RESEARCH PAPER | Education | Volume : 3 | Issue : 9 | Sept 2013 | ISSN - 2249-555X

Bernstein Differential Quadrature Method for Solving the Unsteady State Convection-Diffusion Equation

KEYWORDS: Differential quadrature method, Convection-diffusion, Bernstein polynomial, Accuracy.

A.S.J. Al- Saif	Firas A. Al-Saadawi
Department Mathematics, Education College for pure science, Basrah University, Basrah, Iraq	Directorate of Education in Basrah, Basrah, Iraq

ABSTRACT *In this paper, we proposed a new technique of differential quadrature method to find numerical solutions of the convection-diffusion equation with appropriate initial and boundary conditions. The present technique is based on the Bernstein polynomial formula, which is used to construct the weighted coefficients matrices of differential quadrature method. To demonstrate its usefulness and accuracy, the new proposed method is applied to three test problems, involving different linearity. The results show that the new method is more accurate and convergent than other numerical methods in literature.*

1- Introduction

The study of the general properties of the convection-diffusion equation has attracted the attention of scientific community due to its applications in various fields such as petroleum reservoir simulation, subsurface contaminant remediation, heat conduction, shock waves, acoustic waves, gas dynamics, elasticity, etc[3,14,19]. The studies conducted for solving the convection-diffusion equation in the last half century are still in an active area of research to develop some better numerical methods to approximate its solution. Many researchers solve different types of convection-diffusion equation by different numerical methods[1,3,9,11,13,12,13,16,-

30,31,32]. The best oldest known approximation techniques are the finite difference (FD) and finite element (FE) methods. A relatively new numerical technique is the differential quadrature method (DQM). Despite being a domain discretization method, the differential quadrature method gives accurate results using less discretization points than all the above mentioned methods (FD&FE). DQM depends on the idea of integral quadrature and approximate a spatial partial derivatives as a linear weighted sum of all functional values of the solution at all mesh points [19]. This method was proposed by Bellman and Casti[5] in 1971. One of important keys to DQM lies in the determination of weighting coefficients for the discretization of a spatial derivative of any order, where it play the important role in the accuracy of numerical solutions. Initially, Bellman et al.[6] (1972), suggested two methods to determine the weighting coefficients of the first order derivative. The first method solves an algebraic equation system. The second use a simple algebraic formulation, but with the coordinates of grid points chosen as the roots of the Legendre polynomial. Quan and Chang[22] (1989a) and Shu and Richards[27] (1992) derived a recursive formula to obtain these coefficients directly and irrespective of the number and positions of the sampling points. In their approach, they used the Lagrange polynomials as the trial functions and found a simple recurrence formula for the weighting coefficients. Bert et al.[7] (1993) and Striz et al.[29] (1995) developed the differential quadrature method, which uses harmonic functions instead of polynomial as test function in the quadrature method to handle periodic problems efficiently, and also circumvented the limitation for the number of grid point in the conventional DQM based on polynomial test function. Their study shows that the proper test functions are essential for the computational efficiency and reliability of the DQM. Shu et al.[25] (2001) presented a numerical study of natural convection in an eccentric annulus between a square outer cylinder and a circular inner cylinder using DQM, by using Fourier series expansion as the trial functions to compute weighting coefficients. Krowiak[18] (2008) studied the methods that based on the differential quadrature in vibration analysis of plates, and using the spline functions as the trial functions to compute weighting coefficients. Korkmaz et al.[17] (2011) used the quartic B-spline differential quadrature method, and applied it on the one-dimensional Burger's equation, by using the quartic B-spline functions as the trial functions to compute weighting coefficients. Meral[19] (2013) found the differential quadrature solution of heat- and mass-transfer equations to show the applicability of DQM space- Runge Kutta method time procedure for the one- and two-dimensional problems by using the Lagrange polynomials as the trial functions to compute weighting coefficients. Jiwari[14] (2013) used a numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation, and using the Lagrange polynomials as the trial functions to compute weighting coefficients.

A lot of researchers cared in their studies how to determine the weighting coefficients of the differential quadrature method, after reading of a lot of research and studies about the differential quadrature method. We noticed that Bernstein polynomials are incredibly useful mathematical tools as they are simply defined. They can be calculated quickly on computer systems and represent a tremendous variety of functions. They can be differentiated and integrated easily, and can be pieced together to form spline curves that can approximate any function to any accuracy desired. One of important properties to Bernstein polynomials are surely convergence. Depending on these reasons and according to our humble knowledge that the Bernstein polynomials not yet used to calculate weighting coefficients, the matter that led us to use it in this study.

In this work, we suggest Bernstein polynomials as test functions to compute the weighting coefficients of the spatial derivatives, in order to introduce a new development to the differential quadrature method that is called Bernstein differential quadrature method (BDQM). Using the BDQM for solving convection-diffusion problems excellent numerical results are obtained. Compared with other methods; the new method with a few grid points appears that it has better convergence and accuracy than the other methods in [4,11,12,13].

2- Differential quadrature method

The differential quadrature is a numerical technique used to solve the initial and boundary value problems. This method was proposed by Bellman and Casti[5] in (1971). The DQM is based on the idea that the partial derivative of a field variable at the discrete points in the computational domain is

Volume : 3 | Issue : 9 | Sept 2013 | ISSN - 2249-555X

approximated by a weighted linear sum of the values of the field variable along the line that passes though that point, which is parallel with coordinate direction of the derivative as following [3]:

$$u_x^{(m)}(x_i) = \frac{\partial^m u}{\partial x^m}\Big|_{x=x_i} = \sum_{k=1}^{N} A_{ik}^{(m)}\, u(x_k),\quad i = 1,2,\dots,N \quad ,m = 1,2,\dots,N-1 \qquad (1)$$

where are the discrete points in the variable, is the order derivative of the function, are the function values at these points, and are the weighting coefficients for the order derivative of the function with respect to and is the number of the grid points. There are two key points in the successful application of the DQM: how the weighting coefficients are determined and how the grid points are selected[20]. Many researchers have obtained weighting coefficients implicitly or explicitly using various test functions[15,17,23,25]. Quan and Chang[22] (1989a) and Shu and Richards[27] (1992), employed a set of Lagrange polynomials as the test function to determine the weighting coefficients of derivatives as.

$$A_{ik}^{(1)} = \begin{cases} \frac{1}{L}\,\frac{M^{(1)}(x_i)}{(x_i-x_k)\,M^{(1)}(x_k)} & for\ i \neq k, \\ -\sum\limits_{k=1,i\neq k}^{N} A_{ik}^{(1)}\ , & for\ i = k \end{cases} \quad ;i,k = 1,2,\dots,N \qquad (2)$$

where $M(x) = \prod_{k=1}^{N}(x - x_k)$ and $M^{(1)}(x_i) = \prod_{k=1,i\neq j}^{N}(x_i - x_k)$ and L is length interval.

And, the weighted coefficient of the second order derivative can be obtained as:

$$\left[A_{ik}^{(2)}\right] = \left[A_{ik}^{(1)}\right]\left[A_{ik}^{(1)}\right] = \left[A_{ik}^{(1)}\right]^2 \qquad (3)$$

We can obtain formulas for higher order derivatives by using the higher order weighting coefficients, which are expressed to avoid confusion.

$$A_{ik}^{(m)} = \begin{cases} m\left(A_{ii}^{(m-1)}\,A_{ik}^{(1)} - \frac{A_{ik}^{(m-1)}}{(x_i-x_k)}\right) for\ i \neq k \\ \qquad 2 \le m \le N-1 \quad ;i,k = 1,2,\dots,N \\ -\sum\limits_{k=1,i\neq k}^{N} A_{ik}^{(m)} \qquad for\ i = k \end{cases} \qquad (4)$$

with the same approach, one can derive quadrature along of multi-variables functions.

3- Bernstein differential quadrature method (BDQM)

A Bernstein polynomial, named after Sergei Natanovich Bernstein, is a polynomial in the Bernstein form, that is a linear combination of Bernstein basis polynomials. The Bernstein basis polynomials of -degree are defined on the interval by Singh et al.[28]:

$$B_{k,n}(x) = \binom{n}{k}\, x^k(1-x)^{n-k}, \quad 0 \le k \le n \qquad (5)$$

The general form of Bernstein polynomials of -degree that used to solve differential equation[8,21] are defined on the interval as:

$$B_{k,n}(x) = \binom{n}{k}\,\frac{x^k(L-x)^{n-k}}{L^n}\ , \qquad 0 \le k \le n \qquad (6)$$

were binomial coefficients are given by :

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!} \qquad (7)$$

There are n+1,n^th -degree Bernstein polynomials. For mathematical convenience, we usually set, $B_{k,n}(x)=0$, if k<0 or k>n. These polynomials are quite easy to write down the coefficients that can be obtained from Pascal's triangle. It can easily be shown that each of the Bernstein polynomials is positive and also the sum of all the Bernstein polynomials is unity for all real x∈[0,1] i.e

$$\sum_{k=0}^{n} B_{k,n}(x) = 1 \quad \forall\, x \in [0,1]$$

The Bernstein polynomials can be written to any interval as following[28]:

$$b(x) \cong B_n(f)(x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right) B_{k,n}(x) \qquad (8)$$

were f(k/n) is arbitrary function, for k=0,1,…,n, n≥1. Similar to Lagrange differential quadrature method LDQM to determined weighting coefficients, we can derive the explicit formulation to compute the weighting coefficients A_{ik}^(1) by using Bernstein polynomial as a test functions, which are listed below:

$$\tilde{A}_{ik}^{(1)} = \begin{cases} \frac{1}{L}\,\frac{b^{(1)}(x_i)}{(x_i-x_k)\,b^{(1)}(x_k)} & for\ i \neq k, \\ -\sum\limits_{k=1,i\neq k}^{N} \tilde{A}_{ik}^{(1)}\ , & for\ i = k \end{cases} \quad ;i,k = 1,2,\dots,N \qquad (9)$$

were is length interval [0,L] and

$$b(x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right)\binom{n}{k}\frac{x^k(L-x)^{n-k}}{L^n}\ and\ b^{(1)}(x_i) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right)\binom{n}{k}\frac{(x_i)^k(L-x_i)^{n-k}}{L^n}$$

The weighted coefficients of the second order derivative by using Bernstein polynomial as a test functions can be obtained as

$$\left[\tilde{A}_{ik}^{(2)}\right] = \left[\tilde{A}_{ik}^{(1)}\right]\left[\tilde{A}_{ik}^{(1)}\right] = \left[\tilde{A}_{ik}^{(1)}\right]^2 \qquad (10)$$

The same technique used in the equation (4), can be obtained from weighting coefficients $A\epsilon_{ik}^{(m)}$. In this work, we can calculate the m[th]-order spatial derivatives $A\epsilon_{ik}^{(m)}$ with respect to χ by using Bernstein polynomials.

4- Numerical examples and discussion

In this section , we apply BDQM on three test problems to demonstrate the efficiency of the BDQM. These examples are chosen such that their exact solutions are known.

Problem 1. (Djidjeli et al.[13])

Consider the unsteady state one-dimensional convection-diffusion equation:

$$\frac{\partial u}{\partial t}(x,t) + \beta_x \frac{\partial u}{\partial x}(x,t) = \alpha_x\,\frac{\partial^2 u}{\partial x^2}(x,t) \qquad (x,t) \in [0,L] \times [0,T] \qquad (11)$$

where u(x,t) is a transported variable , β_x is arbitrary constant which show the speed of convection and the diffusion coefficient is positive.

We can approximate the partial derivatives with respect to spatial variable of the one-dimension unsteady state convection-diffusion equation (11) by using BDQM to obtain the system of ordinary differential equations as:

$$\frac{\partial u}{\partial t}\Big|_{ij}^{n} = \alpha_x \sum_{k=1}^{N} \tilde{A}_{ik}^{(2)}\, u_k^n - \beta_x \sum_{k=1}^{N} \tilde{A}_{ik}^{(1)}\, u_k^n \qquad (12)$$

where and are the weighting coefficients of the first and second derivatives with respect to and calculating by Equs. (9) and (10). Approximating the first-order derivatives with respect to the temporal variable by using the forwarded differences and then arranging the terms of Equation (12), we obtain the system of algebraic equations as:

$$u_i^{n+1} = u_i^n + \Delta t\left[\alpha_x \sum_{k=1}^{N} \tilde{A}_{ik}^{(2)}\, u_k^n - \beta_x \sum_{k=1}^{N} \tilde{A}_{ik}^{(1)}\, u_k^n\right] \qquad (13)$$

Equ. (11) has the initial condition:

$$u(x,0) = a\, e^{-xc}\ , \qquad 0 \le x \le 1 \qquad (14)$$

where

$$c = \frac{\beta_x \pm \sqrt{\beta_x^2 + 4\alpha_x b}}{2\alpha_x} > 0$$

and the boundary conditions

$$u(0,t) = ae^{bt} \quad , \qquad u(1,t) = ae^{bt-c}, \qquad t>0 \tag{15}$$

The exact solution is given as:

$$u(x,t) = ae^{bt-x\,c} \quad , \qquad 0 \le x \le 1\,, \qquad t>0 \tag{16}$$

In this problem, we take take $\alpha_x=\beta_x=1, L=1,\ a=1, b=0.1$ and $\Delta t=0.00001$ and use equally space grid points. Table 1 are shows the errors obtained from solving problem 1 by using LDQM and BDQM at (t=0.01 and 1) and $x\in[0,1]$ for different values of h=1/(N-1). Fig. 1 is clarify a comparison between exact solution and numerical solutions of the problem 1. The results confirm that the BDQM has a higher accuracy, good convergence compare with LDQM.

Table 1. Errors obtained for problem 1 with

h	Max error of LDQM				Max error of BDQM			
	t=0.01	CPU	t=1	CPU	t=0.01	CPU	t=1	CPU
0.25	5.537E-07	0.469	1.000E-03	18.99	4.171E-07	0.421	9.974E-04	18.86
0.17	6.643E-07	0.509	9.999E-04	22.43	1.345E-07	0.496	9.663E-04	21.71
0.125	7.275E-07	0.611	3.464E-03	29.07	6.456E-08	0.567	2.671E-03	28.51
0.1	7.684E-07	0.689	8.949E-03	43.94	8.087E-08	0.645	3.735E-03	43.40

We choose the arbitrary function and is the arbitrary constant. In this problem, we will take at the number of grid points respectively. Notice that in the next examples, we chosen the same above arbitrary function with different values of .

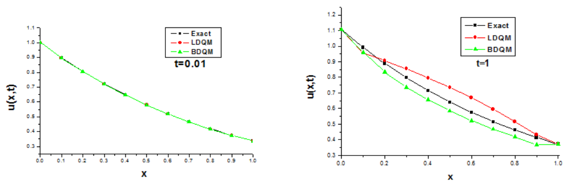


Fig.1 Exact and approximate solutions of the problem 1 with,

Problem 2 (Dehghan and Mohebbi[12])
Consider the unsteady state two-dimensional convection-diffusion equation:

$$\frac{\partial u}{\partial t}(x,y,t) + \beta_x \frac{\partial u}{\partial x}(x,y,t) + \beta_y \frac{\partial u}{\partial y}(x,y,t) - \alpha_x \frac{\partial^2 u}{\partial x^2}(x,y,t) - \alpha_y \frac{\partial^2 u}{\partial y^2}(x,y,t) = 0\,, \qquad (x,y,t) \in [0,L]\times[0,L]\times[0,T] \tag{17}$$

where is a transported variable , and are arbitrary constants which show the speed of convection and the diffusion coefficients and are positive constants. We can approximate the partial derivatives with respect to spatial variable of the two-dimension unsteady state convection-diffusion equation (17) by using BDQM to obtain the system of ordinary differential equations as:

$$\left.\frac{\partial u}{\partial t}\right|_{ij}^{n} = \sum_{k=1}^{N} (\alpha_x \tilde{A}_{ik}^{(2)} - \beta_x \tilde{A}_{ik}^{(1)}) u_{kj}^{n} + \sum_{l=1}^{M} (\alpha_y \tilde{B}_{jl}^{(2)} - \beta_y \tilde{B}_{jl}^{(1)}) u_{il}^{n} \tag{18}$$

where are the weighting coefficients of the first and second spatial derivatives with respect to and and calculating by Equs. (9) and (10). By approximating the first-order derivatives with respect to the temporal variable by using the forward differences and then arranging the terms of Equation (18), we obtain the system of algebraic equations as:

$$u_{ij}^{n+1} = u_{ij}^{n} + \Delta t \left[\sum_{k=1}^{N} (\alpha_x \tilde{A}_{ik}^{(2)} - \beta_x \tilde{A}_{ik}^{(1)}) u_{kj}^{n} + \sum_{l=1}^{M} (\alpha_y \tilde{B}_{jl}^{(2)} - \beta_y \tilde{B}_{jl}^{(1)}) u_{il}^{n} \right] \tag{19}$$

The initial condition of Equation (17) has the following form:

$$u(x,y,0) = a(e^{-x\,c_x} + e^{-y\,c_y}) \quad , \quad 0 \le x,y \le 1 \tag{20}$$

where

$$c_x = \frac{-\beta_x \pm \sqrt{\beta_x^2 + 4b\alpha_x}}{2\alpha_x} > 0 \quad , \qquad c_y = \frac{-\beta_y \pm \sqrt{\beta_y^2 + 4b\alpha_y}}{2\alpha_y} > 0$$

and the boundary conditions are given by:

$$\left.\begin{aligned} &u(0,y,t) = ae^{bt}(1+e^{-y\,c_y}), u(1,y,t) = ae^{bt}(e^{-c_x}+e^{-y\,c_y}) \\ &u(x,0,t) = ae^{bt}(e^{-x\,c_x}+1), u(x,1,t) = ae^{bt}(e^{-x\,c_x}+e^{-c_y}) \ ,t>0 \end{aligned}\right\} \tag{21}$$

The exact solution is given as:

$$u(x,y,t) = ae^{bt}(\,e^{-x\,c_x} + e^{-y\,c_y}) \quad , \quad 0 \le x,y \le 1\,, \quad t>0 \tag{22}$$

In this problem, we take , respectively, , and use equally spaced grid points. Tables 2 and 3 are shows the errors obtained from solving problem 2 by using LDQM and BDQM at and for different values of . Figs.2 and 3 are clarify a comparisons between exact solution and numerical solutions for and respectively. The results confirm that the BDQM more accuracy and less CPU time than the LDQM.

Table 2 Errors obtained for problem 2 with

h	Max of LDQM	CPU		Max of BDQM	CPU
0.25	2.912711E-08	0.514		3.3.735268E-13	0.508
0.17	2.108340E-08	0.615		4.105996E-18	0.607
0.125	1.371521E-08	0.770		5.272405E-15	0.763
0.1	7.726773E-09	1.105		5.703941E-15	1.086

Table 3 Errors obtained for problem 2 with

h	Maxof LDQM	CPU		Max of BDQM	CPU
0.25	2.572117E-06	0.509		3.7.073172E-09	0.503
0.17	1.379575E-05	0.619		1.895764E-08	0.610
0.125	3.223357E-05	0.772		3.894741E-08	0.757
0.1	5.348801E-05	1.047		1.094325E-07	1.015

In this problem, we take for Table 2. and for Table 3. respectively at the number of grid points

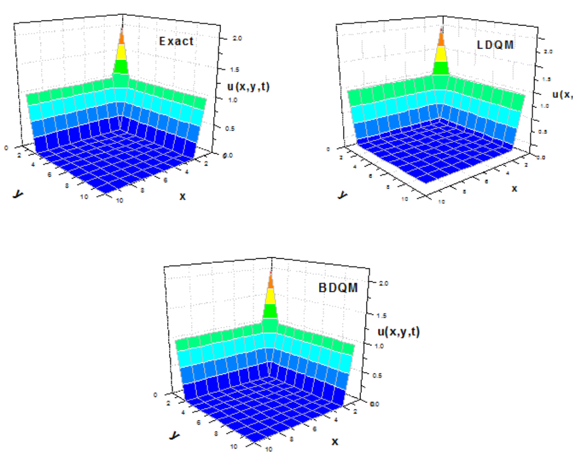
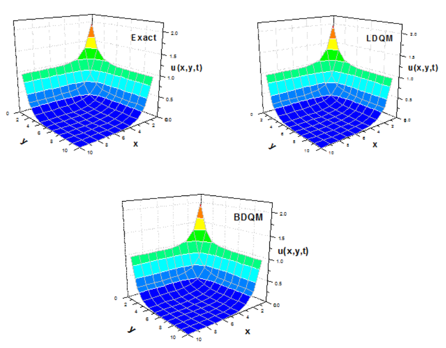


Fig. 2 Exact and approximate solutions of the problem 2 with,



Volume : 3 | Issue : 9 | Sept 2013 | ISSN - 2249-555X

Fig. 3 Exact and approximate solutions of the problem 2 with, , and

Problem 3 (Al-Saif and Al-kanani[4])

Consider the two-dimensional Burger's Equation:

$$\frac{\partial u}{\partial t}+\alpha u\frac{\partial u}{\partial x}+\alpha v\frac{\partial u}{\partial y}-\frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)=0 \qquad (23)$$

$$\frac{\partial v}{\partial t}+\alpha u\frac{\partial v}{\partial x}+\alpha v\frac{\partial v}{\partial y}-\frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)=0 \qquad (24)$$

where are the velocity components to be determined, is constant and is the Reynolds number, We can approximate the partial derivatives with respect to spatial variable of the two dimensional Burger's Equation (23) and (24) by using BDQM to obtain the system of ordinary differential equations as:

$$\left.\frac{\partial u}{\partial t}\right|_{ij}^{n}=\frac{1}{Re}\left(\sum_{k=1}^{N}\tilde{A}_{kj}^{(2)}u_{kj}^{n}+\sum_{l=1}^{M}\tilde{B}_{jl}^{(2)}u_{il}^{n}\right)-u_{ij}^{n}\sum_{k=1}^{N}\alpha\tilde{A}_{ik}^{(1)}u_{kj}^{n}-v_{ij}^{n}\sum_{l=1}^{M}\alpha\tilde{B}_{jl}^{(1)}u_{il}^{n} \qquad (25)$$

$$\left.\frac{\partial v}{\partial t}\right|_{ij}^{n}=\frac{1}{Re}\left(\sum_{k=1}^{N}\tilde{A}_{kj}^{(2)}v_{kj}^{n}+\sum_{l=1}^{M}\tilde{B}_{jl}^{(2)}v_{il}^{n}\right)-u_{ij}^{n}\sum_{k=1}^{N}\alpha\tilde{A}_{ik}^{(1)}v_{kj}^{n}-v_{ij}^{n}\sum_{l=1}^{M}\alpha\tilde{B}_{jl}^{(1)}v_{il}^{n} \qquad (26)$$

where are the weighting coefficients of the first and second spatial derivatives with respect to and and calculating by Equs. (9) and (10). By approximating the first-order derivatives with respect to the temporal variable by using the forwarded differences and then arranging the terms of Equations (25) and (26), we obtain the system of algebraic equations as:

$$u_{ij}^{n+1}=u_{ij}^{n}+\Delta t\left[\sum_{k=1}^{N}\left(\frac{1}{Re}\tilde{A}_{ik}^{(2)}-\alpha u_{ij}^{n}\tilde{A}_{ik}^{(1)}\right)u_{kj}^{n}+\sum_{l=1}^{M}\left(\frac{1}{Re}\tilde{B}_{jl}^{(2)}-\alpha v_{ij}^{n}\tilde{B}_{jl}^{(1)}\right)u_{il}^{n}\right] \qquad (27)$$

$$v_{ij}^{n+1}=v_{ij}^{n}+\Delta t\left[\sum_{k=1}^{N}\left(\frac{1}{Re}\tilde{A}_{ik}^{(2)}-\alpha u_{ij}^{n}\tilde{A}_{ik}^{(1)}\right)v_{kj}^{n}+\sum_{l=1}^{M}\left(\frac{1}{Re}\tilde{B}_{jl}^{(2)}-\alpha v_{ij}^{n}\tilde{B}_{jl}^{(1)}\right)v_{il}^{n}\right] \qquad (28)$$

In this problem, we take , and the initial conditions of Equations (25) and (26) have the following form:

$$u(x,y,0)=\frac{1}{2}-\frac{x+y}{1+x+y}, \qquad v(x,y,0)=\frac{1}{2}+\frac{x+y}{1+x+y} \qquad (29)$$

The exact solutions are given by:

$$u(x,y,t)=\frac{1}{2}-\frac{x+y+t}{1+x+y+t}, \qquad v(x,y,t)=\frac{1}{2}+\frac{x+y+t}{1+x+y+t} \qquad (30)$$

The boundary conditions can be achieved easily from (30) by using For the above problem, we found numerical solutions for and and use equally spaced grid points. Tables 4 and 5 shows the errors obtained in solving problem 3 with the BDQM and LDQM at and for different values of Fig. 4 is clarify a comparison between exact solution and numerical solutions of the problem 3. The results showed that the BDQM has a high accuracy, good convergence and less CPU time comparing with the LDQM.

Table 4. Error obtained by LDQM and BDQM for problem 3 with and

	Max of u				Max of v			
h	LDQM	CPU	BDQM	CPU	LDQM	CPU	BDQM	CPU
0.25	1.943390E-05	0.098	1.877131E-05	0.093	1.943416E-05	0.100	1.877144E-05	0.099
0.17	3.109294E-05	0.106	2.859741E-05	0.096	3.109260E-05	0.110	2.859693E-05	0.102
0.125	4.022089E-05	0.122	3.450607E-05	0.120	4.022089E-05	0.122	3.450572E-05	0.118
0.1	4.733133E-05	0.151	3.707231E-05	0.132	4.733010E-05	0.137	3.707123E-05	0.128

Table 5. Error obtained by LDQM and BDQM for problem 3 with and

h	Maxof u				Max of v			
	LDQM	CPU	BDQM	CPU	LDQM	CPU	BDQM	CPU
0.25	1.637199E-03	0.475	1.001154E-03	0.456	1.637175E-03	0.487	1.001153E-03	0.477
0.17	2.612820E-03	0.558	9.961354E-04	0.543	2.612838E-03	0.554	9.961241E-04	0.547
0.125	3.405341E-03	0.703	9.925080E-04	0.677	3.405345E-03	0.709	9.925192E-04	0.698
0.1	4.022784E-03	0.947	9.934846E-04	0.945	4.021884E-03	0.978	9.934885E-04	0.968

In this problem, we take and respectively at the number of grid points

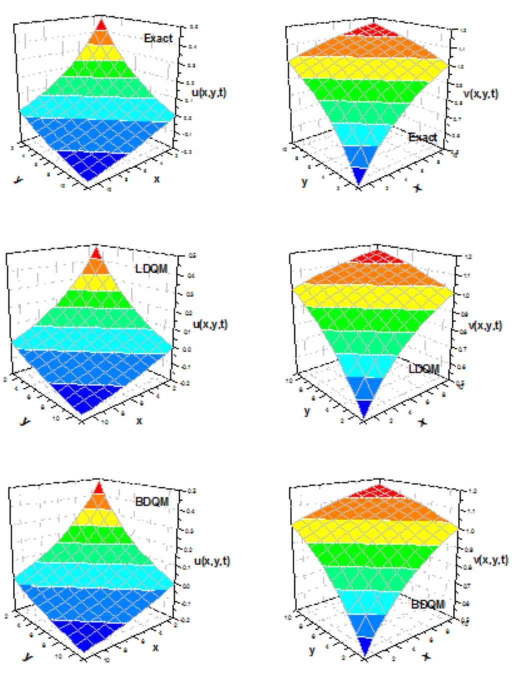


Fig. 4 Exact and approximate solutions of the problem 3 with, t=0.01 andt=0.0001.

5- Comparison with the other methods

We compare the numerical results of BDQM for problems 1, 2 and 3 with the results of other numerical methods such as LDQM, compactly supported radial basis function (RBF)[13], High-order compact boundary value method (HOCBVM)[12], Radial basis function based meshless method (RBFBMM)[11], and Alternating direction implicit formulation of the differential quadrature method (ADI-DQM)[3,4]. The error measurements resulted from the BDQM is more accurate than the methods LDQM, RBF[13], HOCBVM[12], RBFBMM[11] and ADI-DQM[3,4]. Moreover, the number of grid points by using BDQM and LDQM are less than the other methods.

Table 6. Comparison of the numerical results of the problem 1 for different methods at

Method	Number grid points	Max of u
BDQM		3.510230E-03
LDQM		8.278487E-03
RBF		4.413E-01

Table 7. Comparison of the numerical results of the problem 2 for different methods at and

Method	Number grid points	Maxof u
BDQM		5.703941E-15
ADI-DQM[3]		4.110175E-13
LDQM		7.726773E-09

HOCBVM		9.4696E-04
RBFBMM		4.97E-02

Table 8. Comparison of the numerical results of the problem 2 for different methods at and

Method	Number grid points	Max of u
BDQM		1.094325E-07
ADI-DQM[3]		1.131288E-05
LDQM		5.348801E-05
RBFBMM		4.97E-02

Table 9. Comparison of the numerical results of the problem 3 for different methods at .

Method	Number grid points	Max of u	Max of v
BDQM		1.877131E-05	1.877144E-05
LDQM		1.943390E-05	1.943416E-05
ADI-DQM		1.308369E-05	2.824992E-05

6- Stability analysis of BDQM

The stability of numerical schemes is closely related to numerical error. A solution is said to be unstable if errors appear at some stage in the calculations (for example, from erroneous initial conditions or local truncation or round-off errors) are propagated without bound throughout subsequent calculations. Thus a method is stable if small changes in the initial data produce correspondingly small changes in the final results, that is, the difference between the theoretical and numerical solutions remains bounded at a given time t, as time and space steps tend to zero or time step remains fixed at every level and [2]. So stability, means that the numerical solution must be close to the exact solution, meaning that whenever was the error a little the deviation in derivatives, however, this error may accumulate at each time step and affects to the stability of the solution.

Theorem[24]

The system of ODE with a constant coefficient matrix is,

(1) Stable if the roots of the characteristic polynomial are purely imaginary.
(2) Asymptotically stable if the roots have negative real parts.
(3) Unstable if a root has positive real part.

From application of BDQM to the any convection-diffusion equation in this work, we obtained the set ordinary differential equations:

$$[A]\{u\} = \{b\} - \{r\} \tag{31}$$

where is a vector of unknown functional values at all the interior points given by

$$\{u\} = \left[u_{2,2}, u_{2,3}, \dots, u_{2,M-1}, u_{3,2}, \dots, u_{3,M-1}, \dots, u_{N-1,2}, \dots, u_{N-1,M-1}\right]^T$$

and is a known vector which is made up of the non-homogeneous part and the boundary conditions given by

$$\{r\} = \left[r_{2,2}, r_{2,3}, \dots, r_{2,M-1}, r_{3,2}, \dots, r_{3,M-1}, \dots, r_{N-1,2}, \dots, r_{N-1,M-1}\right]^T$$

and is the coefficient matrix containing the weighting coefficients, the dimension of the matrix is by . For the multi-dimensional case, the matrix contains many zero elements, which are irregularly distributed in the matrix.

The stability analysis of the Equation (31) is based on the eigenvalue distribution of the BDQM discretization matrix . If has eigenvalues and corresponding eigenvector , being the size of the matrix, the similarity transformation reduces the system (31) of the form[1,10,26].

$$\frac{d\{U\}}{dt} = [D]\{U\} + \{R\} \tag{32}$$

Here the diagonal matrix [D] is formed from the eigenvalues and from a nonsingular matrix [P] containing the eigenvectors as columns

$$[D] = [P]^{-1}[A][P] \tag{33}$$

Pre-multiplying by the matrix on the both sides Equation (32) and setting

$$\{U\} = [P]^{-1}[u] \tag{34}$$
$$\{R\} = [P]^{-1}[r] \tag{35}$$

Since is a diagonal matrix, Equation (32) is an uncoupled set of ordinary differential equations.
Considering the equation of (32)

$$\frac{dU_i}{dt} = \lambda_i U_i + R_i \tag{36}$$

If is time-independent, then the solution of Equation (36) can be written as

$$U_i = \left(U_i(0) + \frac{R_i}{\lambda_i}\right)e^{\lambda_i t} - \frac{R_i}{\lambda_i} \tag{37}$$

For this case, using Equations (34) and (35), the solution can be obtained as

$$\{u\} = [P][U] = \sum_{i=1}^{N} U_i\,\xi_i = \sum_{i=1}^{N}\left[U_i(0)e^{\lambda_i t} + \frac{R_i}{\lambda_i}\left(e^{\lambda_i t} - 1\right)\right]\xi_i \tag{38}$$

Clearly, the stable solution of $\{u\}$ when $t \to \infty$ requires

$$Real(\lambda_i) \le 0 \quad \text{for all} \quad i \tag{39}$$

where denotes the real part of . This is the stability condition for the system (31).

In this section, we can applied the stability condition (39) on the problems that mentioned in the previous section by using BDQM.

Problem 1.

From the application of BDQM to the Equation (11) and using ,, Equation (11) can be rewritten as:

$$\sum_{k=2}^{N-1} \bar{A}_{ik}^{(2)} u_k - \sum_{k=2}^{N-1} \bar{A}_{ik}^{(1)} u_k = \{b\} - \{r\} \tag{40}$$

where $2 \le i \le N-1$

From Equation (40), we can obtain a system of algebraic equations (31).

This system has the solution (38), and this solution is stable as and the real parts of the eigenvalues of the matrix are:

$Real(\lambda_1) = -17.920, Real(\lambda_2) = -58.152, Real(\lambda_3) = -95.494$

This means that the stability condition (39) is hold.

Problem 2.

From the application of BDQM to the Equation (18) and using , and , Equation (18) can be rewritten as:

$$\sum_{k=2}^{4}(\alpha_x \bar{A}_{ik}^{(2)} - \beta_x \bar{A}_{ik}^{(1)})u_{kj}^n + \sum_{l=2}^{4}(\alpha_y \bar{B}_{jl}^{(2)} - \beta_y \bar{B}_{jl}^{(1)})u_{il}^n = \{b\} - \{r\} \tag{41}$$

where $2 \le i \le 4$, $2 \le j \le 4$

From Equation (41), we can obtain a system of algebraic equations (31).

This system has the solution (38), and this solution is stable as and the real parts of the eigenvalues of the matrix for respectively, are:

$Real(\lambda_1) = -0.231, Real(\lambda_2) = -0.231, Real(\lambda_3) = 0.028,$
$Real(\lambda_4) = -0.103, Real(\lambda_5) = -0.103, Real(\lambda_6) = -0.044,$
$Real(\lambda_7) = -0.044, Real(\lambda_8) = -0.231, Real(\lambda_9) = -0.231.$

and

$Real(\lambda_1) = -5.283, Real(\lambda_2) = -5.283, Real(\lambda_3) = -1.135,$
$Real(\lambda_4) = -2.940, Real(\lambda_5) = -2.940, Real(\lambda_6) = -4.881,$
$Real(\lambda_7) = -5.110, Real(\lambda_8) = -4.178, Real(\lambda_9) = -4.178.$

This means that the stability condition (39) is hold.

Problem 3.

From the application of BDQM to the Equation (23) and us-

Volume : 3 | Issue : 9 | Sept 2013 | ISSN - 2249-555X

ing , and , Equation (23) can be rewritten as:

$$\sum_{k=2}^{4}(\frac{1}{Re}\tilde{A}_{ik}^{(2)} - \alpha u_{ij}^{n}\tilde{A}_{ik}^{(1)})u_{kj}^{n} + \sum_{l=2}^{4}\left(\frac{1}{Re}\tilde{B}_{jl}^{(2)} - \alpha v_{ij}^{n}\tilde{B}_{jl}^{(1)}\right)u_{il}^{n} = \{b\} - \{r\} \quad (42)$$

where $2 \le i \le 4$, $2 \le j \le 4$

From Equation (42), we can obtain a system of algebraic equations (31).

This system has the solution (38), and this solution is stable as and the real parts of the eigenvalues of the matrix are:

$$Real(\lambda_1) = -9.207, Real(\lambda_2) = -4.922, Real(\lambda_3) = -4.750,$$
$$Real(\lambda_4) = -4.476, Real(\lambda_5) = -4.476, Real(\lambda_6) = -0.001,$$
$$Real(\lambda_7) = -0.048, Real(\lambda_8) = -0.189, Real(\lambda_9) = -0.141.$$

This means that the stability condition (39) is hold. When using the equation (24), we will find the same eigenvalues mentioned above of the matrix [A].

Finally, the numerical results of the above problems confirm that the newly developed method BDQM is stable for the grid points . In this work, with the help of symbolic computation software Maple 13, the eigenvalues are computed.

7- Conclusions

In this work, we employed a new technique BDQM to solve convection-diffusion equations successfully. The weighting coefficients for spatial derivatives are computing by use Bernstein polynomials as test functions. The numerical results show that the new method has higher accuracy, good convergence and reasonable stability as well as a less computation workload by using few grid points.

REFERENCE

[1] M. Akman, Differential quadrature method for time-dependent diffusion equation, M. Sc. Thesis, Graduate School of the Middle east, Technical University,(2003). | [2] A. Ali, Mesh free collocation method for numerical solution of initial-boundary-value problems using radial basis functions, Ph.D. thesis, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan, (2009). | | [3] A. S. J. Al-Saif and M. J. Al-kanani, Alternating direction implicit formulation of the differential quadrature method for solving the unsteady state two-dimensional convection-diffusion equation, Gen. Math. Notes, 7(2011), 41-50. | [4] A. S. J. Al-Saif and M. J. Al-kanani, Alternating direction implicit formulation of the differential quadrature method for solving Burger equations, Inter. J. Modern Math. Sc. 30(1)(2012), 1-11. | [5] R. Bellman and J. Casti, Differential quadrature and long-term integration, J. Math. Analy. Appl., 34(1971), 235-238. | [6] 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(1972), 40-52. | | [7] C. W. Bert, X. Wang and A. G. Striz, "Differential quadrature for static and free vibrational analysis of anisotropic plates", Int. J. Solids Structures, 30(1993),1737-1744. | [8] M. I. Bhatti and P. Bracken, Solution of differential equations in a Bernstein polynomial basis, J. Comput. Appl. Math., 205(2007), 272-280. | [9] J. Chan and J. A. Evans, A minimum-residual finite element method for the convection-diffusion equation, Institute for Comput. Engineering and Sciences ICES, 13(13)(2013), 1-19. | | [10] W. Chen," Differential quadrature method and its applications in engineering ", Ph. D. thesis, Shanghai JiaoTong University, China, (1996). | [11] P.P. Chinchapatnam, Radial basis function based meshless methods for fluid flow problems, Ph.D. thesis, Southampton University, Faculty of Engineering Science and Mathematics, UK, (2006). | [12] M. Dehghan and A. Mohebbi, High-order compact boundary value method for the solution of unsteady convection-diffusion problems, Math. Comput. Simul.,79(2008), 683-699. | [13] K. Djidjeli,P. P. Chinchapatnam,P. B. Nair and W. Price, Global and compact meshless schemes for the unsteady convection-diffusion equation, In Proceedings of the International Symposium on Health Care and Biomedical Interaction, 8 page CDROM, (2004). | [14] R. Jiwari, R. c. Mittal and K. K. Sharma, A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burger's equation, Appl. Math. Comput. 219(2013), 6680-6691. | [15] B. Kaya and Y. Arisoy, Differential quadrature solution for one-dimensional aquifer flow, Math. Comput. Appl., 16(2)(2011), 524-534. | [16] B. Kaya, Solution of the advection-diffusion equation using the differential quadrature method , KSCE J. civil Engineering, (1)(2009), 69-75. | [17] A. Korkmaz,A. M. Aksoy and I. Dag, Quartic B-spline differential quadrature method, International J. of Nonlinear Sc. 11(2011),403-411. | [18] A. Krowiak, methods based on the differential quadrature in vibration analysis of plates, J. Theor. Appl. Mech. 46(1)(2008), 123-139. | | [19] G. Meral, Differential quadrature solution of heat-and mass-transfer equations, Appl. Math. Modelling 37(2013),4350-3459. | [20] M. Mozaffari , A. A. Atai and N. Mostafa, Large deformation and mechanics of flexible isotropic membrane ballooning in three dimensions by differential quadrature method, J. Mech. Sc. Tech. 23(2009), 2921-2927. | [21] Y. Ordokhani and S. D. Far, Application of the Bernstein polynomials for solving the nonlinear Fredholm Integro-differential equations, J. Appl. Math. And Bioinformatics,1(2011), 13-31. | [22] J. Quan and C. Chang, New insights in solving distributed system equations by the quadrature methods - I, Comput. Chem. Engng, 13(1989a), 779-788. | | [23] M. Ramezani, M. Shahrezaee, H. Kharazi and L. H. Kashany, Numerical solutions of differential algebraic equation by differential quadrature method, J. Basic. Appl. Sc. Res, 2(11)(2012), 11821-11828. | [24] D.A. Sanchez, Ordinary Differential Equations and Stability Theory: An Introduction, W.H. Freeman and Company, USA, (1968). | | [25] C. Shu , H. Xue and Y. D. Zhu, Numerical study of natural convection in an eccentric annulus between a square outer cylinder and a circular inner cylinder using DQ method, Int. J. Heat and Mass Transfer, 44(2001), 3321-3333. | [26] C. Shu, Differential Quadrature and Its Application in Engineering, springer-verlag, London, (2000). | [27] C. Shu and B. Richards, Application of generalized differential quadrature to solve two-dimensional incompressible Navier-Stokes equations, Int. J. Numer. Meth. In Fluids,15(1992), 791-798. | | [28] A. K. Singh, V. K. Singh and O. P. Singh, The Bernstein operational matrix of integration, Appl. Math. Sc. 3(49)(2009), 2427-2436. | [29] A. G. Striz, X. Wang and C. W. Bert, "Harmonic differential method and applications to structural components", Acta Mechanica, 111(1995), 85-94. | | [30] L. Su and P. Cheng, A weighted average finite difference method for the fractional convection-diffusion Equation, Advances in Mathematical Physics, Article ID 129404(2013),1-5. | | [31] M. Tanaka and Chen, Coupling dual reciprocity BEM and differential quadrature method for time-dependent diffusion problems, Appl. Math. Modeling 25(2001),257-267. | [32] H. Zhu, H. Shu and M. Ding, Numerical solutions of two-dimensional Burger's equations by discrete Adomian decomposition method, Comput. Math. Appl. 60(2010), 840-848. |