ผลเฉลยของสมการเนเวียร์สโต๊กส์สองมิติโดยวิธีการจัดตำแหน่งจุดซึ่งใช้ฟังก์ชันฐานเชิงรัศมี

สมชาติ ฉันทศิริวรรณ[1]

มหาวิทยาลัยธรรมศาสตร์ ศูนย์รังสิต คลองหลวง ปทุมธานี 12121

บทคัดย่อ

เท่าที่ผ่านมาการแก้สมการเนเวียร์สโต๊กส์สองมิติซึ่งมีสามสมการและสามตัวแปรมักใช้วิธีเชิงตัวเลขตามแนวทางตัวแปรปฐมฐานหรือแนวทางกระแสวนและฟังก์ชันสายธาร บทความนี้นำเสนอแนวทางใหม่โดยการกำจัดความดันและส่วนประกอบความเร็วตัวหนึ่งออกจากสมการเนเวียร์สโต๊กส์ ผลที่ได้คือ สมการเชิงอนุพันธ์ย่อยอันดับหนึ่งสมการซึ่งมีตัวแปรที่ไม่ทราบค่าเพียงตัวเดียว นอกจากนี้บทความยังนำเสนอวิธีการจัดตำแหน่งจุดซึ่งใช้ฟังก์ชันฐานเชิงรัศมีสำหรับแก้สมการนี้ ตัวแปรที่ไม่ทราบค่าในสมการคือ ส่วนประกอบความเร็วจะถูกประมาณค่าเป็นผลรวมเชิงเส้นของฟังก์ชันฐาน การหาค่าสัมประสิทธิ์ของการประมาณค่าใช้วิธีการคำนวณซ้ำ วิธีนี้ใช้แก้ปัญหาตัวอย่างซึ่งทราบผลเฉลยแม่นตรง ผลที่ได้แสดงให้เห็นว่าจำนวนครั้งของการคำนวณซ้ำเพื่อให้ได้ผลเฉลยลู่เข้าและความแม่นยำของผลเฉลยขึ้นกับพารามิเตอร์อิสระของฟังก์ชันฐาน

[1] *รองศาสตราจารย์ คณะวิศวกรรมศาสตร์*

Solutions of Two-dimensional Navier-Stokes Equations by a Collocation Method Based on Radial Basis Functions

Somchart Chantasiriwan[1]

Thammasat University, Rangsit Campus, Pathum Thani 12121, Thailand

Abstract

The two-dimensional Navier-Stokes equations consisting of three equations and three unknowns have been solved by conventional methods using the primitive-variable approach and the vorticity-stream function approach. A new approach is proposed in this paper. By getting rid of pressure and one of the velocity components, the Navier-Stokes equations can be reduced to a third-order partial differential equation with one velocity component as the only unknown. A collocation method based on radial basis functions is proposed for solving this equation. Unknown velocity component is approximated as a linear combination of basis functions. Unknown coefficients are determined by an iterative scheme. The proposed method is used to solve a test problem, for which exact solution is known. It is found that the number of iterations required for a convergence and the accuracy of the solution depends on the free parameter of basis functions.

---

[1] *Associate Professor, Faculty of Engineering.*

1. Introduction

The incompressible Navier-Stokes equations are coupled partial differential equations of pressure and velocity components. The two most popular approaches for solving these equations are the primitive-variable approach and the vorticity-stream function approach [1]. In the primitive-variable approach, the Navier-Stokes equations are to be solved for primitive variables (pressure and velocity components). Imposition of boundary conditions in this approach is quite straightforward. The vorticity-stream function approach requires the transformation of the Navier-Stokes equations into equations of derived variables (vorticity and stream function). Pressure and velocity components are determined from these derived variables. An advantage of solving the transformed equations is that there are two equations, which are fewer than the number of equations for the corresponding problem in the primitive-variable approach. By eliminating vorticity from the two equations of the vorticity-stream function approach, only one fourth-order partial differential equation of the stream function remains [2]. A disadvantage of the vorticity-stream function approach is, however, that the imposition boundary condition may be troublesome because actual boundary conditions are usually given in terms of velocity and pressure instead of the stream function.

In this paper, an alternative approach for solving the two-dimensional Navier-Stokes equations is proposed. This approach reduces the number of the Navier-Stokes equations by two without requiring the transformation of the Navier-Stokes equations into equations of derived variables like the vorticity-stream function approach. The number of equations is reduced by eliminating pressure and one of the velocity components from the governing equations. As a result, there is only one equation to be solved. Unlike the vorticity-stream function approach, imposition of boundary conditions in the proposed approach is simple because variables to be solved for are still primitive variables. In addition this approach results in a third-order partial differential equation, which should be easier to solve than the fourth-order partial differential equation of stream function from vorticity-stream function approach [2]. Therefore, this approach has advantages over both the primitive-variable approach and the vorticity-stream function approach. The following sections give details of this approach, a collocation method using this approach, and numerical results of using this method to solve a test problem with known exact solution.

2. Reduction of Equations

The two-dimensional Navier-Stokes equations consist of the momentum equations:

$$u_1 \frac{\partial u_1}{\partial x} + u_2 \frac{\partial u_1}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_1}{\partial y^2}\right) \qquad \textbf{(1)}$$

$$u_1 \frac{\partial u_2}{\partial x} + u_2 \frac{\partial u_2}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 u_2}{\partial x^2} + \frac{\partial^2 u_2}{\partial y^2}\right) \qquad \textbf{(2)}$$

and the continuity equation:

$$\frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y} = 0 \qquad \textbf{(3)}$$

where ρ is density, ν is kinematic viscosity, p is pressure, and u_1 and u_2 are x- and y-components of velocity. Elimination of p from Eqs. (1) and (2) yields

$$u_1\left(\frac{\partial^2 u_1}{\partial x \partial y} - \frac{\partial^2 u_2}{\partial x^2}\right) + u_2\left(\frac{\partial^2 u_1}{\partial y^2} - \frac{\partial^2 u_2}{\partial x \partial y}\right) - \nu\left(\frac{\partial^3 u_1}{\partial x^2 \partial y} + \frac{\partial^3 u_1}{\partial y^3} - \frac{\partial^3 u_2}{\partial x^3} - \frac{\partial^3 u_3}{\partial x \partial y^2}\right) = 0 \qquad \textbf{(4)}$$

Next, Eq. (3) is used to express u_2 in terms of u_1.

$$u_2 = -\int \frac{\partial u_1}{\partial x} dy \qquad \textbf{(5)}$$

Substituting Eq. (5) into Eq.(4) results in one equation with one unknown (u_1). However, this equation is a third-order nonlinear partial differential equation. An appropriate method for solving this equation is a collocation method.

3. Basis Functions

In order to solve Eq. (4) by a collocation method, $\partial u_1/\partial x$ must be approximated as a linear combination of independent basis functions.

$$\frac{\partial u_1}{\partial x}(x, y) = \sum_i a_i \phi(x, y, x_i, y_i) \qquad \textbf{(6)}$$

where a_i are unknown coefficients. The approximation for $\partial u_2/\partial y$ is obtained from Eq. (3).

$$\frac{\partial u_2}{\partial y}(x, y) = -\sum_i a_i \phi(x, y, x_i, y_i) \qquad \textbf{(7)}$$

In order to generate independent functions for Eqs. (6) and (7), it is convenient to use radial basis functions. One of the most popular radial basis function is the multiquadrics:

$$\phi(x, y, x_i, y_i) = \sqrt{(x - x_i)^2 + (y - y_i)^2 + c^2} \qquad \textbf{(8)}$$

where *c* is called the shape parameter. Multiquadrics is an axisymmetric function. A small value of *c* results in a cone-like function. The function is smoother as *c* increases. N independent basis functions can be generated from multiquadrics by choosing N different sets of coordinates (x_i, y_i).

Once the basis functions for the approximations of $\partial u_1/\partial x$ and $\partial u_2/\partial y$ have been selected, the approximations for u_1 and u_2 can be obtained as follows.

$$u_1(x, y) = \sum_i a_i \psi_1(x, y, x_i, y_i) \qquad \textbf{(9)}$$

The approximation for $\partial u_2/\partial x$ is obtained from Eq. (5).
where

$$u_2(x, y) = \sum_i a_i \psi_2(x, y, x_i, y_i) \qquad \textbf{(10)}$$

$$\psi_1(x, y, x_i, y_i) = \left(\frac{x - x_i}{2}\right)\sqrt{(x - x_i)^2 + (y - y_i)^2 + c^2} + \left[\frac{(y - y_i)^2 + c^2}{2}\right]\ln\left[(x - x_i) + \sqrt{(x - x_i)^2 + (y - y_i)^2 + c^2}\right] \quad \textbf{(11)}$$

$$\psi_2(x, y, x_i, y_i) = -\left(\frac{y - y_i}{2}\right)\sqrt{(x - x_i)^2 + (y - y_i)^2 + c^2} - \left[\frac{(x - x_i)^2 + c^2}{2}\right]\ln\left[(y - y_i) + \sqrt{(x - x_i)^2 + (y - y_i)^2 + c^2}\right] \quad \textbf{(12)}$$

4. Collocation Method

Collocation methods based on radial basis functions were popularized by Kansa [3]. They have been used to solve a variety of linear and nonlinear partial differential equations [4-9]. Previously, Mai-Duy and Tran-Cong [10] and Shu et al. [11] solved the two-dimensional Navier-Stokes equations in the vorticity-stream function approach by collocation methods that use radial basis functions. However, equations they solved are second-order partial differential equations, and not a third-order partial differential equation like Eq. (4). In this section, the collocation method for solving Eq. (4) is described. This collocation method is known as the multiquadric collocation method with additional collocation at the boundary. It was used by Chantasiriwan to solve linear partial differential equations [12], and found to give more accurate solutions than the standard multiquadric collocation method that was used by Kansa [3].

Assume that there are a total of N nodes, divided into N_b boundary nodes (indexed by i = 1, 2, ..., N_b) and Ni interior nodes (indexed by i = N_b + 1, N_b + 2, ..., N). Note that N = N_b + Ni. The velocity components u_1 and u_2 at the kth iteration are approximated as

$$u_1^{(k)} = \sum_{i=1}^{N} a_i^{(k)}\psi_1(x, y, x_i, y_i) + \sum_{i=1}^{N_b} a_{i+N}^{(k)}\psi_1'(x, y, x_{i+N_i}, y_{i+N_i}) \quad \textbf{(13)}$$

$$u_2^{(k)} = \sum_{i=1}^{N} a_i^{(k)}\psi_2(x, y, x_i, y_i) + \sum_{i=1}^{N_b} a_{i+N}^{(k)}\psi_2'(x, y, x_{i+N_i}, y_{i+N_i}) \quad \textbf{(14)}$$

Basis functions ψ_1' and ψ_2' must be independent from ψ_1' and ψ_2'. It is assumed that ψ_1' and ψ_2' have similar functional forms as ψ_1 and ψ_2, except for a different shape parameter (d ≠ c).

$$\psi_1'(x, y, x_i, y_i) = \left(\frac{x - x_i}{2}\right)\sqrt{(x - x_i)^2 + (y - y_i)^2 + d^2} + \left[\frac{(y - y_i)^2 + d^2}{2}\right]\ln\left[(x - x_i) + \sqrt{(x - x_i)^2 + (y - y_i)^2 + d^2}\right] \quad \textbf{(15)}$$

$$\psi_2'(x, y, x_i, y_i) = \left(\frac{y - y_i}{2}\right)\sqrt{(x - x_i)^2 + (y - y_i)^2 + d^2} - \left[\frac{(x - x_i)^2 + d^2}{2}\right]\ln\left[(y - y_i) + \sqrt{(x - x_i)^2 + (y - y_i)^2 + d^2}\right] \quad \textbf{(16)}$$

Approximations for derivatives of the velocity components can be obtained by finding derivatives of ψ_1, ψ_2, ψ_1', and ψ_2'. For the purpose of iterative determination of N + N_b unknown coefficients $a_i^{(k)}$ (i = 1, 2, ..., N + N_b), nonlinear terms in Eq. (4) are linearized by a scheme proposed by Ferziger and Peric [13]. For example,

$$u_1^{(k)}\frac{\partial^2 u_1^{(k)}}{\partial x \partial y} = u_1^{(k-1)}\frac{\partial^2 u_1^{(k)}}{\partial x \partial y} + u_1^{(k)}\frac{\partial^2 u_1^{(k-1)}}{\partial x \partial y} - u_1^{(k-1)}\frac{\partial^2 u_1^{(k-1)}}{\partial x \partial y} \quad \textbf{(17)}$$

After linearization, Eq. (4) represents N equations for (x, y) = (x_i, y_i) (i = 1, 2, ..., N). Boundary conditions for u_1 and u_2 yield 2N_b more equations. Therefore, the resulting system of equations can be solved by a least-square method because there are more equations than unknowns. Initially, let $a_i^{(o)} = 0$ (i = 1, 2, ..., N + N_b). The iteration process is continued until the following convergence criterion is satisfied:

$$\left[\frac{1}{N_t}\sum_{i=1}^{N_t}\left(1 - \frac{f_i^{(k-1)}}{f_i^{(k)}}\right)^2\right]^{1/2} < 10^{-5} \quad \textbf{(18)}$$

where *f* represents u_1 or u_2. The value of *k* that yields converged solutions within 1000 iterations is denoted by K. It is considered that no converged solutions can be found if more than 1000 iterations are required.

5. Results and Discussion

The test problem having Dirichlet boundary condition is considered. Let (ξ_i, η_i) be coordinates of a test node in the test problem. Velocity components at the test node can be calculated from Eqs. (13) and (14) after coefficients have been determined. If the exact solution is known, error can be computed from

$$\varepsilon_l = \left[\frac{\sum_{k=1}^{N_t} \left[u_l(\xi_k, \eta_k) - u_{j,exact}(\xi_k, \eta_k) \right]^2}{\sum_{i=1}^{N_t} \left[u_{l,exact}(\xi_k, \eta_k) \right]^2} \right]^{1/2} \quad \textbf{(19)}$$

where l = 1 or 2, and N_t is the number of test nodes.

The domain for the test problem is a 1 × 1 square. There are 64 test nodes located at (0.125i + 0.0625), (0.125j + 0.0625) with integers i and j running from -4 to 3. Let there be N collocation nodes uniformly distributed in the domain, forming a square grid. Fig. 1 shows distributions of test nodes and collocation nodes for N = 81. Exact solutions for this problem are

$$u_{1,exact}(x, y) = \lambda v + e^{\lambda(x+y)} \quad \textbf{(20)}$$

$$u_{2,exact}(x, y) = \lambda v - e^{\lambda(x+y)} \quad \textbf{(21)}$$

where λ is a parameter influencing the smoothness of the solutions. Solutions with a smaller absolute value of λ have a smoother form of function than solutions with a larger absolute value of λ. From these exact solutions, boundary conditions are generated, numerical solutions are determined by the proposed method, and compared with exact solutions. In the following results, it is shown how ε_1, ε_2 and K vary with the shape parameters (c and d) and other parameters.

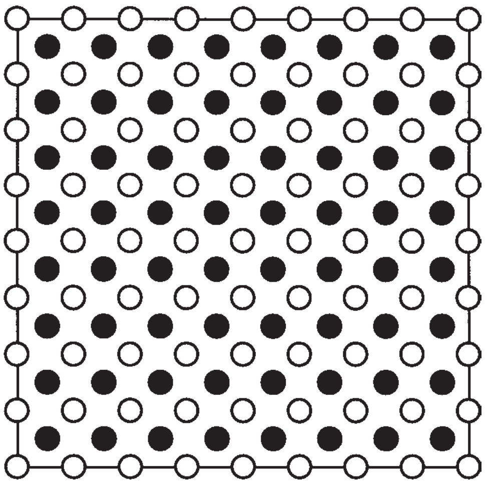
Fig. 1 Distributions of 81 collocation nodes (white circles) and 64 test nodes (black circles) in the domain of the test problem.

Fig. 2 shows influences of the shape parameters c and d on ε_1, ε_2 and K for the case in which λ = -1, ν = 1, and N = 81. It can be seen that solutions are more accurate as c is increased. When c is lower than 0.8, converged solutions are found within a few iterations. If c is larger than 0.9, however, it is found that no converged solution can be found. A large value of c is associated with high condition number of the coefficient matrix of the system of linear equations. Since the computing machine used in this study has a limited precision, an ill-conditioned system of linear equations cannot be solved with high precision. This may be a reason why the proposed method does not converge when a certain value of c is reached. In addition, Fig. 2 also shows that solutions are relatively insensitive to the shape parameter d between c + 0.1 and c + 0.2.

Fig. 3 compares ε_1, ε_2 and K of solutions for three values of N (49, 81, and 121) with $\lambda = -1$ and $\nu = 1$. The shape parameter d is varied with N so that the difference between d and c scales with grid spacing. The range of the shape parameter c in which a converged solution can be obtained is narrower as N is increased. An interesting consequence of this is that solutions with a larger number of nodes and smaller values of shape parameters may not be much more accurate than solutions with a smaller number of nodes and larger values of shape parameters. At the limits of convergence, roughly similar accuracy is obtained regardless of the number of nodes.

The proposed method is also tested with cases in which λ and ν are smaller than -1 and 1, respectively. These cases present a stiffer challenge because their exact solutions are less smooth than the exact solution for which $\lambda = -1$ and $\nu = 1$. Results of two new cases in which $(\lambda, \nu) = (-2, 0.1)$ and (-3, 0.01) are compared with results of the base case in which $(\lambda, \nu) = (-1, 1)$ in Fig. 4. It can be seen that the proposed method can solve the test problem in the two new cases. Behaviors of ε_1, ε_2 and K for two new cases are found to be qualitatively similar to those for the base case.

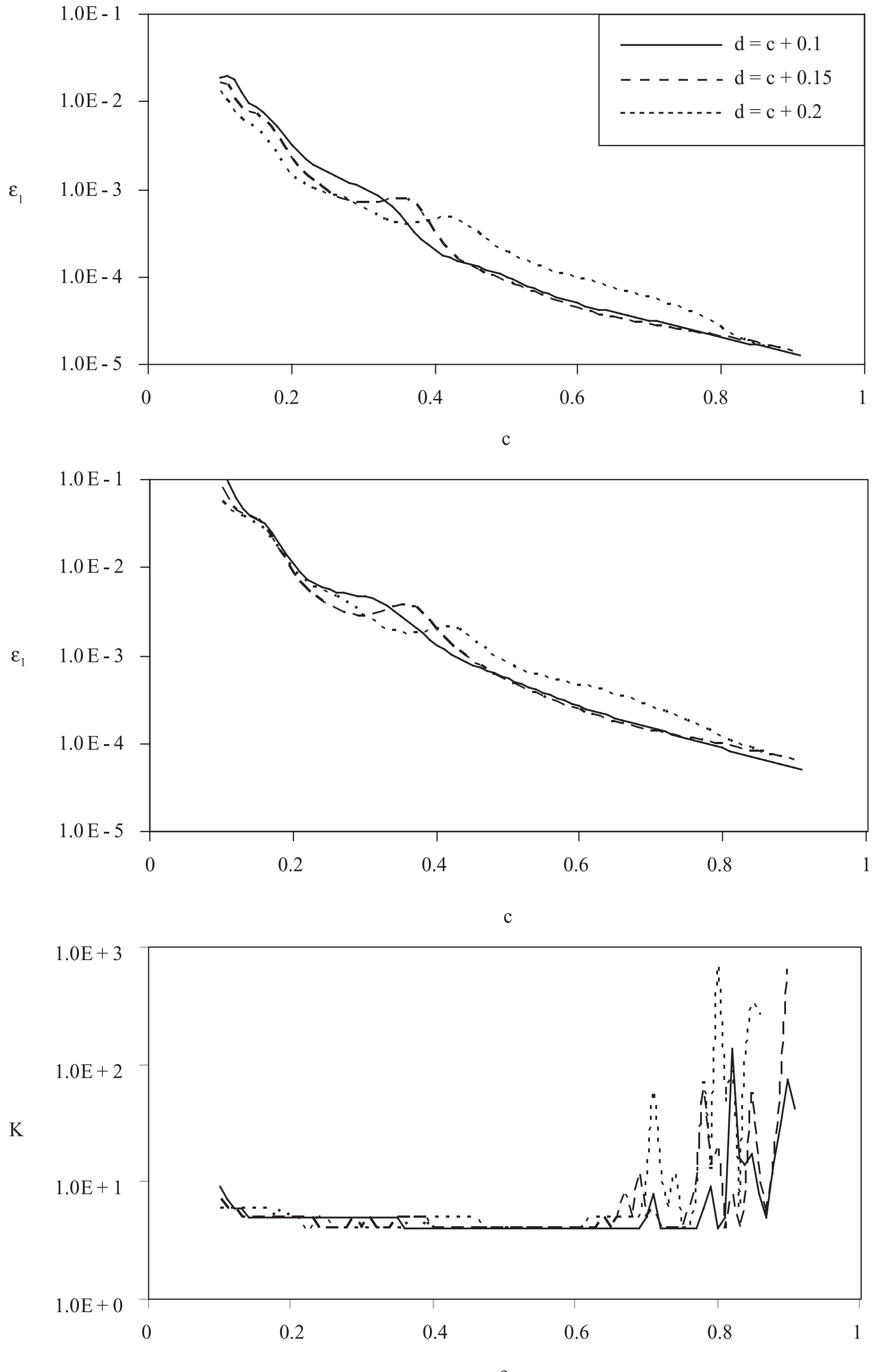

Fig. 2 Variations of errors of velocity components (ε_1 and ε_2) and the number of iterations with the shape parameters c and d for the test problem having parameters λ = -1 and ν = 1. The number of collocation nodes (N) is 81.

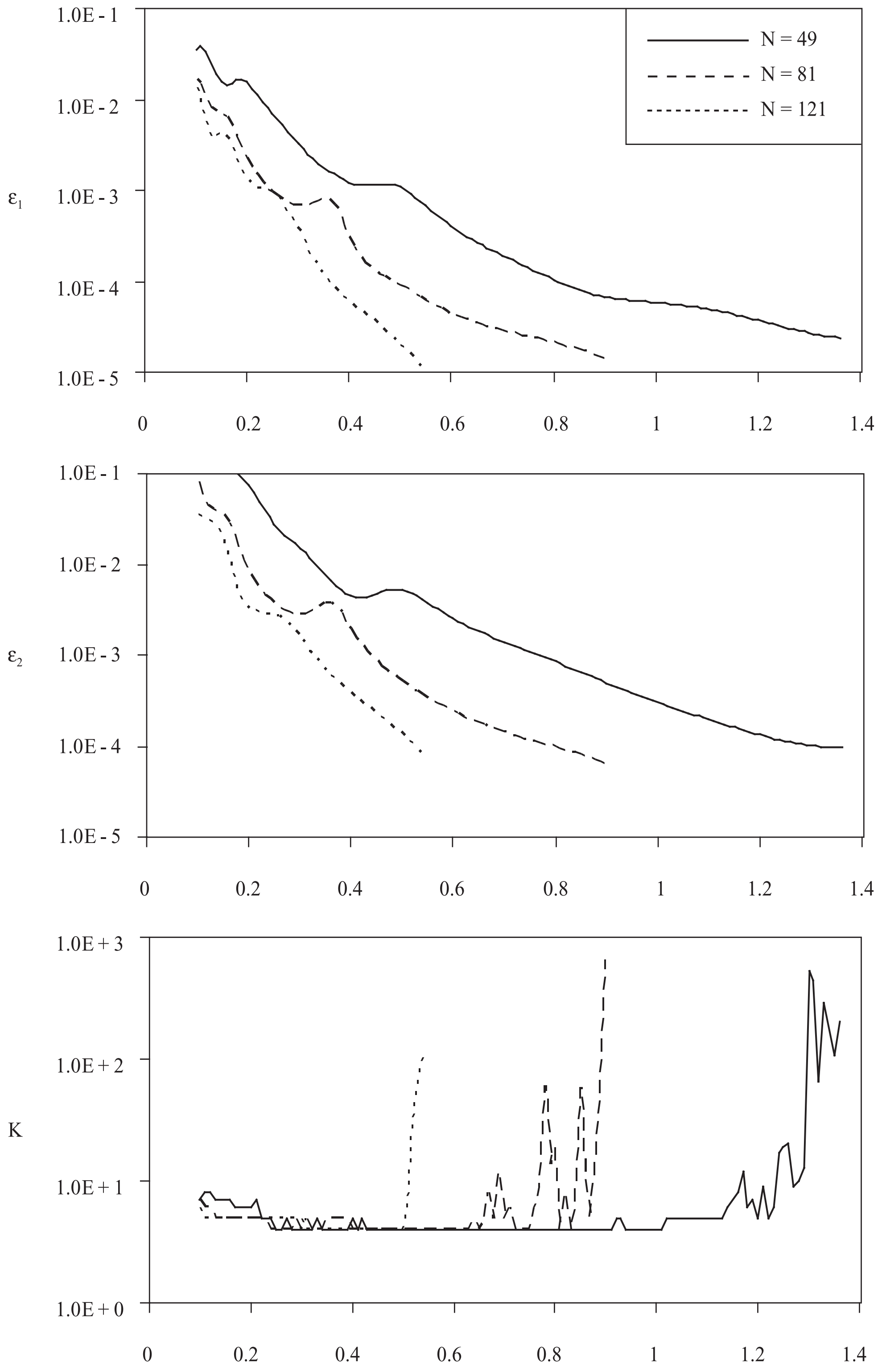

Fig. 3 Variations of errors of velocity components (ε_1 and ε_2) and the number of iterations (K) with the shape parameters c and the number of nodes (N) for the test problem having parameters $\lambda = -1$ and $\nu = 1$. The shape parameter d equals c + 0.2 for N = 49, c + 0.15 for N = 81, and c + 0.12 for N = 121.

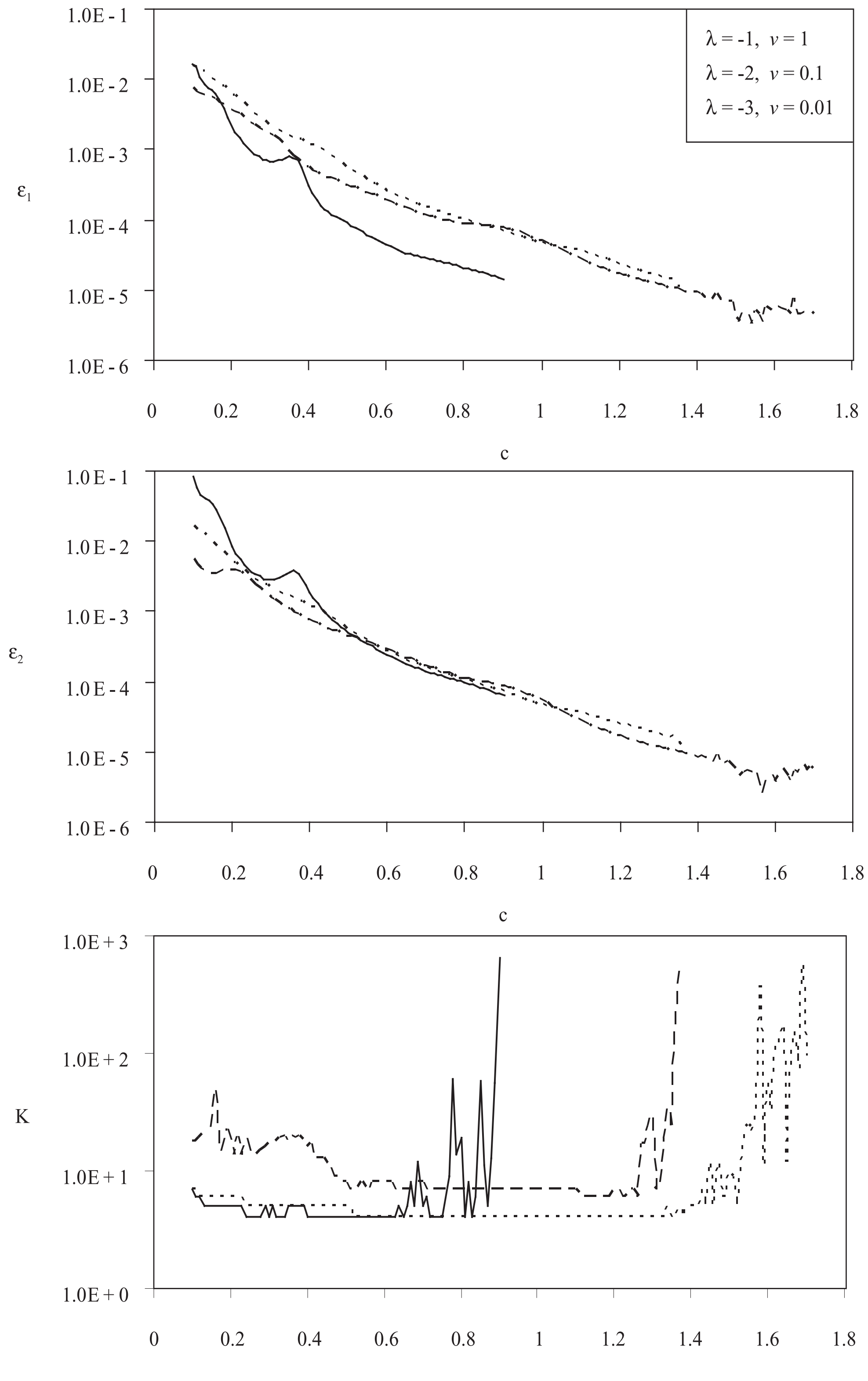

Fig. 4 Variations of errors of velocity components (ε_1 and ε_2) and the number of iterations (K) with the shape parameter c for the test problem having three sets of parameters: $(\lambda,\nu) = (-1, 1), (-2, 0.1)$, and $(-3, 0.01)$. The number of collocation nodes (N) is 81, and the shape parameter d = c + 0.15.

6. Conclusions

This paper presents an alternative approach for solving the Navier-Stokes equations, in which pressure and one of the velocity components are eliminated, and the governing equations are reduced to a third-order partial differential equation of the remaining velocity component. The resulting equations are solved by a meshless collocation method that uses multiquadrics and associated functions as basis functions. Inspection of solutions to the test problem by the proposed method reveals that shape parameters of multiquadrics influence the accuracy of solutions. Small values of the shape parameters result in a low number of required iterations, but the resulting solutions may not be accurate. On the contrary, large values of the shape parameters can yield very accurate solutions provided that converged solutions can be found. Furthermore, converged solutions may not be obtained if the shape parameters are too large.

7. Acknowledgement

This research is financially supported by the Thailand Research Fund.

8. References

1. Tanhill, J.C. Anderson, D.A., and Pletcher, R.H., 1997, *Computational Fluid Mechanics and Heat Transfer (2nd edn.)*, Taylor and Francis, Wahsington DC.

2. Polyanin, A.D. and Zaitsev, V.F., 2003, *Handbook of Nonlinear Partial Differential Equations*, Chapman and Hall/CRC, Boca Raton.

3. Kansa, E.J., 1990, "Multiquadrics - A Scattered Data Approximation Scheme with Applications to Computational Fluid Dynamics - II Solutions to Parabolic, Hyperbolic and Elliptic Partial Differential equations," *Computers and Mathematics with Applications*, Vol. 19, pp. 147-161.

4. Zerroukat, M., Power, H., and Chen, C.S., 1998, "A Numerical Method for Heat Transfer Problems Using Collocation and Radial Basis Functions," *International Journal for Numerical Methods in Engineering*, Vol. 42, pp. 1263-1278.

5. Hon, Y.C. and Mao, X.Z., 1998, "An Efficient Numerical Scheme for Burger's Equation,é *Applied Mathematics and Computation*, Vol. 95, pp. 37-50.

6. Fedoseyev, A.I., Friedman, M.J., and Kansa, E.J., 2000, "Continuation for Nonlinear Elliptic Partial Differential Equations Discretized by the Multiquadric Method," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, Vol. 10, pp. 481-492.

7. Li, J. and Chen, C.S., 2003, "Some Observations on Unsymmetric Radial Basis Function Collocation Methods for Convection-Diffusion Problems," *International Journal for Numerical Methods in Engineering*, Vol. 57, pp. 1085-1094.

8. Zhoua, X., Hon, Y.C., and Cheung, K.F., 2004, "A Grid-free, Nonlinear Shallow-water Model with Moving Boundary," *Engineering Analysis with Boundary Elements*, Vol. 28, pp. 967-973.

9. Young, D.L., Jane, S.C., Lin, C.Y., Chiu, C.L., and Chen, K.C., 2004, "Solutions of 2D and 3D Stokes Laws Using Multiquadrics Method," *Engineering Analysis with Boundary Elements*, Vol. 28, pp. 1233-1243.

10. Mai-Duy, M. and Tran-Cong, T., 2001, "Numerical Solution of Navier-Stokes Equations using Multiquadric Radial Basis Function Networks," *International Journal for Numerical Methods in Fluids*, Vol. 37, pp. 65-86.

11. Shu, C., Ding, H., and Yeo, K.S., 2005, "Computation of Incompressible Navier-Stokes Equations by Local RBF-based Differential

Quadrature Method," *CMES - Computer Modeling in Engineering and Sciences*, Vol. 7, pp. 195-205.

12. Chantasiriwan, S., 2004, "Cartesian Grid Methods Using Radial Basis Functions for Solving Poisson, Helmholtz, and Diffusion-Convection Equations," *Engineering Analysis with Boundary Elements*, Vol. 28, pp. 1417-1425.

13. Ferziger, J.H. and Milovan, P., 1997, *Computational Methods for Fluid Dynamics*, Springer, Berlin.