INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 15, 791–798 (1992)

APPLICATION OF GENERALIZED DIFFERENTIAL QUADRATURE TO SOLVE TWO-DIMENSIONAL INCOMPRESSIBLE NAVIER–STOKES EQUATIONS

CHANG SHU AND BRYAN E. RICHARDS

Department of Aerospace Engineering, University of Glasgow, Glasgow G12 8QQ, Scotland, U.K.

SUMMARY

A global method of generalized differential quadrature is applied to solve the two-dimensional incompressible Navier–Stokes equations in the vorticity–stream-function formulation. Numerical results for the flow past a circular cylinder were obtained using just a few grid points. A good agreement is found with the experimental data.

KEY WORDS Generalized differential quadrature Incompressible flows Navier–Stokes equations

INTRODUCTION

Most engineering problems can be currently simulated by finite difference and finite element methods. Usually, these methods require a large number of grid points for accurate results. More recently, spectral and pseudospectral methods have provided attractive techniques for the solution of smooth engineering problems, using only a few grid points. Amongst the family of these methods, the Chebyshev pseudospectral method is commonly used. This method usually requires a transformation between the physical space and the computational space since Chebyshev collocation points lie in the domain $[1, -1]$, leading to some inconveniences in use. In seeking a more efficient numerical method, the present authors have developed a method, based on the work of Bellman *et al.*,[1] of generalized differential quadrature (GDQ), which can be considered as a global method. GDQ has overcome the difficulty of differential quadrature (DQ) in obtaining the weighting coefficients for the first-order derivative discretization with arbitrary distribution of grid points, and is easier to apply than spectral methods. It is shown in Reference 2 that GDQ can be considered as the highest-order finite difference scheme, and both GDQ and Chebyshev pseudospectral method provide exactly the same weighting coefficients of the first-order derivative when the co-ordinates of grid points are chosen as the roots of a Chebyshev polynominal. This demonstrates that GDQ may have a considerable scope for development since it can be used with arbitrary distributions of grid points. In GDQ, the weighting coefficients of the first-order derivative are given by a simple algebraic formulation, and the weighting coefficients of the second- and higher-order derivative are determined by a recurrence relation. Some basic features of GDQ, such as the error estimations of the derivatives approximation and the influence of distribution of grid points on the stability, have also been analysed in Reference 2. The successful application of GDQ for the solution of a partial differential equation has been shown in

0271-2091/92/190791–08$09.00

Received March 1990
Revised April 1992

References 2 and 3 using considerably few grid points. It will be demonstrated in this paper that GDQ has potential as an attractive tool in CFD, especially in incompressible flow simulations.

GENERALIZED DIFFERENTIAL QUADRATURE

The differential quadrature technique, first presented by Bellman *et al.*,[1] approximates the partial derivatives of a function with respect to a space variable at a given discrete point as a weighted linear sum of all the functional values at all discrete points in the overall domain of that variable. Obviously, the key to this technique is: How to determine the weighting coefficients of any order partial derivative? For the weighting coefficients of the first-order derivative, Bellman *et al.* suggested two ways to carry this out. One solves a set of algebraic equations. Unfortunately, when the number of grid points is large, the matrix of this algebraic equation system is ill-conditioned and its inversion is difficult. This is probably one of the reasons that applications of this scheme so far use only 13 or lesss grid points. The other computes the weighting coefficients by a simple algebraic formulation, but with the co-ordinates of grid points chosen as the roots of a shifted Legendre polynomial. This means that if the number of grid points is specified, the distributions of grid points are the same for different physical problems. This may provide a major drawback and restrict the application of DQ. In order to overcome the drawbacks described above, the technique of generalized differential quadrature was then developed, based on the analysis of a polynomial vector space.

According to Weierstrass polynomial approximation theorem, a continuous function in a domain can be approximated by an infinite polynomial accurately. In practice, a truncated finite polynomial may be used. Some methods, an example being the spectral method, have successfully applied the concept of the high-order polynomial approximation to the solution of a partial differential equation. Following this approach, it is supposed that any smooth function in a domain can be approximated by an $(N-1)$th-order polynomial. It is easy to show that a polynomial of degree less than or equal to $N-1$ constitutes an N-dimensional polynomial vector space V_N with respect to the operation of addition and multiplication. Based on the analysis of a linear polynomial vector space, it has been shown that when the base polynomials are chosen to be x^k, $k=0, 1, \ldots, N-1$, or the co-ordinates of grid points are chosen as the roots of a Legendre polynomial, GDQ provides exactly the same results as DQ. For generality, the Lagrange interpolation polynomials are chosen as the base polynomials, which result in a simple algebraic formulation for calculating the weighting coefficients of the first-order derivative, without any restriction on the choice of grid points. Furthermore, a recurrence relation was obtained for the determination of the weighting coefficients of the second- and higher-order derivatives. For the multi-dimensional case, each direction can be treated as in the one-dimensional case. It has been proved that GDQ can be considered as the highest-order finite difference scheme. Some basic features such as the error estimations, stability and convergence have also been analysed. For details, see Reference 2. Here, for brevity, only the results for the two-dimensional case are given. It is supposed that there are N grid points in the x-direction, $x_1, \ldots, x_N$, and M grid points in the y-direction, $y_1, \ldots, y_M$. Then the nth-order partial derivative of $f(x, y)$ with respect to x and the mth-order partial derivative of $f(x, y)$ with respect to y can be discretized at x_i, y_j as

$$f_x^{(n)}(x_i, y_j) = \sum_{k=1}^{N} W_{ik}^{(n)} f(x_k, y_j), \quad n=1, \ldots, N-1, \tag{1a}$$

$$f_y^{(m)}(x_i, y_j) = \sum_{k=1}^{M} \bar{W}_{jk}^{(m)} f(x_i, y_k), \quad m=1, \ldots, M-1, \tag{1b}$$

$$\text{for } i=1, 2, \ldots, N, \quad j=1, 2, \ldots, M$$

where $w_{ij}^{(n)}$, $\bar{w}_{ij}^{(m)}$ are the weighting coefficients to be determined as follows:

$$w_{ij}^{(1)}=\frac{M^{(1)}(x_i)}{(x_i-x_j)\,M^{(1)}(x_j)}, \qquad i,j=1,2,\ldots,N, \text{ but } j\neq i, \tag{2a}$$

$$\bar{w}_{ij}^{(1)}=\frac{P^{(1)}(y_i)}{(y_i-y_j)\,P^{(1)}(y_j)}, \qquad i,j=1,2,\ldots,M, \text{ but } j\neq i, \tag{2b}$$

where

$$M^{(1)}(x_i)=\prod_{j=1,j\neq i}^{N}(x_i-x_j),$$

$$P^{(1)}(y_i)=\prod_{j=1,j\neq i}^{M}(y_i-y_j),$$

$$w_{ij}^{(n)}=n\left(w_{ii}^{(n-1)}\,w_{ij}^{(1)}-\frac{w_{ij}^{(n-1)}}{x_i-x_j}\right), \tag{3a}$$

$$\text{for } i,j=1,2,\ldots,N, \text{ but } j\neq i, \quad n=2,3,\ldots,N-1,$$

$$\bar{w}_{ij}^{(m)}=m\left(\bar{w}_{ii}^{(m-1)}\,\bar{w}_{ij}^{(1)}-\frac{\bar{w}_{ij}^{(m-1)}}{y_i-y_j}\right), \tag{3b}$$

$$\text{for } i,j=1,2\ldots,M, \quad \text{but } j\neq i; \quad m=2,3,\ldots,M-1$$

$$w_{ii}^{(n)}=-\sum_{j=1,j\neq i}^{N}w_{ij}^{(n)}, \quad i=1,2,\ldots,N, \quad n=1,2,\ldots,N-1, \tag{4a}$$

$$\bar{w}_{ii}^{(m)}=-\sum_{j=1,j\neq i}^{M}\bar{w}_{ij}^{(m)}, \quad i=1,2,\ldots,M, \quad m=1,2,\ldots,M-1. \tag{4b}$$

It is clear from formulation (3) that the weighting coefficients of the second- and higher-order derivatives can be calculated from those of the first-order derivative completely.

When the uniform grid is used, formulation (2) is reduced to

$$w_{ij}^{(1)}=(-1)^{i+j}\frac{(i-1)!(N-i)!}{\Delta x(i-j)\;(j-1)!(N-j)!} \tag{5a}$$

$$\text{for } i,j=1,2,\ldots,N, \text{ but } j\neq i,$$

$$\bar{w}_{ij}^{(1)}=(-1)^{i+j}\frac{(i-1)!(M-i)!}{\Delta y(i-j)\;(j-1)!(M-j)!}, \tag{5b}$$

$$\text{for } i,j=1,2,\ldots,M, \text{ but } j\neq i,$$

where $\Delta x=x_i-x_{i-1}$, $\Delta y=y_i-y_{i-1}$, and when x_i, which is in the domain $[1,-1]$, is chosen to be

$$x_i=\cos(i\pi/N), \qquad i=0,\ldots,N,$$

formulation (2) is reduced to

$$w_{ij}^{(1)}=\frac{(-1)^{i+j}\;\bar{c}_i}{\bar{c}_j\,(x_i-x_j)}, \tag{6}$$

where

$$\bar{c}_i = \begin{cases} 2 & \text{when } i=0,\ N, \\ 1 & \text{otherwise,} \end{cases}$$

which is exactly the same as that given from the pseudospectral Chebyshev method.[4]

Finally, when the functional values at all grid points are obtained, it is easy to determine the functional values in the overall domain in terms of the polynomial approximation, i.e.

$$f(x, y_j) = \sum_{i=1}^{N} f(x_i, y_j)\, r_i(x), \tag{7}$$

$$f(x_i, y) = \sum_{j=1}^{M} f(x_i, y)_j)\, s_j(y), \tag{8}$$

$$f(x, y) = \sum_{i=1}^{N} \sum_{j=1}^{M} f(x_i, y_j)\, r_i(x)\, s_j(y), \tag{9}$$

where $r_i(x)$, $s_j(y)$ are the Lagrange interpolation polynomials along the x- and y-direction, respectively.

FLOW PAST A CIRCULAR CYLINDER

The two-dimensional Navier–Stokes equations are used to simulate this problem: The version of vorticity–streamfunction formulation is written as

$$\omega_t + u\omega_x + v\omega_y = (\omega_{xx} + \omega_{yy})/Re, \tag{10}$$

$$\psi_{xx} + \psi_{yy} = \omega, \tag{11}$$

with $u = \psi_y$ being the horizontal velocity component, $v = -\psi_x$ the vertical velocity component $\omega = u_y - v_x$ the vorticity and Re the Reynolds number (based on the radius of the cylinder and the free-stream velocity V_∞). In this notation, the subscripts x and y denote the derivative in the indicated direction.

In numerical simulation, the physical domain can be mapped into the computational domain by the following transformation:

$$x = e^{\eta} \cos \xi, \quad y = e^{\eta} \sin \xi, \tag{12}$$

where the function e^{η} assures an appropriately clustered grid point distribution close to the cylinder surface. Using (12), equation (10) and (11) can be transformed to

$$e^{2\eta}\omega_t + \psi_\xi \omega_\eta - \psi_\eta \omega_\xi = (\omega_{\xi\xi} + \omega_{\eta\eta})/Re, \tag{13}$$

$$\psi_{\xi\xi} + \psi_{\eta\eta} = e^{2\eta}\omega. \tag{14}$$

To avoid having to deal with the large values of ψ occurring in the far field and also to facilitate the numerical implementation of the far-field boundary conditions, the streamfunction ψ is decomposed into two parts such that

$$\psi = \psi_{in} + \Psi,$$

where ψ_{in} is chosen as the value of the inviscid flow, i.e.

$$\psi_{in} = (e^{\eta} - e^{-\eta}) \sin \xi.$$

Thus, equations (13) and (14) can be reduced to

$$e^{2\eta}\omega_t+(\Psi_\xi+v_1)\omega_\eta-(\Psi_\eta+u_1)\omega_\xi=(\omega_{\xi\xi}+\omega_{\eta\eta})/Re, \tag{15}$$

$$\Psi_{\xi\xi}+\Psi_{\eta\eta}=e^{2\eta}\omega, \tag{16}$$

with

$$v_1=(e^{\eta}-e^{-\eta})\cos\xi$$

$$u_1=(e^{\eta}+e^{-\eta})\sin\xi.$$

On the surface of the cylinder, the no-slip boundary condition gives

$$\left.\begin{aligned}&\Psi=0, \quad \Psi_\eta=-2\sin\xi\\ &\omega=\Psi_{\eta\eta}\end{aligned}\right\} \quad \text{at } \eta=0,$$

and at infinity, the following boundary conditions are used:

$$\left.\begin{aligned}&\Psi=0, \quad \Psi_\eta=0\\ &\omega=e^{-2\eta}\Psi_{\eta\eta}\end{aligned}\right\} \quad \text{at } \eta=\infty.$$

On the branch cut (from the rear point of the cylinder to the outer boundary), the periodic boundary condition is imposed. It is noted that there are two boundary conditions for Ψ and one boundary condition for ω at each solid and infinity boundaries, and there is only one boundary condition for both ω and Ψ at the branch cut line. For numerical simulation, the unbounded far-field boundary is truncated at a finite distance, η_{max}, which is far enough from the cylinder to allow the far-field boundary conditions to be satisfied accurately and is chosen here as 3·0.

Using GDQ described above, all the spatial derivatives can be discretized as

$$(\Psi_\xi)_{ij}=\sum_{k=1}^{N} w_{ik}^{(1)}\Psi_{kj}, \qquad (\Psi_{\xi\xi})_{ij}=\sum_{k=1}^{N} w_{ik}^{(2)}\Psi_{kj},$$

$$(\Psi_\eta)_{ij}=\sum_{k=1}^{M} \bar{w}_{jk}^{(1)}\Psi_{ik}, \qquad (\Psi_{\eta\eta})_{ij}=\sum_{k=1}^{M} \bar{w}_{jk}^{(2)}\Psi_{ik},$$

$$(\omega_\xi)_{ij}=\sum_{k=1}^{N} w_{ik}^{(1)}\omega_{kj}, \qquad (\omega_{\xi\xi})_{ij}=\sum_{k=1}^{N} w_{ik}^{(2)}\omega_{kj},$$

$$(\omega_\eta)_{ij}=\sum_{k=1}^{M} \bar{w}_{jk}^{(1)}\omega_{ik}, \qquad (\omega_{\eta\eta})_{ij}=\sum_{k=1}^{M} \bar{w}_{jk}^{(2)}\omega_{ik}.$$

Substituting the above expressions into equations (15) and (16) yields a set of ordinary differential equations in time for ω and a set of algebraic equations for Ψ. Similarly, the Neumann boundary conditions can also be discretized by GDQ. With N grid points in the ξ-direction and M grid points in the η-direction, after implementing all the boundary conditions for ω and Ψ, the resultant set of $(N-2)\times(M-2)$ ordinary differential equations for ω are then solved by the four-stage Runge–Kutta scheme, and the set of $(N-2)\times(M-4)$ algebraic equations for Ψ are solved by a direct method of LU decomposition.

For steady-state solution of this problem, the most sensitive parameter to check the accuracy of numerical simulation is the calculation of the parameters defining the structure of the wake behind the cylinder. The cylinder and the geometrical parameters of the closed wake are shown in Figure 1. Accurate simulations of the flow past a circular cylinder have demonstrated sensitivity to the boundary conditions imposed. The key factors may be the implementation of reasonable conditions at the far-field boundary and the boundary conditions at the surface of the cylinder. In

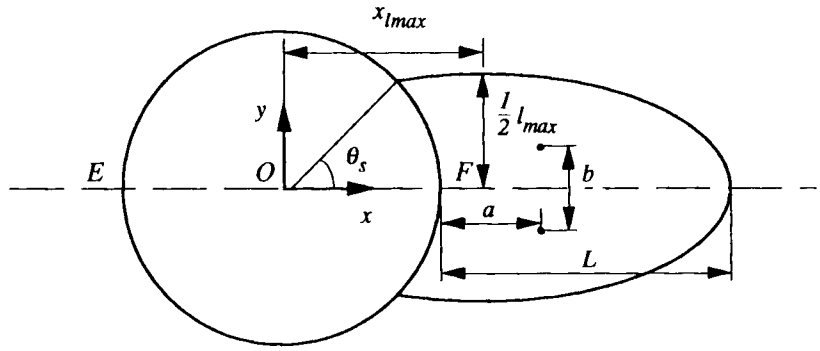

Figure 1. Geometrical parameters of the closed wake behind a circular cylinder

the present computation, all the derivatives included in the boundary conditions for ω and Ψ at the surface of the cylinder were treated by GDQ with $(M-1)$th-order accuracy. At the outer boundary, the inviscid flow ($u=1$, $v=0$) was assumed to provide two boundary conditions for Ψ, where the Neumann boundary condition was treated with $(M-1)$th-order accuracy, and the boundary condition for ω was examined for two cases: in the first case, the outer boundary is assumed to be in the inviscid region, which yields $\omega=0$; in the other case, ω is computed from $\omega=e^{-2\eta}\Psi_{\eta\eta}$, which is discretized by GDQ with $(M-2)$th-order accuracy. Numerical results for Re of 20 and 25 showed that both cases demonstrate nearly the same solutions. This further demonstrates that the outer boundary is in the inviscid region for these low Reynolds numbers. For the steady-state solution, the treatment of the boundary condition along the cut line was examined using two methods. One is to use the symmetric boundary conditions, namely, $\Psi=0$, $\omega=0$; the other is to use the patching technique, which enforces ω, Ψ and their first-order derivatives with respect to the normal direction of the cut line to be continuous. Numerical experiment showed that both cases achieve nearly the same results but require different time steps for satisfying the given convergent criterion. Recommended is the use of $\omega=0$, $\Psi=0$ at the cut lines since this requires less time steps, without losing accuracy. For the present computation, the outer boundary condition was set to the value of inviscid flow, the boundary condition on the cut line was set to $\omega=0$, $\Psi=0$, and the mesh size used is 25×21. Numerical results were obtained within 1 min CPU time on the IBM 3090 for each Reynolds number. Figure 2 shows the streamlines for $Re=25$, the values of streamlines being $\pm3{\cdot}0$, $\pm2{\cdot}0$, $\pm1{\cdot}0$, $\pm0{\cdot}5$, $\pm0{\cdot}15$, $\pm5{\cdot}0\times10^{-3}$, $\pm5{\cdot}0\times10^{-4}$, $\pm1{\cdot}0\times10^{-4}$, 0·0. The symmetric eddy pair is clearly shown in this figure. Table I gives the details of the parameters of the wake eddy pair. Also included in Table I are the experimental data,[5,6] and other numerical results.[7,8] It is seen from Table I that the present results are closer to the experimental data than those of Gresho *et al.*, although these authors put the outer boundary further away from the surface of the cylinder than in the present work and use a larger number of grid points. The present results are, thus, more accurate than other numerical results, even though the outer boundary was closer to the cylinder surface and fewer grid points were used. It is seen that, on the one hand, GDQ appears to be a robust, efficient numerical technique; on the other hand, the treatment of the boundary condition on the surface of the cylinder may be critically important in numerical simulation. The major difference between the present approach and other numerical techniques is the treatment of the Neumann boundary conditions, with high-order accuracy in the present approach and low-order accuracy in other approaches.

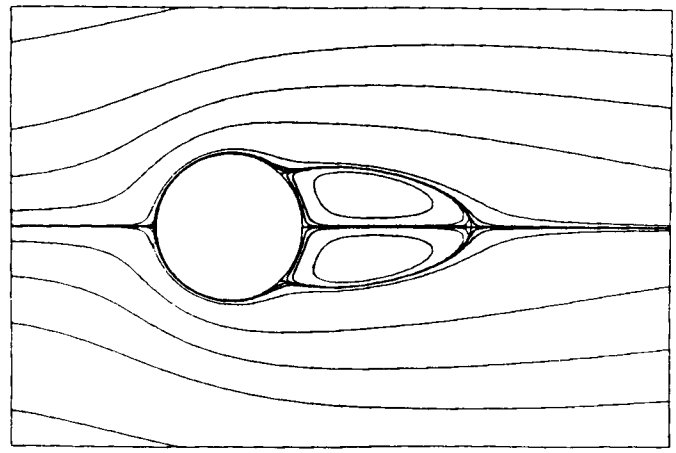

Figure 2. Streamlines past a circular cylinder, $Re=25$

Table I. Geometric parameters of the closed wake behind a cylinder

Re	References	L	a	b	x_{lmax}	l_{max}	θ_s	C_D
20	Experimental*	0·93	0·33	0·47	0·66	0·80	44·8°	2·1243
	Present	0·92	0·352	0·41	0·68	0·74	43.7°	2·1220
	Dennis *et al.*[8]	0·94					43.7°	2.0450
25	Experimental	1·21	0.44	0.51	0.75	0.85	48°	1.8176
	Present	1.21	0·424	0·475	0·73	0·82	46·6°	1·8336
	Gresho *et al.*[7]	1·15	0·38	0·47	0·67	0·81	45°	2·2600

* The drag coefficient C_D is from Reference 5; other parameters are from Reference 6 with $\lambda=0$, where λ is the ratio between the cylinder and the tank diameters.

CONCLUSIONS

The global method of generalized differential quadrature was applied to simulate the incompressible steady flow past a circular cylinder. It has been demonstrated that accurate numerical results can be obtained by GDQ using considerably few grid points, and require much less storage and computational effort compared to the conventional low-order finite difference and finite element methods, in which a large number of grid points is usually used. For the simulation of the flow past a circular cylinder, two boundary conditions, which are from the two components of the velocity, are applied for the stream function at each of the solid and outer boundaries. Since all the spatial derivatives included in both the governing equations and the boundary conditions were discretized by GDQ with high-order accuracy, numerical results are very accurate compared to the experimental data. It is expected that GDQ may have extensive applications in CFD, especially in the incompressible flow simulations.

REFERENCES

1. R. Bellman, B. G. Kashef and J. Casti, 'Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations', *J.Comput. Phys.*, **10**, 40–52 (1972).

2. C. Shu, 'Generalized differential–integral quadrature and application to the simulation of incompressible viscous flows including parallel computation', *Ph.D. Thesis*, University of Glasgow, 1991.
3. C. Shu and B. E. Richards, 'High resolution of natural convection in a square cavity by genralized differential quadrature', in *Proc. 3rd int. Conf. on Advances in Numer. Methods in Engineering: Theory and Applications, Vol. II*, Swansea, U.K., 1990, pp. 978–985.
4. U. Ehrenstein and R. Peyret, 'A Chebyshev collocation method for the Navier–Stokes equations with application to double-diffusive convection', *Int. j. numer. methods fluids*, **9**, 427–452 (1989).
5. D. J. Tritton, 'Experiments on the flow past a circular cylinder at low Reynolds numbers', *J. Fluid Mech.*, **6**(4), 547–567 (1959).
6. M. Coutanceau and R. Bouard, 'Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation, part I: steady flow', *J. Fluid Mech.*, **79**, 231–256 (1977).
7. P. M. Gresho, S. T. Chan, R. L. Lee and C. D. Upson, 'A modified finite element method for solving the time-dependent, incompressible Navier–Stokes equations, part II: applications', *Int. j. numer. methods fluids*, **4**, 619–640 (1984).
8. S. C. R. Dennis and G. Chang, 'Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100', *J. Fluid Mech.*, **42**, 471–489 (1970).