Computers and Mathematics with Applications 73 (2017) 1576–1592

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Spline-based DQM for multi-dimensional PDEs: Application to biharmonic and Poisson equations in 2D and 3D

M. Ghasemi

Department of Mathematics, University of Kurdistan, P.O. Box 416, Sanandaj, Iran

ARTICLE INFO

Article history:
Received 28 September 2016
Received in revised form 31 December 2016
Accepted 5 February 2017
Available online 23 February 2017

Keywords:
Differential quadrature method
B-spline
Multi-dimensional PDEs
Poisson equation
Biharmonic equation

ABSTRACT

The idea of differential quadrature is used to construct a new algorithm for the solution of differential equations. To determine the weighting coefficients of DQM, B-spline basis functions of degree r are used as test functions. The method is constructed on a set of points mixed from grid points and mid points of a uniform partition. Using the definition of B-splines interpolation as alternative, some error bounds are obtained for DQM. The method is successfully implemented on nonlinear boundary value problems of order m. Also the application of the proposed method to approximate the solution of multi-dimensional elliptic PDEs is included in the paper. As test problem, some examples of biharmonic and Poisson equations are solved in 2D and 3D. The results are compared with some existing methods to show the efficiency and performance of the proposed algorithm. Also some examples of time dependent PDEs are solved to compare the results with other existing spline based DQ methods.

1. Introduction

The biharmonic problem is a fourth order partial differential equation which arises in modeling some complex phenomena in sciences and engineering. It plays an important role in different scientific disciplines, but it is difficult to solve due to the existing fourth order derivatives. It arise in several areas of mechanics such as two dimensional theory of elasticity and the deformation of elastic and elasto-plastic plates [1,2]. The term biharmonic is used to indicate the fact that, the function describing such a processes should satisfies the Laplace's equation twice explicitly. Many of the applications of biharmonic equation stem from the consideration of mechanical and physical processes involving solids and fluids. One of the earliest applications of the biharmonic equation deals with the classical theory of flexure of elastic plates which was studied by J. Bernoulli, Euler, Lagrange, Poisson and some other scientists. The mathematical theory of thin plates which is generally attributed to Poisson and Kirchhoff, has been extensively applied to the stress analysis of structural plates composed of both metallic and non-metallic materials. In two dimensional space, the biharmonic equation can be used to model the deflections arising in rectangular orthotropic symmetric laminate plates. Also there are many other physical problems such as bending the clamped thin elastic isotropic plates, equilibrium of an elastic body under conditions of plane strain or plane stress, or creeping flow of a very viscous incompressible fluid, which can be formulated in terms of the two-dimensional biharmonic equation for one scalar function with prescribed values of the function and its normal derivative at the boundary [1]. The reduction of the analysis of two-dimensional problem in the classical theory of elasticity to the solution of the biharmonic equation is due to Airy, who used the calculations in the design of a structural support system for an astronomical telescope. The work of Green, Stokes, Kelvin and others in the areas relating to three-dimensional

E-mail address: mohamad.ghassemi@gmail.com.

http://dx.doi.org/10.1016/j.camwa.2017.02.006
0898-1221/

problems in the mathematical theory of elasticity also touch upon formulations which involve the biharmonic operator. These developments essentially lay the foundations to the study of the mathematical theory of elasticity which forms an important aspect of the mechanics of deformable media. There exist useful books devoted to the history of theory of elasticity and strength of materials that contain a lot of interesting results on the 2D biharmonic problem [3].

The Poisson's equation, named after the mathematician Simeon Denis Poisson, is an elliptic partial differential equation with wide applications in mechanical engineering and theoretical physics. It frequently appears in many fields such as fluid dynamics [4], acoustics [5], heat transfer [6], electromagnetics [7], electrostatics mechanical engineering [8]. Due to its importance, many efforts have been made in the past few decades on studding the numerical techniques to approximate the solution of Poisson equation [9–12].

Many different numerical approaches are available to approximate the solution of differential equations which are based on discretizing the problem's domain by a set of grid points and finding the numerical solution on the sub-domains. Examples of such methods are finite difference, finite element and collocation, which convert the problem into a system of algebraic equations to approximate the solution. An application of spline-based collocation method to study the free vibration of layered cylindrical shells, laminated cross-ply plates, antisymmetric angle-ply-laminated plates and multi-layered circular cylindrical shell can be found in [13–16]. One of the efficient numerical methods which become an increasingly popular numerical technique for the rapid and efficient solution of partial differential equations is differential quadrature method (DQM). This approach was first time introduced by Bellman and co-authors for the solution of time dependent partial differential equations [17]. Due to its high accuracy and low computational efforts, the differential quadrature method became a good alternative to FDM and FEM. The differential quadrature method was initiated from the idea of integral quadrature. By defining a partition on the desired domain, the differential quadrature method approximates the function's derivatives at any grid point by a weighted linear sum of the function values at all of the points of the partition. The weighting coefficients of the linear sum can be calculated in several ways. Bellman et al. [17], introduced two algorithms to compute weighting coefficients. The first one was based on solving a simple linear system while the second one proposed some iterative formulas with grid points chosen as the roots of the shifted Legendre polynomial which was a little inefficient when the number of grid points was increased. It was because of its ill-conditioned coefficient matrix of Vandermonde type. Some efforts have been made by Civan [18], to overcome the deficiency of the first approach which was based on BP algorithm for the solution of Vandermonde systems. In [19,20], two new approaches were introduced to obtain weighting coefficients which were based on the analysis of linear vector spaces. Shu [19], obtained the first order weighting coefficients such that they did not depend on the choice of grid points. Also the higher order weighting coefficients which have been determined by a recurrence relation. Also in [21,22], some new formulations have been proposed to obtain the first and second order weighting coefficients based on Fourier series expansion. The idea of using differential quadrature method was extended to the case of cardinal spline functions by Zhong and co-authors [23–25]. In [26,27] the DQ method based on quartic and cubic B-splines respectively, was constructed to approximate the solution of nonlinear Burgers' equation. Mittal and co-authors [28–30], Korkmaz et al. [31–33], Bashan et al. [34] and many other researchers extended the idea of spline-based differential quadrature method for the solution of different kinds of boundary value problems. More recently, Barrera et al. used differential quadrature method based on spline quasi-interpolation to construct a new algorithms to approximate the derivatives of functions [35,36]. They obtained optimal convergent approximations for derivatives of sufficiently smooth functions. Also a comprehensive review on all the methods which used DQM to discretize the differential equations can be found in [37].

In the current work we propose a new spline-based differential quadrature algorithm to approximate the solution of boundary value problems. By introducing a uniform partition on the problem's domain, we construct a set of data points mixed from grid points and mid points of the partition. Then the method is constructed on the new set of points. We used a B-spline basis function of degree r to fined the weighting coefficients of the method. In order to find error bounds, we consider two cases with $r = 2s - 1$ and $r = 2s$. In each case we find an error bound based on the degree of spline and the order of the problem. Also the method is applied to approximate the solution of biharmonic problem as well as Poisson equation in multi-dimensions. Obviously the new algorithm is much more efficient and easy to use, comparing to the previous approaches.

The paper is organized as follows: In Section 2 at first we introduce the DQM briefly, then we use the B-spline basis functions to obtain weighting coefficients. In the following, using B-spline interpolation as an alternative we find some error bounds for the derivatives of the function. Section 3 is devoted the implementation of the new approach. At first the method is used to an ordinary differential equation with boundary conditions. Then we solve the biharmonic and Poisson equations respectively. In Section 4, some problems in 2D and 3D will be solved using the proposed algorithm to show the efficiency and good performance of the method computationally. Also in order to compare the method with some other existing methods, some problems of time dependent partial differential equations will be solved. Finally in the Appendix we will represent some error bounds for B-spline DQM used to approximate the solution of arbitrary order boundary value problems.

2. B-spline differential quadrature method

Let us consider the uniform partition Δ on the finite interval $[a, b]$ with step size $h = \frac{b-a}{n}$ in the following form

$$\Delta \equiv \left\{a = x_0 < x_1 < \cdots < x_n = b\right\},$$

which divide the interval into n subintervals I_i with $I_i = [x_{i-1}, x_i)$, $i = 1, \dots, n-1$, and $I_n = [x_{n-1}, x_n]$. Let Π_r be the space of polynomials of degree at most $r \geq 0$ and define

$$SP_r(\Delta) = \left\{f \in C^{r-1}[a,b] : f\big|_{I_i} \in \Pi_r,\ i = 1, 2, \dots, n\right\}.$$

Let $\tilde{\Delta} \equiv \{x_i\}_{i=-r+1}^{n+r-1}$ be the extended partition with

$$x_{-r+1} < \cdots < x_{-1} < a, \qquad b < x_{n+1} < \cdots < x_{n+r-1}.$$

It can be proved [38], that a B-spline is the unique nonzero spline of smallest compact support with the knots at $\tilde{\Delta}$. By the help of extended partition $\tilde{\Delta}$, we can find a unique B-spline basis for $SP_r(\Delta)$ in the form of $\{B_{r,i}(x)\}_{i=-\sigma}^{n+\delta}$ where $\sigma = [\frac{r-1}{2}]$ and $\delta = [\frac{r}{2}]$ and [ ] indicates the integer part. It is obvious that the dimension of $SP_r(\Delta)$ is $n + 1 + \sigma + \delta = n + r$. If we define the zero degree B-spline $B_{0,i}(x)$ as

$$B_{0,i}(x) = \begin{cases} 1 & x_i \leq x \leq x_{i+1}, \\ 0 & \text{otherwise}, \end{cases}$$

then the higher degree B-splines for $r \geq 1$, can be constructed by the following recurrence formula [39]

$$B_{r,i}(x) = \frac{x - x_i}{x_{r+i} - x_i} B_{r-1,i}(x) + \frac{x_{r+i+1} - x}{x_{r+i+1} - x_{i+1}} B_{r-1,i+1}(x). \tag{2.1}$$

The differential quadrature method approximates the derivatives of a sufficiently smooth function at any grid point of the partition by the linear combination of the function values at all of the grid points of the partition. This approximation can be written in the following form

$$\frac{d^\gamma}{dx^\gamma} u(x)\Big|_{x_i} = \sum_{j=0}^{n} a_{i,j}^{(\gamma)} u(x_j), \quad i = 0, \dots, n, \tag{2.2}$$

where $a_{i,j}^{(\gamma)}$ are unknowns representing the weighting coefficients corresponds to the γth order derivative of the function. The way of obtaining weighting coefficients is the main key in implementation of DQ method based on various kinds of basis functions. Most of the previous methods are based on Shu's approach [40]. We will construct a new approach independent of Shu's recurrence relation for finding the weighting coefficients using B-spline basis functions. Using the basis functions (2.1) as test function in (2.2), we have

$$B_{r,k}^{(\gamma)}(x_i) = \sum_{j=0}^{n} a_{i,j}^{(\gamma)} B_{r,k}(x_j), \quad 0 \leq i \leq n, \ -\sigma \leq k \leq n + \delta, \tag{2.3}$$

which is a linear system to find $a_{i,j}^{(\gamma)}$ for $\gamma = 1, \dots, r$. If we fix γ and i in the system above, it contains $n + r$ equations with $n + 1$ unknowns $a_{i,j}^{(\gamma)}$, $j = 0, \dots, n$. The number of equations is greater than the number of unknowns, so the system may have no solution. It is possible to have a square system by adding $r - 1$ extra nodes to the grid points. Since adding extra points outside the domain is often difficult to handle, we will use the mid points of the partition as extra points. To do this, let $\Gamma \equiv \{\tau_i\}_{i=1}^{n}$ be the set of mid points of Δ and define the new set of points

$$\Theta \equiv \{t_{-\sigma}, \dots, t_{n+\delta}\} = \{\tau_1, \dots, \tau_\sigma\} \cup \{x_0, \dots, x_n\} \cup \{\tau_{n-\delta+1}, \dots, \tau_n\},$$

which contains $n + r$ points mixed from grid points and mid points of the partition. Now for $r \geq 0$ and $\gamma = 1, \dots, r$, using the points of Θ, we have

$$B_{r,k}^{(\gamma)}(t_i) = \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} B_{r,k}(t_j), \quad -\sigma \leq i \leq n + \delta, \ -\sigma \leq k \leq n + \delta, \tag{2.4}$$

which for fixed i, form a square system of order $n + r$ with the unknowns $a_{i,j}^{(\gamma)}$, $j = -\sigma, \dots, n + \delta$. Now suppose that $\bar{u}^{(\gamma)}(x)$ is the approximation of the γth derivative for the sufficiently smooth function $u(x)$ at the grid points of the partition, then by solving the above systems and finding $a_{i,j}^{(\gamma)}$, we can construct the following discrete approximations to the derivatives of $u(x)$

$$u^{(\gamma)}(t_i) \simeq \bar{u}^{(\gamma)}(t_i) = \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} u(t_j), \quad -\sigma \leq i \leq n + \delta, \ \gamma = 1, \dots, r. \tag{2.5}$$

Substituting the above approximations into the differential equation, the problems will be reduced to a system of algebraic equations which can be solved to obtain an approximation of $u(t_j)$.

Theorem 2.1. *Let $s(x)$ be the periodic interpolating spline of degree $r = 2s - 1$ for $f \in C^{r+2}[a, b]$ over the uniform mesh Δ, then for $x \in [a, b]$ we have*

$$\left\| s^{(\gamma)}(x) - f^{(\gamma)}(x) \right\| = O(h^{2s-\gamma}), \quad \gamma = 0, 1, \dots, 2s - s. \tag{2.6}$$

Also the local error bounds for $x_i \in \Delta$ are as follows

$$\begin{aligned}
&\left|s^{(\gamma)}(x_i) - f^{(\gamma)}(x_i)\right| = O(h^{2s}), && \gamma = 1,\\
&\left|s^{(\gamma)}(x_i) - f^{(\gamma)}(x_i)\right| = O(h^{2s-\gamma}), && \gamma = 2, 4, \ldots, 2s-2,\\
&\left|s^{(\gamma)}(x_i) - f^{(\gamma)}(x_i)\right| = O(h^{2s+1-\gamma}), && \gamma = 3, 5, \ldots, 2s-3.
\end{aligned} \tag{2.7}$$

Proof. See [41, Corollary 2]. □

Theorem 2.2. *Let $u(x) \in C^{r+2}[a, b]$ and suppose that $\bar{u}^{(\gamma)}(t_i)$ is the spline-based DQM approximation of $\bar{u}^{(\gamma)}(t_i)$ by (2.5). If we use the odd degree B-spline basis functions $B_{r,i}(x)$ with $r = 2s - 1$ in (2.4), then the following error bounds hold*

$$\begin{aligned}
&\left|u^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i)\right| = O(h^{2s}), && \gamma = 1,\\
&\left|u^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i)\right| = O(h^{2s-\gamma}), && \gamma = 2, 4, \ldots, 2s-2,\\
&\left|u^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i)\right| = O(h^{2s+1-\gamma}), && \gamma = 3, 5, \ldots, 2s-3.
\end{aligned} \tag{2.8}$$

Proof. Let $s(x) \in SP_r(\Delta)$ be the B-spline interpolant of $u(x)$ defined in the following form

$$s(x) = \sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(x)$$

where c_k are some constant coefficients. Using the triangular inequality we have

$$\left|u^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i)\right| \le \left|u^{(\gamma)}(x_i) - s^{(\gamma)}(x_i)\right| + \left|s^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i)\right|. \tag{2.9}$$

For the first term using (2.7) we have

$$\left|u^{(\gamma)}(x_i) - s^{(\gamma)}(x_i)\right| = \begin{cases} O(h^{2s}) & \gamma = 1,\\ O(h^{2s-\gamma}) & \gamma = 2, 4, \ldots, 2s-2,\\ O(h^{2s+1-\gamma}) & \gamma = 3, 5, \ldots, 2s-3. \end{cases} \tag{2.10}$$

For the second term we can write

$$s^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i) = \sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}^{(\gamma)}(x_i) - \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} u(t_j),$$

where by using (2.4) we have

$$\begin{aligned}
s^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i) &= \sum_{k=-\sigma}^{n+\delta} c_k \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} B_{r,k}(t_j) - \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} u(t_j)\\
&= \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} \sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(t_j) - \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} u(t_j)\\
&= \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)} \left(\sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(t_j) - u(t_j)\right)\\
&= \sum_{j=1}^{\sigma} a_{i,j}^{(\gamma)} \left(\sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(\tau_j) - u(\tau_j)\right) + \sum_{j=0}^{n} a_{i,j}^{(\gamma)} \left(\sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(x_j) - u(x_j)\right)\\
&\quad + \sum_{j=n-\delta+1}^{n} a_{i,j}^{(\gamma)} \left(\sum_{k=-\sigma}^{n+\delta} c_k B_{r,k}(\tau_j) - u(\tau_j)\right)\\
&= \sum_{j=1}^{\sigma} a_{i,j}^{(\gamma)} \big(s(\tau_j) - u(\tau_j)\big) + \sum_{j=0}^{n} a_{i,j}^{(\gamma)} \big(s(x_j) - u(x_j)\big) + \sum_{j=n-\delta+1}^{n} a_{i,j}^{(\gamma)} \big(s(\tau_j) - u(\tau_j)\big).
\end{aligned} \tag{2.11}$$

Also an application of (2.6) for $\gamma = 0$ gives

$$\begin{aligned}
&s(x_j) - \bar{u}(x_j) = O(h^{2s}), \quad x_i \in \Delta,\\
&s(\tau_j) - \bar{u}(\tau_j) = O(h^{2s}), \quad \tau_i \in \Gamma,
\end{aligned}$$

which can be substituted into (2.11) to get

$$s^{(\gamma)}(x_i) - \bar{u}^{(\gamma)}(x_i) = \Big(\sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(\gamma)}\Big) O(h^{2s}) = O(h^{2s}). \tag{2.12}$$

Finally substituting (2.10) and (2.12) into (2.9) we get the results and the proof is complete. □

Theorem 2.3. *Let $u(x) \in C^{r+2}[a, b]$ and suppose that $\bar{u}^{(\gamma)}(t_i)$ is the spline-based* DQM *approximation of $\bar{u}^{(\gamma)}(t_i)$ by (2.5). If we use the even degree B-spline basis functions $B_{r,i}(x)$ with $r = 2s$ in (2.4), then we have*

$$\begin{aligned}
\left|s^{(\gamma)}(t_i) - u^{(\gamma)}(t_i)\right| &= O(h^{2s}), & \gamma &= 1,\\
\left|s^{(\gamma)}(t_i) - u^{(\gamma)}(t_i)\right| &= O(h^{2s+2-\gamma}), & \gamma &= 2, 4, \ldots, 2s-2,\\
\left|s^{(\gamma)}(t_i) - u^{(\gamma)}(t_i)\right| &= O(h^{2s+1-\gamma}), & \gamma &= 3, 5, \ldots, 2s-3.
\end{aligned} \tag{2.13}$$

Proof. The proof is in a similar manner with Theorem 2.2 using the results in [42,43]. □

3. Implementation of the method

Consider the following differential equations

$$\mathcal{L}u(x) \equiv u^{(m)}(x) = f(x, u(x), u'(x), \ldots, u^{(m-1)}(x)), \quad a \le x \le b, \tag{3.1}$$

subjected to the boundary conditions

$$\mathcal{B}u \equiv \sum_{q=0}^{m-1} \big(\alpha_{p,q} u^{(q)}(a) + \beta_{p,q} u^{(q)}(b)\big) = \lambda_p, \quad 0 \le p \le m-1. \tag{3.2}$$

Substituting (2.5) into (3.1)–(3.2) and using the notation $u_j \equiv u(t_j)$ we can convert the boundary value problem into a system of algebraic equations with the unknowns u_j, $-\sigma \le j \le n+\delta$, in the following form

$$\sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(m)} u_j = f\Big(t_i, u_i, \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(1)} u_j, \ldots, \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(m-1)} u_j\Big), \quad 0 \le i \le n, \tag{3.3}$$

$$\sum_{q=0}^{m-1} \Big(\alpha_{p,q} \sum_{j=-\sigma}^{n+\delta} a_{0,j}^{(q)} u_j + \beta_{p,q} \sum_{j=-\sigma}^{n+\delta} a_{n,j}^{(q)} u_j\Big) = \lambda_p, \quad 0 \le p \le m-1. \tag{3.4}$$

The above system contains $n+m+1$ equations with $n+r$ unknowns, but in order to uniquely solve the system it is required that the number of unknowns and equations are equal. If $r > m+1$, we need to add $r-m-1$ extra relations to have a square system. To do this we may use (3.1) at mid points as follows

$$\sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(m)} u_j = f\Big(t_i, u_j, \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(1)} u_j, \ldots, \sum_{j=-\sigma}^{n+\delta} a_{i,j}^{(m-1)} u_j\Big), \quad i \in \{-\sigma, \ldots, n+\delta\} \backslash \{0, \ldots, n\}. \tag{3.5}$$

Solving the system we can find the values u_j, $-\sigma \le j \le n+\delta$, which are approximate values to the solution of the problem at t_j.

Corollary 3.1. *Let u_j be the* DQM *approximation to the solution of (3.1)–(3.2) at grid points by (2.5) which is based on B-spline basis functions of odd degree with $r = 2s-1$. If the exact solution satisfies $u(x) \in C^{r+2}[a, b]$, then we have*

$$(\mathcal{L}u - f)_{t_j} = \begin{cases} O(h^{2s-m}), & m \text{ is even} \\ O(h^{2s+1-m}), & m \text{ is odd.} \end{cases} \tag{3.6}$$

$$(\mathcal{B}u - \lambda)_{t_0,t_n} = \begin{cases} O(h^{2s+1-m}), & m \text{ is odd} \\ O(h^{2s+2-m}), & m \text{ is even.} \end{cases} \tag{3.7}$$

Corollary 3.2. *Let u_j be the* DQM *approximation to the solution of (3.1)–(3.2) at grid points by (2.5) which is based on B-spline basis functions of even degree with $r = 2s$. If the exact solution satisfies $u(x) \in C^{r+2}[a, b]$, then we have*

$$(\mathcal{L}u - f)_{t_j} = \begin{cases} O(h^{2s+2-m}), & m \text{ is even} \\ O(h^{2s+1-m}), & m \text{ is odd.} \end{cases} \tag{3.8}$$

$$(\mathcal{B}u - \lambda)_{t_0,t_n} = \begin{cases} O(h^{2s+3-m}), & m \text{ is odd} \\ O(h^{2s+2-m}), & m \text{ is even.} \end{cases} \tag{3.9}$$

3.1. The biharmonic problem

Let Ω be a bounded domain in $\mathbb{R}^d$ defined as

$$\Omega \equiv \{x = (x_1, \ldots, x_d) \in \mathbb{R}^d,\ a_i \le x_i \le b_i,\ i = 1, \ldots, d\}, \tag{3.10}$$

and consider the problem of finding u such that

$$\begin{aligned} \Delta^2 u(x) &= f(x), \quad x \in \Omega \\ u(x) &= g_1(x), \quad x \in \partial\Omega \\ \nabla u(x) \cdot \mathbf{n} &= g_2(x), \quad x \in \partial\Omega, \end{aligned} \tag{3.11}$$

where the biharmonic operator Δ^2 in $\mathbb{R}^d$ is defined as follows

$$\Delta^2 \equiv \sum_{\iota=1}^{d} \frac{\partial^4}{\partial x_\iota^4} + \sum_{\iota=1}^{d} \sum_{\substack{\eta=1 \\ \eta\neq\iota}}^{d} \frac{\partial^4}{\partial x_\iota^2 \partial x_\eta^2}.$$

Let us define the uniform partition Δ_ι of the size $h_\iota = \frac{b_\iota - a_\iota}{n_\iota}$, on $[a_\iota, b_\iota]$ as follows

$$\Delta_\iota \equiv \{a_\iota = x_{\iota,0} < \cdots < x_{\iota,n_\iota} = b_\iota\}, \quad \iota = 1, \ldots, d,$$

and also let $\Gamma_\iota \equiv \{\tau_{\iota,i}\}_{i=1}^{n_\iota}$, be the set of mid points of Δ_ι. We need to define the set of points Θ_ι on $[a_\iota, b_\iota]$ in the following form

$$\Theta_\iota \equiv \{t_{\iota,-\sigma}, \ldots, t_{\iota,n_\iota+\delta}\} = \{\tau_{\iota,1}, \ldots, \tau_{\iota,\sigma}\} \cup \{x_{\iota,0}, \ldots, x_{\iota,n_\iota}\} \cup \{\tau_{\iota,n_\iota-\delta+1}, \ldots, \tau_{\iota,n_\iota}\}.$$

Also for later use we need to define the set of d-tuples Σ in the following form

$$\Sigma = \{(x_1, \ldots, x_d) |\, x_i \in \Delta_i\}.$$

Similar to the one-dimensional case, we need to find the weighting coefficients $a^{(\gamma)}_{l,j_l,p}$ correspond to each variable x_l, which is needed to construct the method in higher dimensions. If $v(x)$ is an arbitrary function defined on Ω, then for $\gamma \in \mathbb{N}$, the weighting coefficients are defined as

$$\frac{\partial^\gamma}{\partial x_\iota^\gamma} v\big(t_{1,j_1}, \ldots, t_{d,j_d}\big) = \sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p} v(t_{1,j_1}, \ldots, t_{\iota,p}, \ldots, t_{d,j_d}), \quad -\sigma \le j_\iota \le n_\iota + \delta. \tag{3.12}$$

Now if we use the B-spline basis functions $B_{r,k}(x)$, we have the following systems of linear equations which can be solved for $\gamma = 1, \ldots, r$,

$$B^{(\gamma)}_{r,k}(t_{\iota,j_\iota}) = \sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p} B_{r,k}(t_{\iota,p}), \quad -\sigma \le j_\iota \le n_\iota + \delta,\ 1 \le \iota \le d. \tag{3.13}$$

Then using the above weighting coefficients, the spline based DQM approximation to the partial derivatives of the multivariate function $u(x_1, \ldots, x_d)$ is defined as

$$\frac{\partial^\gamma}{\partial x_\iota^\gamma} \bar{u}(t_{1,j_1}, \ldots, t_{d,j_d}) = \sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p} u(t_{1,j_1}, \ldots, t_{\iota,p}, \ldots, t_{d,j_d}), \quad -\sigma \le j_\iota \le n_\iota + \delta. \tag{3.14}$$

Also the DQ approximation to the cross-derivatives of u with respect to different variables x_ι and x_η, $\iota \neq \eta$, can be defined in the following form

$$\frac{\partial^{\alpha+\beta}}{\partial x_\iota^\alpha \partial x_\eta^\beta} \bar{u}(t_{1,j_1}, \ldots, t_{d,j_d}) = \sum_{p=-\sigma}^{n_\iota+\delta} \sum_{q=-\sigma}^{n_\eta+\delta} a^{(\alpha)}_{\iota,j_\iota,p} a^{(\beta)}_{\eta,j_\eta,q} u(t_{1,j_1}, \ldots, t_{\iota,p}, t_{\eta,q}, \ldots, t_{d,j_d}), \quad -\sigma \le j_i \le n_i + \delta. \tag{3.15}$$

Now we are able to construct an approximation to the solution of biharmonic problem. Substituting (3.14) and (3.15) into (3.11) we have

$$\begin{aligned} &\sum_{\iota=1}^{d} \sum_{p=-\sigma}^{n_\iota+\delta} \Big\{ a^{(4)}_{\iota,j_\iota,p} u(t_{1,j_1}, \ldots, t_{\iota,p}, \ldots, t_{d,j_d}) + \sum_{\substack{\eta=1 \\ \eta\neq\iota}}^{d} \sum_{q=-\sigma}^{n_\eta+\delta} a^{(2)}_{\iota,j_\iota,p} a^{(2)}_{\eta,j_\eta,q} u(t_{1,j_1}, \ldots, t_{\iota,p}, t_{\eta,q}, \ldots, t_{d,j_d}) \Big\} \\ &\quad = f(t_{1,j_1}, \ldots, t_{d,j_d}), \quad 0 \le j_i \le n_i. \end{aligned} \tag{3.16}$$

Also the boundary conditions will be translated to the following relations

$$\begin{aligned}
&u(t_{1,j_1},\dots,t_{\iota,0},\dots,t_{d,j_d}) = g_1(t_{1,j_1},\dots,a_\iota,\dots,t_{d,j_d}),\\
&u(t_{1,j_1},\dots,t_{\iota,n_\iota},\dots,t_{d,j_d}) = g_1(t_{1,j_1},\dots,b_\iota,\dots,t_{d,j_d}),\\
&\sum_{p=-\sigma}^{n_\iota+\delta} a^{(1)}_{\iota,j_\iota,p} u(t_{1,j_1},\dots,t_{\iota,0},\dots,t_{d,j_d}) = g_2(t_{1,j_1},\dots,a_\iota,\dots,t_{d,j_d}),\\
&\sum_{p=-\sigma}^{n_\iota+\delta} b^{(1)}_{\iota,j_\iota,p} u(t_{1,j_1},\dots,t_{\iota,n_\iota},\dots,t_{d,j_d}) = g_2(t_{1,j_1},\dots,b_\iota,\dots,t_{d,j_d}),
\end{aligned} \tag{3.17}$$

for $\iota = 1,\dots,d$ and $-\sigma \le j_i \le n_i + \delta$. Relations (3.16) and (3.17) together form a system of equations with the unknowns $u(t_{1,j_1},\dots,t_{d,j_d})$ which are the approximate values to the exact solution of the problem.

Lemma 3.1. *Let $s(x), x \in \Omega$ be the periodic d-variate spline interpolation of degree $r = 2s-1$ for $f \in C^{r+2}[\Omega]$ over the partition Σ. Then for $x \in \Omega$ we have*

$$\left\| \frac{\partial^\xi s}{\partial x_\iota^\xi}(x) - \frac{\partial^\xi f}{\partial x_\iota^\xi}(x) \right\| = O(h_\iota^{2s-\xi}), \quad \xi = 0,1,\dots,2s-s. \tag{3.18}$$

Also the local error bounds for $x \in \Sigma$ are as follows

$$\left| \left(\frac{\partial^\gamma s}{\partial x_\iota^\gamma} - \frac{\partial^\gamma f}{\partial x_\iota^\gamma}\right)(t_{1,j_1},\dots,t_{d,j_d}) \right| = O(h_\iota^{2s-\alpha}), \tag{3.19}$$

$$\left| \left(\frac{\partial^{\alpha+\beta} s}{\partial x_\iota^\alpha \partial x_\eta^\beta} - \frac{\partial^{\alpha+\beta} f}{\partial x_\iota^\alpha \partial x_\eta^\beta}\right)(t_{1,j_1},\dots,t_{d,j_d}) \right| = O(h_\iota^{2s-\alpha}) + O(h_\eta^{2s-\beta}), \tag{3.20}$$

where γ, α and β are even.

Proof. Relations (3.18) and (3.19) are trivial extensions of Theorem 2.1 for multi-dimensional case. In order to prove (3.20), we will use a technique similar to the approach presented in [44]. Let us suppose that $\mathbb{I}_x$, is the one-dimensional spline interpolation operator, then for $f \in C[a,b]$ we have $s = \mathbb{I}_x f$. Also for a for multivariate function $f \in C(\Omega)$, we need to define the two-dimensional interpolation operator $\mathbb{I}_{x_\iota,x_\eta}$ with respect to the variables x_ι and x_η. It is trivial that $\mathbb{I}_{x_\iota,x_\eta} = \mathbb{I}_{x_\iota}(\mathbb{I}_{x_\eta} f)$, thus we have

$$\begin{aligned}
\frac{\partial^{\alpha+\beta} s}{\partial x_\iota^\alpha \partial x_\eta^\beta} - \frac{\partial^{\alpha+\beta} f}{\partial x_\iota^\alpha \partial x_\eta^\beta} &= D^\alpha_{x_\iota} D^\beta_{x_\eta} \mathbb{I}_{x_\iota,x_\eta} f - D^\alpha_{x_\iota} D^\beta_{x_\eta} f = D^\alpha_{x_\iota} \mathbb{I}_{x_\iota} D^\beta_{x_\eta} \mathbb{I}_{x_\eta} f - D^\alpha_{x_\iota} D^\beta_{x_\eta} f\\
&= (D^\alpha_{x_\iota}\mathbb{I}_{x_\iota} - D^\alpha_{x_\iota})(D^\beta_{x_\eta}\mathbb{I}_{x_\eta} f - D^\beta_{x_\eta} f) + D^\alpha_{x_\iota}(D^\beta_{x_\eta}\mathbb{I}_{x_\eta} f - D^\beta_{x_\eta} f) + (D^\beta_{x_\iota}\mathbb{I}_{x_\iota} - D^\beta_{x_\iota})(D^\beta_{x_\eta} f).
\end{aligned}$$

Taking the absolute value and using (3.19) we have

$$\begin{aligned}
\left| \left(\frac{\partial^{\alpha+\beta} s}{\partial x_\iota^\alpha \partial x_\eta^\beta} - \frac{\partial^{\alpha+\beta} f}{\partial x_\iota^\alpha \partial x_\eta^\beta}\right)(t_{1,j_1},\dots,t_{d,j_d}) \right| &= O(h_\iota^{2s-\alpha}) O(h_\eta^{2s-\beta}) + O(h_\eta^{2s-\beta})(D^\alpha_{x_\iota} f) + O(h_\iota^{2s-\alpha})(D^\beta_{x_\eta} f)\\
&= O(h_\iota^{2s-\alpha}) + O(h_\eta^{2s-\beta}). \quad \square
\end{aligned}$$

Lemma 3.2. *Let $s(x), x \in \Omega$ be the periodic d-variate spline interpolation of degree $r = 2s$ for $f \in C^{r+2}[\Omega]$ over the partition Σ, then for $x \in \Omega$ we have*

$$\left\| \frac{\partial^\xi s}{\partial x_\iota^\xi}(x) - \frac{\partial^\xi f}{\partial x_\iota^\xi}(x) \right\| = O(h_\iota^{2s+1-\xi}), \quad \xi = 0,1,\dots,2s. \tag{3.21}$$

Also the local error bounds for $x \in \Sigma$ are as follows

$$\left| \left(\frac{\partial^\gamma s}{\partial x_\iota^\gamma} - \frac{\partial^\gamma f}{\partial x_\iota^\gamma}\right)(t_{1,j_1},\dots,t_{d,j_d}) \right| = O(h_\iota^{2s+2-\alpha}), \tag{3.22}$$

$$\left| \left(\frac{\partial^{\alpha+\beta} s}{\partial x_\iota^\alpha \partial x_\eta^\beta} - \frac{\partial^{\alpha+\beta} f}{\partial x_\iota^\alpha \partial x_\eta^\beta}\right)(t_{1,j_1},\dots,t_{d,j_d}) \right| = O(h_\iota^{2s+2-\alpha}) + O(h_\eta^{2s+2-\beta}), \tag{3.23}$$

where γ, α and β are even.

Proof. The proof is in a similar manner with Lemma 3.1. $\square$

Theorem 3.1. *Let $u(t_{1,j_1},\dots,t_{d,j_d})$ be the* DQM *approximation to the solution of biharmonic problem (3.11) based on odd degree B-spline basis functions $B_{r,k}$ with $r=2s-1$, $s>2$. If the exact solution satisfies $u(x)\in C^{r+2}[\Omega]$, then*

$$(\Delta^2 u - f)\Big|_x = O(h_1^{2s-4})+\cdots+O(h_d^{2s-4}),\quad x\in\Sigma. \tag{3.24}$$

Proof. Let s be the d-variate spline interpolation to the solution of (3.11). By adopting the following representation for s

$$s(x_1,\dots,x_d)=\sum_{i_1}\cdots\sum_{i_d} c_{i_1\cdots i_d}B_{r,i_1}(x_1)\cdots B_{r,i_d}(x_d),$$

and assuming that γ is an even positive integer, we have

$$\begin{aligned}
&\left[\frac{\partial^\gamma \bar u}{\partial x_\iota^\gamma}-\frac{\partial^\gamma s}{\partial x_\iota^\gamma}\right](t_{1,j_1},\dots,t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}u(t_{1,j_1},\dots,t_{\iota,p},\dots,t_{d,j_d})-\sum_{i_1}\cdots\sum_{i_d}c_{i_1\cdots i_d}B_{r,i_1}(t_{1,j_1})\cdots B^{(\gamma)}_{r,i_\iota}(t_{\iota,j_\iota})\cdots B_{r,i_d}(t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}u(t_{1,j_1},\dots,t_{\iota,p},\dots,t_{d,j_d})-\sum_{i_1}\cdots\sum_{i_d}c_{i_1\cdots i_d}B_{r,i_1}(t_{1,j_1})\cdots \sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}B_{r,i_\iota}(t_{\iota,p})\cdots B_{r,i_d}(t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}\left\{u(t_{1,j_1},\dots,t_{\iota,p},\dots,t_{d,j_d})-\sum_{i_1}\cdots\sum_{i_d}c_{i_1\cdots i_d}B_{r,i_1}(t_{1,j_1})\cdots B_{r,i_\iota}(t_{\iota,p})\cdots B_{r,i_d}(t_{d,j_d})\right\}\\
&=\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}\left\{u(t_{1,j_1},\dots,t_{\iota,p},\dots,t_{d,j_d})-s(t_{1,j_1},\dots,t_{\iota,p},\dots,t_{d,j_d})\right\}\\
&=\left(\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\gamma)}_{\iota,j_\iota,p}\right)O(h_l^{2s-\gamma})=O(h_l^{2s-\gamma})
\end{aligned} \tag{3.25}$$

where the last two equalities are the results of Lemma 3.1. Now we want to find some bounds on DQM cross-derivatives approximations. Let α and β be even positive integers, then we have

$$\begin{aligned}
&\left[\frac{\partial^{\alpha+\beta} \bar u}{\partial x_\iota^\alpha\partial x_\eta^\beta}-\frac{\partial^{\alpha+\beta} s}{\partial x_\iota^\alpha x_\eta^\beta}\right](t_{1,j_1},\dots,t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta}\sum_{q=-\sigma}^{n_\eta+\delta} a^{(\alpha)}_{\iota,j_\iota,p}a^{(\beta)}_{\eta,j_\eta,q}u(t_{1,j_1},\dots,t_{\iota,p},t_{\eta,q},\dots,t_{d,j_d})\\
&\quad-\sum_{i_1}\cdots\sum_{i_d}c_{i_1\cdots i_d}B_{r,i_1}(t_{1,j_1})\cdots B^{(\alpha)}_{r,i_\iota}(t_{\iota,j_\iota})\cdots B^{(\beta)}_{r,i_\eta}(t_{\eta,j_\eta})\cdots B_{r,i_d}(t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta}\sum_{q=-\sigma}^{n_\eta+\delta} a^{(\alpha)}_{\iota,j_\iota,p}a^{(\beta)}_{\eta,j_\eta,q}u(t_{1,j_1},\dots,t_{\iota,p},t_{\eta,q},\dots,t_{d,j_d})\\
&\quad-\sum_{i_1}\cdots\sum_{i_d}c_{i_1\cdots i_d}B_{r,i_1}(t_{1,j_1})\cdots\sum_{p=-\sigma}^{n_\iota+\delta} a^{(\alpha)}_{\iota,j_\iota,p}B_{r,i_\iota}(t_{\iota,p})\cdots\sum_{q=-\sigma}^{n_\eta+\delta} a^{(\beta)}_{\eta,j_\eta,q}B_{r,i_\eta}(t_{\eta,q})\cdots B_{r,i_d}(t_{d,j_d})\\
&=\sum_{p=-\sigma}^{n_\iota+\delta}\sum_{q=-\sigma}^{n_\eta+\delta} a^{(\alpha)}_{\iota,j_\iota,p}a^{(\beta)}_{\eta,j_\eta,q}\left\{u(t_{1,j_1},\dots,t_{\iota,p},t_{\eta,q},\dots,t_{d,j_d})-s(t_{1,j_1},\dots,t_{\iota,p},t_{\eta,q},\dots,t_{d,j_d})\right\}\\
&=\left(\sum_{p=-\sigma}^{n_\iota+\delta}\sum_{q=-\sigma}^{n_\eta+\delta} a^{(\alpha)}_{\iota,j_\iota,p}a^{(\beta)}_{\eta,j_\eta,q}\right)[O(h_\iota^{2s-\alpha})+O(h_\eta^{2s-\beta})]=O(h_\iota^{2s-\alpha})+O(h_\eta^{2s-\beta}).
\end{aligned} \tag{3.26}$$

Finally by assuming $\gamma=4$ and $\alpha=\beta=2$ in relations (3.25) and (3.26) we can obtain (A.3). □

Theorem 3.2. *Let $u(t_{1,j_1},\dots,t_{d,j_d})$ be the* DQM *approximation to the solution of biharmonic problem (3.11) based on even degree B-spline basis functions $B_{r,k}(x)$ with $r=2s$, $s\ge 2$. If the exact solution satisfies $u(x)\in C^{r+2}[\Omega]$, then*

$$(\Delta^2 u - f)\Big|_x = O(h_1^{2s-2})+\cdots+O(h_d^{2s-2}),\quad x\in\Sigma. \tag{3.27}$$

Proof. Using Lemma 3.2 and similar to the approach used in Theorem 3.1 we can obtain the above error bounds. □

3.2. The Poisson equation

Let Ω be defined as (3.10) and consider the problem of finding the numerical solution of Poisson equation

$$\begin{aligned} \nabla^2 u(x) &= f(x), \quad x \in \Omega \\ u(x) &= g(x), \quad x \in \partial\Omega \end{aligned} \tag{3.28}$$

which the operator ∇^2 is defined as follows

$$\nabla^2 = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \cdots + \frac{\partial^2}{\partial x_d^2}. \tag{3.29}$$

Also let Δ_ι, Γ_ι, Θ_ι and Σ be the same as notations we already defined. Substituting (3.14) with $\gamma = 2$ into the Poisson equation (3.28) we have

$$\sum_{\iota=1}^{d} \sum_{p=-\sigma}^{n_\iota+\delta} a^{(2)}_{\iota,j_\iota,p} u(t_{1,j_1}, \ldots, t_{\iota,p}, \ldots, t_{d,j_d}) = f(t_{1,j_1}, \ldots, t_{d,j_d}), \quad 0 \le j_i \le n_i. \tag{3.30}$$

The Dirichlet boundary conditions can be written as follows

$$\begin{cases} u(t_{1,j_1}, \ldots, t_{\iota,0}, \ldots, t_{d,j_d}) = g_1(t_{1,j_1}, \ldots, a_\iota, \ldots, t_{d,j_d}), \\ u(t_{1,j_1}, \ldots, t_{\iota,n_\iota}, \ldots, t_{d,j_d}) = g_1(t_{1,j_1}, \ldots, b_\iota, \ldots, t_{d,j_d}), \end{cases} \quad -\sigma \le j_i \le n_i + \delta, \tag{3.31}$$

for $\iota = 1, \ldots, d$. Relations (3.30) and (3.31) form a system of linear equations with the unknowns $u(t_{1,j_1}, \ldots, t_{d,j_d})$, which are the desired approximations to the exact solution at grid points.

Theorem 3.3. *Let $u(t_{1,j_1}, \ldots, t_{d,j_d})$ be the* DQM *approximation to the solution of Poisson problem* (3.28) *based on odd degree B-spline basis functions $B_{r,k}$ with $r = 2s - 1$, $s > 1$. If the exact solution satisfies $u(x) \in C^{r+2}[\Omega]$, then*

$$(\nabla^2 u - f)\Big|_x = O(h_1^{2s-2}) + \cdots + O(h_d^{2s-2}), \quad x \in \Sigma. \tag{3.32}$$

Proof. Using (3.25) for $\gamma = 2$ we can find the above error bound for Poisson equations. □

Theorem 3.4. *Let $u(t_{1,j_1}, \ldots, t_{d,j_d})$ be the* DQM *approximation to the solution of Poisson problem* (3.28) *based on even degree B-spline basis functions $B_{r,k}$ with $r = 2s$, $s \ge 1$. If the exact solution satisfies $u(x) \in C^{r+2}[\Omega]$, then*

$$(\nabla^2 u - f)\Big|_x = O(h_1^{2s}) + \cdots + O(h_d^{2s}), \quad x \in \Sigma. \tag{3.33}$$

Remark 3.1. It should be mentioned that it is possible to make some changes in the proposed algorithm. We constructed our method on grid points x_i (relation (2.3)), then by adding some mid points we tried to obtain a uniquely solvable system (relation (2.4)). Based on the problem's type and the degree of B-spline basis function we can choose another arrangement of the points. The following relations can be used to obtain the weighting coefficients

$$B^{(\gamma)}_{r,k}(\tau_i) = \sum_{j=0}^{n} a^{(\gamma)}_{i,j} B_{r,k}(\tau_j), \quad 1 \le i \le n, \ -\sigma \le k \le n + \delta, \tag{3.34}$$

which is not a square system. Thus to be able to solve the system uniquely, we need r extra relations. In this case we can construct (3.34) at some near boundary grid points x_i as follows. Let us define the set of points $\bar{\Theta}$ in the following form

$$\bar{\Theta} \equiv \{\bar{t}_{-\sigma}, \ldots, \bar{t}_{n+\delta}\} = \{x_0, \ldots, x_\sigma\} \cup \{\tau_1, \ldots, \tau_n\} \cup \{n_{n-\delta+1}, \ldots, x_n\}$$

where $\bar{\sigma} = [\frac{r+1}{2}]$, $\bar{\delta} = [\frac{r}{2}]$ and $\bar{\delta} + \bar{\sigma} = r$. Then for $r \ge 0$, we can use the following system of linear equations to find the weighting coefficients $a^{(\gamma)}_{i,j}$ for $\gamma = 1, \ldots, r$, using B-spline basis functions

$$B^{(\gamma)}_{r,k}(t_i) = \sum_{j=-\bar{\sigma}}^{n+\bar{\delta}} a^{(\gamma)}_{i,j} B_{r,k}(\bar{t}_j), \quad \bar{t}_i \in \bar{\Theta}, \ -\bar{\sigma} \le k \le n + \bar{\delta}. \tag{3.35}$$

The accuracy of two proposed algorithm are exactly the same and based on the problem's type we can choose a strategy. It should be noted that in order to solve the problems using even degree B-splines, it is better to use the second approach.

Table 1
Maximum absolute errors (MAE) for Example 4.1.

n	$r=3$	Order	$r=4$	Order	$r=5$	Order	$r=6$	Order	[9]	[10]
2^4	1.62(−4)		2.71(−07)		1.03(−07)		4.98(−10)		3.08(−4)	3.08(−4)
2^5	4.06(−5)	1.99	1.71(−08)	3.98	6.53(−09)	3.98	7.32(−12)	6.08	8.02(−5)	8.02(−5)
2^6	1.01(−5)	1.99	1.07(−09)	3.99	4.12(−10)	3.99	1.03(−13)	6.14	2.02(−5)	2.02(−5)
2^7	2.54(−6)	1.99	6.71(−11)	3.99	2.50(−11)	4.03	6.58(−15)	3.97	5.08(−6)	5.08(−6)

Table 2
Relative errors (RE) for Example 4.1.

n	$r=3$	Time	$r=4$	Time	$r=5$	Time	$r=6$	Time
2^4	3.20(−3)	0.41	5.41(−6)	0.43	2.88(−05)	0.40	9.93(−09)	0.45
2^5	8.03(−4)	1.27	3.38(−7)	1.31	1.28(−07)	1.65	1.45(−10)	1.82
2^6	2.00(−4)	3.92	2.11(−8)	4.55	8.13(−09)	5.25	2.05(−12)	5.39
2^7	5.01(−5)	175	1.33(−9)	182	5.12(−10)	191	3.16(−14)	197

4. Numerical experiments

In order to test the applicability of the proposed algorithm, we solved some examples of multi-dimensional elliptic PDEs including 2D and 3D Poisson and biharmonic equations. Also some examples of time-dependent PDEs have been solved to compare the method with similar previously used DQM approaches. We approximated the solutions by using several choices of B-spline basis functions. The problems have been solved with $r = 3, 4, 5, 6$, and the maximum absolute errors, the relative errors, the CPU runtimes and the practical orders of convergence are tabulated in Tables 1–13. Without lose of generality, it is supposed that $n_1 = \cdots = n_d = n$ and also $h_1 = \cdots = h_d = h$. It should be noted that in the tables, errors are reported in the form of $a.b(-c)$ which means $a.b \times 10^{-c}$. For Poisson equation with $r = 3$ and 5, according to Theorem 3.3, it is expected to obtain $O(h^2)$ and $O(h^4)$ approximations respectively. Also for $r = 4$ and 6, by the help of Theorem 3.4, the approximations should be of orders 4 and 6 respectively. For the biharmonic problem, with $r = 4$ and 5 it is expected to obtain $O(h^2)$ and for $r = 6$ the approximations should be of order 4. It is obvious that the numerical results are in a good agreement with the theoretical results. The log-plot of the errors and the plot o the numerical solutions are presented in Figs. 1–4.

If we denote by E_1 and E_2, the errors obtained with $n = N_1$ and N_2 respectively, then the practical orders of convergence can be calculated by

$$Order = \frac{\log(E_1/E_2)}{\log(N_2/N_1)}.$$

If we indicate the exact solution by u and the DQM approximation by $\bar{u}$, then the maximum absolute error and the relative error can be obtained as follows

$$MAE = \|u - \bar{u}\|_\infty$$
$$RE = \frac{\|u - \bar{u}\|_\infty}{\|u\|_\infty}$$

where $\|u\|_\infty$ means the maximum absolute value at all grid points of the partition

$$\|u\|_\infty = \max\left|u(t_{0,j_0}, \ldots, t_{d,j_d})\right|, \quad 0 \le j_i \le n_i,\ 0 \le i \le d.$$

All the programs are written in Mathematic 9.1, and have been run on a system with Intel Core i7-2670 2.20 GHz and 8 GB RAM.

Example 4.1. Let us suppose that $d = 2$ and

$$\Omega = \{(x, y) \in \mathbb{R}^2,\ 0 \le x \le 1,\ 0 \le y \le 1\}.$$

Consider the following 2D Poisson equation

$$\begin{aligned} u_{xx} + u_{yy} &= \sin(\pi x)\sin(\pi y), && (x, y) \in \Omega, \\ u(x, y) &= 0, && (x, y) \in \partial\Omega, \end{aligned}$$

with the exact solution $u(x, y) = -\frac{1}{2\pi^2}\sin(\pi x)\sin(\pi y)$. We solved this problem using various degrees of B-spline basis functions. The maximum absolute errors as well as the practical orders of convergence are reported in Table 1. The numerical orders of convergence in Table 1 verify the theoretical results as well. Our results have been compared with the results in [9,10] to show the efficiency and high accuracy of the proposed method. In Table 2 we reported the relative errors as well as CPU run-times in seconds. The numerical solution of the problem with $r = 6$ and $n = 64$ is plotted in Fig. 1. Also the log-plot of the errors is presented in Fig. 2.

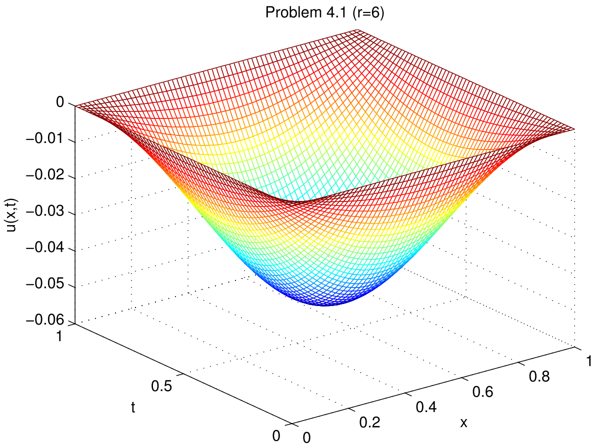

Fig. 1. The graph of the numerical solution for problems 4.1 with $r = 6$ and $n = 64$.

Table 3
Maximum absolute errors (MAE) for Example 4.2.

n	$r = 3$	Order	$r = 4$	Order	$r = 5$	Order	$r = 6$	Order	[9]
08	2.53(−6)		2.72(−6)		1.00(−6)		2.79(−08)		–
16	9.09(−8)	4.79	1.08(−7)	4.65	6.91(−8)	3.85	3.26(−10)	6.41	1.34(−4)
32	2.33(−8)	1.95	6.87(−9)	3.97	4.39(−9)	3.97	5.21(−12)	5.96	3.53(−5)

Table 4
Relative errors (RE) for Example 4.2.

n	$r = 3$	Time	$r = 4$	Time	$r = 5$	Time	$r = 6$	Time
08	7.51(−5)	0.56	8.53(−5)	1.21	1.00(−5)	1.67	8.78(−07)	2.19
16	2.69(−6)	8.47	5.41(−6)	13.3	2.04(−6)	19.1	9.79(−09)	25.6
32	6.73(−7)	127	3.41(−7)	139	1.33(−7)	183	1.58(−10)	216

Example 4.2. Let us suppose that $d = 3$ and

$$\Omega = \{(x, y, z) \in \mathbb{R}^3,\ 0 \le x \le 1,\ 0 \le y \le 1,\ 0 \le z \le 1\}.$$

Consider the following 3D Poisson equation

$$\begin{aligned} u_{xx} + u_{yy} + u_{zz} &= \sin(\pi x)\sin(\pi y)\sin(\pi z), && (x, y, z) \in \Omega,\\ u(x, y, z) &= 0, && (x, y, z) \in \partial\Omega, \end{aligned}$$

with the exact solution $u(x, y, z) = -\frac{1}{3\pi^2}\sin(\pi x)\sin(\pi y)\sin(\pi z)$. We solved the problem using various degrees B-spline basis functions. The maximum absolute errors and the practical orders of convergence are tabulated in Table 3. The practical orders of convergence in Table 3 verify the theoretical results as well. Our results have been compared with the results in [9] to show the efficiency of the method. The relative errors of the problem and the CPU runtimes in second are tabulated in Table 4. The log-plot of the errors is presented in Fig. 2.

Example 4.3. Let us suppose that $d = 3$ and

$$\Omega = \{(x, y, z) \in \mathbb{R}^3,\ 0 \le x \le 1,\ 0 \le y \le 1,\ 0 \le z \le 1\}.$$

Consider the following 3D Poisson equation

$$u_{xx} + u_{yy} + u_{zz} = 0, \quad (x, y, z) \in \Omega,$$

subjected to the following boundary conditions

$$\begin{aligned} \frac{\partial u}{\partial x} &= \frac{\sin(\pi y)\sin(\pi z)}{\sinh(\pi\sqrt{2})}\Big[2\sqrt{2}\pi\sinh(\pi\sqrt{2}x) - \sqrt{2}\pi\sinh\big(\pi\sqrt{2}(1-x)\big)\Big],\\ \frac{\partial u}{\partial y} &= \frac{\pi\sin(\pi y)\sin(\pi z)}{\sinh(\pi\sqrt{2})}\Big[2\sinh(\pi\sqrt{2}x) + \sinh\big(\pi\sqrt{2}(1-x)\big)\Big],\\ \frac{\partial u}{\partial z} &= \frac{\pi\sin(\pi y)\sin(\pi z)}{\sinh(\pi\sqrt{2})}\Big[2\sinh(\pi\sqrt{2}x) + \sinh\big(\pi\sqrt{2}(1-x)\big)\Big], \end{aligned}$$

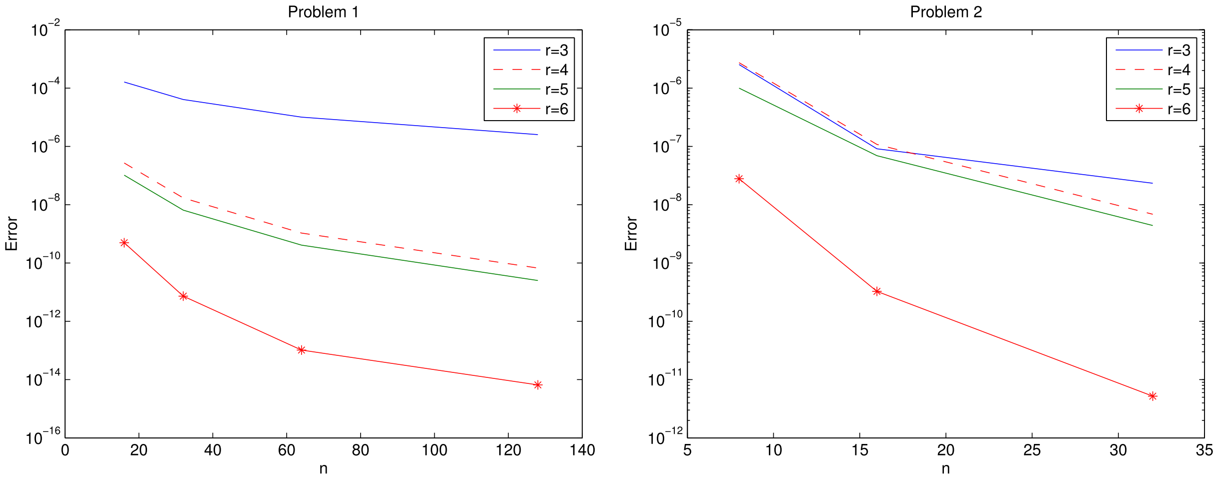

Fig. 2. The log-plot of the errors for problems 4.1 and 4.2.

Table 5
Maximum absolute errors (MAE) for Example 4.3.

n	DQM $r=3$	DQM $r=4$	DQM $r=5$	DQM $r=6$	[11] 7-points	[11] 15-points	[11] 19-points	[12] LSFEM	[12] Galerkin
04	1.2(−3)	3.8(−4)	5.7(−4)	–	5.8(−2)	7.9(−3)	5.1(−3)	7.6(−2)	7.0(−2)
08	9.0(−5)	4.8(−5)	2.1(−5)	4.4(−6)	1.5(−2)	4.9(−4)	3.2(−4)	1.7(−2)	1.6(−2)
16	1.0(−5)	3.4(−6)	1.0(−6)	3.3(−8)	3.9(−3)	3.0(−5)	2.0(−5)	4.3(−3)	3.9(−3)

Table 6
Relative (RE) errors for Example 4.3.

n	$r=3$	Time	$r=4$	Time	$r=5$	Time	$r=6$	Time
04	2.1(−3)	0.23	4.5(−4)	0.48	5.8(−4)	0.45	–	
08	1.1(−4)	0.93	5.0(−5)	1.21	2.1(−5)	1.60	4.6(−6)	2.18
16	1.0(−5)	10.2	3.5(−6)	13.3	1.1(−6)	18.8	3.3(−8)	26.2

on $\partial\Omega$. The exact solution of the problem can be written as follows

$$u(x,y,z)=\frac{\sin(\pi y)\sin(\pi z)}{\sinh(\pi\sqrt{2})}\left[2\sinh(\pi\sqrt{2}x)+\sinh\big(\pi\sqrt{2}(1-x)\big)\right].$$

The problem has been solved by DQM, based on various degrees B-spline basis functions. The maximum absolute errors are tabulated in Table 5. Our results have been compared with the results in [11,12] to show the efficiency and good performance of our proposed method. Also the relative errors of the problem and the CPU runtimes in second are tabulated in Table 6.

Example 4.4. Let us suppose that $d=2$ and

$$\Omega=\{(x,y)\in\mathbb{R}^2,\ 0\le x\le 1,\ 0\le y\le 1\}.$$

Consider the following 2D biharmonic equation

$$\frac{\partial^4 u}{\partial x^4}+2\frac{\partial^4 u}{\partial x^2\partial y^2}+\frac{\partial^4 u}{\partial y^4}=\sin(2\pi x)\sin(2\pi y),\quad (x,y)\in\Omega,$$

subjected to the following Dirichlet and Neumann boundary conditions

$$\begin{aligned} &u(x,y)=0, && (x,y)\in\partial\Omega,\\ &u_x(x,y)=\frac{1}{32\pi^2}\sin(2\pi y), && 0\le y\le 1,\ x=0,1,\\ &u_y(x,y)=\frac{1}{32\pi^2}\sin(2\pi x), && 0\le x\le 1,\ y=0,1. \end{aligned}$$

The exact solution of the problem can be written as $u(x,y)=\frac{1}{64\pi^4}\sin(2\pi x)\sin(2\pi y)$. We solved this problem by DQM, based on various degrees B-spline basis functions. The maximum absolute errors and the practical orders of convergence

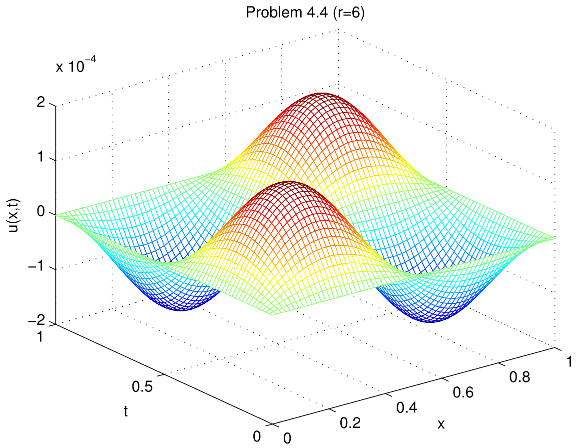

Fig. 3. The graph of the numerical solution for problems 4.4 with $r = 6$ and $n = 64$.

Table 7
Maximum absolute errors (MAE) for Example 4.4.

n	$r = 4$	Order	$r = 5$	Order	$r = 6$	Order	[9]
08	1.03(−5)		2.33(−6)		6.97(−08)		1.78(−4)
16	2.30(−6)	2.17	5.74(−7)	2.02	4.41(−09)	3.99	1.67(−4)
32	5.86(−7)	1.97	1.46(−7)	1.96	2.83(−10)	3.97	1.66(−4)
64	1.46(−7)	2.01	3.61(−8)	2.02	1.85(−11)	3.95	1.65(−4)

Table 8
Relative errors (RE) for Example 4.4.

n	$r = 4$	Time	$r = 5$	Time	$r = 6$	Time
08	6.48(−2)	0.32	1.45(−2)	0.35	5.67(−4)	0.39
16	1.68(−2)	0.79	3.58(−3)	0.89	3.99(−5)	0.93
32	3.80(−3)	6.81	9.15(−4)	7.98	2.39(−6)	8.12
64	9.19(−4)	131	2.28(−4)	143	1.49(−7)	148

are tabulated in Table 7. The practical orders verify the theoretical results as well. Our results have been compared with the results in [9] to show the efficiency of the method. Also the relative errors for this problem and the CPU runtimes are reported in Table 8. The numerical solution of the problem with $r = 6$ and $n = 64$ is plotted in Fig. 3. The log-plot of the errors is presented in Fig. 4.

Example 4.5. Let us suppose that $d = 3$ and

$$\Omega = \{(x, y, z) \in \mathbb{R}^3,\ 0 \le x \le 1,\ 0 \le y \le 1,\ 0 \le z \le 1\}.$$

Consider the following 3D biharmonic equation

$$\frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4} + \frac{\partial^4 u}{\partial z^4} + 2\frac{\partial^4 u}{\partial x^2 \partial y^2} + 2\frac{\partial^4 u}{\partial y^2 \partial z^2} + 2\frac{\partial^4 u}{\partial x^2 \partial z^2} = \sin(\pi x)\sin(\pi y)\sin(\pi z), \quad (x, y, z) \in \Omega,$$

subjected to the Dirichlet and Neumann boundary conditions

$$u = 0, \quad \frac{\partial^2 u}{\partial x^2} = \frac{\partial^2 u}{\partial y^2} = \frac{\partial^2 u}{\partial z^2} = 0, \quad (x, y, z) \in \partial\Omega.$$

The exact solution of the problem can be written as

$$u(x, y, z) = \frac{1}{9\pi^4}\sin(\pi x)\sin(\pi y)\sin(\pi z).$$

We solved this problem by DQM, based on various degrees B-spline basis functions. The maximum absolute errors and the practical orders of convergence are tabulated in Table 9. The practical orders verify the theoretical results as well. Our results have been compared with the results in [9] to show the efficiency of the method. The relative errors for this problem as well as the CPU runtimes are reported in Table 10. The log-plot of the errors is presented in Fig. 4.

Table 9
Maximum absolute errors (MAE) for Example 4.5.

n	$r=4$	Order	$r=5$	Order	$r=6$	Order	[9]
08	2.74(−5)		4.87(−6)		3.83(−08)		1.26(−6)
16	1.30(−5)	1.07	1.22(−6)	1.99	2.37(−09)	4.01	4.07(−7)
32	6.37(−6)	0.94	3.03(−7)	2.00	1.47(−10)	4.01	–

Table 10
Relative errors (RE) for Example 4.5.

n	$r=4$	Time	$r=5$	Time	$r=6$	Time
08	2.55(−2)	3.73	4.27(−4)	6.34	1.56(−5)	9.71
16	1.37(−2)	94.1	1.07(−4)	113	9.59(−7)	162

Table 11
Maximum absolute errors (MAE) for Example 4.6.

t	Our method $r=3$	DQM in [29]
0.1	1.81×10^{-6}	2.51×10^{-5}
1.0	7.27×10^{-4}	3.12×10^{-3}

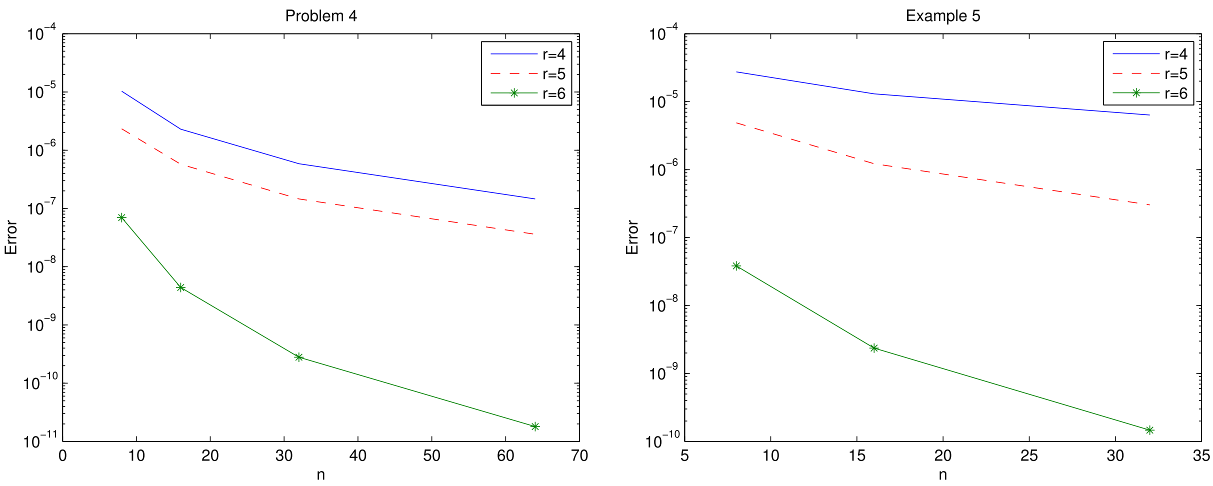

Fig. 4. The log-plot of the errors for problems 4.4 and 4.5.

Example 4.6 (*2D Time-dependent problem*)**.** Let us suppose that $d=2$ and

$$\Omega = \{(x,y) \in \mathbb{R}^2,\ 0 \le x, y \le 1\}.$$

Consider the following 2D wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{1}{2}\left(\frac{\partial^2 u}{\partial x^4} + \frac{\partial^2 u}{\partial y^2}\right), \quad (x,y) \in \Omega,\ t > 0,$$

with initial condition

$$u(x,y,0) = 0, \quad (x,y) \in \Omega,$$

and Dirichlet boundary conditions

$$u(x,y,t) = 0, \quad (x,y) \in \partial\Omega,\ t > 0.$$

The exact solution is given by $u(x,y,t) = \sin(\pi x)\sin(\pi y)\sin(\pi t)$. This problem has been solved in [29] using cubic B-spline differential quadrature method. We solved this problem with $n = 10$ and reported the maximum absolute errors at $t = 0.1$ and $t = 1.0$ in Table 11 to be compared with the results in [29]. In order to solve the problem in time direction we used the implicit fourth order Runge–Kutta method.

Table 12
Maximum absolute errors (MAE) at $t = 0.1$ for Example 4.7-A.

$n = 16$	Our method $r = 3$	DQM [28]	Collocation [45]	FDM [46]
$\theta = 40,\ \beta = 4$	1.08×10^{-6}	4.35×10^{-4}	1.71×10^{-3}	1.86×10^{-3}
$\theta = 20,\ \beta = 4$	1.73×10^{-6}	2.20×10^{-4}	4.44×10^{-4}	4.49×10^{-3}
$\theta = 10,\ \beta = 5$	1.85×10^{-6}	1.84×10^{-4}	6.44×10^{-4}	6.75×10^{-4}

Table 13
Maximum absolute errors (MAE) at $t = 3.6$ for Example 4.7-B.

n	Our method $r = 3$	[32] Meth. 1	[32] Meth. 2	[32] Meth. 3
021	2.01×10^{-3}	3.10×10^{-3}	3.29×10^{-3}	11.6×10^{-3}
041	6.31×10^{-4}	1.61×10^{-3}	1.68×10^{-3}	1.48×10^{-3}
081	1.23×10^{-4}	7.70×10^{-4}	8.30×10^{-4}	7.60×10^{-4}
161	4.02×10^{-5}	2.80×10^{-4}	3.50×10^{-4}	4.30×10^{-4}

Example 4.7 (*1D Time-dependent problem*)**.** A: Consider the dissipated nonlinear one-dimensional wave equation

$$\frac{\partial^2 u}{\partial t^2} + 2\theta \frac{\partial u}{\partial t} + \beta^2 u = \frac{\partial^2 u}{\partial x^2} + (3 + \beta^2 - 4\theta)e^{-2t}\sinh(x), \quad x \in (0, 1),\ t > 0,$$

subjected to the initial conditions

$$u(x, 0) = \sinh(x), \qquad u_t(x, 0) = -2\sinh(x),$$

along with the boundary conditions

$$u(0, t) = 0, \qquad u(1, t) = \sinh(1)e^{-2t}.$$

The exact solution is $u(x, t) = e^{-2t}\sinh(x)$. We solved this problem to compare our results with two the other methods based on the same basis functions. This problem has been solved in [28,45] using cubic B-spline. In [28] the cubic B-spline basis functions have been used to construct a differential quadrature algorithm. Also in [45] a collocation procedure based on cubic B-spline has been used to approximate the solution. We solved this problem with $r = 3$ using cubic basis functions and reported the maximum absolute errors in Table 12 for different values of θ and β. However all the three methods use the same basis functions but our method is more accurate compared to other ones. We also compared the results with those in [46] which show the efficiency of our method.

B: Next we considered the nonlinear Burgers' equation

$$u_t + uu_x = \epsilon u_{xx}, \quad x \in (0, 1.2),\ t > 1,$$

subjected to the initial condition

$$u(x, 1) = \frac{x}{1 + \exp\left(\frac{x^2 - 1/4}{4\epsilon}\right)}, \quad x \in (0, 1.2), \tag{4.1}$$

along with the homogeneous Dirichlet boundary conditions. The exact solution can be written as

$$u(x, t) = \frac{x/t}{1 + \sqrt{t/\tau}\exp\left(x^2/(4\epsilon t)\right)}, \tag{4.2}$$

where $\tau = \exp\left(\frac{1}{8\epsilon}\right)$. This problem has been solved in [32] using differential quadrature method based on cubic B-spline. We solved this problem with $\epsilon = 0.005$ for various values of n and compared the maximum absolute errors with the results in [32]. The results are tabulated in Table 13.

5. Conclusion

In this paper a new method based on differential quadrature is developed to the solution of differential equations. At first the B-spline basis functions of degree r have been used as test functions to find the weighting coefficients of the method. A set of points combined of grid points and mid points of the uniform partition has been used to constructing the algorithm. The method is applied to the solution of nonlinear boundary problem of order m. Also some error bounds are obtained using the definition of spline collocation. The implementation of the method for multi-dimensional PDEs including biharmonic and Poisson equations are presented in the paper as well. Also the method was successfully examined to approximate the solution of test problems including 2D and 3D biharmonic and Poisson equations. The practical orders of convergence which are reported are in good agreement with theoretical results. Also the CPU runtimes and relative errors have been tabulated to be compared with other existing methods. Finally, to be able to compare the method with other existing spline based DQM algorithms, we solve some problems of time dependent partial differential equations.

Acknowledgment

The research was supported by University of Kurdistan by project number 4/31300.

Appendix

Here we will present some error bounds for differential quadrature method based on B-spline functions of degree r for a boundary value problem of order θ. Consider the following boundary value problem

$$\begin{aligned} Lu &= f \quad x \in \Omega \\ Bu &= g \quad x \in \partial\Omega \end{aligned} \tag{A.1}$$

where L and B are differential operators of order θ and $\theta - 1$ respectively. If we use our proposed algorithm based on the rth degree B-spline basis functions with $r > \theta - 1$ to approximate the solution of (A.1), then we have the following results.

Theorem A.1. *Let $u(x_i)$ be the* DQM *approximation to the solution of* (A.1) *based on odd degree B-spline basis functions $B_{r,k}(x)$ on the uniform partition Δ with the step size h. If the exact solution satisfies $u(x) \in C^{r+2}[\Omega]$, then we have*

$$(Lu - f)\big|_{x_i} = O\left(h^{r+1-2\left[\frac{\theta}{2}\right]}\right), \quad x_i \in \Delta \tag{A.2}$$

where $[\theta]$ is the integer part of θ.

Theorem A.2. *Let $u(x_i)$ be the* DQM *approximation to the solution of* (A.1) *based on even degree B-spline basis functions $B_{r,k}(x)$ on the uniform partition Δ with the step size h. If the exact solution satisfies $u(x) \in C^{r+2}[\Omega]$, then we have*

$$(Lu - f)\big|_{x_i} = O\left(h^{r-2\left[\frac{\theta-1}{2}\right]}\right), \quad x_i \in \Delta, \tag{A.3}$$

where $[\theta]$ is the integer part of θ.

References

[1] S. Timoshenko, A Course in the Theory of Elasticity, Naukova Dumka, Kiev, 1972.
[2] S. Lurie, V. Vasiliev, The Biharmonic Problem of the Theory of Elasticity, Gordon and Breach Publishers, London, 1995.
[3] S.P. Timoshenko, History of Strength of Materials, With a Brief Account of the History of Theory of Elasticity and Theory of Structures, McGraw-Hill, New York, 1953.
[4] C.A.J. Fletcher, Computational Techniques for Fluid Dynamics, Springer, New York, 1991.
[5] A.D. Pierce, Acoustics: An Introduction to Its Physical Principles and Applications, Acoustical Society of America, Woodbury, New York, 1989.
[6] I. Harari, E. Turkel, Accurate finite difference methods for time-harmonic wave propagation, J. Comput. Phys. 119 (1995) 252–270.
[7] U. Schumann, R.A. Sweet, Fast Fourier transforms for direct solution of Poissons equation with staggered boundary conditions, J. Comput. Phys. 75 (1988) 123–127.
[8] J. Castillo, G. Miranda, Mimetic Discretization Methods, CRC Press, 2013.
[9] Z. Shi, Y.Y. Cao, Q.J. Chen, Solving 2D and 3D Poisson equations and biharmonic equations by the Haar wavelet method, Appl. Math. Model. 36 (2012) 5143–5161.
[10] Z. Shi, Y.Y. Cao, A spectral collocation method based on Haar wavelets for Poisson equations and biharmonic equations, Appl. Math. Model. 54 (2011) 2858–2868.
[11] M.M. Gupta, J. Kouatchou, Symbolic derivation of finite difference approximations for the three-dimensional Poisson equation, Numer. Methods Partial Differential Equations 14 (5) (1998) 593–606.
[12] E.C. Romao, M.D. Campos, L.F.M. Moura, Application of the galerkin and least-squares finite element methods in the solution of 3D Poisson and Helmholtz equations, Comput. Math. Appl. 62 (2011) 4288–4299.
[13] K.K. Viswanathan, P.V. Navaneethakrishnan, Free vibration study of layered cylindrical shells by collocation with splines, J. Sound Vib. 260 (2003) 807–827.
[14] K.K. Viswanathan, S.K. Lee, Free vibration of laminated cross-ply plates including shear deformation by spline method, Int. J. Mech. Sci. 49 (2007) 352–363.
[15] K.K. Viswanathan, K.S. Kim, Free vibration of antisymmetric angle-ply-laminated plates including transverse shear deformation: Spline method, Int. J. Mech. Sci. 50 (2008) 1476–1485.
[16] K.K. Viswanathan, K.S. Kim, J.H. Lee, H.S. Koh, J.B. Lee, Free vibration of multi-layered circular cylindrical shell with cross-ply walls, including shear deformation by using spline function method, J. Mech. Sci. Tech. 22 (2008) 2062–2075.
[17] R. Bellman, B.G. Kashef, J. Casti, Differential quadrature: A technique for the rapid solution of nonlinear partial differential equations, J. Comput. Phys. 10 (1972) 40–52.
[18] F. Civan, Differential cubature for multidimensional problems, in: W.G. Vogt, Mickle M.H. (Eds.), Proc. 20th Ann. Pittsburgh Conf. Model. and Simul., Instrument Society of American, North Carolina, 1989, pp. 1843–1847.
[19] C. Shu, B.E. Richards, High resolution of natural convection in a square cavity by generalized differential quadrature, in: Proc. 3rd Conf. Adv. Numer. Meth. Eng., Theory and Appl., Swansea, UK, 2, 1990, pp. 978–985.
[20] C. Shu, Generalized differential-integral quadrature and application to the simulation of incompressible viscous flows including parallel computation (Ph.D. thesis), Univ of Glasgow, UK, 1991.
[21] C. Shu, YT. Chew, Fourier expansion-based differential quadrature and its application to helmholtz eigenvalue problems, Commun. Numer. Methods. Eng. I3 (8) (1997) 643–653.
[22] C. Shu, H. Xue, Explicit computation of weighting coefficients in the harmonic differential quadrature, J. Sound Vib. 204 (3) (1997) 549–555.
[23] H. Zhong, Spline-based differential quadrature for fourth order equations and its application to Kirchhoff plates, Appl. Math. Model. 28 (2004) 353–366.
[24] Q. Guo, H. Zhong, Non-linear vibration analysis of beams by a spline-based differential quadrature method, J. Sound Vib. 269 (2004) 413–420.
[25] H. Zhong, M. Lan, Solution of nonlinear initial-value problems by the spline-based differential quadrature method, J. Sound Vib. 296 (2006) 908–918.

[26] A. Korkmaz, A.M. Aksoy, I. Dag, Quartic B-spline differential quadrature method, Int. J. Nonlinear Sci. 11 (4) (2011) 403–411.
[27] G. Arora, B.K. Singh, Numerical solution of Burgers' equation with modiffied B-spline differential quadrature method, Appl. Math. Comput. 224 (2013) 166–177.
[28] R.C. Mittal, S. Dahiya, Numerical simulation on hyperbolic diffusion equations using modified cubic B-spline differential quadrature methods, Comput. Math. Appl. 70 (5) (2015) 737–749.
[29] R.C. Mittal and, Sumita Dahiya, Numerical solutions of differential equations using modified b-spline differential quadrature method, in: Mathematical Analysis and Its Applications, in: Springer Proceedings in Mathematics and Statistics, 143, 2015, pp. 509–523.
[30] R.C. Mittal, S. Dahiya, A study of quintic b-spline based differential quadrature method for a class of semi-linear Fisher–Kolmogorov equations, Alex. Eng. J. 55 (3) (2016) 2893–2899.
[31] A. Korkmaz, I. Dag, Cubic B-spline differential quadrature methods for the advection-diffusion equation, Internat. J. Numer. Methods Heat Fluid Flow 22 (2012) 1021–1036.
[32] A. Korkmaz, I. Dag, Cubic B-spline differential quadrature methods and stability for Burgers equation, Int. J. Comput. Aided Eng. Soft. 30 (3) (2013) 320–344.
[33] A. Korkmaz, I. Dag, Numerical simulations of boundary-forced RLW equation with cubic B-spline based differential quadrature methods, Arab. J. Sci. Eng. 38 (2013) 1151–1160.
[34] A. Bashan, B. Karakoc, T. Geyikli, Approximation of the KdVB equation by the quintic B-spline differential quadrature method, Kuwait J. Sci. 42 (2) (2015) 67–92.
[35] D. Barrera, P. Gonzalez, F. Ibnez, M.J. Ibnez, A general spline differential quadrature method based on quasi-interpolation, J. Comput. Appl. Math. 275 (2015) 465–479.
[36] D. Barrera, P. Gonzalez, F. Ibnez, M.J. Ibnez, On spline-based differential quadrature, J. Comput. Appl. Math. 275 (2015) 272–280.
[37] F. Tornabene, N. Fantuzzi, F. Ubertini, E. Viola, Strong formulation finite element method based on differential quadrature: A survey, Appl. Mech. Rev. 67 (2) (2015) 020801–1–55.
[38] P.M. Prenter, Spline and Variational Methods, Wiley, New York, 1975.
[39] C. De Boor, A Practical Guide to Splines, Springer, New York, 2001.
[40] C. Shu, Differential Quadrature and its Application in Engineering, Athenaeum Press Ltd., Great Britain, 2000.
[41] T.R. Lucas, Asymptotic expansions for interpolating periodic splines, SIAM J.Numer. Anal. 19 (5) (1982) 1051–1066.
[42] D. Meek, Some new linear relations for even degree polynomial splines on a uniform mesh, BIT 20 (1980) 382–384.
[43] F. Dubeau, J. Savoie, Periodic even degree spline interpolation on a uniform partition, J. Approx. Theory 4 (1985) 43–54.
[44] C.C. Christara, Quadratic spline collocation methods for elliptic partial differential equations, BIT 34 (1994) 33–61.
[45] M. EL-Gamel, A. El-Shenawy, The solution of a time dependent problemby the B-spline method, Appl. Math. Comput. 267 (2014) 254–265.
[46] R. Mohanty, An unconditionally stable difference scheme for the one space dimensional linear hyperbolic equation, Appl. Math. Lett. 17 (2004) 101–105.