

## GDQ METHOD FOR NATURAL CONVECTION IN A SQUARE CAVITY USING VELOCITY–VORTICITY FORMULATION

D. C. Lo, D. L. Young, and K. Murugesan

*Department of Civil Engineering and Hydrotech Research Institute,  
National Taiwan University, Taipei, Taiwan, Republic of China*

*This article describes a compact numerical algorithm based on the generalized differential quadrature (GDQ) method for the numerical analysis of natural convection in a differentially heated square cavity. The velocity–vorticity form of the Navier–Stokes equations and energy equation are used to represent the mass, momentum, and energy conservations of the fluid medium in the cavity. The GDQ form of the governing equations and the vorticity definition at the boundaries are solved by a coupled solution algorithm using a global matrix scheme for all the field variables. The vorticity values at the boundary are correctly imposed using the GDQ method, which approximates a given space derivative with higher-order accuracy compared to the existing schemes based on Taylor’s series expansion. This has assured a divergence-free solution for the flow field by satisfying the continuity constraint, though the pressure term is not used directly in the present formulation. The proposed algorithm is validated for a lid-driven cavity flow for Reynolds number  $Re = 100, 400,$  and  $1,000,$  and the predicted velocity profiles are in excellent agreement with the benchmark solutions. The algorithm is then used to compute the average Nusselt number and flow parameters for natural convection in a square cavity for Rayleigh number  $Ra = 10^3, 10^4, 10^5,$  and  $10^6.$  These results are in better agreement with the benchmark solutions than the results obtained by other numerical schemes, which used much finer grids compared to the present scheme.*

### 1. INTRODUCTION

In the governing equations used to represent natural-convection problems, the fluid momentum conservation equations are coupled to the energy equation through the buoyancy term. In order to compute the temperature field correctly, the fluid momentum conservation equations have to be satisfied exactly, because it is the moving parcel of the fluid which carries the energy from one region to another region. Generally, the Navier–Stokes equations in velocity–pressure form are employed to represent the flow field. Gresho and Sani [1] pointed out that the pressure appearing in the incompressible Navier–Stokes equations is not a

Received 21 November 2003; accepted 2 July 2004.

We acknowledge financial support from the National Science Council, Taiwan, and it is greatly appreciated.

Address correspondence to D. L. Young, Department of Civil Engineering and Hydrotech Research Institute, National Taiwan University, Taipei, Taiwan, Republic of China. E-mail: dlyoung@hy.ntu.edu.tw

## NOMENCLATURE

$g$	acceleration due to gravity	$u_{\max}$	maximum horizontal velocity
$L$	length of square cavity	$v_{\max}$	maximum vertical velocity
$Nu_0$	Nusselt number on the vertical boundary at $x = 0$	$x, y$	dimensionless Cartesian coordinates
$\overline{Nu_0}$	average Nusselt number throughout the cavity	$\alpha$	thermal diffusivity
Pr	Prandtl number	$\beta$	thermal expansion coefficient
Ra	Rayleigh number	$\Gamma$	boundary of the computational domain
Re	Reynolds number	$\mu$	dynamic viscosity
$T$	dimensionless temperature	$\nu$	kinematic viscosity
$\Delta T$	temperature difference	$\rho$	mass density
$u, v$	dimensionless velocities in the $x$ and $y$ directions	$\omega$	dimensionless vorticity in the $z$ direction
		$\Omega$	computational domain

thermodynamic variable, but it has to be handled in such a way that the continuity constraint is satisfied because the pressure is not an explicit variable in the continuity equation. Without using the pressure term, the fluid momentum conservation equations can be expressed either in vorticity–stream function form or in velocity–vorticity form. Though there is no need to specify the pressure boundary conditions in the above two formulations, the continuity constraint can be satisfied only when the vorticity definition are properly imposed on the boundary. Napolitano et al. [2] presented a review of various methods for computing the vorticity boundary conditions in the vorticity–stream function formulation. Though this formulation is free from the pressure term and requires less computational effort, it is very difficult to extend this approach for solving three-dimensional Navier–Stokes equations.

The velocity–vorticity form of the Navier–Stokes equations pioneered by Fasel [3] has been found to be an effective alternative formulation to represent the Navier–Stokes equations for the solution of both two- and three-dimensional flow problems [4, 5] without involving the pressure term. An impressive property of this formulation is that its numerical formulation is independent of whether or not the reference frame is inertial. Since the vorticity is one of the field variables, this formulation can be used to study vortex-dominated flows. The main disadvantage of this formulation is that the number of field variables increases to six for the case of three-dimensional Navier–Stokes equations. Numerical solution of the velocity–vorticity equations have been attempted using the finite-difference method with a staggered grid [4] and also by the finite-element method [6, 7]. Guevremont et al. [7] employed quadratic elements for the velocity components and linear elements for the vorticity to solve three-dimensional velocity–vorticity equations using the finite-element method by a coupled numerical algorithm. Though a coupled solution scheme was employed, they used the Newton method to linearize the governing equations. Lo and Young [8] established a time-accurate computational fluid dynamics (CFD) finite-element algorithm for two-dimensional incompressible Navier–Stokes equations by velocity–vorticity formulation and obtained numerical results for Reynolds number  $Re = 100$  to 20,000 for a lid-driven cavity flow. They extended their work to study two-dimensional free surface flows at higher Reynolds numbers [9].

In the velocity–vorticity formulation the incompressibility constraint has to be satisfied by enforcing the definition of vorticity aptly at the boundaries of a computational domain. This demands the accurate computation of vorticity values at the boundaries in order to obtain a divergence-free flow field. Generally, the vorticity boundary values are computed externally and enforced at the boundaries before the solution of the vorticity transport equations. Wong and Baker [10] used a second-order-accurate Taylor’s series expansion scheme to compute the vorticity boundary values in their study on three-dimensional flow problems using the finite-element method. Based on our previous work [8, 9] on the velocity–vorticity formulation, we found the generalized differential quadrature (GDQ) method to be an efficient scheme to enforce the vorticity definition suitably at the boundaries, because the GDQ method uses higher-order polynomials with weighting functions to approximate the partial space derivatives of a function. Since the vorticity is defined in terms of velocity gradients, the GDQ method can be used to compute the vorticity values at the boundary more accurately compared to other schemes. The differential quadrature (DQ) method developed by Bellman et al. [11, 12] has been improved [13] and well established. Shu and co-workers [14–16] applied the GDQ method to simulate two- and three-dimensional incompressible viscous flows using vorticity–stream function and primitive-variable forms of the Navier–Stokes equations.

In the present work, we propose a new numerical algorithm based on the GDQ method for solving two-dimensional Navier–Stokes equations in velocity–vorticity form for the study of natural convection in a square cavity. The use of the GDQ method assures the accurate computation of the vorticity boundary values, so a divergence-free solution is guaranteed. The governing equations in the form of a vorticity transport equation, velocity Poisson equations, and an energy equation are approximated using the GDQ method. The resulting coefficient matrices and the load vectors of all the field variables are combined to form a global matrix scheme, resulting in a coupled solution algorithm. The vorticity definition expressed in terms of velocity gradients is approximated by the GDQ method to evaluate the vorticity values at the boundaries. The resulting coefficient matrices of velocities for the vorticity definition are incorporated into the global matrix at the boundary points. This procedure results in an implicit way of computing the vorticity boundary values and hence there is no need to specify the vorticity boundary values externally. The present method is also easy to implement from the discretization point of view, since it does not require a staggered grid as required by the finite-difference methods. The proposed algorithm is validated by solving a lid-driven square cavity problem at  $Re = 100, 400, \text{ and } 1,000$ . The efficiency of the present scheme to solve coupled problems is demonstrated by solving a differentially heated square cavity problem for Rayleigh number  $Ra = 10^3, 10^4, 10^5, \text{ and } 10^6$ . The application of the algorithm to flow and natural-convection problems is discussed in the following sections.

## 2. GENERALIZED DIFFERENTIAL QUADRATURE METHOD

The GDQ method replaces a given partial space derivative of a function  $f(x)$  by a linear weighted sum of the function values at the discrete sample points considered along the coordinate direction. As a result, the given partial differential equation reduces to a set of algebraic equations. Hence the GDQ method can be

used to obtain numerical solution of partial differential equations. Details about this method can be obtained elsewhere [13]. For a function of two variables  $f(x, y)$ , the  $l$ th-order derivatives with respect to  $x$  and the  $m$ th-order derivatives with respect to  $y$  can be obtained as [13]

$$f_x^{(l)}(x_i, y_j) = \sum_{k=1}^L A_{i,k}^{(l)} f(x_k, y_j) \quad l = 1, 2, \dots, L-1 \quad (1a)$$

$$f_y^{(m)}(x_i, y_j) = \sum_{k=1}^M B_{j,k}^{(m)} f(x_i, y_k) \quad m = 1, 2, \dots, M-1 \quad (1b)$$

for  $i = 1, 2, \dots, L; j = 1, 2, \dots, M$

where  $L, M$  are the number of grid points in the  $x, y$  directions, respectively, and  $A_{i,k}^{(l)}, B_{j,k}^{(m)}$  are the respective weighting coefficients. For the first-order derivatives, the weighting coefficients  $A_{i,k}^{(1)}, B_{j,k}^{(1)}$  can be determined as follows:

$$A_{i,j}^{(1)} = \frac{L^{(1)}(x_i)}{(x_i - x_j)L^{(1)}(x_j)} \quad i, j = 1, 2, \dots, L, \quad \text{but } j \neq i \quad (2a)$$

$$B_{i,j}^{(1)} = \frac{M^{(1)}(y_i)}{(y_i - y_j)M^{(1)}(y_j)} \quad i, j = 1, 2, \dots, M, \quad \text{but } j \neq i \quad (2b)$$

in which

$$L^{(1)}(x_i) = \prod_{j=1, j \neq i}^L (x_i - x_j) \quad M^{(1)}(y_i) = \prod_{j=1, j \neq i}^M (y_i - y_j) \quad (3)$$

Similarly, the weighting coefficients for the second- and higher-order derivatives can be obtained as

$$A_{i,j}^{(l)} = l \left( A_{i,i}^{(l-1)} A_{i,j}^{(1)} - \frac{A_{i,j}^{(l-1)}}{x_i - x_j} \right) \quad \text{for } i, j = 1, 2, \dots, L, \\ \text{but } j \neq i, \quad l = 2, 3, \dots, L-1 \quad (4a)$$

$$B_{i,j}^{(m)} = m \left( B_{i,i}^{(m-1)} B_{i,j}^{(1)} - \frac{B_{i,j}^{(m-1)}}{y_i - y_j} \right) \quad \text{for } i, j = 1, 2, \dots, M, \\ \text{but } j \neq i, \quad m = 2, 3, \dots, M-1 \quad (4b)$$

When  $j = i$ , the weighting coefficients are written as

$$A_{i,i}^{(l)} = - \sum_{j=1, j \neq i}^L A_{i,j}^{(l)} \quad i = 1, 2, \dots, L, \quad l = 1, 2, \dots, L-1 \quad (5a)$$

$$B_{i,i}^{(m)} = - \sum_{j=1, j \neq i}^M B_{i,j}^{(m)} \quad i = 1, 2, \dots, M, \quad m = 1, 2, \dots, M - 1 \quad (5b)$$

It should be noted from the above equations that the weighting coefficients for the second- and higher-order derivatives can be computed from the first-order derivatives themselves. The main advantage of the GDQ method is that the  $l$ th-order and  $m$ th-order derivatives of a function can be approximated with  $(L - l)$ th-order and  $(M - m)$ th-order accuracy when  $L$  and  $M$  are the number of grid points considered in the  $x$  and  $y$  coordinate directions, respectively.

### 2.1. Incompressible Viscous Flow Problem

**Governing equations and boundary conditions.** For a steady, two-dimensional, and incompressible viscous flow, the Navier–Stokes equations in velocity–vorticity form are written as

Velocity Poisson equations

$$\nabla^2 u = - \frac{\partial \omega}{\partial y} \quad (6)$$

$$\nabla^2 v = \frac{\partial \omega}{\partial x} \quad (7)$$

Vorticity transport equation

$$u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \frac{1}{\text{Re}} \nabla^2 \omega \quad (8)$$

In Eqs. (6)–(8),  $u$  and  $v$  are the velocity components in the  $x$  and  $y$  directions, respectively, and  $\omega$  is the vorticity defined for a 2-D incompressible flow in a solution domain  $\Omega$  surrounded by a boundary  $\Gamma$ . We seek a solution in the domain  $\Omega$ , which satisfies the Dirichlet boundary conditions of velocity given as

$$\mathbf{u} = \mathbf{u}_b \quad (9)$$

and the corresponding vorticity values on the boundary can be obtained using the definition given as

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \quad (10)$$

**Numerical solution of the momentum equations using the GDQ.** The spatial derivatives of the velocity Poisson equations (6) and (7) and the vorticity transport equation (8) can be numerically approximated by the GDQ method using Eqs. (1)–(5). For a given set of grid points considered in the coordinate directions, the GDQ approximation of the governing equations results in the following algebraic equations:

$$\sum_{k=1}^L A_{i,k}^{(2)} u_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} u_{i,k} + \sum_{k=1}^M B_{j,k}^{(1)} \omega_{i,k} = 0 \quad (11)$$

$$\sum_{k=1}^L A_{i,k}^{(2)} v_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} v_{i,k} - \sum_{k=1}^L A_{i,k}^{(1)} \omega_{k,j} = 0 \quad (12)$$

$$u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} \omega_{k,j} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} \omega_{i,k} - \frac{1}{\text{Re}} \left( \sum_{k=1}^L A_{i,k}^{(2)} \omega_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} \omega_{i,k} \right) = 0 \quad (13)$$

The velocity boundary conditions can be computed as

$$\begin{aligned} u_{1,j} = 0 & & u_{L,j} = 0 & & u_{i,1} = 0 & & u_{i,M} = 1 \\ v_{1,j} = 0 & & v_{L,j} = 0 & & v_{i,1} = 0 & & v_{i,M} = 0 \end{aligned} \quad (14)$$

for  $i = 1, \dots, L$  and  $j = 1, \dots, M$

The vorticity boundary conditions expressed in terms of velocity gradients by Eq. (10) can also be approximated by the GDQ method as follows:

$$\omega_{i,j} - \sum_{k=1}^L A_{i,k}^{(1)} v_{k,j} + \sum_{k=1}^M B_{j,k}^{(1)} u_{i,k} = 0 \quad (15)$$

for  $i = 1, \dots, L$ , and  $j = 1$  or  $j = M$

for  $j = 2, \dots, M - 1$ , and  $i = 1$  or  $i = L$

The GDQ approximation of the governing equations expressed by the algebraic equations (11)–(13) are coupled and nonlinear equations. By combining the respective coefficient matrices of the flow variables, Eqs. (11)–(13) can be expressed as a global matrix scheme as follows:

$$\begin{bmatrix} A_1 & 0 & A_2 \\ 0 & B_1 & B_2 \\ 0 & 0 & C_1 \end{bmatrix} \begin{Bmatrix} u \\ v \\ \omega \end{Bmatrix} = \begin{Bmatrix} f_u \\ f_v \\ f_\omega \end{Bmatrix} \quad (16)$$

where

$$\begin{aligned} [A_1] &= \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} & [A_2] &= \sum_{k=1}^M B_{j,k}^{(1)} \\ [B_1] &= \sum_{k=1}^L A_{i,k}^{(1)} + \sum_{k=1}^M B_{j,k}^{(1)} & [B_2] &= - \sum_{k=1}^L A_{i,k}^{(2)} \\ [C_1] &= u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} - \frac{1}{\text{Re}} \left( \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} \right) \end{aligned}$$

The specification of velocity boundary conditions using Eq. (14) is straightforward. However, the vorticity boundary conditions represented by Eq. (15) can be specified either explicitly as adopted by Napolitano et al. [2] and Wong and Baker [10] or implicitly. Since the vorticity as well as the velocities, the gradients of which are used in the vorticity definition, are the field variables to be computed, an implicit computation of vorticity boundary values makes the computational

algorithm simple. In order to achieve an implicit computation of the vorticity boundary values, the vorticity definition expressed by Eq. (15) is used to modify the coefficients of velocities and vorticity in Eq. (16) at the solid boundaries. This results in a coupled solution algorithm, in which the vorticity boundary values will be computed implicitly. Further, the use of the GDQ method allows the computation of the vorticity boundary values with higher-order accuracy, which is essential to satisfy the incompressibility constraint. In the above global coefficient matrix, only the nonzero entries of the coefficient matrices of the velocities and vorticity are evaluated and stored in a compressed column-vector storage form. A biconjugate gradient iterative scheme [17] is used to solve the final equations.

## 2.2. Natural Convection in a Square Cavity

**Governing equations and boundary conditions.** In the case of natural-convection problems, the momentum transport equation includes the buoyancy force generated as a result of difference in density of the fluid caused by the temperature difference. The buoyancy term is computed based on the Boussinesq approximation. By taking the curl of the primitive variable form of the governing equations for natural convection and making use of the vorticity definition, the velocity–vorticity form of the natural-convection equations can be obtained as follows:

Vorticity transport equation

$$u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \text{Pr} \nabla^2 \omega + \text{Ra Pr} \frac{\partial T}{\partial x} \quad (17)$$

Energy equation

$$u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = \nabla^2 T \quad (18)$$

in which  $L$ ,  $\alpha/L$ ,  $\alpha/L^2$ ,  $L^2/\alpha$ , and  $T = (\theta - \theta_c)/(\theta_h - \theta_c)$  are used as scale factors for length, velocity, vorticity, time, and temperature in the above nondimensional form of governing equations. The other nondimensional parameters are defined as  $\text{Ra} = g\beta\Delta TL^3/\alpha\nu$  and  $\text{Pr} = \nu/\alpha$ . The velocity Poisson equations (6) and (7) remain unaffected, as they represent the kinematic condition of the flow field. Equations (17) and (18) with  $\omega$  as vorticity and  $T$  as the scalar temperature field are the final form of the governing equations to be solved along with the velocity Poisson equations (6) and (7) in the domain  $\Omega$  with the specified boundary conditions for velocity, vorticity, and temperature on the solid boundary  $\Gamma$ . The boundary conditions for velocity and vorticity have already been defined by the expressions given by Eqs. (9) and (10).

The Dirichlet and Neumann boundary conditions for temperature are as follows:

$$T = T_b \quad (19a)$$

$$\frac{\partial T}{\partial y} = 0 \quad (19b)$$

**Numerical solution of the governing equations for natural convection by the GDQ method.** Following the steps as discussed in Section 2.1 for the flow equations, the GDQ approximation of the velocity Poisson equations (6) and (7), the vorticity transport equation (17), and the energy equation (18) are expressed as

$$\sum_{k=1}^L A_{i,k}^{(2)} u_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} u_{i,k} + \sum_{k=1}^M B_{j,k}^{(1)} \omega_{i,k} = 0 \quad (20)$$

$$\sum_{k=1}^L A_{i,k}^{(2)} v_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} v_{i,k} - \sum_{k=1}^L A_{i,k}^{(1)} \omega_{k,j} = 0 \quad (21)$$

$$u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} \omega_{k,j} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} \omega_{i,k} - \text{Pr} \left( \sum_{k=1}^L A_{i,k}^{(2)} \omega_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} \omega_{i,k} \right) - \text{Ra Pr} \sum_{k=1}^L A_{i,k}^{(1)} T_{k,j} = 0 \quad (22)$$

$$u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} T_{k,j} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} T_{i,k} - \left( \sum_{k=1}^L A_{i,k}^{(2)} T_{k,j} + \sum_{k=1}^M B_{j,k}^{(2)} T_{i,k} \right) = 0 \quad (23)$$

The algebraic equations (20)–(23) are the GDQ form of the governing equations for the velocities, vorticity, and temperature. By combining the coefficient matrices of all the field variables, the following global matrix form is obtained for the coupled solution algorithm:

$$\begin{bmatrix} D_1 & 0 & D_2 & 0 \\ 0 & E_1 & E_2 & 0 \\ 0 & 0 & F_1 & F_2 \\ 0 & 0 & 0 & G_1 \end{bmatrix} \begin{Bmatrix} u \\ v \\ \omega \\ T \end{Bmatrix} = \begin{Bmatrix} f_u \\ f_v \\ f_\omega \\ f_T \end{Bmatrix} \quad (24)$$

where

$$[D_1] = \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} \quad [D_2] = \sum_{k=1}^M B_{j,k}^{(1)}$$

$$[E_1] = \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} \quad [E_2] = - \sum_{k=1}^L A_{i,k}^{(1)}$$

$$[F_1] = u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} - \text{Pr} \left( \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} \right)$$

$$[F_2] = -\text{Ra Pr} \sum_{k=1}^L A_{i,k}^{(1)}$$

$$[G_1] = u_{i,j} \sum_{k=1}^L A_{i,k}^{(1)} + v_{i,j} \sum_{k=1}^M B_{j,k}^{(1)} - \left( \sum_{k=1}^L A_{i,k}^{(2)} + \sum_{k=1}^M B_{j,k}^{(2)} \right)$$



The expressions (14) and (15) for the velocity and the vorticity boundary conditions that have already been derived in the previous section remain the same. The vorticity values at the boundaries are evaluated implicitly by the procedure discussed in Section 2.1. The Dirichlet temperature boundary conditions at the left and the right walls of the square cavity can be represented as

$$T_{1,j} = 1 \quad T_{L,j} = 0 \tag{25}$$

The adiabatic boundary conditions on the top and the bottom sides of the square cavity can be expressed in the GDQ form as

$$T_{i,1} = \frac{1}{B_{1,1}^{(1)}B_{M,M}^{(1)} - B_{M,1}^{(1)}B_{1,M}^{(1)}} \left[ \sum_{k=2}^{M-1} (B_{1,M}^{(1)}B_{M,k}^{(1)} - B_{M,M}^{(1)}B_{1,k}^{(1)})T_{i,k} \right] \tag{26a}$$

$$T_{i,M} = \frac{1}{B_{1,M}^{(1)}B_{M,1}^{(1)} - B_{M,M}^{(1)}B_{1,1}^{(1)}} \left[ \sum_{k=2}^{M-1} (B_{M,k}^{(1)}B_{1,1}^{(1)} - B_{1,k}^{(1)}B_{M,1}^{(1)})T_{i,k} \right] \tag{26b}$$

for  $i = 2, \dots, L - 1, j = 1, 2, \dots, M$

### 3. RESULTS AND DISCUSSION

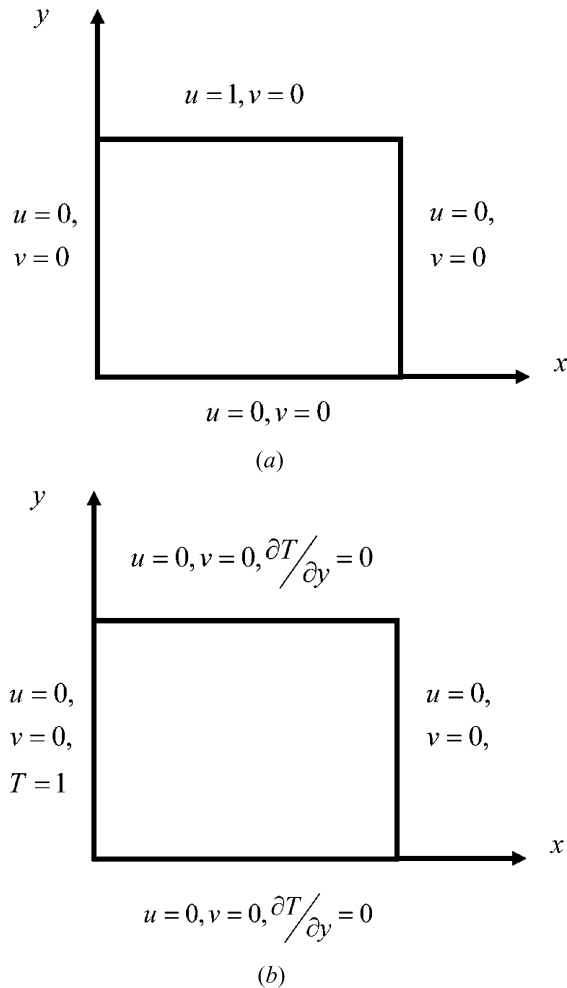
The present numerical algorithm based on the velocity–vorticity formulation and the GDQ method is validated for a lid-driven cavity flow at  $Re = 100, 400,$  and  $1,000$ . Later the algorithm is applied to solve a natural-convection problem in a differentially heated square cavity, in which the momentum equation is coupled to the energy equation through the buoyancy term. Numerical results for natural-convection at  $Ra = 10^3-10^6$  are obtained and compared with the benchmark solutions. The present algorithm allows the prediction of temperature field without using the pressure term. We used the following expression to compute the mesh-point distributions in both  $x$  and  $y$  directions in the GDQ numerical procedure:

$$x_i = \frac{\cos[\pi/(2N)] - \cos[(2i - 1)\pi/(2N)]}{\cos[\pi/(2N)] - \cos[(2N - 1)\pi/(2N)]} \quad i = 1, 2, \dots, N \tag{27}$$

where  $N$  is the number of grid points considered in a given coordinate direction.

#### 3.1. Validation Results for Lid-Driven Cavity Flow

A program in FORTRAN has been developed to validate the code for the solution of two-dimensional incompressible Navier–Stokes equations in velocity–vorticity form for a lid-driven square cavity as shown in Figure 1a. The top lid of the cavity is assumed to move with a unit velocity in the  $x$  direction. Figure 2a shows the comparison of the present results for  $u$ – $y$  and  $x$ – $v$  plots obtained using different grids with those of Ghia et al. [18] obtained with a grid of size  $129 \times 129$  for  $Re = 100$ . As can be seen from the above figure, even with a coarser grid of size  $15 \times 15$ , the present predictions are in close agreement with the results of Ghia et al.



**Figure 1.** (a) Cavity flow problem with boundary conditions. (b) Natural-convection problem with boundary conditions.

[18], who used a grid of size  $129 \times 129$ . The comparison of the present results obtained using two different grids for  $Re = 400$  with the results of Ghia et al. [18] are shown in Figure 2b. It is noted that the present results obtained with a finer grid of size  $21 \times 21$  are in excellent agreement with the predictions of Ghia et al. [18]. As the Reynolds number increases, a finer mesh has to be used in order to capture the thin boundary layers developed along the wall boundaries. Hence, for the case of  $Re = 1,000$  we used a  $31 \times 31$  grid size to get the flow solutions. Figure 2c depicts the comparisons of the present results with those of Ghia et al. [18] obtained with a grid of size  $129 \times 129$  for  $Re = 1,000$ . The present numerical results predicted using a mesh of size  $31 \times 31$  are in close agreement with the results of Ghia et al. [18]. It can be observed from the above figures that the present results obtained even with a coarse mesh show excellent agreement with the benchmark solutions for  $Re$  in the

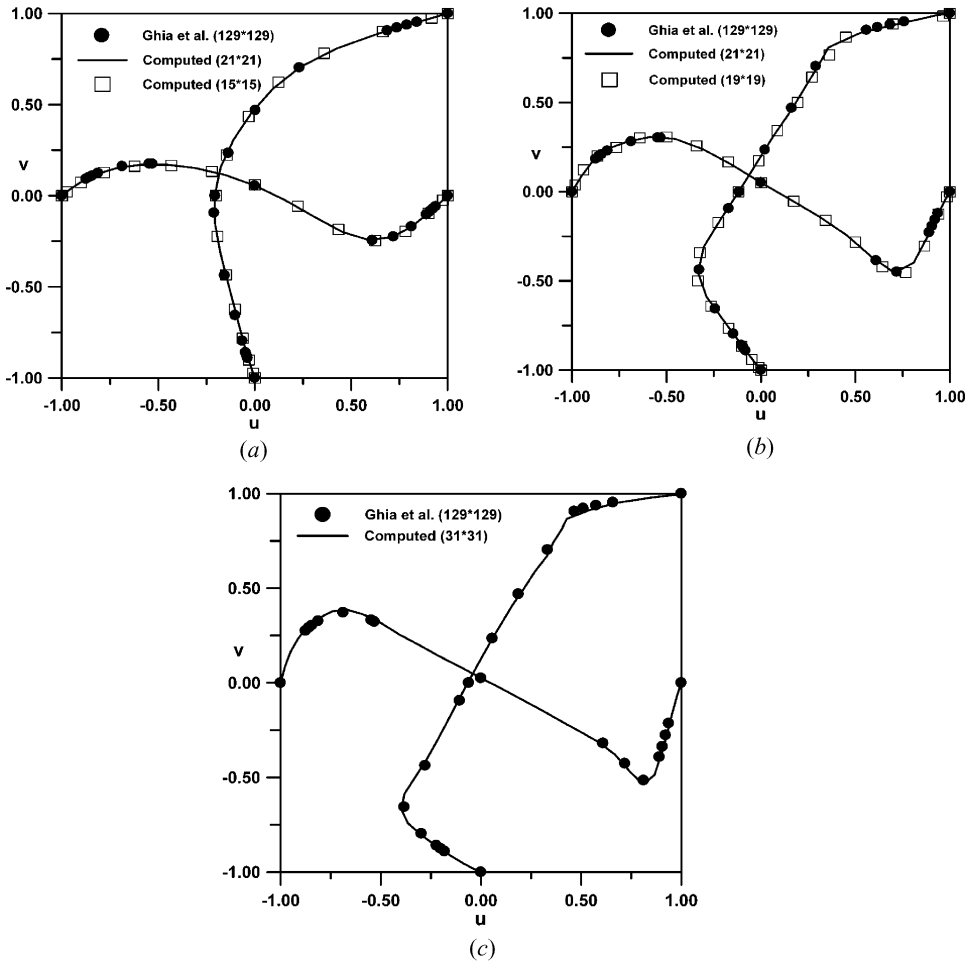


Figure 2. Velocity profiles at horizontal and vertical line: (a)  $Re = 100$ ; (b)  $Re = 400$ ; (c)  $Re = 1,000$ .

range 100–1,000. The changes in the velocity vector distribution with increase in Reynolds number are shown in Figures 3a, 3b, and 3c for  $Re = 100$ , 400, and 1,000, respectively. The expected flow behavior depicted in the above figures demonstrates the efficiency of the present coupled numerical algorithm to solve the Navier–Stokes equations in velocity–vorticity form.

### 3.2. Mesh Dependence Study

The GDQ method approximates a given partial space derivative with an order equal to  $(L - l)$ , where  $L$  is the number of grid points in the given coordinate direction and  $l$  is the order of derivative. This indicates that the GDQ method requires only a coarser mesh compared to the finite-element and the finite-difference methods to achieve a given numerical accuracy. In order to demonstrate this

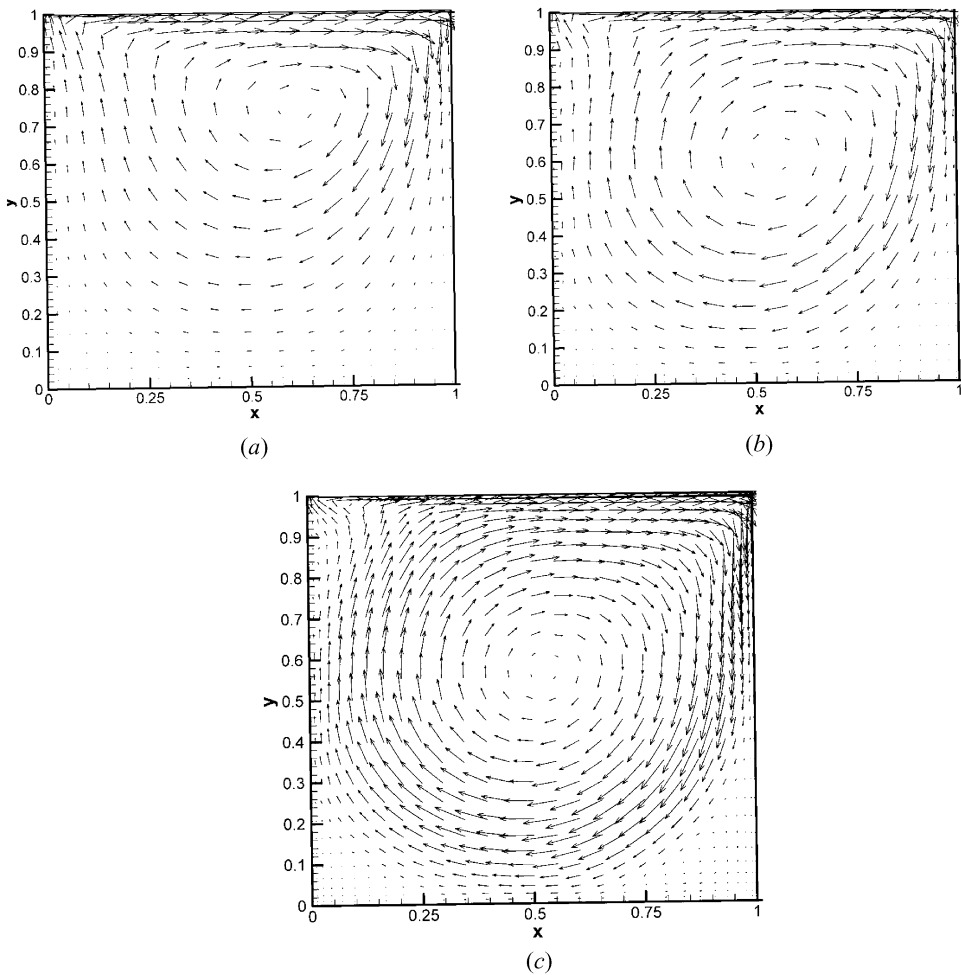


Figure 3. Velocity distribution for: (a)  $Re = 100$ ; (b)  $Re = 400$ ; (c)  $Re = 1,000$ .

property of the GDQ method and verify the consistency in the numerical accuracy with refinement of mesh, a mesh dependence study is carried out using the square cavity problem at  $Re = 100$ ,  $400$ , and  $1,000$ . The minimum  $u$  velocity, and the minimum and maximum  $v$ -velocities, are considered as the comparison parameters. The present numerical predictions of these parameters are compared with the results of Ghia et al. [18] and two commercial codes: (1) the FIDAP with the finite-element method (FEM) [19], (2) the TASC flow using the finite-volume method (FVM) [19] as found in [20], and (3) the BDIM [20]. Tables 1, 2, and 3 show the comparisons of the present results with those of the benchmark solutions for  $Re = 100$ ,  $400$ , and  $1,000$ , respectively. Observing the values in Table 1, it is easily understood that the present scheme based on the GDQ method can predict the benchmark solutions for  $Re = 100$  using a much coarser grid,  $15 \times 15$ , compared to other numerical schemes, which required a finer mesh. Similarly, for

**Table 1.** Numerical results of lid-driven square cavity for  $Re = 100$ 

Velocity mesh size	Bench. [18] (129 × 129)	GDQ ( $v - \omega$ ) (15 × 15)	FEM [19] (21 × 21)	FVM [19] (21 × 21)	BDIM [20] (21 × 21)
$(u)_{\min}$	-0.211	-0.211	-0.178	-0.191	-0.213
Error (%)		0.47	15.64	10.47	1.42
$(v)_{\min}$	-0.245	-0.243	-0.217	-0.233	-0.259
Error (%)		0.82	11.3	4.9	5.71
$(v)_{\max}$	0.175	0.172	0.152	0.16	0.177
Error (%)		1.71	13.14	8.57	1.14

**Table 2.** Numerical results of lid-driven square cavity for  $Re = 400$ 

Velocity mesh size	Bench. [18] (129 × 129)	GDQ ( $v - \omega$ ) (21 × 21)	FEM [19] (41 × 41)	FVM [19] (41 × 41)	BDIM [20] (41 × 41)
$(u)_{\min}$	-0.327	-0.329	-0.309	-0.237	-0.327
Error (%)		0.61	5.5	27.5	0
$(v)_{\min}$	-0.450	-0.455	-0.430	-0.387	-0.458
Error (%)		1.11	4.44	14	1.78
$(v)_{\max}$	0.302	0.306	0.284	0.230	0.306
Error (%)		1.32	5.96	23.84	1.32

**Table 3.** Numerical results of lid-driven square cavity for  $Re = 1,000$ 

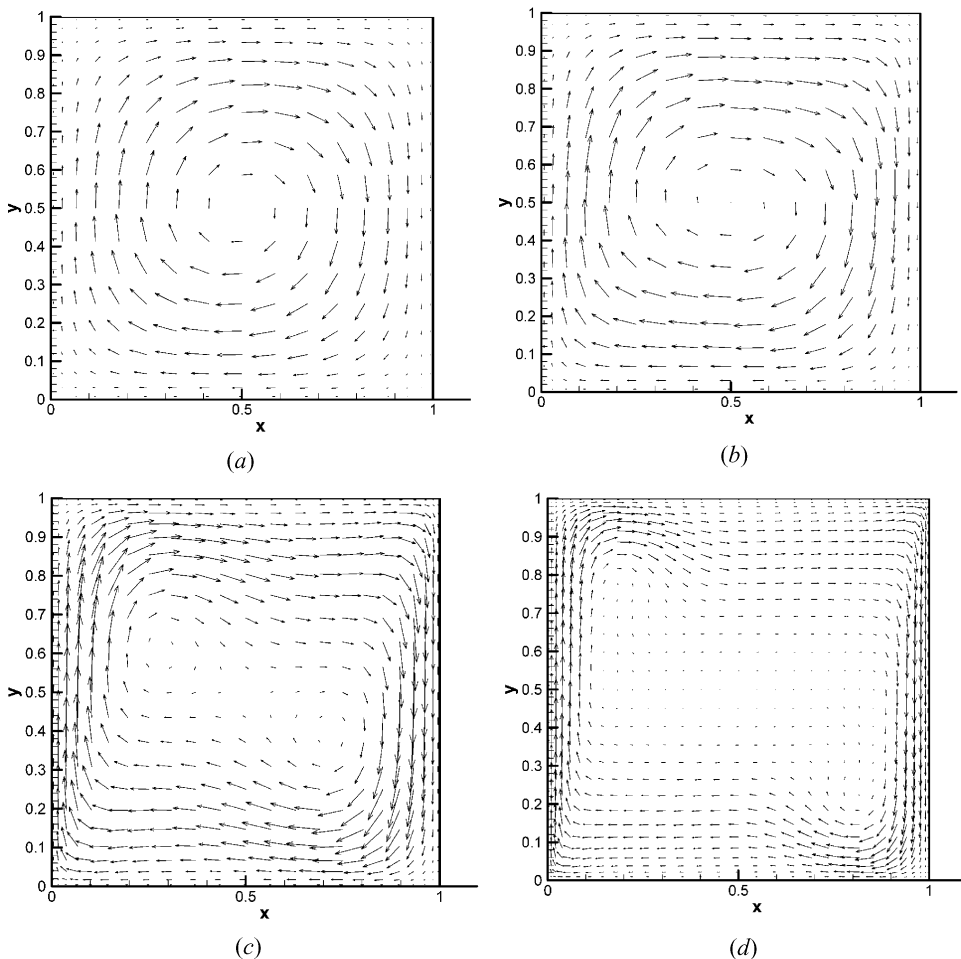
Velocity mesh size	Bench. [18] (129 × 129)	GDQ ( $v - \omega$ ) (31 × 31)	FEM [19] (41 × 41)	FVM [19] (41 × 41)	BDIM [20] (41 × 41)
$(u)_{\min}$	-0.383	-0.392	-0.264	-0.243	-0.389
Error (%)		2.35	30.29	36.55	1.57
$(v)_{\min}$	-0.516	-0.526	-0.359	-0.411	-0.533
Error (%)		1.94	30.43	20.35	3.29
$(v)_{\max}$	0.371	0.380	0.211	0.227	0.371
Error (%)		2.69	43.13	38.81	0

$Re = 400$ , Table 2 indicates that the GDQ method requires only a  $21 \times 21$  grid, whereas other numerical schemes have used finer meshes of the order of  $41 \times 41$  and higher, in order to get the same results. It can be observed that even with such finer meshes, the FEM [19] and the FVM [19] could not predict results as close to the benchmark solutions as are predicted by the present numerical scheme. The above observations hold good for all three comparison parameters shown in the above tables. Comparison of results for  $Re = 1,000$ , shown in Table 3, indicates that the present results obtained using a grid of size  $31 \times 31$  are in excellent agreement with those of the benchmark solutions, which had used much finer meshes ( $129 \times 129$ ) than the present method. The tabular values indicate that the present coupled numerical scheme based on the velocity–vorticity equations and the GDQ method is capable of predicting the benchmark solutions for a cavity flow with less than 3% error using meshes much coarser than other numerical schemes.

### 3.3. Natural Convection in a Square Cavity

After successfully validating the present algorithm for a lid-driven cavity problem, natural convection in a differentially heated square cavity is analyzed. The velocities at all the boundaries are assumed to be equal to zero. Temperatures equal to 1 and 0 are assumed at the left and the right walls, respectively. Adiabatic conditions are assumed on the top and bottom sides of the cavity. In order to verify the ability of the present numerical algorithm to satisfy the momentum conservation and the continuity constraint along with heat transfer, results on flow and temperature fields are obtained for  $Ra = 10^3, 10^4, 10^5,$  and  $10^6$ .

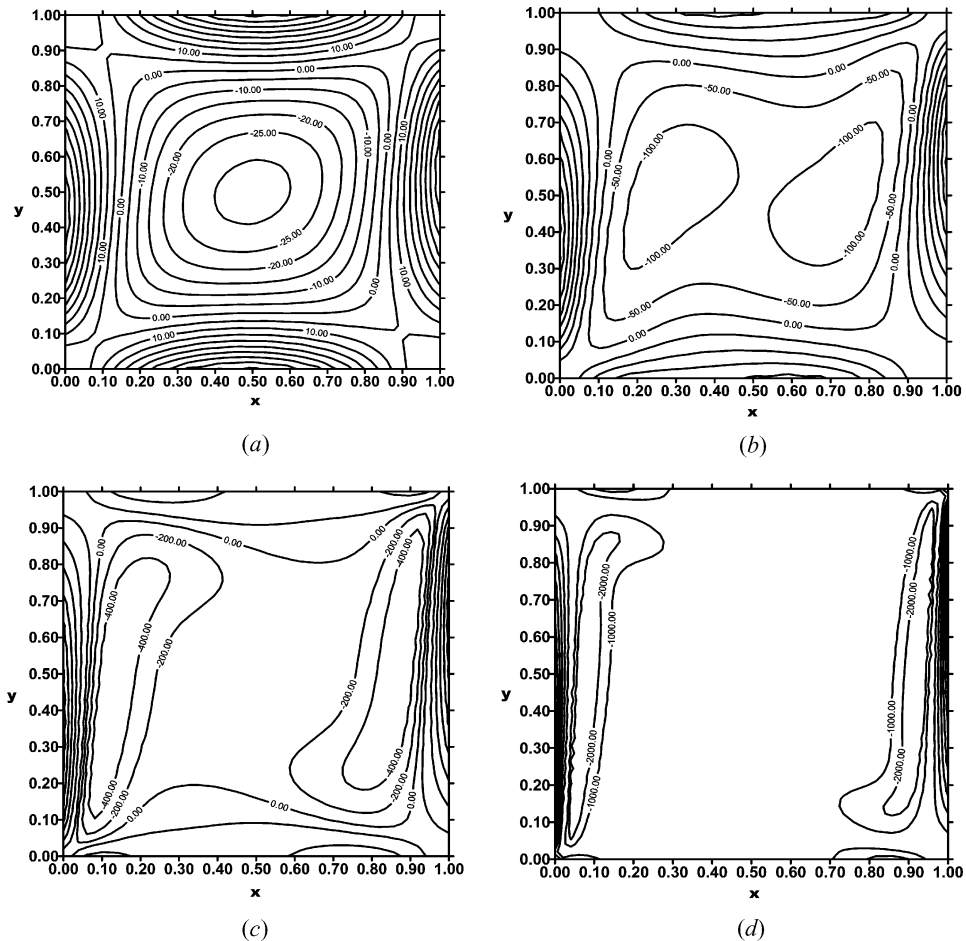
Figures 4a–4d show the velocity vector distributions on the  $x$ – $y$  plane for  $Ra = 10^3, 10^4, 10^5,$  and  $10^6$ , respectively. Since the left wall is at a higher temperature than the right wall, the density of fluid near the left wall decreases compared to



**Figure 4.** Velocity distribution at different  $Ra$  number for: (a)  $Ra = 10^3$ ; (b)  $Ra = 10^4$ ; (c)  $Ra = 10^5$ ; (d)  $Ra = 10^6$ .

the density of the fluid adjacent to the right wall, resulting in a clockwise rotation of the fluid inside the cavity as seen from the above figures. At low values of Rayleigh numbers, that is, at  $10^3$  and  $10^4$ , the convective effect is too small and hence the inertial forces do not make significant contribution to the heat transport between the end walls. As the Rayleigh number increases, the buoyancy force increases, resulting in a strong circulation of the fluid inside the cavity and thus increasing convective heat transport as observed in Figures 4c and 4d. Vorticity distributions inside the cavity for different values of Rayleigh number display the effect of buoyancy force on the vorticity field generated as a result of convection inside the cavity.

Figures 5a–5d show the vorticity distributions on the  $x$ – $y$  plane for  $Ra = 10^3$ ,  $10^4$ ,  $10^5$ , and  $10^6$ , respectively. For  $Ra = 10^3$ , a somewhat concentric vorticity distributions about the center of the cavity is observed in Figure 5a, because the heat



**Figure 5.** Vorticity distribution at different  $Ra$  number for: (a)  $Ra = 10^3$ ; (b)  $Ra = 10^4$ ; (c)  $Ra = 10^5$ ; (d)  $Ra = 10^6$ .

penetration across the fluid has not yet set the complete convection process. The vorticity values are also small compared to those at high Rayleigh numbers. As the Rayleigh number increases, the onset of convection separates the two zones of the fluid circulating near the hot and the cold walls, as observed in Figures 5*b–5d*. This trend is enhanced at  $Ra = 10^6$ , as seen in Figure 5*d*. It is to be noted from Figures 5*a–5d* that the strength of vorticity increases with increase in the Rayleigh number, as expected. Since the vorticity is a function of velocity gradient and also it is generated at the boundary of the cavity, the GDQ method could predict the vorticities very accurately, and these are demonstrated in the above figures.

Figures 6*a–6d* represent the temperature distributions inside the cavity for  $Ra = 10^3, 10^4, 10^5$ , and  $10^6$ , respectively. As can be seen from these figures, the present numerical scheme can accurately predict the changes in the temperature of the fluid, varying from 1.0 at the left wall to 0.0 at the right wall. At low Rayleigh number value, the heat transfer is purely by means of diffusion because the buoyancy force generated is not strong enough to initiate fluid convection, as noted in Figure 6*a* for  $Ra = 10^3$ . As the Rayleigh number increases, the convection inside the cavity also increases, and this results in diminishing thermal boundary layers at the hot and the cold walls as observed in Figures 6*c* and 6*d*. Higher temperature gradients observed near the end walls reveal the strength of convective heat transfer at high Rayleigh number flows. The expected trends of temperature distribution could be obtained because the fluid momentum conservation and the continuity constraint have been correctly satisfied on the computational domain by the present algorithm.

### 3.4. Comparison of $u_{\max}$ , $v_{\max}$ , and Nusselt Number for Natural Convection

The numerical accuracy of the present algorithm to predict the flow and heat transport parameters is tested by comparing the following parameters with benchmark solutions of de Vahl Davis [21] and others [15, 22, 23]:

1. Maximum horizontal velocity  $(u)_{\max}$  on the vertical midplane and its location
2. Maximum vertical velocity  $(v)_{\max}$  on the horizontal midplane and its location
3. The Nusselt number on the vertical boundary at  $x = 0$ , evaluated as

$$Nu_0 = \int_0^1 \left( uT - \frac{\partial T}{\partial x} \right) dy \quad (28)$$

4. The average Nusselt number throughout the cavity, evaluated as

$$\overline{Nu}_0 = \int_0^1 Nu_0 dx \quad (29)$$

Numerical results of de Vahl Davis [21], the GDQ method applied to the primitive-variable form of the momentum equations [15], the finite-difference method [22], and the finite-volume method [23] are considered for the purpose of comparison.



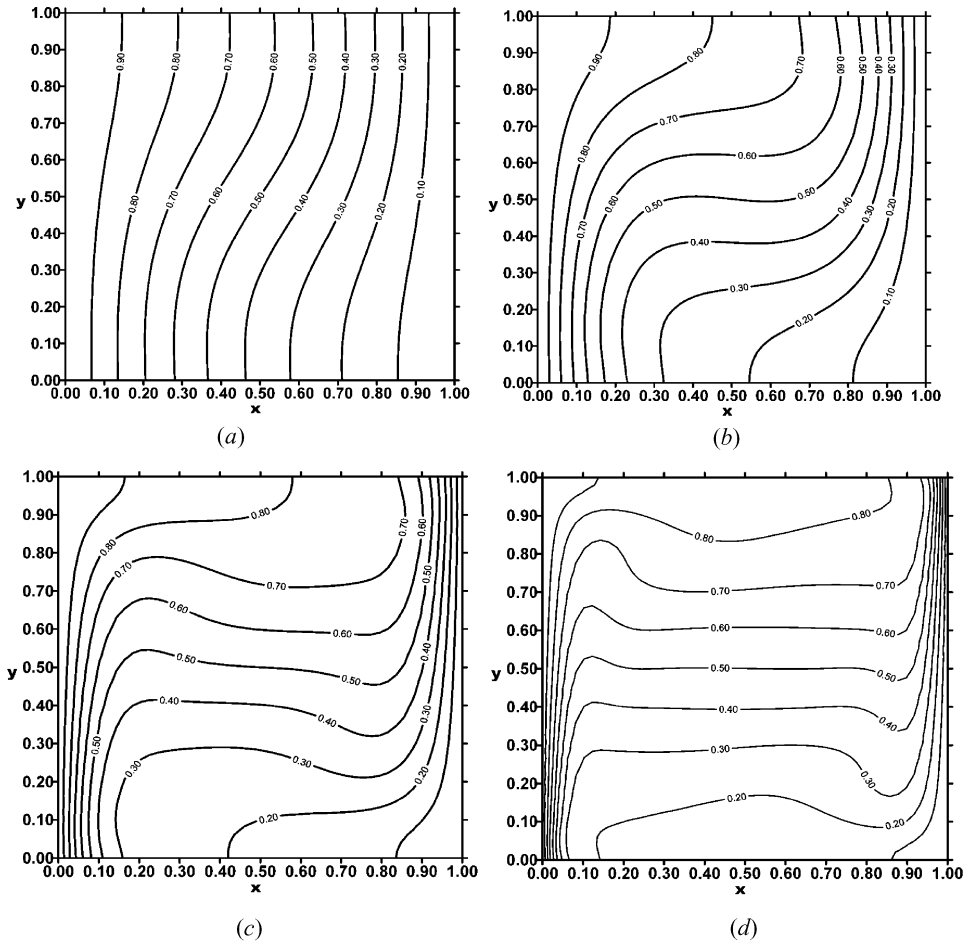


Figure 6. Temperature distribution at different Ra number for: (a)  $Ra = 10^3$ ; (b)  $Ra = 10^4$ ; (c)  $Ra = 10^5$ ; (d)  $Ra = 10^6$ .

Tables 4–7 show the comparisons of the present results with the above references for  $Ra = 10^3, 10^4, 10^5,$  and  $10^6$ , respectively. The expected trend of increase in the values of  $u_{max}, v_{max},$  and average Nusselt number with increase in Rayleigh number has been correctly predicted by the present algorithm. From the tabulated results it can be observed that the present GDQ method produced the benchmark solutions with a much coarser mesh compared to the mesh used in the benchmark solutions for all four values of Rayleigh number. Further, the present coupled numerical scheme using the velocity–vorticity equations predicts flow and heat transfer parameters much closer to the benchmark solutions, compared to the results of Shu and Wee [15], who used the GDQ method to solve the primitive form of the momentum equations, though we used the same grid size as used in [15]. For all four values of Rayleigh number considered here, the present numerical scheme predicts flow and heat transfer parameters much closer to the values of the benchmark

**Table 4.** Numerical results of natural convection in a square cavity for  $Ra = 10^3$ 

Mesh size	Bench. [21] (81 × 81)	GDQ (present) (11 × 11)	GDQ [15] (11 × 11)	FD [22] (41 × 41)	FV [23] (21 × 21)
$(u)_{\max}$	3.649	3.645	3.648	3.642	3.649
Error (%)		0.11	0.027	0.192	0
$y$	0.813	0.815	0.81	0.8	0.813
Error (%)		0.246	0.369	1.6	0
$(v)_{\max}$	3.697	3.697	3.696	3.699	3.690
Error (%)		0	0.027	0.054	0.189
$x$	0.178	0.179	0.18	0.175	0.179
Error (%)		0.562	1.11	1.685	0.562
$Nu_0$	1.117	1.115	1.118	1.117	1.109
Error (%)		0.18	0.09	0	0.716
$\overline{Nu}_0$	1.118	1.116	1.118	1.117	1.113
Error (%)		0.179	0	0.089	0.447

solutions compared to the results obtained by the finite-difference method [22] and the finite-volume method [23], which used finer meshes than the present scheme. This demonstrates that the present GDQ method is a very powerful numerical scheme to solve flow and heat transfer problems when the momentum equations in velocity–vorticity form are solved as a fully coupled system of equations. As an extension of the present study, simulation of natural convection in a cubical cavity is being carried out, and the results will be communicated as a separate article.

#### 4. CONCLUSIONS

A numerical scheme based on the GDQ method has been developed for solving the velocity–vorticity form of two-dimensional Navier–Stokes equations for flow and heat transfer in a square cavity. The vorticity boundary conditions are computed directly from the vorticity definition from the global matrix obtained as a result of

**Table 5.** Numerical results of natural convection in a square cavity for  $Ra = 10^4$ 

Mesh size	Bench. [21] (81 × 81)	GDQ (present) (19 × 19)	GDQ [15] (19 × 19)	FD [22] (41 × 41)	FV [23] (21 × 21)
$(u)_{\max}$	16.178	16.180	16.182	16.265	16.189
Error (%)		0.012	0.025	0.538	0.068
$y$	0.823	0.821	0.82	0.825	0.822
Error (%)		0.243	0.365	0.243	0.122
$(v)_{\max}$	19.617	19.612	19.628	19.662	19.606
Error (%)		0.025	0.056	0.229	0.056
$x$	0.119	0.118	0.12	0.125	0.12
Error (%)		0.840	0.840	5.042	0.840
$Nu_0$	2.238	2.235	2.244	2.239	2.310
Error (%)		0.134	0.268	0.045	3.217
$\overline{Nu}_0$	2.243	2.239	2.244	2.245	2.248
Error (%)		0.178	0.045	0.089	0.223

**Table 6.** Numerical results of natural convection in a square cavity for  $Ra = 10^5$ 

Mesh size	Bench. [21] (81 × 81)	GDQ (present) (25 × 25)	GDQ [15] (25 × 25)	FD [22] (41 × 41))	FV [23] (40 × 40)
$(u)_{\max}$	34.722	34.698	34.721	35.156	34.632
Error (%)		0.069	0.003	1.25	0.259
$y$	0.855	0.854	0.85	0.85	0.851
Error (%)		0.0117	0.585	0.585	0.468
$(v)_{\max}$	68.590	68.215	68.462	68.138	68.090
Error (%)		0.547	0.187	0.659	0.729
$x$	0.066	0.067	0.07	0.075	0.073
Error (%)		1.515	6.06	13.64	10.606
$Nu_0$	4.509	4.512	4.518	4.479	N/A
Error (%)		0.067	0.2	0.665	N/A
$\overline{Nu}_0$	4.519	4.515	4.519	4.522	4.537
Error (%)		0.089	0	0.066	0.398

coupling the vorticity definition through the velocities with the velocity Poisson equations, the vorticity transport equation, and the energy equation. Further, the vorticity boundary conditions are computed with much higher-order accuracy by the GDQ method compared to the schemes based on the Taylor's series expansion, which is generally employed to compute the vorticity boundary conditions. Numerical results obtained for  $Re = 100, 400, \text{ and } 1,000$  for a lid-driven cavity flow show good agreement with the benchmark solutions. The higher-order-accurate GDQ method is capable of achieving the benchmark solutions with a maximum error less than 3% using a coarser mesh compared to other numerical schemes.

Natural convection in a differentially heated square cavity was also studied by the coupled solution algorithm. Velocity, vorticity, and temperature distributions computed for  $Ra = 10^3, 10^4, 10^5, \text{ and } 10^6$  show the expected flow and heat transfer behaviors. The present coupled numerical scheme predicted average Nusselt

**Table 7.** Numerical results of natural convection in a square cavity for  $Ra = 10^6$ 

Mesh size	Bench. [21] (81 × 81)	GDQ (present) (33 × 33)	GDQ [15] (33 × 33)	FD [22] (81 × 81)	FV [23] (72 × 72)
$(u)_{\max}$	64.630	64.789	64.855	65.332	64.834
Error (%)		0.246	0.348	1.086	0.316
$y$	0.850	0.853	0.85	0.85	0.850
Error (%)		0.352	0	0	0
$(v)_{\max}$	219.360	220.256	220.072	221.658	220.599
Error (%)		0.408	0.325	1.048	0.565
$x$	0.0379	0.038	0.04	0.038	0.038
Error (%)		0.264	5.541	0.264	0.264
$Nu_0$	8.817	8.825	8.822	8.763	8.825
Error (%)		0.091	0.057	0.612	0.091
$\overline{Nu}_0$	8.8	8.816	8.814	8.829	N/A
Error (%)		0.182	0.159	0.33	N/A

numbers closer to the benchmark solutions with less than 2% error using a grid much coarser than those used by other numerical schemes. Numerical results obtained by the present method for flow as well as heat transfer problems suggest that the higher-order-accurate GDQ method can be well exploited when the Navier–Stokes equations in velocity–vorticity form are used and solved as coupled equations.

## REFERENCES

1. P. M. Gresho and R. L. Sani, On Pressure Boundary Conditions for the Incompressible Navier–Stokes Equations, *Int. J. Numer. Meth. Fluids*, vol. 7, pp. 1111–1145, 1987.
2. M. Napolitano, G. Pascazio, and L. Quartapelle, A Review of Vorticity Conditions in the Numerical Solution of the  $\zeta$ – $\omega$  Equations, *Comput. Fluids*, vol. 28, pp. 139–185, 1999.
3. H. Fasel, Investigation of the Stability of Boundary Layers by a Finite-Difference Model of the Navier–Stokes Equations, *J. Fluid Mech.*, vol. 78, pp. 355–383, 1976.
4. G. Guj and F. Stella, A Vorticity-Velocity Method for the Numerical Solution of 3D Incompressible Flows, *J. Comput. Phys.*, vol. 106, pp. 286–298, 1993.
5. W. Z. Shen and T. P. Loc, Numerical Method for Unsteady 3D Navier–Stokes Equations in Velocity-Vorticity Form, *Comput. Fluids*, vol. 26, pp. 193–216, 1997.
6. G. Guevremont, W. G. Habashi, and M. M. Hafez, Finite Element Solution of the Navier–Stokes Equations by a Velocity-Vorticity Method, *Int. J. Numer. Meth. Fluids*, vol. 10, pp. 461–475, 1990.
7. G. Guevremont, W. G. Habashi, P. L. Kotiuga, and M. M. Hafez, Finite Element Solution of the 3D Compressible Navier–Stokes Equations by a Velocity-Vorticity Method, *J. Comput. Phys.*, vol. 107, pp. 176–187, 1993.
8. D. C. Lo and D. L. Young, Two-Dimensional Incompressible Flows by Velocity-Vorticity Formulation and Finite Element Method, *Chinese J. Mech.*, vol. 17, pp. 13–20, 2001.
9. D. C. Lo and D. L. Young, Arbitrary Lagrangian-Eulerian Finite Element Analysis of Free Surface Flow Using a Velocity-Vorticity Formulation, *J. Comput. Phys.*, vol. 195, pp. 175–201, 2004.
10. K. L. Wong and A. J. Baker, A 3D Incompressible Navier–Stokes Velocity-Vorticity Weak Form Finite Element Algorithm, *Int. J. Numer. Meth. Fluids*, vol. 38, pp. 99–123, 2002.
11. R. E. Bellman and B. G. Kashef, Differential Quadrature: A Technique for the Rapid Solution of Nonlinear Partial Differential Equations, *J. Comput. Phys.*, vol. 10, pp. 40–52, 1972.
12. R. E. Bellman, *Methods of Non-linear Analysis*, vol. 12, chap. 16, Academic Press, New York, 1973.
13. C. Shu and B. E. Richards, Application of Generalized Differential Quadrature to Solve 2-Dimensional Incompressible Navier–Stokes Equations, *Int. J. Numer. Meth. Fluids*, vol. 15, pp. 791–798, 1992.
14. C. Shu, B. C. Khoo, and K. S. Yeo, Numerical Solutions of Incompressible Navier–Stokes Equations by Generalized Differential Quadrature, *Finite Elements Anal. Design*, vol. 18, pp. 83–97, 1994.
15. C. Shu and K. H. A. Wee, Numerical Simulation of Natural Convection in a Square Cavity by SIMPLE-Generalized Differential Quadrature Method, *Comput. Fluids*, vol. 31, pp. 209–226, 2002.
16. C. Shu, L. Wang, and Y. T. Chew, Numerical Computation of Three-Dimensional Incompressible Navier–Stokes Equations in Primitive Variable Form by DQ Method, *Int. J. Numer. Meth. Fluids*, vol. 43, pp. 345–368, 2003.

17. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, chap. 2, Cambridge, University Press, 1992.
18. U. Ghia, K. N. Ghia, and C. T. Shin, High-Re Solutions for Incompressible Flow Using the Navier–Stokes Equations and a Multigrid Method, *J. Comput. Phys.*, vol. 48, pp. 387–411, 1982.
19. TASCflow, Version 2.4.1-3, Advanced Scientific Computing, 1996.
20. M. Ramšak and L. Škerget, Mixed Boundary Elements for Laminar Flows, *Int. J. Numer. Meth. Fluids*, vol. 31, pp. 861–877, 1999.
21. G. de Vahl Davis, Natural Convection of Air in a Square Cavity, a Bench Mark Numerical Solution, *Int. J. Numer. Meth. Fluids*, vol. 3, pp. 249–264, 1983.
22. J. C. Kalita, D. C. Dalal, and A. K. Dass, Fully Compact Higher-Order Computation of Steady-State Natural Convection in a Square Cavity, *Phys. Rev. E*, vol. 64, pp. 066703-1–066703-13, 2001.
23. H. S. Mahdi and R. B. Kinney, Time-Dependent Natural Convection in a Square Cavity: Application of a New Finite Volume Method, *Int. J. Numer. Meth. Fluids*, vol. 11, pp. 57–86, 1990.

Copyright of Numerical Heat Transfer: Part B -- Fundamentals is the property of Taylor & Francis Ltd and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.