evaluation points. However, the same nomenclature is used in this paper.

It should be noted that the actual numerical values of field variables at nodal points, $u(x_J)$, $J = 1,2,\ldots,N_{point}$, are obtained as follows:

$$u(x_J) = \sum_{I=1}^{n} N_I(x_J)\hat{u}_I \tag{34}$$

4.1. Generation of evaluation points by quadtree algorithm

In Eq. (28), the evaluation points and the weight factors should be determined in background cells. To deal with complex geometries of analysis domain in meshfree framework, simple construction of background cells is essential. In this regard, several schemes could be considered to generate evaluation points, such as quadtree/octree algorithm, Delaunay triangulation, Voronoi cell and so on [29,30]. However, for the effectiveness of meshfree method, the solution accuracy should not be degraded even

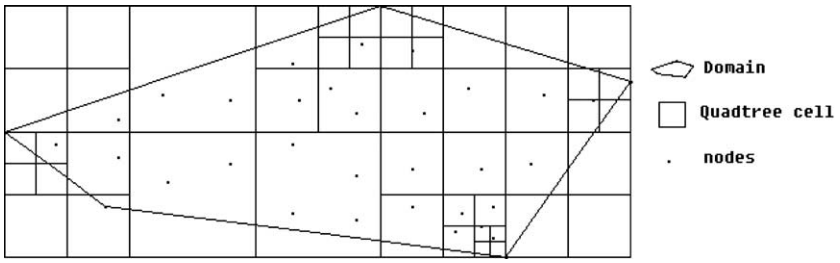

Fig. 1. Quadtree algorithm.

Table 1
Gauss quadrature rule for quadtree algorithm

	No. of nodes in cell	Quadrature order (rule 1)	Quadrature order (rule 2)
The background cell cuts boundaries	0	2×2	2×2
	1	2×2	2×2
The background cell does not cut boundaries	0	2×2	1×1
	1	2×2	1×1
	2	3×3	2×2
	3	3×3	2×2
	4	3×3	2×2

with those simple integration schemes. The LSMFM satisfies this requirement since it is robust to integration errors [24].

In the present work, the quadtree algorithm is used in the two-dimensional geometry model. Quadtree algorithm is a well-known algorithm in computational geometry, which is easy to construct evaluation points for the complex model with Gaussian quadrature rule. The quadtree algorithm used in the present work is as follows (Fig. 1):

(i) Determine the maximum number of nodes $N_{\max}$ in each background cell. In this work, different $N_{\max}$ is used. $N_{\max} = 4$ is used if the cell does not cut boundaries, or else $N_{\max} = 1$.
(ii) Give a rectangular background domain to cover the computational domain.
(iii) Initial square background cells are defined according to ratio between width and height of computational domain, here called as mother cells.
(iv) If a mother cell contains more than $N_{\max}$ nodes, then divide it into four squares with equal size, called as daughter cells. Loop this step until the number of nodes in the each cell is less than or equal to $N_{\max}$.
(v) Generate the evaluation points in background cells in terms of the given Gaussian quadrature rule as in Table 1. It should be pointed out the higher-order quadrature will be used for the background cells which cut boundaries. Therefore the loss for the integration from such background cells can be reduced.
(vi) The evaluation points outside computational domains are not considered in calculation.

5. Numerical examples

5.1. Error estimates on the LSMFM for the Stokes problem

The Stokes problem has been studied for many years. The Stokes operator is a basic ingredient of the incompressible Navier–Stokes equation. Therefore, the preliminary study on the Stokes problem will be

carried out by LSMFM. In this section, the convergence characteristic and approximation accuracy will be investigated.

For the LSMFM, the convergence characteristics depend on the regularity of differential operator **L** and the approximation properties of the shape function. Here, general results on the convergence characteristics of the first-order least-squares formulation are given. The details can be found in References [31–33].

For the Stokes problem, linear first-order differential operator **L** is strictly elliptic [31]. For the strictly elliptic problems, if the MLS shape functions are employed, a priori error estimates in L^2 and H^1 norm can be expressed as follows: there exist positive constants C_0 and C_1 such that

$$\left\|u-u^h\right\|_{H^1(\Omega)} C_1 h^m |u|_{H^{m+1}(\Omega)} \tag{35a}$$

and

$$\left\|u-u^h\right\|_{L^2(\Omega)} C_0 h^{m+1} |u|_{H^{m+1}(\Omega)}, \tag{35b}$$

where m is the degree of complete polynomial in the basis and h is the nodal spacing.

In the numerical example, the following L^2 and H^1 error norms are used as error measures

$$\left\|u-u^h\right\|_{L^2} = \left(\int_\Omega (u-u^h)^2 \,\mathrm{d}\Omega\right)^{1/2}, \tag{36a}$$

$$\left\|u-u^h\right\|_{H^1} = \left(\int_\Omega \left[(u-u^h)^2 + (u_{,x}-u^h_{,x})^2 + (u_{,y}-u^h_{,y})^2\right] \mathrm{d}\Omega\right)^{1/2}. \tag{36b}$$

The corresponding relative error norms are

$$\left\|u-u^h\right\|^{\mathrm{rel}}_{L^2} = \frac{\|u-u^h\|_{L^2}}{\|u\|_{L^2}}, \tag{37a}$$

$$\left\|u-u^h\right\|^{\mathrm{rel}}_{H^1} = \frac{\|u-u^h\|_{H^1}}{\|u\|_{H^1}}. \tag{37b}$$

In the present study, a cavity model will be presented for the error estimate on the Stokes problem. It has a unit square domain ($0 \leqslant x \leqslant 1$, $0 \leqslant y \leqslant 1$) with the polynomial divergence-free velocity field [1,31,34] as

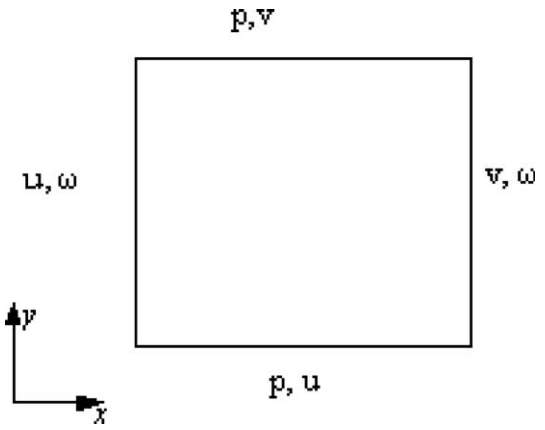

Fig. 2. Boundary conditions for the Stokes problem.

$$u(x,y) = x^2(1-x)^2(2y - 6y^2 + 4y^3),$$
$$u(x,y) = y^2(1-y)^2(-2x + 6x^2 - 4x^3)$$

and the pressure field is

$$p(x,y) = x^2 - y^2 - 0.25$$

with the vorticity field as

$$\omega(x,y) = -x^2(1-x)^2(2 - 12y + 12y^2) + y^2(1-y)^2(-2 + 12x - 12x^2).$$

The boundary conditions are shown in Fig. 2.

In practical applications, the domain of background cells may not coincide with the complex computational domain, background cells would cut the boundaries of the computational domain arbitrarily such that the integration for the discretized system becomes more inaccurate in the vicinity of boundaries. In order to investigate the effect of inaccurate integration on the solution, we construct background cells with the quadtree algorithm in which the background domain does not coincide with the computational domain. Besides this, in general, the solutions will be affected by the nodal distribution, Gauss quadrature rule and the size of nodal support in the case of the present linear basis functions and weight function. Therefore different schemes will be designed to analyze the effect of the above parameters on the convergence rate and the accuracy of the solution. Irregular nodal distribution, inclined background domain and uniform nodal distribution with simple background cell will be devised. Irregular nodal distributions are presented in Fig. 3. The inclined background domain is built and shown in Fig. 4 to test the effect of inaccurate integration on the results, where the same nodal distributions in Fig. 3 are used. For the quite simple geometry, the simple nodal distribution will be constructed, where the nodes coincide with the vertices of the background cells. It is presented in Fig. 5. According to the number of the background cells, nodes and boundary conditions, the reduced integration, i.e., one-point integration, can be also used in the simple background cells. Table 2 lists the six schemes at different nodal distributions and quadrature rules in present numerical example. For Scheme 1 and Scheme 2, the background domain coincides with the computational domain and irregular nodal distributions are investigated. The inclined background domain and irregular nodal distributions are used in Scheme 3 and Scheme 4. Simple background cells and uniform nodal distributions will be constructed in Scheme 5 and Scheme 6. Quadrature rules in Table 1 will be employed in the leading four schemes. The uniform quadrature rule will be used in Scheme 5 and Scheme 6. In present study, equal-order MLS approximation is taken with the linear basis function. The size of nodal support can be changed by the proportional coefficient α, here $\alpha = 1.5$ is used. The nodal spacing d is defined as the maximum distance from the closest nodes to the given node in four quadrants. The boundary conditions are enforced by the penalty method. The residual of the linear system for all variables is used as the stopping criterion of MFEBEJCG iteration.

$$\text{residual} = \sqrt{\sum_{i=1}^{N_{\text{point}} \times N_{\text{dof}}} r_i^2} < 10^{-6}, \tag{38a}$$

where i denotes the ith degree of freedom.

Here let us define the following relative error of each variable in L^2-norm between two schemes, i.e., between Scheme 1 and Scheme 3.

$$\text{Relative error} = \frac{\left\|\mathbf{U}_i^{\text{Scheme1}} - \mathbf{U}_i^{\text{Scheme3}}\right\|_2}{\left\|\mathbf{U}_i^{\text{Scheme1}}\right\|_2} \quad \text{with } \left\|\mathbf{U}_i^{\text{Scheme1}}\right\|_2 = \sqrt{\sum_{i=1}^{N_{\text{point}}} (\mathbf{U}_i^{\text{Scheme1}})^2},$$

where i denotes velocity, pressure and vorticity, respectively.

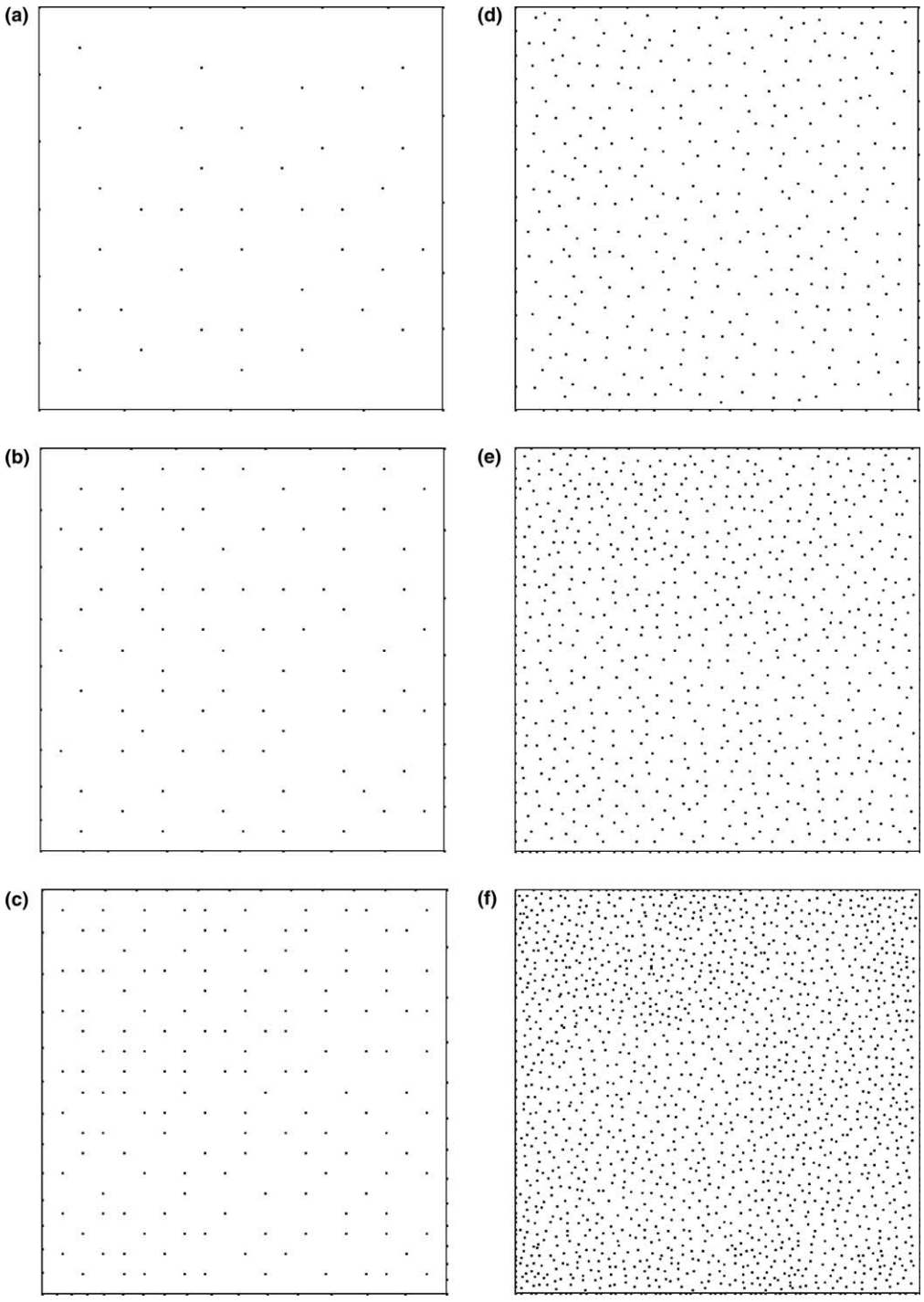

Fig. 3. Irregular nodal distributions.

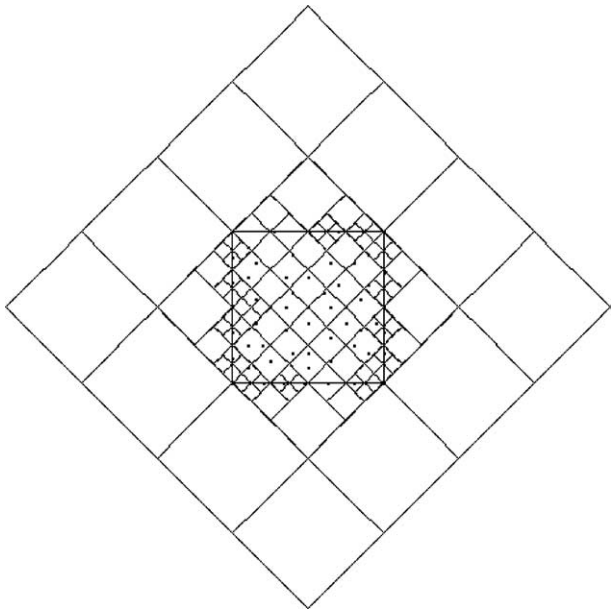

Fig. 4. Inclined background domain which is not coincident with the computational domain.

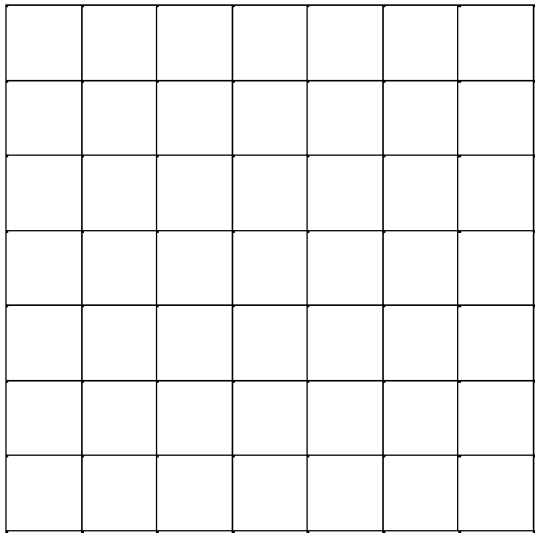

Fig. 5. Regular nodal distribution with simple background cells in which nodes are at the vertices of background cells.

The above definition on relative error can be also employed for the comparison between Scheme 2 and Scheme 4.

In present study, all error norms for the velocity, the pressure and the vorticity in Eqs. (36a) and (36b) are plotted with respect to the number of nodes.

In Fig. 6(a)–(f), it can be found that the convergence rates for all variables in L^2 norm and H^1 norm are consistent with the error estimate theory. The uniform nodal distributions with simple background cells

Table 2
Schemes for error estimates

Scheme No.	Nodal distribution	Algorithms to construct background cells	The relationship between the background domain and the computational domain	Quadrature rules
1	Irregular	Quadtree	Coincident	Rule 1 in Table 1
2	Irregular	Quadtree	Coincident	Rule 2 in Table 1
3	Irregular	Quadtree	Not coincident	Rule 1 in Table 1
4	Irregular	Quadtree	Not coincident	Rule 2 in Table 1
5	Uniform	Simple way	Coincident	1×1
6	Uniform	Simple way	Coincident	2×2

have higher rate of convergence than the irregular nodal distributions with quadtree algorithm. For the uniform nodal distribution with simple background cells, the convergence rate for velocity and vorticity in L^2 norm using the one-point quadrature is the same as those using higher-order quadratures, but the convergence rate for pressure at one-point quadrature is lower than that at higher-order quadrature. The convergence rate of each variable in H^1 norm using one-point quadrature is higher than those using higher-order quadratures.

In comparison with Scheme 1 and Scheme 2, on the whole, it can be found that the results at the lower-quadrature with quadtree algorithm will be more accurate than those at higher-order quadrature. In comparison with Scheme 3 and Scheme 4, the results at low-order quadrature may not accurate than those at higher-lower quadrature except the solutions for velocity and vorticity in the relative H^1 error norm. This may be resulted from the effect of inaccurate integration, which will combine with quadrature rule in the background cells to influence the solutions. Therefore lower-order quadrature rule is suggested for the irregular nodal distributions.

Comparing Scheme 1 with Scheme 3, and Scheme 2 with Scheme 4, it can be found that the accuracy of the solution is obviously degraded because some background cells cut boundaries. From Figs. 7 and 8, it can be seen that the discrepancy will become lower with respect to the number of nodes. For the irregular nodal distribution in Fig. 3(f), the maximum relative error between Scheme 1 and Scheme 3 is 0.19%, the minimum value is 0.012%; in comparison with Scheme 2 and Scheme 4, the maximum error and the minimum relative error are 1.06% and 0.046%, respectively. Thus the effects of inaccurate integration on the accuracy of the solution will be quite small if the present quadtree algorithm and quadrature rule are used for the large number of nodes.

In comparison among Schemes 1, 2, 5 and 6, it is obviously seen that the results for uniform nodal distribution are more accurate than those from irregular nodal distributions except those in the relative L^2 error norm for velocity. For the uniform nodal distributions in Schemes 5 and 6, the solutions for each variable at one-point quadrature have the better approximation accuracy than those at higher-order quadrature. Namely, the residual of governing equations will be more close to zero if one-point quadrature is employed for present size of nodal support. For the For the Stokes problem with quite simple geometry, where computational domain coincide with the background domain and the simple background cells can be constructed, it is remarkable that the LSMFM gives more accurate solution using one-point quadrature rule for the suitable size of nodal support if the necessary condition for existence of the solution is satisfied.

It is well known that the accuracy will be improved with respect to the size of nodal support in meshfree method. But too large size of nodal support will lead to the bad approximation accuracy because the shape functions are strongly non-polynomial. At the same time, the computation cost will be very higher. Therefore, the relative large size of nodal support is suggested.

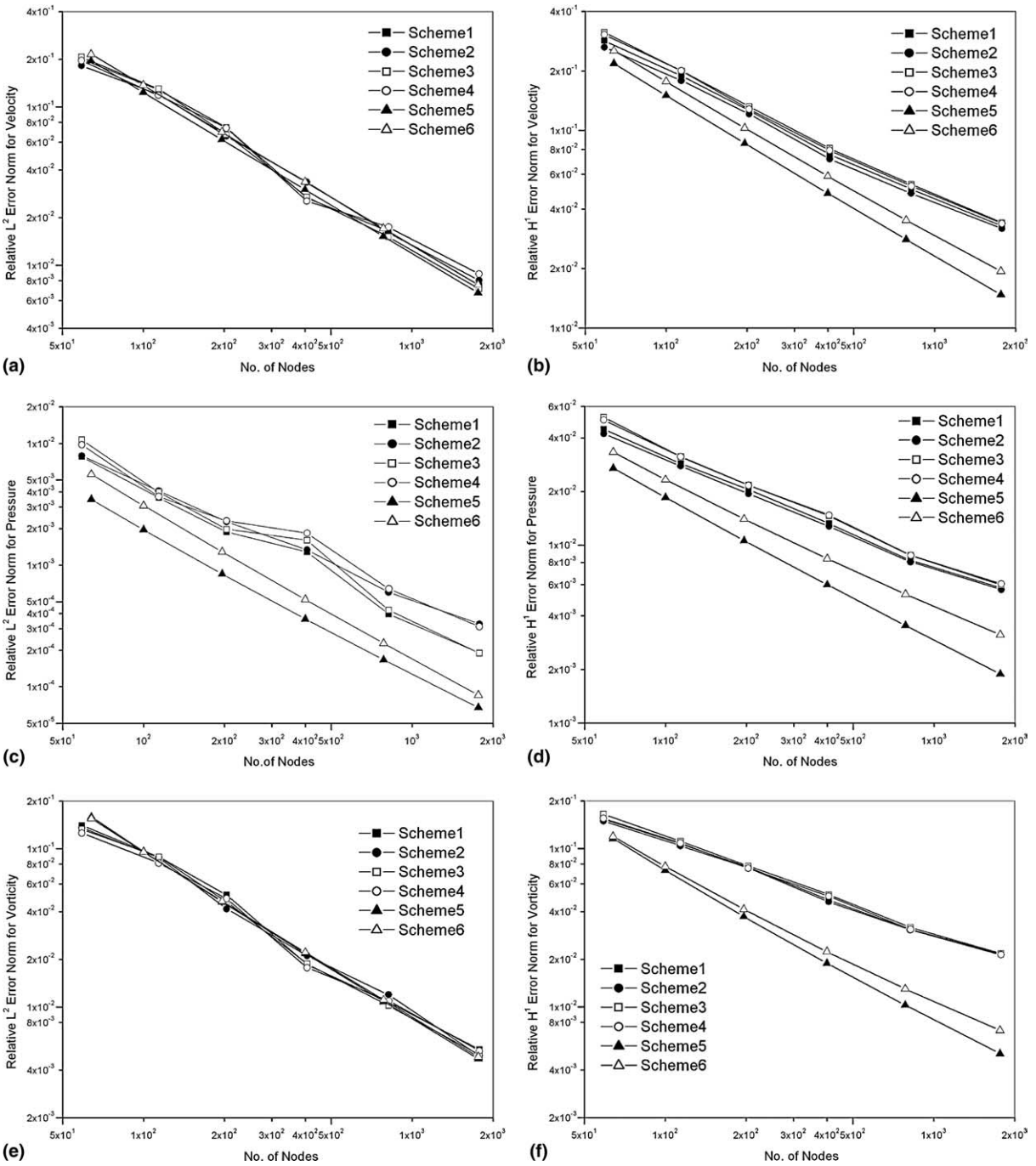

Fig. 6. Relative L^2 and H^1 error norms and convergence rates of velocity, pressure and vorticity.

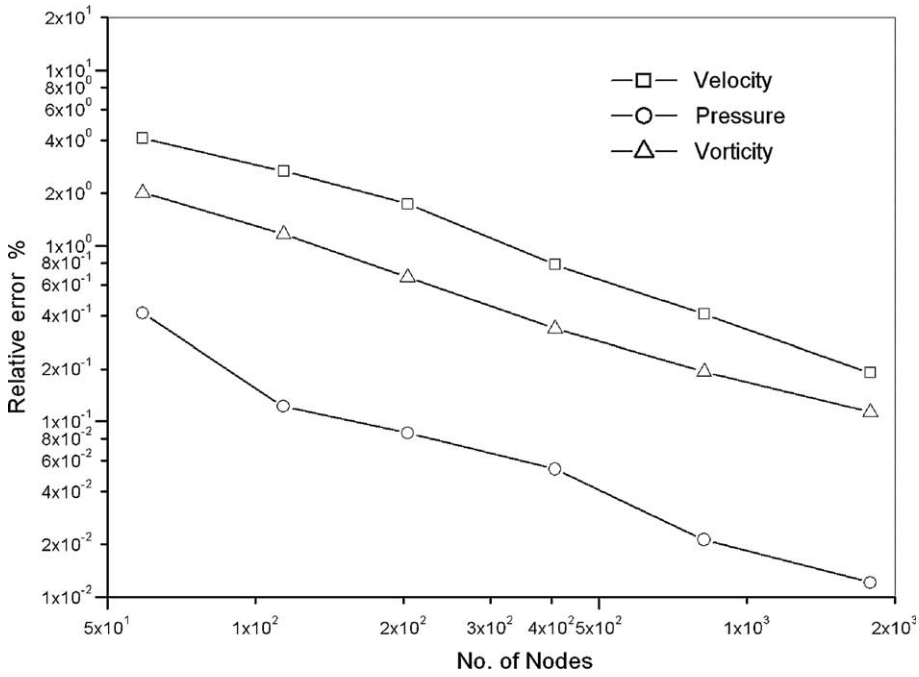

Fig. 7. Relative error between Scheme 1 and 3.

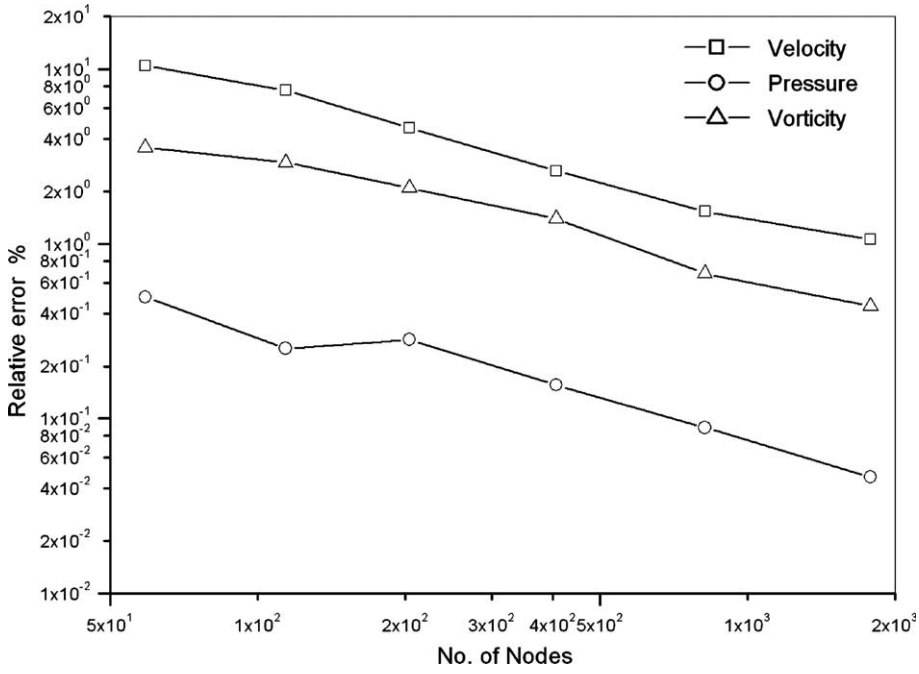

Fig. 8. Relative error between Scheme 2 and 4.

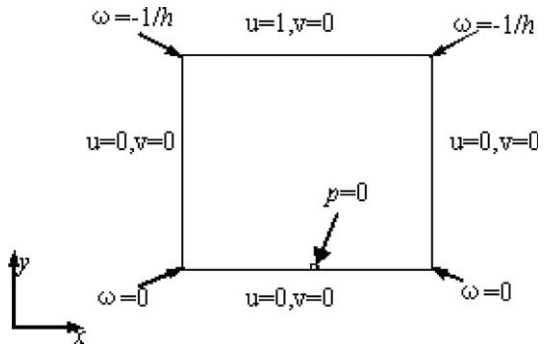

Fig. 9. The boundary conditions for the cavity flow.

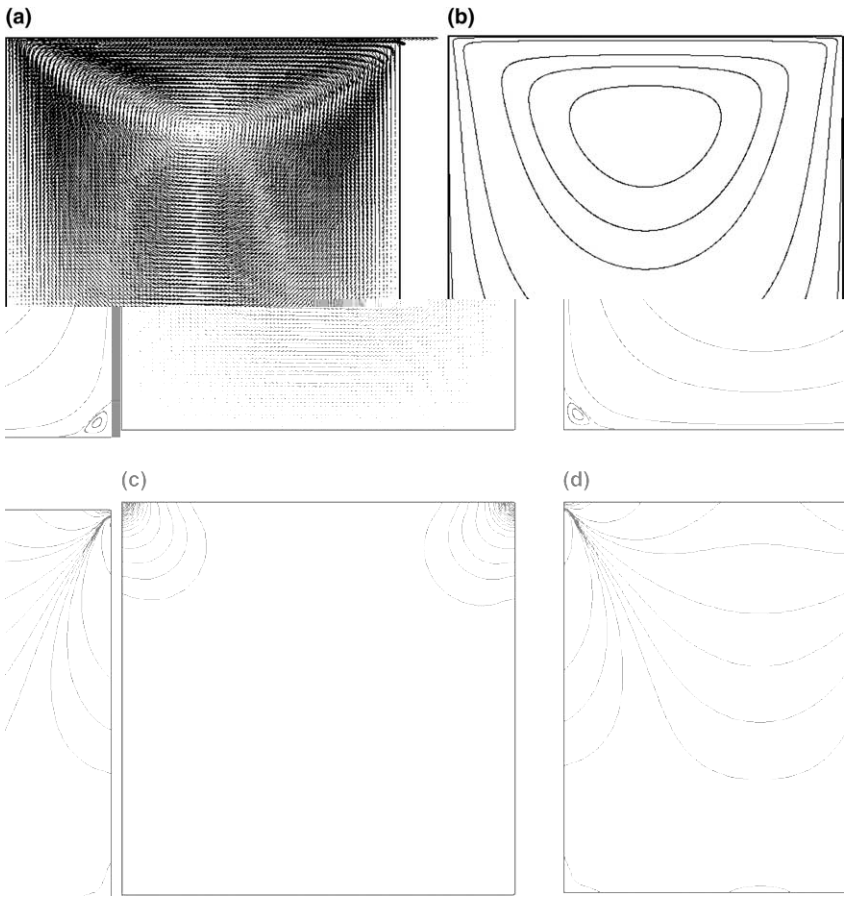

Fig. 10. The results for cavity-driven flow: (a) velocity, (b) streamline, (c) pressure contour and (d) vorticity contour.

5.2. Cavity flow for Stokes problem

In present study, driven-cavity flow in unit square domain ($0 \leqslant x \leqslant 1$, $0 \leqslant y \leqslant 1$) can be also used to test the LSMFM for the Stokes problem [34–36].

The boundary conditions are shown in Fig. 9. On the top side $u = 1$ and $v = 0$ are given, and no slip boundary conditions, i.e., $u = 0$, $v = 0$, are prescribed on the left, right and bottom sides. At the mid-point of the bottom side, the pressure condition $p = 0$ is specified. We have $\omega = -\partial u/\partial y + \partial v/\partial x = -1/h + 0 = -1/h = -100$ at the two upper corners, where h is the closest distance from the nodes on the left side or the right side to the corresponding node at the upper corner, $\omega = 0$ are given at the two bottom corners. As an initial value for the iterative process of matrix-free element-by-element Jacobi preconditioned conjugate method (MFMBMJCG), $u = v = p = \omega = 0$ are taken.

As mentioned in section 5.1, owing to the quite simple geometry for the cavity flow problem, simple uniform square background cells can be built, where the vertices of the cells coincide with the nodes. 10,201 nodes and 10,000 cells are constructed. The linear basis and one-point Gauss quadrature rule are used.

Table 3
The location and the value of the stream function (ψ) for the eddy centers

	Shankar [36]			LSMFM		
	x	y	ψ	x	y	ψ
Primary eddy	0.5	0.24	−0.10005	0.50	0.761	−0.09811
Corner eddy	0.038	0.038	2.227×10^{-6}	0.0358	0.0378	1.937×10^{-6}

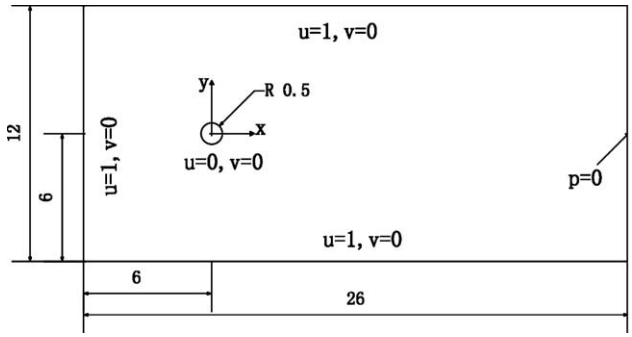

Fig. 11. The geometry and boundary conditions for the flow past a circular cylinder.

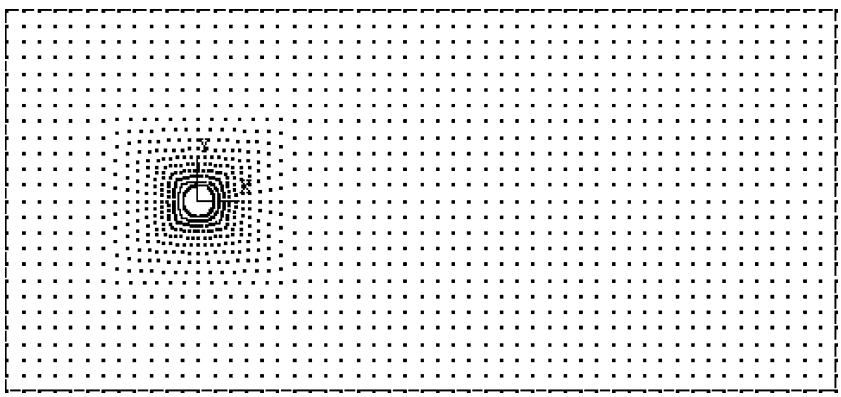

Fig. 12. The nodal distribution for the flow past a circular cylinder.

Equal-order MLS approximation is employed. The value of α used is 1.5. The boundary conditions are enforced by the penalty method. The same stopping criterion of MFEBEJCG iteration in Eq. (38) is adopted in this example.

Fig. 10 presents the computed results of velocity, streamlines, and the contours of vorticity and pressure.

The comparisons between the results from Shankar [36] and solutions of LSMFM are listed in Table 3. Here, the location of the eddy centers and the value of the stream function at the eddy center are listed. The results of LSMFM agree well with those of Shankar.

5.3. The flow past a circular cylinder

The study for the flow past a circular cylinder has long history in experimental and numerical analysis [37–51]. It has been used as a benchmark to investigate a new method for the steady or unsteady incompressible viscous flow problem. In present study, only steady flow past a circular cylinder at low Reynolds numbers ($\leqslant 40$) will be investigated by LSMFM.

Figs. 11 and 12 show the geometry dimension, boundary conditions and nodal distribution. $u = 1$ and $v = 0$ are given on the inlet, top and bottom sides. $u = 0$, $v = 0$, are prescribed along the surface of the circular cylinder. At the mid-point of the outflow side, the pressure condition $p = 0$ is specified. In calculation, $u = 1$, $v = 0$, $p = 0$, $\omega = 0$ in the whole computation domain except along the surface of the cylinder with $u = 0$, $v = 0$ are taken as an initial trial solution.

In the square domain surrounding the cylinder ($-3 \leqslant x \leqslant 3$, $-3 \leqslant y \leqslant 3\leqslant$), radial-type nodal distribution is constructed. The uniform nodal distribution is built in the other domain.

In the numerical example, the different number of nodes and proportional coefficient of the nodal support α will be used for the computation at different Reynolds numbers. For the computation at $Re = 10$ will be done with 9341 nodes and α is 1.2. For $Re = 20$, 26,896 nodes is built and α is 1.2. The solution for $Re = 40$ is based on 34,038 nodes and α is 1.5. Background cells are constructed by quadtree

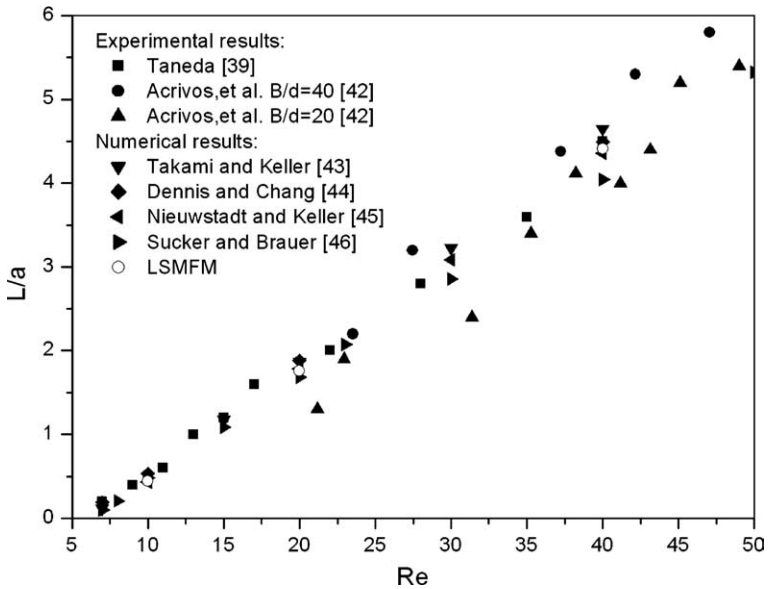

Fig. 13. The wake length at different Reynolds numbers.

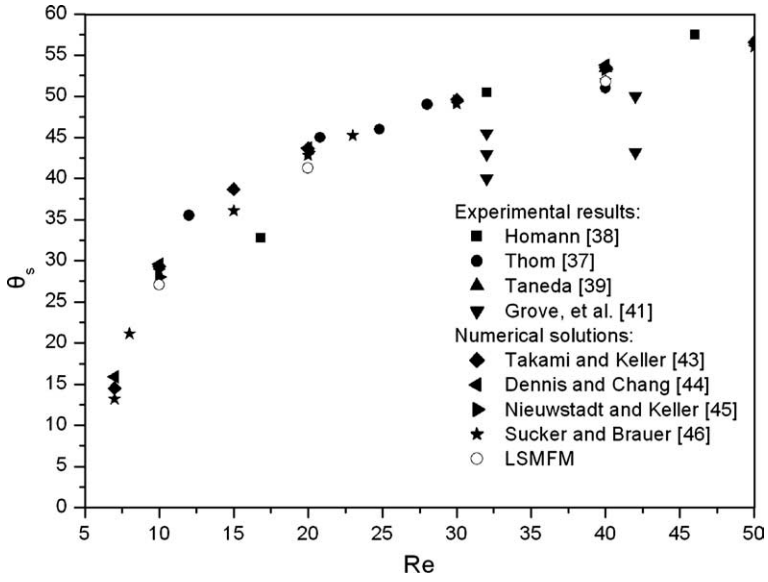

Fig. 14. The separation angle at different Reynolds numbers.

algorithm with quadrature rule2 in Table 1. Equal-order MLS approximation is employed. Newton's method is used to linearize the convective term. The boundary conditions are enforced by the penalty method.

The stopping criterion of MFEBEJCG iteration is the same as that in Eq. (38a). The stopping criterion for Newton's linearization is

$$\frac{\left\|\mathbf{U}^{k+1}-\mathbf{U}^{k}\right\|_2}{\left\|\mathbf{U}^{k}\right\|_2}<10^{-5}, \tag{38b}$$

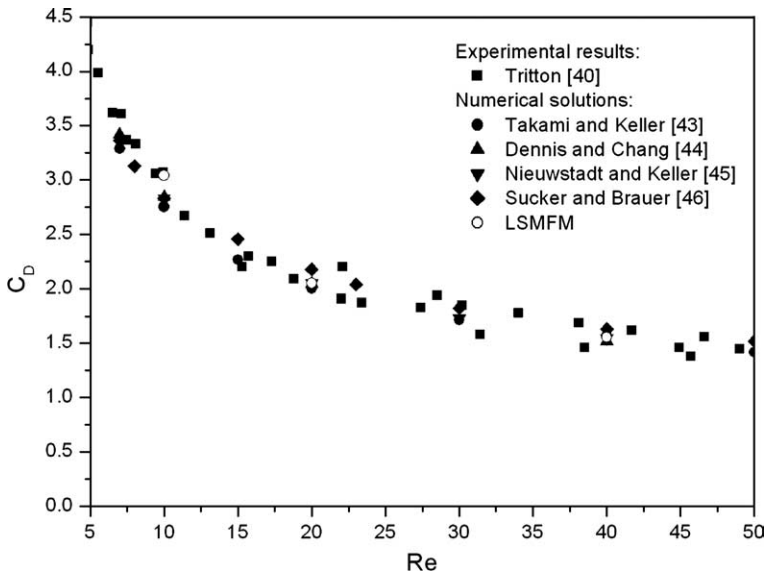

Fig. 15. The drag coefficient at different Reynolds numbers.

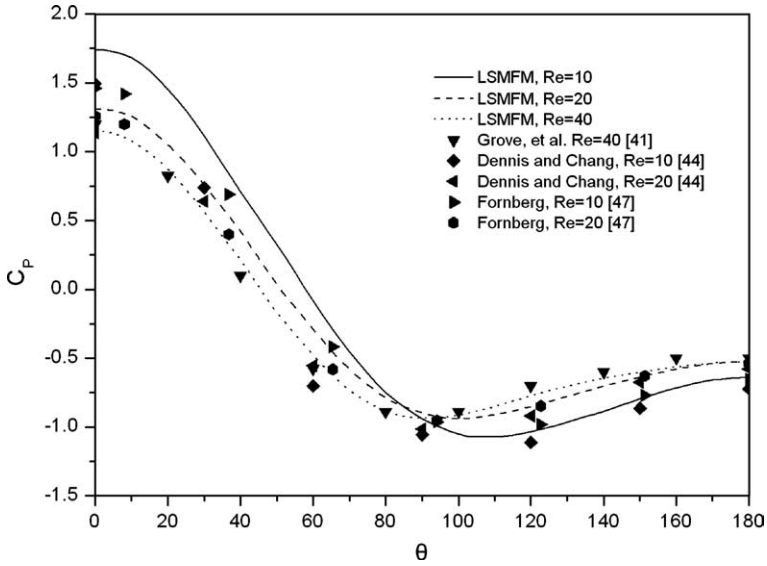

Fig. 16. The distribution of the pressure coefficient on the cylinder surface.

where $\|\mathbf{U}^k\|_2 = \sqrt{\sum_{i=1}^{N_{\text{point}} \times N_{\text{dof}}} (\mathbf{U}_i^k)^2}$ is the L^2-norm of all unknown variables, the subscripts k and $k+1$ denote the k-th and the $(k+1)$-th linearization step, respectively.

For the fluid flow past a circular cylinder, the characteristic quantities usually include the wake length L, the separation angle θ_s, the drag coefficient C_D and the pressure coefficient C_p. The wake length, L, is defined as the distance from the rear of the cylinder to the end of the separated region. The most accurate way is to use zero-value contour of the x-component of the velocity to get the position of stagnation point

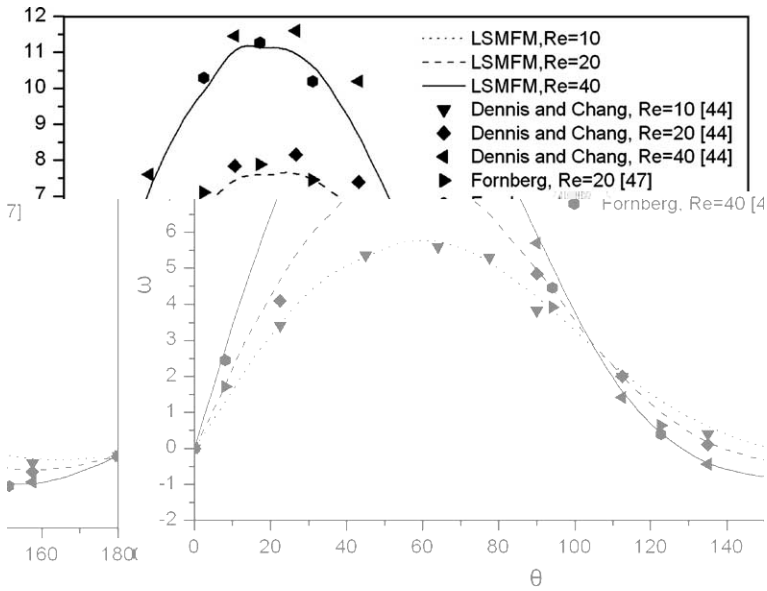

Fig. 17. The vorticity distribution on the surface of the cylinder.

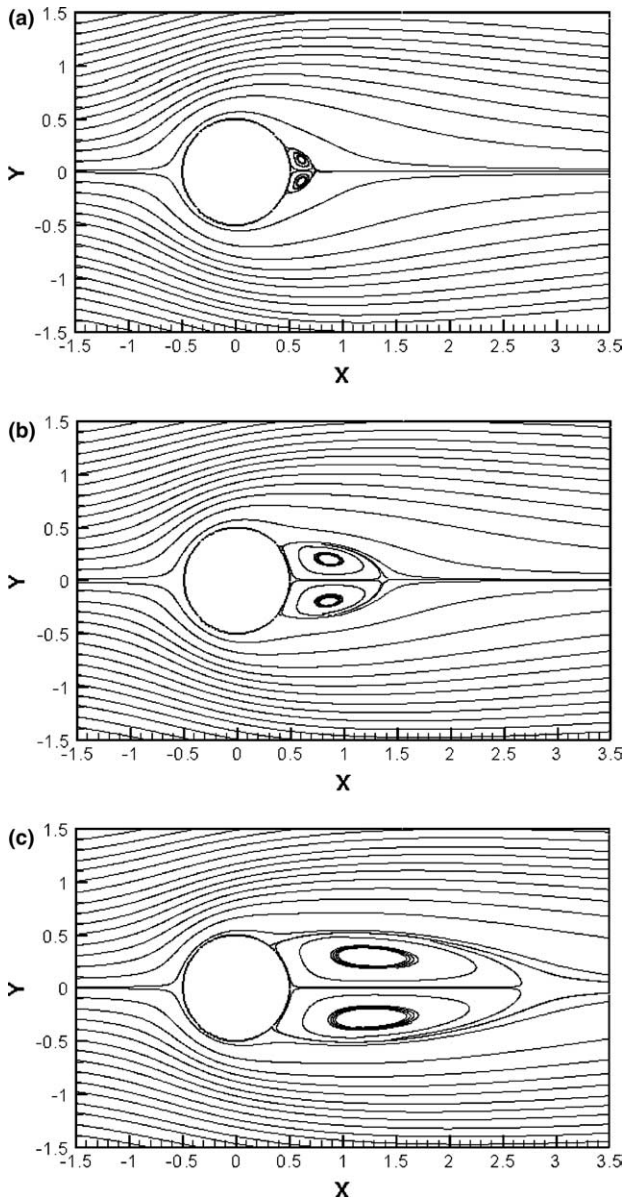

Fig. 18. The streamline at different Reynolds numbers: (a) The streamline at $Re = 10$. (b) The streamline at $Re = 20$. (c) The streamline at $Re = 40$.

along the x-axial. The separation angle, θ_s, is determined from zero-vorticity at the surface of the cylinder. The drag coefficient C_D is computed by

$$C_D = F_D/(\rho V_0^2 a), \tag{39}$$

where F_D is drag force, ρ the characteristic fluid density, a the radius of the circular cylinder.

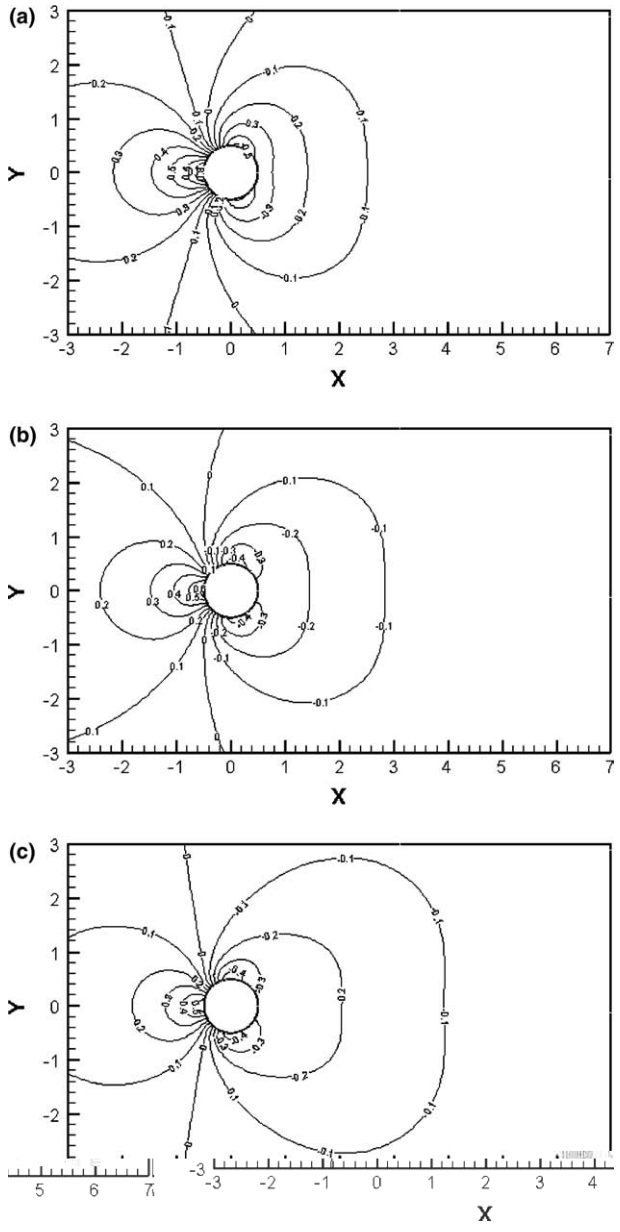

Fig. 19. The pressure contour at different Reynolds numbers: (a) The pressure contour at $Re = 10$. (b) The pressure contour at $Re = 20$. (c) The pressure contour at $Re = 40$.

The drag force on the circular cylinder can be expressed

$$F_{\mathrm{D}} = \oint_s -p\cos\phi \,\mathrm{d}s + \oint_s \tau \sin\phi \,\mathrm{d}s, \tag{40}$$

where p is the pressure along the cylinder surface, τ the shear force on the cylinder surface, ϕ the angle between the normal to the surface element and the flow direction, s the total surface area of the circular cylinder.

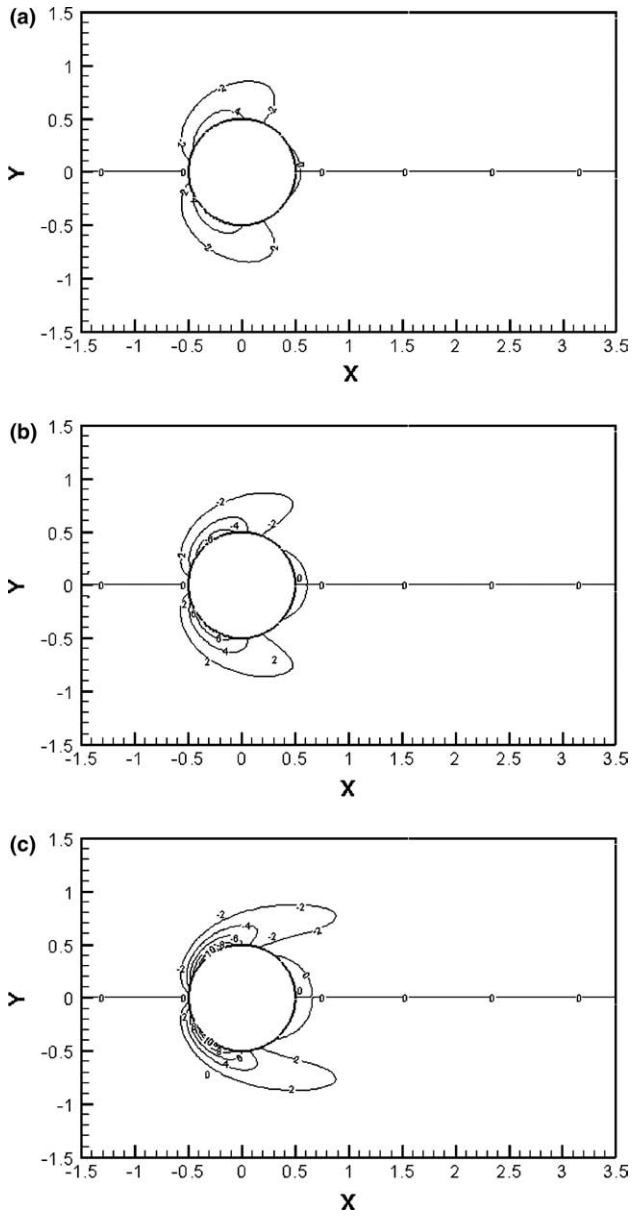

Fig. 20. The vorticity contour at different Reynolds numbers. (a) The vorticity contour at $Re = 10$. (b) The vorticity contour at $Re = 20$. (c) The vorticity contour at $Re = 40$.

The pressure coefficient is defined by

$$C_p = 2(p - p_\infty)/(\rho V_0^2), \tag{41}$$

where p_∞ is the uniform pressure at large distance.

The comparisons of the wake length L, the separation angle θ_s, the drag coefficient C_D, the distribution of the pressure coefficient C_p and the vorticity ω along the cylinder surface by LSMFM with other

experimental and numerical results are presented in Figs. 13–17. t should be pointed out that the orientation angle, θ, starts from the front position of the cylinder surface at the x-axis to the rear point of the cylinder surface at the x-axis in anticlockwise. From these results, it can be seen that the good agreements are obtained. Figs. 18–20 present the streamline, pressure contour and vorticity contour at different Reynolds numbers, respectively.

6. Concluding remarks

The least-squares meshfree method has been presented for the numerical analysis of incompressible viscous flow. The presented method employs the first-order velocity–pressure–vorticity formulation for two-dimensional steady incompressible Navier–Stokes problem. For a second-order system, at least quadratic completeness for MLS shape functions is required to ensure convergence since the second derivative of the variable is involved, so the quadratic basis function and large size of nodal supports should be used. But for first-order velocity–pressure–vorticity system, linear basis function can be used. Also, costly solutions of Poisson equation are avoided. However, the discretized system must be solved efficiently due to the increase of unknown variables and additional equations. For LSMFM, the resulting system is symmetric and positive definite, and can be efficiently solved by iterative methods for large-scale problems. All examples in the present work are computed by matrix-free element-by-element Jacobi preconditioned conjugate method. All governing equations are computed in the fully coupled manner. No special treatments, such as upwinding or adjustable parameters are required. Equal-order MLS approximation is employed for all unknown variables and Gauss quadrature integration in background cells is performed. Quadtree algorithm is used to construct background cells with arbitrary nodal distributions, which can be applied to more complicated problems.

For LSMFM, the numerical results are affected by the nodal distribution, the size of nodal support and the order of Gauss quadrature in background cells. The numerical study on the error estimates in the Stokes problem illustrates the effects of the size of nodal support and the order of Gauss quadrature on the convergence rate and the approximation accuracy. It should be noted that low-order quadrature in the background cells should be employed regardless of nodal distributions and the ways to construct background cells. For the Stokes problem in quite simple geometry, where computational domain coincides with the background domain and the simple background cells can be constructed, LSMFM gives excellent results using one-point quadrature in the background cell at the suitable size of nodal support. Even in case that the background domain does not coincide with the computational domain, the effects of inaccurate integration on the accuracy of the solution would be quite small if the quadtree algorithm and corresponding quadrature rule in present study were employed for the large number of nodes.

The results of the cavity flow for the Stokes problem and the flow past a circular cylinder at low Reynolds numbers for two-dimensional steady incompressible Navier–Stokes problem further show that the LSMFM can be successfully employed the incompressible viscous flow. The presented LSMFM preserves the useful meshfree properties. For the discretization of analysis domain, only nodal distribution is required and integration is performed with simply constructed background cells. Also the implementation of an adaptive strategy could be facilitated in this framework. The adaptive analysis using LSMFM on CFD problems and the application on the unsteady incompressible viscous flow will be presented in future.

Acknowledgment

The postdoctoral fellowship to the first author from Brain Korea 21 Project by the Ministry of Education of Korea is gratefully acknowledged.

References

[1] J.T. Oden, O.P. Jacquotte, Stability of some mixed finite element methods for Stokesian flows, Comput. Methods. Appl. Mech. Engrg. 43 (1984) 231–248.
[2] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, Springer, Berlin, 1996.
[3] T.J.R. Hughes, W.K. Liu, A. Brooks, Finite element analysis of incompressible viscous flows by the penalty function formulation, J. Comput. Phys. 30 (1979) 1–75.
[4] T.J.R. Hughes, Recent progress in the development and understanding of SUPG methods with special reference to the compressible Euler and Navier–Stokes equations, Comput. Methods. Appl. Mech. Engrg. 7 (1987) 1261–1295.
[5] B.N. Jiang, The Least-Squares Finite Element Method-Theory and Applications in Computational Fluid Dynamics and Electromagnetics, Springer, Berlin, 1998.
[6] L.B. Lucy, A numerical approach to the testing the fission hypothesis, Astron. J. 8 (1977) 1013–1024.
[7] R.A. Gingold, J.J. Monaghan, Kernal estimates as a basis for general particle methods in hydrodynamics, J. Comput. Phys. 46 (1982) 429–453.
[8] T. Liszka, J. Orkisz, The finite difference method at arbitrary irregular grids and its application in applied mechanics, Comput. Struct. 11 (1980) 83–95.
[9] T. Belytschko, Y.Y. Lu, L. Gu, Element-free Galerkin methods, Int. J. Numer. Methods Engrg. 37 (1994) 229–256.
[10] T. Belytschko, Y. Krongauz, M. Fleming, D. Organ, W.K. Liu, Smoothing and accelerated computation in the element free Galerkin method, J. Comput. Appl. Math. 74 (1996) 111–126.
[11] W.K. Liu, S. Jun, S. Li, J. Adee, T. Belytschko, Reproducing kernel particle methods for structural dynamics, Int. J. Numer. Methods Engrg. 38 (1995) 1655–1679.
[12] W.K. Liu, S. Jun, Y.F. Zhang, Reproducing kernel particle methods, Int. J. Numer. Methods Fluids 20 (1995) 1081–1106.
[13] I. Babuska, J.M. Melenk, The partition of unity method, Int. J. Numer. Methods Engrg. 40 (1997) 727–758.
[14] C.A. Duarte, J.T. Oden, *Hp* clouds – a meshless method to solve boundary-value problems, Technical Report 95-05, Texas Institute for Computational and Applied Mathematics, Austin, 1995.
[15] C.A. Duarte, J.T. Oden, An *h–p* adaptive method using clouds, Comput. Methods. Appl. Mech. Engrg. 139 (1996) 237–262.
[16] S.N. Atluri, T. Zhu, A new meshless local Petrov–Galerkin (MLPG) approach in computational mechanics, Comput. Mech. 22 (1998) 117–127.
[17] B. Nayroles, G. Touzot, P. Villon, Generalizing the finite element method: diffuse element approximation and diffuse elements, Comput. Mech. 10 (1992) 307–318.
[18] W.K. Liu, S. Jun, D.T. Sihling, Y.J. Chen, W. Hao, Multiresolution reproducing kernel particle method for computational fluid dynamics, Int. J. Numer. Methods Fluids 24 (1997) 1391–1415.
[19] H. Sadat, S. Couturier, Performance and accuracy of a meshless method for laminar natural convection, Numer Heat Tranf. B 37 (2000) 455–467.
[20] G. Yagawa, M. Shirazaki, Parallel computing for incompressible flow using a nodal-based method, Comput. Mech. 23 (1999) 209–217.
[21] M. Cheng, G.R. Liu, A novel finite point method for flow simulation, Int. J. Numer. Methods Fluids 39 (2002) 1161–1178.
[22] D.W. Kim, Y.S. Kim, Point collocation methods using the fasting moving least square reproducing kernel approximation, Int. J. Numer. Methods Engrg. 56 (2003) 1445–1464.
[23] S.H. Park, S.K. Youn, The least-squares meshfree method, Int. J. Numer. Methods Engrg. 52 (2001) 997–1012.
[24] S.H. Park, K.C. Kwon, S.K. Youn, A study on the convergence of least-squares meshfree method under inaccurate integration, Int. J. Numer. Methods Engrg. 56 (2003) 1397–1419.
[25] S.H. Park, K.C. Kwon, S.K. Youn, A posterior error estimates and an adaptive scheme of least-squares meshfree method, Int. J. Numer. Methods Engrg. 58 (2003) 1213–1250.
[26] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, P. Krysl, Meshless methods: an overview and recent developments, Comput. Methods. Appl. Mech. Engrg. 139 (1996) 3–47.
[27] P. Krysl, T. Belytschko, ESFLIB: A library to compute the element free Galerkin shape functions, Comput. Methods. Appl. Mech. Engrg. 190 (2001) 2181–2205.
[28] L.Q. Tang, T.T.H. Tsang, An efficient least-squares finite element method for incompressible flows and transport processes, Int. J. Comput. Fluid Dyn. 4 (1995) 21–39.
[29] P.L. Gorge, Automatic Mesh Generation – Application to Finite Element Methods, Wiley, Masson, 1991.
[30] M.D. Berg, M.V. Kreveld, M. Overmars, O. Schwwarzkopf, Computational Geometry – Algorithms and Applications, Springer, Berlin, 1997.
[31] B.N. Jiang, On the least-squares method, Comput. Methods. Appl. Mech. Engrg. 152 (1998) 239–257.
[32] S.H. Park, A study on error estimates and an adaptive scheme of least-squares meshless method, Ph.D. Dissertation, Korea Advanced Institute of Science and Technology, 2001.

[33] W.K. Liu, S. Li, T. Belytschko, Reproducing least square kernel Galerkin method, (I) Methodology and convergence, Comput. Methods. Appl. Mech. Engrg. 143 (1997) 113–154.
[34] B.N. Jiang, C.L. Chang, Least-squares finite elements for the Stokes problems, Comput. Methods. Appl. Mech. Engrg. 78 (1990) 297–311.
[35] F. Pan, A. Acrivos, Steady flows in rectangular cavities, J. Fluid. Mech. 28 (1967) 643–655.
[36] P.N. Shankar, The eddy structure in Stokes flow in a cavity, J. Fluid. Mech. 250 (1993) 371–383.
[37] A. Thom, The flow past circular cylinders at low speeds, Proc. Roy. Soc. A 141 (1933) 651–669.
[38] F. Homann, Der Einfluss grosser Zähigkeit bei der strömung um den Zylinder und um die Kugel, Z. Angew. Math. Mech. 16 (1936) 153. Translation: The effect of high viscosity on the flow around a cylinder and around a sphere, NACA TM 1334 (1952).
[39] S. Taneda, Experimental investigation of the wakes behind cylinders and plates at low Reynolds numbers, J. Phys. Soc. Jpn. 11 (1956) 302–306.
[40] D.J. Tritton, Experiments on the flow past a circular cylinder at low Reynolds numbers, J. Fluid. Mech. 6 (1959) 547–567.
[41] A.S. Grove, F.H. Shair, E.E. Petersen, A. Acrivos, An experimental investigation of the steady separated flow past a circular cylinder, J. Fluid. Mech. 19 (1964) 60–80.
[42] A. Acrivos, L.G. Leal, D.D. Snowden, F. Pan, Further experiments on steady separated flows past bluff objects, J. Fluid. Mech. 34 (1968) 25–48.
[43] H. Takami, H.B. Keller, Steady two-dimensional viscous flow of an incompressible fluid past a circular cylinder, Phys. Fluids 12 (Suppl. II.) (1969), II-51.
[44] S.C.R Dennis, G.Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, J. Fluid. Mech. 42 (1970) 471–489.
[45] F. Nieuwstadt, H.B. Keller, Viscous flow past circular cylinders, Comput. Fluids 1 (1973) 59.
[46] D. Sucker, H Brauer, Fluiddynamik bei der angeströmten Zylindern, Wärme- und Stoffübertragung. 8 (1975) 149.
[47] B. Fornberg, A numerical study of steady viscous flow past a circular cylinder, J. Fluid. Mech. 98 (1980) 819–855.
[48] M. Braza, P. Chassaing, H.Ha. Minh, Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder, J. Fluid Mech. 165 (1986) 79–130.
[49] A. Shklyar, A. Arbel, Numerical method for calculation of the incompressible flow in general curvilinear co-ordinates with double staggered grid, Int. J. Numer. Meth. Fluids 41 (2003) 1273–1294.
[50] A.L.F Lima E Silva, A. Silveira-Neto, J.J.R. Damasceno, Numerical simulation of two-dimensional flow over a circular cylinder using the immersed boundary method, J. Comput. Phys. 189 (2003) 351–370.
[51] H. Ding, C. Shu, K.S. Yeo, D. Xu, Simulation of incompressible viscous flows past a circular cylinder by hybrid FD scheme and meshless least square-based finite difference method, Comput. Methods. Appl. Mech. Engrg. 193 (2004) 727–744.