Applied Mathematical Modelling 36 (2012) 158–167

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

Differential quadrature method (DQM) and Boubaker Polynomials Expansion Scheme (BPES) for efficient computation of the eigenvalues of fourth-order Sturm–Liouville problems

Uğur Yücel [a], Karem Boubaker [b,*]

[a] Department of Mathematics, Faculty of Science and Letters, Pamukkale University, Denizli 20020, Turkey
[b] Department of Physcics and Chemistry, ESSTT, Tunis University, 63 Rue Sidi Jabeur, 5100 Mahdia, Tunisia

ARTICLE INFO

Article history:
Received 11 January 2011
Received in revised form 11 May 2011
Accepted 16 May 2011
Available online 26 May 2011

Keywords:
Sturm–Liouville problems
Eigenvalue problems
Differential quadrature method (DQM)
Boubaker Polynomials Expansion Scheme (BPES)
Boundary-value problems

ABSTRACT

The differential quadrature method (DQM) and the Boubaker Polynomials Expansion Scheme (BPES) are applied in order to compute the eigenvalues of some regular fourth-order Sturm–Liouville problems. Generally, these problems include fourth-order ordinary differential equations together with four boundary conditions which are specified at two boundary points. These problems concern mainly applied-physics models like the steady-state Euler–Bernoulli beam equation and mechanicals non-linear systems identification. The approach of directly substituting the boundary conditions into the discrete governing equations is used in order to implement these boundary conditions within DQM calculations. It is demonstrated through numerical examples that accurate results for the first kth eigenvalues of the problem, where $k = 1, 2, 3, \ldots$, can be obtained by using minimally $2(k + 4)$ mesh points in the computational domain. The results of this work are then compared with some relevant studies.

© 2011 Elsevier Inc. All rights reserved.

1. Introduction

The Sturm–Liouville boundary value problems for ordinary differential equations play a very important role in both theory and applications. These problems have been used in order to describe a large number of physical, biological and chemical phenomena. One can cite the Sturm–Liouville analytical model of dirt transport in the industrial washing of wool, developed by Caunce et al. [1], the one-dimensional heat and mass diffusion modelling software presented by Barouh and Mikhailov [2] as well as a panoply of boundary valued models [3–7]. Most of these models have been based on expressing solutions as linear combination of eigenvalues, extracted through appropriate methods like, among others, Rayleigh Iterative Scheme (RIS), Optimal Monte Carlo (MAO) algorithm, Implicitly Shifted QR algorithm (ISQR), and Boundary Element Method (BEM). In this context, Alibeigloo and Kani [3] used the differential quadrature method in order to develop an approach combining the state space method and the differential quadrature method (DQM) for studying free vibrations in multilayered shells with embedded piezoelectric layers. Similarly, Eftekhari and Khani [4] combined the finite element method and the differential quadrature element method (DQEM), in order to solve a system of linear second-order ordinary differential equations in time, namely: a sample moving load problem, while Peng et al. [5] focused on a semi-analytic approach for studying geometrically nonlinear vibration of circular plates. In this study [5], Linstedt–Poincaré perturbation method was carried out

* Corresponding author.
E-mail address: mmbb11112000@yahoo.fr (K. Boubaker).

0307-904X/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.apm.2011.05.030

in order to approximate solutions in the conjoint space- time domain. Several models of clamped and simply supported circular plates have ben successfully tested.

Recently, Yücel [6] considered a special kind of boundary value problem known as a Sturm–Liouville problem. It is equivalent to a second-order ordinary differential equation of the form

$$-y'' + q(x)y = \lambda y, \tag{1}$$

with main boundary conditions of the type:

$$y(0) = y(\pi) = 0. \tag{2}$$

The differential quadrature method (DQM) [7] was used for determining the eigenvalues of this problem. It was shown that the (DQM) produces highly accurate results for the eigenvalues of the problem (1) and (2) when compared with other published results. For more details see [6] and references therein.

The present work is undertaken to explore the efficiency and accuracy of the BPES and DQM methods in the computation of eigenvalues of fourth-order Sturm–Liouville problems. The problem at hand is more complicated than the second order case in that there are four boundary conditions to be implemented. Having two boundary conditions at each boundary point of the computational domain is a challenging problem in the application of the DQM.

We will apply the DQM and BPES for finding the eigenvalues of the following fourth-order non-singular Sturm–Liouville problem:

$$y^{(4)} - (s(x)y')' + q(x)y = \lambda y, \quad a < x < b, \tag{3}$$

where the functions $q(x)$, $s(x)$, and $s'(x)$ are in $L^1(a,b)$, and the interval (a,b) is finite. We consider the above equation with four boundary conditions specified at both ends of the domain (a,b), two boundary conditions at the end $x = a$, and other two boundary conditions at the end $x = b$. Basically, there are three types of boundary conditions commonly used with Eq. (3) in applications. In this work, we will only consider the following two types of boundary conditions and their combinations:

$$y = 0 \quad \text{and} \quad \frac{dy}{dx} = 0, \tag{4}$$

for the clamped end, and

$$y = 0 \quad \text{and} \quad \frac{d^2y}{dx^2} = 0, \tag{5}$$

for the simply supported end.

It is well known that the eigenvalues of the problem (3)–(5) are bounded from below. They can be ordered: $\lambda_0 \leqslant \lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_k \leqslant \cdots$, where $\lambda_k \to \infty$ as $k \to \infty$, and where each eigenvalue has multiplicity at most 2. For a more extensive exposition of the theory see [8,9].

Numerically, not much work was done on fourth-order problems compared to second-order. In 1997, Greenberg and Marletta [10] released a software package, named SLEUTH (Sturm–Liouville Eigenvalues Using Theta Matrices), dealing with the computation of eigenvalues of fourth-order Sturm–Liouville problems, which is the only code available in this regard. This situation contrasts with the availability of many software packages dealing with the second-order case, like SLEIGN [11], SLEIGN2 [12], and SLEGDGE [13].

There is a continued interest in the numerical solution of the fourth-order Sturm–Liouville problems with the aim to improve convergence rates and ease of implementation of different algorithms. Chanane [14,15] introduced a novel series representation for the boundary/characteristic function associated with fourth order Sturm–Liouville problems using the concepts of Fliess series and iterated integrals. The fourth power of the zeros of this characteristic function are the eigenvalues of the problem. Few examples were provided and the results were in agreement with the output of SLEUTH [10]. Recently, Attili and Lesnic [16] used the Adomian decomposition method (ADM) to solve fourth-order eigenvalue problems. Syam and Siyyam [17] developed a numerical technique for finding the eigenvalues of fourth-order non-singular Sturm–Liouville problems. More recently, Chanane [18] has enlarged the scope of the Extended Sampling Method [19] which was devised initially for second-order Sturm–Liouville problems to fourth-order ones. Abbasbandy and Shirzadi [20] applied the homotopy analysis method (HAM) to numerically approximate the eigenvalues of the second and fourth-order Sturm–Liouville problems.

In the literature, to the best knowledge of the author of this paper, there is no study on the DQM applications to fourth-order Sturm–Liouville problems. On the other hand, the DQM is an efficient discretization technique for obtaining accurate numerical solutions using a considerably small number of grid points. Bellman et al. [21] introduced this method in the early seventies for solving linear and nonlinear partial differential equations. Then the method was improved by [22–24]. DQM has shown good performance in solving initial and boundary value problems [25–27].

In the DQM, derivatives of a function with respect to a coordinate direction are expressed as linear weighted sums of all the functional values at all mesh (grid) points along that direction. These weighting coefficients are determined using test functions. Among the different test functions, the Lagrange interpolation polynomial is widely employed since it has no grid points limitation. This leads to polynomial-based differential quadrature (PDQ) which is suitable for most engineering problems. Concerning problems with periodic behaviours, polynomial approximation may not be the best choice for the true solution. In contrast, Fourier series expansion can be the best approximation giving the Fourier expansion-based differential quadrature (FDQ). The ease for computation of weighting coefficients in explicit formulations [7,23] for both cases, is based on the analysis of function approximation and linear vector space.

In this paper, both the DQM and BPES methods are used for obtaining eigenvalues of the considered problem (3)–(5). The paper is organized as follows. We summarize the DQM in Section 2. The Boubaker Polynomials Expansion Scheme is given in Section 3. Application of DQM to the fourth order Sturm–Liouville problems is developed in Section 4. Several numerical examples are discussed in Section 5 and some conclusions are drawn in Section 6.

2. Differential quadrature method

The DQM was presented for the first time [21] in the framework of solving differential equations. This method refers to the quadrature method in deriving the derivatives of a function. It follows that the partial derivative of a function with respect to a space variable can be approximated by a weighted linear combination of function values at some intermediate points in that variable.

In order to show the mathematical representation of DQM, we consider a single variable function $f(x)$ on the domain $a \leqslant x \leqslant b$; then the nth order derivative of the function $f(x)$ at an intermediate point (grid point) x_i can be written as:

$$\left.\frac{d^n f}{dx^n}\right|_{x=x_i} = \sum_{j=1}^{N} w_{ij}^{(n)} f(x_j) \quad i = 1, 2, \ldots, N, \; n = 1, 2, \ldots, N-1, \tag{6}$$

where $w_{ij}^{(n)}$ is the weighting coefficient of the nth derivative and N is the number of grid points in the whole domain $(a = x_1, x_2, \ldots, x_N = b)$.

As it can be seen from (6), two important factors control the quality of the approximation resulting from the application of DQM. These factors are the values of the weighting coefficients and the positions of the discrete variables. Once the weighting coefficients are determined, the bridge to link the derivatives in the governing differential equation and the functional values at the mesh points is established. In other words, with the weighting coefficients, one can easily use the functional values to compute the derivatives. Note that for multi-dimensional problems each derivative is approximated in the respective direction similarly.

In order to determine the weighting coefficients in Eq. (6), $f(x)$ must be approximated by some test functions. The primary requirements for the choices of the test functions are of differentiability and smoothness.

2.1. Polynomial-based differential quadrature (PDQ)

If the test functions are chosen as the Lagrange interpolation polynomials, the weighting coefficients of the first- and second-order derivatives in explicit formulations are available in [7], and they are given, respectively, by

$$w_{ij}^{(1)} = \frac{M^{(1)}(x_i)}{(x_i - x_j) M^{(1)}(x_j)}, \quad \text{for} \quad i \neq j, \quad i, j = 1, 2, \ldots, N w_{ii}^{(1)} = -\sum_{\substack{j=1 \\ i \neq 1}}^{N}, \tag{7}$$

$$w_{ij}^{(2)} = 2 w_{ij}^{(1)} \left(w_{ii}^{(1)} - \frac{1}{x_i - x_j} \right), \quad \text{for} \quad j \neq i, \quad i, j = 1, 2, \ldots, N$$

$$w_{ii}^{(2)} = -\sum_{\substack{j=1 \\ i \neq j}}^{N} w_{ij}^{(2)}, \tag{8}$$

where

$$M^{(1)}(x_k) = \prod_{\substack{m=1 \\ m \neq k}}^{N} (x_k - x_m) \tag{9}$$

and x_i, $i = 1, 2, \ldots, N$, are the coordinates of grid points which may be chosen arbitrarily. The weighting coefficients for the third- and higher-order derivatives may be obtained by the matrix multiplication approach, described in detail in [7].

2.2. Fourier expansion-based differential quadrature (FDQ)

When the function $f(x)$ is approximated by a Fourier series expansion, the explicit formulae for computing the weighting coefficients of the first- and second-order derivatives are available in [7], and they are given, respectively, by

$$\begin{aligned} w_{ij}^{(1)} &= \frac{\pi}{2L}\frac{P(x_i)}{\sin\left[(x_i-x_j)\frac{\pi}{2L}\right]P(x_j)}, \quad \text{for} \quad j\neq i, \quad i,j=1,2,\ldots,N, \\ w_{ii}^{(1)} &= -\sum_{\substack{j=1\\ i\neq j}}^{N} w_{ij}^{(1)}, \end{aligned} \tag{10}$$

$$\begin{aligned} w_{ij}^{(2)} &= w_{ij}^{(1)}\left\{2w_{ii}^{(1)} - \frac{\pi}{L}\cot\left[(x_i-x_j)\frac{\pi}{2L}\right]\right\}, \quad \text{for} \quad j\neq i, \quad i,j=1,2,\ldots,N, \\ w_{ii}^{(2)} &= -\sum_{\substack{j=1\\ i\neq j}}^{N} w_{ij}^{(2)}, \end{aligned} \tag{11}$$

where L is the length of the interval (physical domain) and

$$P(x_k) = \prod_{\substack{m=0\\ m\neq k}}^{N} \sin\left[(x_k-x_m)\frac{\pi}{2L}\right]. \tag{12}$$

For higher order derivatives we use the matrix multiplication approach to compute the weighting coefficients.

2.3. Choice of the grid point distributions

The selection of locations of the sampling points plays a significant role in the accuracy of the solution of the differential equations. Using equally spaced points (uniform grid) can be considered to be a convenient and easy selection method. For a domain specified by $a \leqslant x \leqslant b$ and discretized by N points, then the coordinate of any point i can be evaluated by

$$x_i = a + \frac{i-1}{N-1}(b-a). \tag{13}$$

Quite frequently, the DQM delivers more accurate solutions with a set of unequally spaced points (non-uniform grid). The so-called Chebyshev–Gauss–Lobatto points, which were first used by [23] and whose advantage has been discussed by [25], are well accepted in the DQM as follows:

$$x_i = a + \frac{1}{2}\left(1-\cos\left(\frac{i-1}{N-1}\pi\right)\right)(b-a), \tag{14}$$

for a domain $a \leqslant x \leqslant b$ again.

2.4. Implementation of boundary conditions

Proper implementation of the boundary conditions is also very important for the accurate numerical solution of differential equations. Essential and natural boundary conditions can be approximated by DQM. Using the DQM for solving differential equations, we actually satisfy the governing equations at each sampling point of the domain, so we have one equation for each point, for each unknown. To satisfy the boundary conditions, at the boundary points, the boundary condition equations are satisfied instead of the governing equations. In other words, in the resulting system of algebraic equations from DQM, each boundary condition replaces the corresponding field equation. This procedure is straightforward when there is one boundary condition at each boundary and when we have distributed the sampling points so that there is one point at each boundary.

Note that we have two boundary conditions specified at both ends given by Eqs. (4) and (5). The fact of imposing two conditions at the same point is a big and real challenge for the DQM, because in the DQM we have only one quadrature equation at one point while two boundary conditions are to be implemented. To eliminate the difficulties in implementing two conditions at a single boundary point, four approaches have been introduced. These are the δ-technique, the modified weighting coefficient matrix approach, the approach of directly substituting the boundary conditions into the discrete governing equations, and the general approach. The details of these approaches can be found in [7] and references therein.

In this work, we will use the approach of directly substituting the boundary conditions into the discrete governing equations. This approach was proposed by Shu and Du [26] to implement the simply supported, clamped conditions and their combinations. The essence of the approach is that the Dirichlet condition is implemented at the boundary point while the derivative condition is discretized by the DQM. This will be described in more detail in Section 4.

3. Boubaker Polynomials Expansion Scheme method

The Boubaker polynomials are integer-coefficient polynomial sequences which have been associated to several applied physics problems [28–37]. The first monomial definition of the Boubaker polynomials appeared in a physical study that yielded an analytical solution to heat equation inside a spray pyrolysis model. This monomial definition is traduced by

$$B_n(X) = \sum_{p=0}^{\xi(n)} \left[\frac{(n-4p)}{(n-p)} C_{n-p}^{p}\right] . (-1)^p \cdot X^{n-2p} \tag{15}$$

where

$$\xi(n) = \left\lfloor \frac{n}{2} \right\rfloor = \frac{2n + ((-1)^n - 1)}{4} \quad (\text{The symbol}: \lfloor\ \rfloor \text{ designates the floor function}).$$

The Boubaker polynomials have also a recursive relation:

$$\begin{cases} B_m(X) = X \times B_{m-1}(X) - B_{m-2}(X), & \text{for } m > 2, \\ B_2(X) = X^2 + 2; \\ B_1(X) = X; \\ B_0(X) = 1. \end{cases} \tag{16}$$

The characteristic differential equation of the Boubaker polynomials is:

$$\begin{cases} A_n y'' + B_n y' - C_n y = 0, \\ \text{with}: \begin{cases} A_n = (x^2-1)(3nx^2+n-2), \\ B_n = 3x(nx^2+3n-2), \\ C_n = -n(3n^2x^2+n^2-6n+8), \end{cases} \end{cases} \tag{17}$$

3.1. Application of the Boubaker Polynomials Expansion Scheme

Recently, it has been demonstrated [30–36] that each 4q-order Boubaker polynomial has got exactly $2q-1$ real positive roots, which are contained exclusively in the domain]0; 2[. The arithmetical properties of the minimal real positive root denoted α_n gave the fundaments of the Boubaker Polynomials Expansion Scheme (BPES), which was used in different applied physics studies.

According to the BPES definition, for a complex function $f(x)$ of a real argument x defined in the domain $[-a; a]$, the $4n$-Boubaker Polynomials Expansion Scheme (BPES) is performed by applying the expression:

$$f(x) = \frac{1}{2N_0} \sum_{q=1}^{N_0} \xi_q \cdot B_{4q}\left(x \frac{\alpha_q}{a}\right), \tag{18}$$

where α_q is $4q$-Boubaker polynomial minimal root, N_0 is a prefixed integer, and ξ_q ($q = 1, \ldots, N_0$) are complex coefficients.

According to this formulation, a weak solution to the equation:

$$\Im(f(x)) = Z_0, \tag{19}$$

where $\Im$ is a known linear operator, Z_0 is a given complex number, is obtained by calculating the set of complex coefficients ξ_n ($n = 1, \ldots, N_0$) which minimizes the real functional $\Lambda(x)$:

$$\Lambda(x) = \left| \Im\left(\frac{1}{2N_0} \sum_{q=1}^{N_0} \xi_q \cdot B_{4q}\left(x \frac{\alpha_q}{a}\right)\right) - Z_0 \right|. \tag{20}$$

While solving a Dirichlet–Newmann boundary-type differential equation, the advantage of the BPES lies in embedding the exogenous boundary condition thanks to the $4q$-Boubaker polynomials properties [28–30].

In order to show the mathematical representation of the BPES, we consider the expansion:

$$f(x) = \frac{1}{2N_0} \sum_{k=1}^{N_0} \xi_k \times B_{4k}\left(x \times \frac{r_k}{b-a}\right), \tag{21}$$

where B_{4k} are the $4k$-order Boubaker polynomials, is the normalized time ($x \in [0, L]$), r_k are B_{4k} minimal positive roots, N_0 is a prefixed integer, and $\xi_k|_{k=1 \cdots N_0}$ are unknown pondering real coefficients. Consequently, it comes for Eq. (3) that:

$$\frac{1}{2N_0}\left(\frac{r_k}{b-a}\right)^4\sum_{k=1}^{N_0}\xi_k\times\frac{d^4B_{4k}}{dx^4}\left(x\times\frac{r_k}{b-a}\right)-\frac{1}{2N_0}s'(x)\times\sum_{k=1}^{N_0}\xi_k\times\frac{dB_{4k}}{dx}\left(x\times\frac{r_k}{b-a}\right)$$
$$-\frac{1}{2N_0}s(x)\left(\frac{r_k}{b-a}\right)^2\sum_{k=1}^{N_0}\xi_k\times\frac{d^2B_{4k}}{dx^2}\left(x\times\frac{r_k}{b-a}\right)-(q(x)-\lambda)\sum_{k=1}^{N_0}\zeta_k\times B_{4k}\left(x\times\frac{r_k}{b-a}\right)=0. \tag{22}$$

Concerning the boundary conditions expressed through Eqs. (4) and (5), the BPES protocol ensures their validity regardless main equation features. In fact, thanks to Boubaker polynomials first derivatives properties [28–30].

The BPES solution is obtained by determining the set of coefficients where $\tilde{\xi}_k\big|_{k=1\cdots N_0}$ that minimizes the absolute difference $\varDelta_{N_0}$:

$$\begin{cases}\varDelta_{N_0}=\left|\left(\frac{1}{2N_0}\sum\limits_{k=1}^{N_0}\tilde{\xi}_k\times\varLambda_k\right)-\left(\frac{1}{2N_0}\sum\limits_{k=1}^{N_0}\tilde{\xi}_k\times\varLambda'_k\right)\right|\\ \text{with}:\\ \varLambda_k=\left(\frac{r_k}{b-a}\right)^4\int_a^b\frac{dB_{4k}}{dx^4}\left(x\times\frac{r_k}{b-a}\right)dx,\\ \varLambda'_k=\int_a^b(s'(x)\times\frac{dB_{4k}}{dx}(X)+s(x)\left(\frac{r_k}{b-a}\right)^2\frac{d^2B_{4k}}{dx^2}(X)+(q(x)-\lambda)\sum\limits_{k=1}^{N_0}\xi_k\times B_{4k}(X))dx,\\ X=x\times\frac{r_k}{b-a}.\end{cases} \tag{23}$$

4. Application of the differential quadrature method

In this section, the DQM is applied to solve Eq. (3) with Eqs. (4) and (5). For the numerical computation, the continuous solution is approximated by the functional values at discrete points. Now, we assume that the computational domain $a\leqslant x\leqslant b$ is divided into $N-1$ intervals with coordinates of the grid points given as $a=x_1,x_2,\ldots,x_N=b$. By applying the PDQ or FDQ method, Eq. (3) can be discretized as

$$\sum_{k=1}^{N}w_{ik}^{(4)}y_k-s_i\sum_{k=1}^{N}w_{ik}^{(2)}y_k-s'_i\sum_{k=1}^{N}w_{ik}^{(1)}y_k+q_iy_i=\lambda y_i, \tag{24}$$

where N is the number of grid points in the x-direction, $w_{ik}^{(n)}$, n = 1, 2, 4, the weighting coefficients of the nth order derivative, and s_i, s'_1, q_i, y_i the functional values at the grid point x_i. With the coordinates of mesh points given by Eq. (13) and (14), the PDQ or FDQ weighting coefficients can be easily computed. When the PDQ method is used, $w_{ik}^{(1)}$ and $w_{ik}^{(2)}$ are computed by Eqs. (7) and (8), while for the FDQ approach, the weighting coefficients $w_{ik}^{(1)}$, $w_{ik}^{(2)}$ are computed by (10) and (11). The weighting coefficients for the fourth order derivative $w_{ik}^{(4)}$ are computed by matrix multiplication technique for both the PDQ and FDQ.

For the proper implementation of the boundary conditions, we now describe the approach of directly substituting the boundary conditions into the discrete governing equations. For any combination of Eqs. (4) and (5) at the two ends, the discrete boundary conditions using DQM can be written as

$$y_1=0, \tag{25a}$$

$$\sum_{k=1}^{N}w_{1k}^{(n_0)}y_k=0, \tag{25b}$$

$$y_N=0, \tag{25c}$$

$$\sum_{k=1}^{N}w_{Nk}^{(n_1)}y_k=0, \tag{25d}$$

where n_0 and n_1 may be taken as either 1 or 2. By choosing the values of n_0 and n_1, the above equations give us four sets of boundary conditions. Eqs (25a) and (25c) can be easily substituted into Eq. (24). This is not the case for Eqs. (25b) and (25d). However, one can couple these two equations together to give two solutions, y_2 and y_{N-1}, as

$$y_2=\frac{1}{AX}\sum_{k=3}^{N-2}AX1_ky_k, \tag{26a}$$

$$y_{N-1}=\frac{1}{AX}\sum_{k=3}^{N-2}AXN_ky_k, \tag{26b}$$

where

$$AX = w^{(n_0)}_{1N-1} w^{(n_1)}_{N2} - w^{(n_0)}_{12} w^{(n_1)}_{NN-1}, \tag{27a}$$

$$AX1_k = w^{(n_0)}_{1k} w^{(n_1)}_{NN-1} - w^{(n_0)}_{1N-1} w^{(n_1)}_{Nk}, \tag{27b}$$

$$AXN_k = w^{(n_0)}_{12} w^{(n_1)}_{Nk} - w^{(n_0)}_{1k} w^{(n_1)}_{N2}. \tag{27c}$$

According to Eqs. (26), y_2 and y_{N-1} are expressed in terms of $y_3, y_4, \ldots, y_{N-2}$, and can be easily substituted into Eq. (24). We should note that Eqs. (25) provides four boundary conditions. In total, we have N unknowns $y_1, \ldots, y_N$. In order to close the system, the discretized Eq. (24) has to be applied at $(N - 4)$ mesh points. This can be achieved by applying Eq. (24) at the interior grid points $x_3, x_4, \ldots, x_{N-2}$. Substituting Eqs. (25a), (25c), and (26) into Eq. (24) gives

$$\sum_{k=3}^{N-2} C^4_{ik} y_k - s_i \sum_{k=3}^{N-2} C^2_{ik} y_k - s'_i \sum_{k=3}^{N-2} C^1_{ik} y_k + q_i y_i = \lambda y_i \quad \text{for } i = 3, 4, \ldots, N-2, \tag{28}$$

where

$$C^1_{ik} = w^{(1)}_{ik} + \frac{w^{(1)}_{i2} AX1_k + w^{(1)}_{iN-1} AXN_k}{AX}, \tag{29a}$$

$$C^2_{ik} = w^{(2)}_{ik} + \frac{w^{(2)}_{i2} AX1_k + w^{(2)}_{iN-1} AXN_k}{AX}, \tag{29b}$$

$$C^4_{ik} = w^{(4)}_{ik} + \frac{w^{(4)}_{i2} AX1_k + w^{(4)}_{iN-1} AXN_k}{AX}. \tag{29c}$$

It can be seen that Eq. (28) has $(N - 4)$ equations with $(N - 4)$ unknowns, which can be written in matrix form as

$$[A]\{y\} = \lambda\{y\}, \tag{30}$$

where $[A]$ is a $(N - 4) \times (N - 4)$ matrix, $\{y\}$ a vector of $(N - 4)$ unknowns.

Equation (30) is an eigenvalue equation system. We can obtain the λ values from the eigenvalues of matrix $[A]$. This can be done by using various methods. In this work, we use a FORTRAN IMSL Routine called DEVLRG. Routine DEVLRG computes the eigenvalues of a real matrix. The matrix is first balanced. Elementary or Gauss similarity transformations with partial pivoting are used to reduce this balanced matrix to a real upper Hessenberg matrix. A hybrid double-shifted LR-QR algorithm is used to compute the eigenvalues of the Hessenberg matrix.

Note that it is necessary to analyze the error resulting from the approximation of a function and its derivatives. Shu [22] has given a thorough error analysis in his PhD thesis. Therefore it will not be discussed here in this work.

We also note that the PDQ method is an extension of finite difference methods, and is actually the highest order finite difference scheme [7]. Eq. (6) can be applied to both interior points and boundary points, and can also be applied to a uniform mesh or a non-uniform mesh. As the highest order finite difference scheme, the PDQ method is a global approximation approach since it uses all the functional values in the whole computational domain.

5. Numerical results

In this section, to demonstrate the efficiency and accuracy of the BPES and DQM methods, as attempted earlier by tempted by Malekzadeh [38], Yücel [6], Yildirim et al. [30] and Robati and Barani [39]. We will present three of our numerical results of fourth-order Sturm–Liouville problems using the method outlined in the previous sections.

Example 4.1. We first consider the following sample fourth-order eigenvalue problem

$$\begin{cases} y^{(4)} = \lambda y, & 0 < x < 1, \\ y(0) = y'(0) = 0, & y(1) = y''(1) = 0, \end{cases} \tag{31}$$

which corresponds to the case $s(x) = q(x) = 0$, $a = 0$, and $b = 1$ in Eq. (3). This problem has been considered by several authors [16,17]. It has been also considered by [27], but with λ^2 instead of λ on the right-hand side of the differential equation associated in elasticity, to the steady-state Euler–Bernoulli beam equation for the deflection y of a vibrating beam. The exact eigenvalues in the latter case can be obtained by solving

$$\tanh(\sqrt{\lambda}) - \tan(\sqrt{\lambda}) = 0. \tag{32}$$

This analytical solution is commonly available in vibration textbooks (see for example [32]).

Table 1
Relative errors of the DQM results for several values of the number of grid points *N*.

k	λ_k	ε_k $(N = 20)$		ε_k $(N = 30)$	
	Exact	PDQ	FDQ	PDQ	FDQ
1	237.72106753	7.59×10^{-9}	1.47×10^{-5}	7.59×10^{-9}	3.24×10^{-9}
2	2496.48743786	4.44×10^{-8}	1.12×10^{-5}	4.45×10^{-8}	4.83×10^{-8}
3	10867.58221698	1.94×10^{-9}	1.43×10^{-5}	1.71×10^{-8}	2.21×10^{-8}
4	31780.09645408	4.50×10^{-8}	1.29×10^{-5}	2.36×10^{-8}	1.72×10^{-8}
5	74000.849349156	3.97×10^{-5}	2.21×10^{-5}	2.99×10^{-8}	1.86×10^{-8}
6	148634.47728577	1.43×10^{-4}	1.86×10^{-5}	4.77×10^{-8}	6.41×10^{-8}
7	269123.43482664	4.08×10^{-3}	4.15×10^{-5}	9.61×10^{-10}	2.24×10^{-8}
8	451247.99471928	1.11×10^{-2}	2.79×10^{-5}	1.74×10^{-8}	5.36×10^{-8}
9	713126.24789600	9.02×10^{-2}	4.86×10^{-5}	3.16×10^{-6}	8.40×10^{-8}
10	1075214.10347396	2.06×10^{-2}	3.15×10^{-4}	9.31×10^{-6}	1.17×10^{-7}

We apply here both the PDQ and FDQ methods with the grid point distribution given by Eq. (14) to compute the eigenvalues of the problem (31). Since we have clamped end condition on the left end and simply supported end condition on the right end of the computational domain (0, 1), we take $n_0 = 1$ and $n_1 = 2$ in Eqs. (25b) and (25d), respectively. The performance of the DQM is measured by the relative error ε_k which is defined as

$$\varepsilon_k = \left| \frac{\lambda_k - \lambda_k^{(\mathrm{DQM})}}{\lambda_k} \right| \quad k = 1, 2, 3, \ldots, \tag{33}$$

where $\lambda_k^{(\mathrm{DQM})}$ indicates kth algebraic eigenvalues obtained by DQM and λ_k are the exact eigenvalues obtained by squaring the solutions of Eq. (31). It should be noted that we use, here in this work, *Maple 12* to obtain the solutions of the nonlinear equation (32).

Table 1 lists the relative errors of the DQM results with different number of mesh points N. It should be noted that since the equation system to be solved has the dimension $(N - 4) \times (N - 4)$, the minimum number of grid points N to be used in the calculations has to be five. Here, it is interesting to also note that in order to have good approximations to the first kth eigenvalues, at least $2(k + 4)$ grid points have to be used. In other words, $N = 2(k + 4)$. It can be observed from Table 1 that the accuracy of the computed eigenvalues by the FDQ method is better than the PDQ approach for the eigenvalues of higher index. In both methods, the computed values for the lower eigenvalues have a better accuracy than those for the higher eigenvalues. As the number of grid points further increased, the accuracy of the DQM results, especially for the higher eigenvalues, can be further improved as shown in Table 1.

We study two examples originally introduced in Chanane [19] and reproduced in Attili and Lesnic [16] and Syam and Siyyam [17], albeit without proper reference and in Chanane [18].

Example 4.2. We now consider the following fourth order eigenvalue problem related to mechanicals non-linear systems identification [14]

$$\begin{cases} y^{(4)} - 0.02x^2y'' - 0.04xy' + (0.0001x^4 - 0.02)y = \lambda y, \quad 0 < x < 5, \\ y(0) = y''(0) = 0, \quad y(5) = y''(5) = 0, \end{cases} \tag{34}$$

which corresponds to the case $s(x) = 0.02x^2$, $q(x) = 0.0001x^4 - 0.02$, $a = 0$, and $b = 5$ in Eq. (3). The PDQ and FDQ methods with the grid point distribution given by Eq. (14) are applied to compute the eigenvalues of the problem (34), along with the BPES. In this case, we have simply supported end conditions on both ends of the computational domain (0, 5). Therefore, we take $n_0 = 2$ and $n_1 = 2$ in Eqs. (25b) and (25d), respectively.

Table 2
Comparison of eigenvalues of Example 4.2.

k	λ_k					
	Attili and Lesnic [16]	Syam and Siyyam [17]	Chanane [18]	PDQ $N = 30$	BPES $N_0 = 28$	FDQ $N = 30$
1	0.21505086437	0.21505086437	0.21505086437	0.21505086437	0.2150534833	0.21505086264
2	2.75480993468	2.75480993468	2.75480993468	2.75480993468	2.7548119066	2.75480993579
3	13.2153515405	13.2153515406	13.2153515406	13.2153515406	13.215354539	13.2153515488
4	40.9508197591	40.9508197591	40.9508193487	40.9508197591	40.950824112	40.9508197584
5	99.0534781381	99.0534781381	–	99.0534780633	99.053483098	99.0534780408
6	204.354493489	204.354493489	–	204.355732256	204.35573676	204.355732287

Table 3
Comparison of eigenvalues of Example 4.3.

k	λ_k				
	Attili and Lesnic [16]	Chanane [18]	PDQ $N = 30$	BPES $N_0 = 28$	FDQ $N = 30$
1	0.8669025023997106	0.866902502399465	0.866902502399437	0.866904356009764	0.866902502602292
2	6.357686448145815	6.357686448174460	6.357686448146623	6.357689457641119	6.357686448439836
3	23.992746850281375	23.99274697506674	23.99274685030316	23.99274982234511	23.99274686509660
4	64.97866759571622	64.97863591597007	64.97866759501693	64.97866711123549	64.97866761311830
5	144.28062803844648	–	144.2806269273482	144.2806276621998	144.2806272956158
6	280.58602048195377	–	280.6009632780809	280.600963292234	280.6009637443962

Table 2 shows the comparison of our results obtained using $N = 30$, for the first six eigenvalues of the problem (34), with the results of Attili and Lesnic [16], Syam and Siyyam [17], and Chanane [18]. It can be seen from Table 2 that the results of BPES, PDQ and FDQ methods are in excellent egreement with the results of [16–18]. We should note here that only the first four computed eigenvalues were reported in [18].

Example 4.3. Our second example is the following fourth order eigenvalue problem [14]

$$\begin{cases} y^{(4)} - 0.02x^2y'' - 0.04xy' + (0.0001x^4 - 0.02)y = \lambda y, & 0 < x < 5, \\ y(0) = y'(0) = 0, \quad y(5) = y'(5) = 0, \end{cases} \tag{35}$$

which is the same as the problem given in Example 4.2, except the boundary conditions. Here, we have clamped end conditions on both ends of the computational domain $(0, 5)$ and therefore we take $n_0 = 1$ and $n_1 = 1$ in Eqs. (25b) and (25d), respectively. We again apply both the PDQ and FDQ methods with the grid point distribution given by Eq. (14) to numerically compute the eigenvalues of the problem (35) along with the BPES, by fixing $a = 0$ and $b = 5$. Table 3 lists the first six computed eigenvalues using $N = 30$. We also have the results of Attili and Lesnic [16] and Chanane [18] on the second and third column of Table 3, respectively. As shown in the table, there is excellent agreement between the results of this work and the results of [16,18]. We should note again here that only the first four computed eigenvalues were reported in [18].

6. Conclusions

In this work, BPES, PDQ and FDQ methods have been applied in order to compute the eigenvalues of fourth-order Sturm–Liouville applied physics problems. A particular attention in the DQM calculations has been given to the implementation of the boundary conditions. Having two boundary conditions at a single boundary point of the computational domain (a, b) to be implemented is a challenging problem for the DQM. To eliminate this difficulty, several approaches have been proposed in the literature. Here, we have used the approach of direct substitution of boundary conditions into discrete governing equations. The dimension of the equation system using this approach is $(N - 4) \times (N - 4)$ where N is the number of grid points.

Through the test example which has exact solution, it was found that in order to obtain accurate numerical results using the DQM for the first kth eigenvalues of the problem, where $k = 1, 2, 3, \ldots$, the minimum number of grid points, N must be equal to $2(k + 4)$. It was also found that as the number of grid points is further increased to above $2(k + 4)$, the accuracy of the DQM results for both approaches can be further improved.

Computed eigenvalues obtained by using the BPES, PDQ and FDQ methods are also compared with other published works in the literature. Excellent agreements are observed between the results of present work and the results of previously published works [16–18]. Therefore, we conclude that the BPES and DQM produces accurate results for the eigenvalues of the fourth-order Sturm–Liouville problems considered in this work. We also suggest the DQM and BPES approaches for the numerical solution of the fourth-order problems since the latter methods gives the possibility of inherenting exogeneous boundary conditions. Indeed, it will be interesting to see how the method works for the sixth-order Sturm–Liouville problems. This will be considered in a future work.

Acknowledgements

The author (U. Yücel) greatfully acknowledges the support of Scientific Research Projects Coordination Unit, Pamukkale University, Denizli, Turkey.

References

[1] J.F. Caunce, S.I. Barry, G.N. Mercer, A spatially dependent model for washing wool, Appl. Math. Modell. 32 (2008) 389–404.
[2] V.A. Barouh, M.D. Mikhailov, One-dimensional heat and mass diffusion modeling, Appl. Math. Model. 13 (1989) 568–573.
[3] A. Alibeigloo, A.M. Kani, 3D free vibration analysis of laminated cylindrical shell integrated piezoelectric layers using the differential quadrature method, Appl. Math. Model. 34 (2010) 4123–4137.

[4] S.A. Eftekhari, M. Khani, A coupled finite element-differential quadrature element method and its accuracy for moving load problem, Appl. Math. Model. 34 (2010) 228–237.

[5] J.S. Peng, Y.Q. Yuan, J. Yang, S. Kitipornchai, A semi-analytic approach for the nonlinear dynamic response of circular plates, Appl. Math. Model. 33 (2009) 4303–4313.

[6] U. Yücel, Approximations of Sturm–Liouville eigenvalues using differential quadrature (DQ) method, J. Comput. Appl. Math. 192 (2006) 310–319.

[7] C. Shu, Differential Quadrature and Its Application in Engineering, Springer-Verlag, London, 2000.

[8] L. Greenberg, An oscillation method for fourth order self-adjoint two-point boundary value problems with nonlinear eigenvalues, SIAM J. Math. Anal. 22 (1991) 1021–1042.

[9] L. Greenberg, M. Marletta, Oscillation theory and numerical solution of fourth order Sturm–Liouville problems, IMA J. Numer. Anal. 15 (1995) 319–356.

[10] L. Greenberg, M. Marletta, Algorithm 775: the code SLEUTH for solving fourth order Sturm–Liouville problems, ACM Trans. Math. Software 23 (1997) 453–493.

[11] P.B. Bailey, M.K. Gordon, L.F. Shampine, Automatic solution of the Sturm–Liouville problem, ACM Trans. Math. Software 4 (1978) 193–208.

[12] P.B. Bailey, W.N. Everitt, A. Zeal, Computing Eigenvalues of Singular Sturm–Liouville Problems, Results in Mathematics, vol. 20, Birkhauser, Basel, 199l.

[13] C. Fulton, S. Pruess, Mathematical Software for Sturm–Liouville Problems, NSF Final Report for Grants DMS88-13113 and DMS88-00839, Computational Mathematics Division, 1991.

[14] B. Chanane, Eigenvalues of fourth order Sturm–Liouville problems using Fliess series, J. Comput. Appl. Math. 96 (1998) 91–97.

[15] B. Chanane, Fliess series approach to the computation of the eigenvalues of fourth order Sturm–Liouville problems, Appl. Math. Lett. 15 (2002) 459–463.

[16] B.S. Attili, D. Lesnic, An efficient method for computing eigenelements of Sturm–Liouville fourth-order boundary value problems, Appl. Math. Comput. 182 (2006) 1247–1254.

[17] M.I. Syam, H.I. Siyyam, An efficient technique for finding the eigenvalues of fourth-order Sturm–Liouville problems, Chaos Solitons Fract. 39 (2009) 659–665.

[18] B. Chanane, Accurate solutions of fourth order Sturm–Liouville problems, J. Comput. Appl. Math. 234 (2010) 3064–3071.

[19] B. Chanane, Sturm–Liouville problems with parameter dependent potential and boundary conditions, J. Comput. Appl. Math. 212 (2008) 282–290.

[20] S. Abbasbandy, A. Shirzadi, A new application of the homotopy analysis method: solving the Sturm–Liouville problems, Commun. Nonlinear Sci. Numer. Simulat. doi:10.1016/j.cnsns.2010.04.004.

[21] 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.

[22] C. Shu, Generalized Differential-integral Quadrature and Application to the Simulation of Incompressible Viscous Flows Including Parallel Computation, Ph.D. Thesis, University of Glasgow, UK, 1991.

[23] C. Shu, B.E. Richards, Application of generalized differential quadrature to solve two-dimensional incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 15 (1992) 791–798.

[24] C. Shu, B.C. Khoo, Y.T. Chew, K.S. Yeo, Numerical studies of unsteady boundary layer flows past an impulsively started circular cylinder by GDQ and GIQ approaches, Comput. Methods Appl. Mech. Eng. 135 (1996) 229–241.

[25] C.W. Bert, M. Malik, Differential quadrature method in computational mechanics: a review, Appl. Mech. Rev. 49 (1996) 1–28.

[26] C. Shu, H. Du, Implementation of clamped and simply supported boundary conditions in the GDQ free vibration analysis of beams and plates, Int. J. Solids Struct. 34 (1997) 819–835.

[27] M. Malik, H. Huy Dang, Vibration analysis of continuous systems by differential transformation, Appl. Math. Comput. 96 (1998) 17–26.

[28] P. Barry, A. Hennessy, Meixner-Type results for Riordan arrays and associated integer sequences, Section 6: the Boubaker polynomials, J. Integer Seq. 13 (2010) 1–34.

[29] M. Agida, A.S. Kumar, A Boubaker Polynomials Expansion Scheme solution to random Love equation in the case of a rational kernel, El. J. Theor. Phys. 7 (2010) 319–326.

[30] A. Yildirim, S.T. Mohyud-Din, D.H. Zhang, Analytical solutions to the pulsed Klein-Gordon equation using Modified Variational Iteration Method (MVIM) and Boubaker Polynomials Expansion Scheme (BPES), Comput. Math. Appl. (2010), doi:10.1016/j.camwa.2009.12.026.

[31] S. Slama, J. Bessrour, M. Bouhafs, K.B. Ben Mahmoud, Numer. Heat Transfer A 55 (2009) 401–404.

[32] S. Slama, M. Bouhafs, K.B. Ben Mahmoud, A Boubaker polynomials solution to heat equation for monitoring A3 point evolution during resistance spot welding, Int. J. Heat Technol. 26 (2) (2008) 141–146.

[33] S. Tabatabaei, T. Zhao, O. Awojoyogbe, F. Moses, Cut-off cooling velocity profiling inside a keyhole model using the Boubaker polynomials expansion scheme, Int. J. Heat Mass Transfer 45 (2009) 1247–1255.

[34] S. Fridjine, M. Amlouk, A new parameter: an ABACUS for optimizing functional materials using the Boubaker polynomials expansion scheme, Mod. Phys. Lett. B 23 (2009) 2179–2182.

[35] A. Belhadj, J. Bessrour, M. Bouhafs, L. Barrallier, Experimental and theoretical cooling velocity profile inside laser welded metals using keyhole approximation and Boubaker polynomials expansion, J. Therm. Anal. Calorimetry 97 (2009) 911–920.

[36] A.S. Kumar, An analytical solution to applied mathematics-related Love's equation using the Boubaker Polynomials Expansion Scheme, J. Franklin Inst. 347 (2010) 1755–1761.

[37] A. Milgram, The stability of the Boubaker polynomials expansion scheme (BPES)-based solution to Lotka–Volterra problem, J. Theor. Biol. 271 (2011) 157–158.

[38] P. Malekzadeh, A two-dimensional layerwise-differential quadrature static analysis of thick laminated composite circular arches, Appl. Math. Model. 33 (2009) 850–1861.

[39] A. Robati, G.A. Barani, Modeling of water surface profile in subterranean channel by differential quadrature method (DQM), Appl. Math. Model. 33 (2009) 1295–1305.