Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. 7, No. 4, pp. 430-440

DOI: 10.4208/aamm.2013.m337
August 2015

Local RBFs Based Collocation Methods for Unsteady Navier-Stokes Equations

Xueying Zhang[1,*], Xin An[1] and C. S. Chen[2,3]

[1] *College of Science, Hohai University, Nanjing, Jiangsu 210098, China*
[2] *Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, U.S.A*
[3] *Department of Engineering Mechanics, Hohai University, Nanjing, Jiangsu 210098, China*

Received 4 September 2013; Accepted (in revised version) 19 October 2014

Abstract. The local RBFs based collocation methods (LRBFCM) is presented to solve two-dimensional incompressible Navier-Stokes equations. In avoiding the ill-conditioned problem, the weight coefficients of linear combination with respect to the function values and its derivatives can be obtained by solving low-order linear systems within local supporting domain. Then, we reformulate local matrix in the global and sparse matrix. The obtained large sparse linear systems can be directly solved instead of using more complicated iterative method. The numerical experiments have shown that the developed LRBFCM is suitable for solving the incompressible Navier-Stokes equations with high accuracy and efficiency.

AMS subject classifications: 65M10, 35L03

Key words: Navier-Stokes equations, local collocation method, particular solutions, radial basis functions (RBFs), large sparse linear systems.

1 Introduction

Due to avoid the high cost of mesh generation, the interest in meshless methods to solve partial differential equations (PDEs) has generated considerably in the past decades. In contrary to the mesh based methods like the the finite difference method (FDM), the finite volume method (FVM) and the finite element method (FEM), meshless methods use a set of uniform or random points to discrete derivatives. Those points are not necessarily interconnected in the form of a grid. Among them a class of meshless methods based on radial basis functions (RBFs) seem much more appealing and have been considered

*Corresponding author.
Email: zhangxy.math@gmail.com (X. Y. Zhang)

 ©2015 Global Science Press

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

as a prospective numerical methods for solving partial differential equations (PDEs). Originally, RBFs were proposed for scattered data and function interpolation by Hardy [1]. However, due to the fact that methods based on RBFs are truly meshfree and spatial dimension independent, which can easily be applied to solve high dimensional problems, researchers got more and more interested in the implementation of RBFs in PDEs solvers. The first trial was successfully done by Kansa [2,3]. Franke [4] showed that Hardy's multiquadric (MQ) and Duchon's thin-plate spline (TPS) ranked the best in accuracy, efficiency, and stability. Recently, an increasing number of researchers extended RBFs' methods to solve various ordinary and partial differential equations including regularized long wave (RLW) equation [5], Hirota-Satsuma coupled KdV equations [6], the solution of 2D biharmonic equations [7], the case of heat transfer equations [8], incompressible Navier-Stokes equations [9,10], natural convection problems [11], elliptic problems with variable coefficients [12, 13], scalar wave equations [14], natural convection problems, hyperbolic partial differential equations [15] and so on.

In general, the RBFs used in the collocation technique are globally supported. The resultant matrices are often dense and even ill-conditioned, and thus restricting our ability to solve large-scale problems. To circumvent this issue, considerable efforts have been made such as domain decomposition [17], multigrid approach and compactly supported RBFs [18], the improved truncated singular valued decomposition [19], etc. However, among all those dealings localized methods are more direct and simple. Currently many localized meshless methods have been developed. For example, Shu et al. [9, 10] applied the local RBF-based differential quadrature (LRBFDQ) method to simulate incompressible Navier-Stokes equations in 2003 and then Lee et al. [20], Sarler et al. [21] proposed local radial basis function collocation method on the basis of global RBFCM and applied it to solve convective-diffusive problems. Recently, Chen et al. [16] presented the localized method of approximated particular solutions (LMAPS). By adopting those methods, we can successfully avoid severe ill-conditioning problem and form a sparse coefficient matrix as approximations by RBF interpolation are all conducted within local domain. In this paper, motivated by the idea of local RBFs based meshless method, we will present readers the so-called local radial basis function collocation method (LRBFCM), and apply it to solve two-dimensional, time-dependent, incompressible Navier-Stokes equations in vorticity-stream function form.

The organization of the paper is as follows. In Section 2, the formulations of the LRBFCM are listed for solving two-dimensional, time-dependent Navier-Stokes equations in vorticity-stream function form in detail. In Section 3, numerical examples are given to demonstrate the effectiveness of the proposed procedure. In Section 4, we draw some conclusions.

2 Formulation of the LRBFCM for incompressible N-S equations

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

2.1 Governing equations

We consider the two dimensional unsteady incompressible Navier-Stokes equation in the vorticity-stream function form [10], which can be written as

$$\frac{\partial \omega}{\partial t}+\mathbf{u}\cdot\nabla\omega=\frac{1}{Re}\Delta\omega, \tag{2.1a}$$

$$\Delta\psi=\omega, \tag{2.1b}$$

where Re is Reynolds number, ω is the vorticity, ψ is the stream function, $\mathbf{u}=(u,v)$, u and v denote the components of velocity in the x and y directions, which can be calculated from the stream function

$$u=\frac{\partial \psi}{\partial y},\quad v=-\frac{\partial \psi}{\partial x}. \tag{2.2}$$

Right side of Eq. (2.1) contains second-order Laplacian operator which can be approximated by RBFs based interpolation. Note that Eq. (2.1b) is Poisson equation whose solution can be obtained by using the RBFs as particular solutions.

2.2 Local approximation of derivatives

The ill-conditioning of the interpolation matrix and the large scale dense matrix have limited the development of global RBFs based meshless methods. To mitigate these difficulties, local schemes based on local supporting domain seem to be more efficient, less sensitive to changes of the value of the shape parameter, thereby avoiding the ill-conditioning problem.

We denote $\mathbf{x}=(x,y)$ and $\{\mathbf{x}_j\}_{j=1}^{n_s}$ be the interpolation points inside the influence domain. For any point $\mathbf{x}_p$, its supporting domain Ω_p contains the nearest n_s interpolation points $\{\mathbf{x}_j\}_{j=1}^{n_s}$ to $\mathbf{x}_p$. The value of a function $f(\mathbf{x})$ at $\mathbf{x}_p$ can be approximated by a linear combination of n_s radial basis functions (RBFs) in the following form

$$f(\mathbf{x}_p)\simeq\hat{f}(\mathbf{x}_p)=\sum_{j=1}^{n_s}a_j\phi\big(\|\mathbf{x}_p-\mathbf{x}_j\|\big),\quad \mathbf{x}_p\in\Omega_p, \tag{2.3}$$

where ϕ is a RBF, $\|\cdot\|$ is the Euclidean norm, and $\{a_j\}_{j=1}^{n_s}$ are coefficients to be determined. We prefer to choose multiquadric (MQ) RBFs that have been found to provide very accurate approximation in most of the applications and have been widely used by researchers. The MQ-RBFs can be written as

$$\phi(r)=\sqrt{r^2+c^2}, \tag{2.4}$$

where $r=\|\mathbf{x}-\mathbf{x}_i\|$, $\mathbf{x}=(x,y)$ represents the coordinates of a node in the physical space. The shape parameter c is crucial to the accuracy of the approximation that plays a very important role on numerical errors and condition number of a matrix. The optimal c depends

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

on the number of the nodes in supporting region, the scale of supporting region and the distribution of supporting nodes. The general behavior of c is such that the larger the shape parameter, the MQ becomes increasingly flat, thus the smaller the approximation error. However, increasing the shape parameter results in a more ill-conditioned matrix and therefore increases the rounding errors and instability. In order to obtain more accurate RBFs approximation, one has to balance the scale of the supporting region and the sensitivity of shape parameter. The scale of supporting region for each reference node is normalized by the following coordinate transformation [9]

$$\bar{x}=\frac{x}{R_i},\quad \bar{y}=\frac{y}{R_i}, \tag{2.5}$$

where R_i is the radius of the minimal circle enclosing all knots in the supporting region for each reference point i. Note that the shape parameter c for traditional MQ-RBFs is equivalent to $\bar{c}R_i$. With the above transformation, the local supporting region is transferred to a unit circle or a unit sphere. In the application, once $\bar{c}$ is chosen as a constant, the optimal c is only related to number and distribution of supporting nodes. The selection for a suitable shape parameter is still one of the challenging research topics.

We note that Eq. (2.3) can be rewritten as

$$\hat{f}(\mathbf{x}_p)=\boldsymbol{\theta}_{n_s}\mathbf{a}_{n_s}, \tag{2.6}$$

where

$$\boldsymbol{\theta}_{n_s}=\left[\phi(\|\mathbf{x}_p-\mathbf{x}_1\|),\cdots,\phi(\|\mathbf{x}_p-\mathbf{x}_{n_s}\|)\right],$$

and $\mathbf{a}_{n_s}=[a_1,a_2,\cdots,a_{n_s}]^T$. From Eq. (2.3), by direct differentiation, we have

$$\frac{\partial f(\mathbf{x}_p)}{\partial x}\simeq\frac{\partial \hat{f}(\mathbf{x}_p)}{\partial x}=\sum_{j=1}^{n_s}a_j\frac{\partial\phi}{\partial x}(\|\mathbf{x}_p-\mathbf{x}_j\|)=\Xi_{n_s}^x\mathbf{a}_{n_s},\qquad \mathbf{x}_p\in\Omega_p, \tag{2.7a}$$

$$\frac{\partial f(\mathbf{x}_p)}{\partial y}\simeq\frac{\partial \hat{f}(\mathbf{x}_p)}{\partial y}=\sum_{j=1}^{n_s}a_j\frac{\partial\phi}{\partial y}(\|\mathbf{x}_p-\mathbf{x}_j\|)=\Xi_{n_s}^y\mathbf{a}_{n_s},\qquad \mathbf{x}_p\in\Omega_p, \tag{2.7b}$$

and

$$\Delta f(\mathbf{x}_p)\simeq\Delta\hat{f}(\mathbf{x}_p)=\sum_{j=1}^{n_s}a_j\Phi\left(\|\mathbf{x}_p-\mathbf{x}_j\|\right)=\boldsymbol{\Theta}_{n_s}\mathbf{a}_{n_s},\quad \mathbf{x}_p\in\Omega_p, \tag{2.8}$$

where

$$\Delta\phi(\|\mathbf{x}-\mathbf{x}_j\|)=\Phi(\|\mathbf{x}-\mathbf{x}_j\|), \tag{2.9a}$$

$$\Xi_{n_s}^x=\left[\frac{\partial\phi}{\partial x}(\|\mathbf{x}_p-\mathbf{x}_1\|),\cdots,\frac{\partial\phi}{\partial x}(\|\mathbf{x}_p-\mathbf{x}_{n_s}\|)\right], \tag{2.9b}$$

$$\Xi_{n_s}^y=\left[\frac{\partial\phi}{\partial y}(\|\mathbf{x}_p-\mathbf{x}_1\|),\cdots,\frac{\partial\phi}{\partial y}(\|\mathbf{x}_p-\mathbf{x}_{n_s}\|)\right], \tag{2.9c}$$

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

and

$$\mathbf{\Theta}_{n_s}=[\Phi(\|\mathbf{x}_p-\mathbf{x}_1\|),\cdots,\Phi(\|\mathbf{x}_p-\mathbf{x}_{n_s}\|)]. \tag{2.10}$$

Furthermore,

$$\frac{\partial\phi}{\partial x}=\frac{x-x_i}{\sqrt{r^2+c^2}},\quad \frac{\partial\phi}{\partial y}=\frac{y-y_i}{\sqrt{r^2+c^2}}. \tag{2.11}$$

Note that $\Delta=(1/r)(d/dr(r(d/dr)))$. From Eq. (2.9a), we have

$$\Phi(r)=\frac{r^2+2c^2}{(r^2+c^2)^{1.5}}. \tag{2.12}$$

Since $\{\mathbf{x}_j\}_{j=1}^{n_s}\subset\Omega_p$, Eq. (2.6) holds for every $\{\mathbf{x}_j\}_{j=1}^{n_s}$. By collocation method, we have the following matrix system

$$\mathbf{\Phi}_{n_s}\mathbf{a}_{n_s}=\hat{\mathbf{f}}_{n_s}, \tag{2.13}$$

where $\hat{\mathbf{f}}_{n_s}=[\hat{f}(\mathbf{x}_1),\cdots,\hat{f}(\mathbf{x}_{n_s})]^T$ and $\mathbf{\Phi}_{n_s}=[\phi(\|\mathbf{x}_j-\mathbf{x}_i\|)]_{1\le i,j\le n_s}$. It follows that

$$\mathbf{a}_{n_s}=\mathbf{\Phi}_{n_s}^{-1}\hat{\mathbf{f}}_{n_s}. \tag{2.14}$$

Substituting Eq. (2.14) into Eqs. (2.7a)-(2.8), we have

$$\frac{\partial\hat{f}(\mathbf{x}_p)}{\partial x}=\left(\Xi_{n_s}^{x}\mathbf{\Phi}_{n_s}^{-1}\right)\hat{\mathbf{f}}_{n_s},\quad \frac{\partial\hat{f}(\mathbf{x}_p)}{\partial y}=\left(\Xi_{n_s}^{y}\mathbf{\Phi}_{n_s}^{-1}\right)\hat{\mathbf{f}}_{n_s}, \tag{2.15}$$

and

$$\Delta\hat{f}(\mathbf{x}_p)=\left(\mathbf{\Theta}_{n_s}\mathbf{\Phi}_{n_s}^{-1}\right)\hat{\mathbf{f}}_{n_s}. \tag{2.16}$$

It can be seen that the approximations of spatial derivatives are evaluated based on RBFs within local supporting domain containing a set of arbitrarily distributed points. It is required to search the nearest neighbor points to define local supporting domain for each reference point.

2.3 The discretization of N-S equations

Using Eqs. (2.15)-(2.16), the governing equations Eqs. (2.1)-(2.2) are discretized as

$$\frac{d\omega_i^{(n)}}{dt}=\frac{1}{Re}\sum_{j=1}^{n_s}a_{ij}\omega_j^{(n)}-\left(u_i^{(n)}\sum_{j=1}^{n_s}c_{ij}\omega_j^{(n)}+v_i^{(n)}\sum_{j=1}^{n_s}b_{ij}\omega_j^{(n)}\right), \tag{2.17a}$$

$$\sum_{j=1}^{n_s}a_{ij}\psi_j^{(n+1)}=\omega_i^{(n+1)}, \tag{2.17b}$$

$$u_i^{(n+1)}=\sum_{j=1}^{n_s}b_{ij}\psi_j^{(n+1)},\quad v_i^{(n+1)}=-\sum_{j=1}^{n_s}c_{ij}\psi_j^{(n+1)}, \tag{2.17c}$$

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

where n is the time step index, n_s is the number of the interpolation points inside the influence domain Ω_i for any point $\mathbf{x}_i \in \Omega$, $i = 1,2,\cdots,N$.

It is noted that the coefficients a_{ij}, b_{ij} and c_{ij} in Eqs. (2.17a)-(2.17c) can be calculated by the coefficient matrices in Eqs. (2.15)-(2.16). Once the weighting coefficients are obtained, we reformulate local coefficient matrices in global and sparse form by inserting zeros in proper positions. For further details of the reformulation, refer to [12, 16]. Thus Eqs. (2.17b) and (2.17c) can be transformed into matrix formulation

$$W^{(n+1)} = A\Psi^{(n+1)}, \quad U^{(n+1)} = B\Psi^{(n+1)}, \quad V^{(n+1)} = -C\Psi^{(n+1)}, \tag{2.18}$$

where U and V are $n \times 1$ vectors. Ultimately, reformulating local coefficient matrices, we have the following discretization for time-dependant equation (2.17a)

$$\frac{dW^{(n)}}{dt} = \frac{1}{Re} AW^{(n)} - \left(\mathbb{U}C + \mathbb{V}B\right) W^{(n)}, \tag{2.19}$$

where A, B and C are $n \times n$ sparse matrices, $\mathbb{U}$ and $\mathbb{V}$ are $n \times n$ diagonal matrices in which all of the diagonal elements come from the elements of the vectors U and V. Further, we obtain $\Psi^{(n+1)}$ from Eq. (2.18) as follows

$$\Psi^{(n+1)} = A^{-1} W^{(n+1)}. \tag{2.20}$$

The resultant equations can be solved with implementation of accurate boundary conditions. The general solution procedure of the LRBFCM for the governing equations is shown below:

Step 1 Discretize the governing equations (2.1)-(2.2) into Eqs. (2.17a)-(2.17c) with the computed weighting coefficients for the related derivatives, then reformulate them in global form, see Eqs. (2.18)-(2.20).

Step 2 Suppose the initial-boundary conditions of streamline function, vorticity and velocity are known at t^n, solve the time-dependent discretized equation (2.19) to obtain the new vorticity distribution.

Step 3 Determine the updated vorticity boundary values using the definition of vorticity.

Step 4 Solve the discretized Laplace equation (2.20) to obtain the new distribution for ψ.

Step 5 Calculate the derivatives of streamline function to obtain velocity distribution, also update the velocity boundary conditions.

Step 6 Go to step 2 until a required time step is reached.

3 Stokes flow in a square cavity

We apply the presented methods to solve a unsteady incompressible Navier-Stokes problem in square cavity to examine the efficiency and accuracy of the proposed local RBFs

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

based collocation method. Maximum absolute error (MAE), the maximum relative error (MRE) and the root mean square error (RMSE) are computed to validate the efficiency and accuracy of the LRBFCM. These error notations are defined as

$$MAE=\max_{1\le j\le N}\left|u(x_j,t^n)-\hat{u}(x_j,t^n)\right|,\quad MRE=\max_{1\le j\le N}\frac{\left|u(x_j,t^n)-\hat{u}(x_j,t^n)\right|}{\left|u(x_j,t^n)\right|},\tag{3.1}$$

and

$$RMSE=\left[\frac{1}{N}\sum_{j=1}^{N}\left(u(x_j,t^n)-\hat{u}(x_j,t^n)\right)^2\right]^{\frac{1}{2}},\tag{3.2}$$

where $u(x_j,t^n)$ and $\hat{u}(x_j,t^n)$ represent the exact and numerical solutions of a given partial differential equation.

The computational domain $\Omega\cup\partial\Omega$ for the problem is $0\le x\le\pi/2$, $0\le y\le\pi/2$ and $0<t<+\infty$ and Re is chosen to be 1. $\Omega\cup\partial\Omega$ is a regular domain where the interior and boundary nodes are distributed uniformly. The number of the approximation nodes in the local domain is 5. The Eqs. (2.17a)-(2.17c) are subjected to the Dirichlet boundary conditions

$$\psi(x,y,t)=\cos(y)e^{-2t},\qquad \text{at } x=0,\qquad 0\le y\le\frac{\pi}{2},\tag{3.3a}$$

$$\psi(x,y,t)=0,\qquad \text{at } x=\frac{\pi}{2},\qquad 0\le y\le\frac{\pi}{2},\tag{3.3b}$$

$$\psi(x,y,t)=\cos(x)e^{-2t},\qquad \text{at } y=0,\qquad 0\le x\le\frac{\pi}{2},\tag{3.3c}$$

$$\psi(x,y,t)=0,\qquad \text{at } y=\frac{\pi}{2},\qquad 0\le x\le\frac{\pi}{2}.\tag{3.3d}$$

The initial condition for ψ is given by

$$\psi(x,y,0)=\cos(x)\cos(y),\quad (x,y)\epsilon\Omega\cup\partial\Omega.\tag{3.4}$$

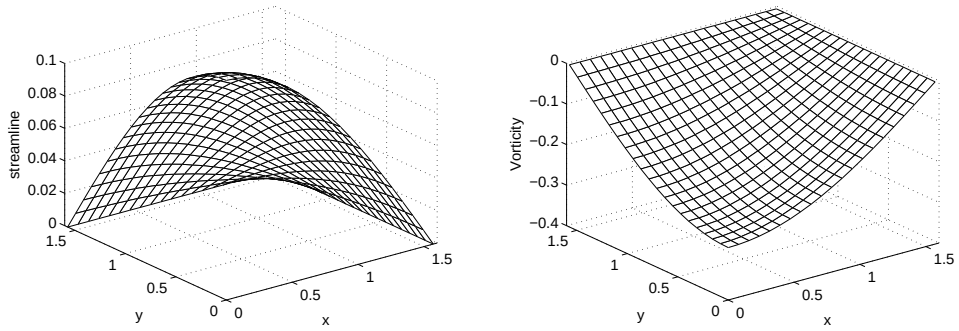
The updating of boundary condition for vorticity ω can be derived from Eq. (2.1b)

$$\omega|_{wall}=\left(\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial y^2}\right)\Big|_{wall}.\tag{3.5}$$

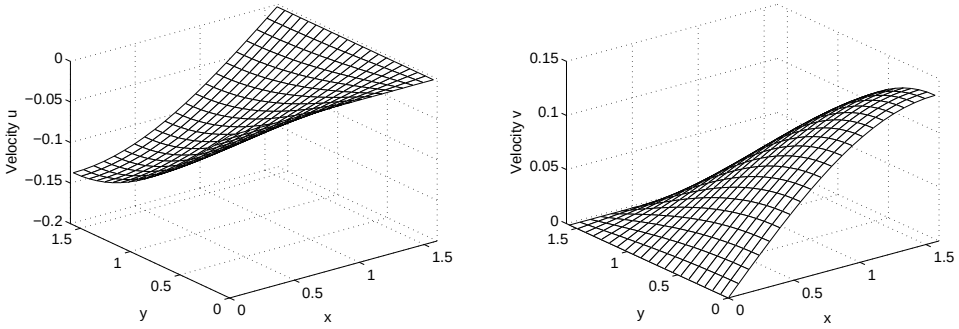
The derivatives can be approximated by the conventional one-sided second order finite difference method. In order to achieve that, three layers of orthogonal grid near and including the boundary are placed near the boundary [9]. Like the governing equations, boundary conditions for vorticity involves the derivatives of stream function can also be discretized by the local method of collocation particular solutions. The analytical solutions for the problem can be found in [10], we don't repeat them here.

Fig. 1 shows that the distributions of streamline, vorticity and velocity with $N=441$ and $c=50$ at $t=1.0$. Here c is the ratio of the shape parameter in Eq. (2.4) to r_0 where r_0

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press



(a) Streamline (b) Vorticity



(c) Velocity u (d) Velocity v

Figure 1: The profiles of streamline, vorticity and velocity with $N=441$, $c=50$ at $t=1$.

is the maximum distance from $\mathbf{x}_p$ to all nodal points in the supporting domain Ω_p, the below is same. In other words,

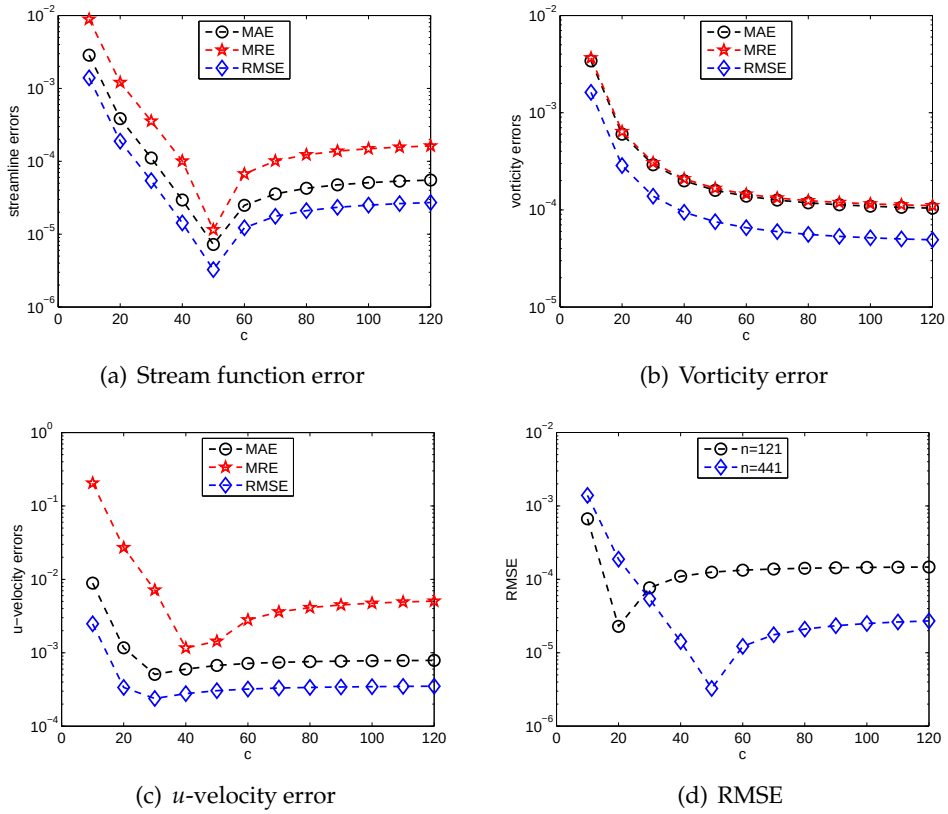
$$r_0 = \max_{1\le j\le n_s} \left\| \mathbf{x}_p - \mathbf{x}_j^{[p]} \right\|. \tag{3.6}$$

Instead of iterative method, it took only less computational complexity to obtain the results for each time step. So the computational efficiency is also remarkable.

Fig. 2 shows the profiles of various errors versus shape parameters c with $N=121$, $N=441$ at $t=0.1$. It can be observed that the accuracy of the numerical solution of different variables is improved as the value of the shape parameter c increases. When c comes to a certain value, the errors show a tendency toward stabilization. Moreover, as shown in Fig. 2(d) the optimal shape parameter c should increase with the number of nodes.

Table 1 shows the MAE and RMSE errors by the LRBFCM for this problem. As we have seen from Table 1, the accuracy of the proposed method is improved as N increases while the computational efficiency decreases. Both MAE and RMSE errors for streamline function ψ achieve at most 10^{-7} scale with mesh points $N=961$. There are practically no

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press



(a) Stream function error (b) Vorticity error

(c) u-velocity error (d) RMSE

Figure 2: The profiles of various errors versus shape parameters c with $N=121$, $N=441$ at $t=0.1$.

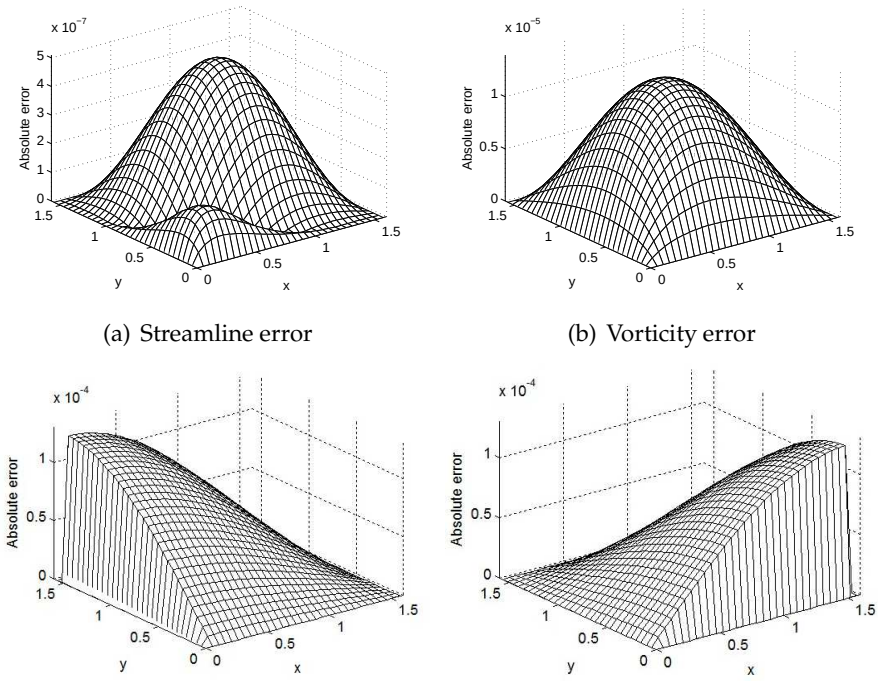
differences for velocity errors in x and y directions with same mesh size. It is also seen that the accuracy for ω and ψ is higher than that for u and v. This is because ω and ψ are directly computed from the Eqs. (2.19) and (2.20) while u and v are indirectly computed from the stream function using the Eq. (2.18).

The obtained absolute errors for different variables are shown in Fig. 3. In the computer implementation for the problem, 441 interior nodes and 80 boundary nodes are chosen. The errors show that the present scheme can achieve good agreement with the exact solutions. It can be also seen that the errors in most of the irregular domain are going down smoothly.

Table 1: The obtained errors by the LRBFCM scheme.

N	c	t	MAE			RMSE		
			stream	vorticity	velocity	stream	vorticity	velocity
121	20	1.0	$1.3e-5$	$7.4e-5$	$3.9e-4$	$6.1e-6$	$3.5e-5$	$1.6e-4$
441	50	1.0	$3.0e-6$	$5.3e-5$	$1.1e-4$	$1.3e-6$	$2.6e-5$	$5.1e-5$
961	65	0.5	$5.0e-7$	$1.3e-5$	$1.2e-4$	$2.1e-7$	$6.7e-6$	$5.8e-5$

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press



(a) Streamline error (b) Vorticity error

(c) Velocity error in x direction (d) Velocity error in y direction

Figure 3: The error profiles obtained by the LRBFCM with $N=961$, $c=65$ at $t=0.5$.

4 Concluding remarks

In this paper, a local RBFs-based method of collocation particular solution is developed for solving two-dimensional, time-dependent, incompressible Navier-Stokes equations. In order to overcome the instability of the ill-posed problem, the weight coefficients can be obtained by solving low order linear systems within local supporting domains. The proposed numerical scheme is validated by comparisons of the obtained numerical results with the corresponding analytical solution. In general, the presented LRBFCM appears to be attractive for being being able to efficiently solve Navier-Stokes system containing Laplacian operator.

References

[1] R. L. HARDY, *Multiquadric equations of topography and other irregular surfaces*, J. Geophys. Res., 76 (1971), pp. 1905–1915.

[2] E. J. KANSA, *A scattered scattered data approximation scheme with applications to computational fluiddynamics, I. Surface approximations and partial derivative estimates*, Comput. Math. Appl., 19 (1990), pp. 127–145.

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press

[3] E. J. KANSA, *Multiquadrics, a scattered data approximation scheme with applications to computational fuid dynamics, II. Solutions to parabolic, hyperbolic and elliptic partial differential equations,* Comput. Math. Appl., 19 (1990), pp. 147–161.
[4] R. FRANKE, *Scatterd data interpolation: tests of some methods,* Math. Comput., 38 (1982), pp. 181–199.
[5] S. U. ISLAM, S. HAQB AND A. ALI, *A meshfree method for the numerical solution of the RLW equation*, J. Comput. Appl. Math., 223 (2009), pp. 997–1012.
[6] A. J. KHATTAK, S. I. A. TIRMIZI AND S. U. ISLAM, *Application of meshfree collocation method to a class of nonlinear partial differential equations*, Eng. Anal. Bound. Elem., 33 (2009), pp. 661–667.
[7] N. MAI-DUY, D. HO-MINH AND T. TRAN-CONG, *A Galerkin approach incorporating integrated radial basis function networks for the solution of 2D biharmonic equations*, Int. J. Comput. Math., 86 (2009), pp. 1746–1759.
[8] K. PARAND, S. ABBASBANDY, S. KAZEM AND A. REZAEI, *Comparison between two common collocation approaches based on radial basis functions for the case of heat transfer equations arising in porous medium*, Commun. Nonlinear Sci. Numer. Simul., 16 (2011), pp. 1396–1407.
[9] C. SHU, H. DING AND K. S. YEO, *Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stokes equations,* Comput. Methods Appl. Mech. Eng., 192 (2003), pp. 941–954.
[10] C. SHU, Q. YAO AND K. S. YEO, *Block-marching in time with DQ discretization: an efficient method for time-dependent problems,* Comput. Methods Appl. Mech. Eng., 191 (2002), pp. 4587–4597.
[11] G. KOSEC AND B. SARLER, *Solution of a low Prandtl number natural convection benchmark by a local meshless method*, Int. J. Numer. Methods Heat Fluid Flow, 23 (2012), pp. 189–204.
[12] C. S. CHEN, C. M. FAN AND P. H. WEN, *The method of particular solutions for solving elliptic problems with variable coefficients,* Int. J. Comput. Methods, 8 (2011), pp. 545–559.
[13] E. LARSSON AND B. FORNBERG, *A numerical study of radial basis functions based solution methods for elliptic pdes*, Comput. Math. Appl., 46 (2003), pp. 891–902.
[14] P. H. WEN AND C. S. CHEN, *The method of particular solutions for solving scalar wave equations*, Int. J. Numer. Methods Biomed. Eng., 26 (2010), pp. 1878–1889.
[15] S. U. ISLAM, R. VERTNIK AND B. SARLER, *Local radial basis function collocation method along with explicit time stepping for hyperbolic partial differential equations,* Appl. Numer. Math., 67 (2013), pp. 136–151.
[16] GUANGMING YAO, C. S. CHEN AND JOSEPH KOLIBAL, *A localized approach for the method of approximate particular solutions,* Comput. Math. Appl., 61 (2011), pp. 2376–2387.
[17] J. C. LI AND Y. C. HON, *Domain decomposition for radialbasis meshless methods,* Numer. Meth. Part. Differ. Eq., 20(3) (2004), pp. 450–462.
[18] C. S. CHEN, M. A. GOLBERG, M. GANESH AND A. H.-D. CHENG, *Multilevel compact radial functions basedcomputational schemes for some elliptic problems,* Comput. Math. Appl., 43 (2002), pp. 359–378.
[19] N. A. LIBRE, A. EMDADI AND E. J. KANSA ET AL., *A stabilized rbf collocation scheme for neumann type boundary value problems,* Comput. Model. Eng. Sci., 24 (2008), pp. 61–80.
[20] C. K. LEE, X. LIU AND S. C. FAN, *Local muliquadric approximation for solving boundary value problems*, Comput. Mech., 30 (2003), pp. 395–409.
[21] B. SARLER AND R. VERTNIK, *Meshfree explicit local radial basis function collocation method for diffusion problems,* Comput. Math. Appl., 21 (2006), pp. 1269–1282.

https://doi.org/10.4208/aamm.2013.m337 Published online by Cambridge University Press