

RESEARCH

Open Access



Extended cubic B-spline method for solving a linear system of second-order boundary value problems

Ahmed Salem Heilat^{*}, Nur Nadiah Abd Hamid and Ahmad Izani Md. Ismail

^{*}Correspondence:
ahmed_heilat@yahoo.com
School of Mathematical
Sciences, Universiti Sains
Malaysia, 11800 Penang,
Malaysia

Abstract

A method based on extended cubic B-spline is proposed to solve a linear system of second-order boundary value problems. In this method, two free parameters, λ_1 and λ_2 , play an important role in producing accurate results. Optimization of these parameters are carried out and the truncation error is calculated. This method is tested on three examples. The examples suggest that this method produces comparable or more accurate results than cubic B-spline and some other methods.

Keywords: Boundary value problem, System, Linear, Extended cubic B-spline

Background

It is well-known that many real life phenomena in physics and engineering can be modelled by systems of linear and nonlinear differential equations. One class of these systems is of second order boundary value problems. The existence of solution to such system was studied in Chen et al. (2005), Cheng and Zhong (2005), Thompson and Tisdell (2002). Consider the following linear system of second-order boundary value problems:

$$\begin{cases} u''(x) + a_1(x)u'(x) + a_2(x)u(x) + a_3(x)v''(x) + a_4(x)v'(x) + a_5(x)v(x) = f_1(x) \\ v''(x) + b_1(x)v'(x) + b_2(x)v(x) + b_3(x)u''(x) + b_4(x)u'(x) + b_5(x)u(x) = f_2(x) \\ u(0) = u(1) = 0, v(0) = v(1) = 0, \end{cases} \quad (1)$$

where $a \leq x \leq b$, $f_1(x)$ and $f_2(x)$ are continuous functions, and $a_i(x)$ and $b_i(x)$, for $i = 1, 2, 3, 4, 5$, are real-valued functions of x that are smooth enough.

There are many studies on the solutions of linear and nonlinear systems of second-order boundary value problems approximately. Amongst others are variational iteration, reproducing kernel, sinc-collocation, modified homotopy analysis, continuous genetic algorithm, He's homotopy perturbation, Laplace homotopy analysis, homotopy perturbation-reproducing kernel, and local radial basis function based differential quadrature methods (Lu 2007; Geng and Cui 2007; Dehghan and Saadatmandi 2007; Bataineh et al. 2009; Arqub and Abo-Hammour 2014; Saadatmandi et al. 2009; Ogunlaran and Ademola 2015; Geng and Cui 2011; Dehghan and Nikpour 2013). The main purpose of our

present study is to apply a spline function in solving Eq. (1). This equation had already been treated using cubic B-spline, cubic B-spline scaling functions, sinc-collocation, and spline collocation approaches (Caglar and Caglar 2009; Dehghan and Lakestani 2008; El-Gamel 2012; Khuri and Sayfy 2009).

In 2003, Han and Liu proposed an extension of cubic B-spline of degree four with one free parameter, λ . This parameter is introduced within the basis function in order to increase the flexibility of the spline curve (Han and Liu 2003). Then, Xu and Wang generalized the extension to degree five and six (Gang and Guo-Zhao 2008). Our goal is to apply the simplest B-spline extension, that is, extended cubic B-spline of degree four, in solving Eq. (1). Linear and singular boundary value problems has already been solved using extended cubic B-spline of degree four and an approach of optimizing λ has been proposed (Hamid et al. 2011; Goh et al. 2011). The results are promising and thus become the motivation of this study.

In this paper, extended cubic B-spline will be discussed along with the extended cubic B-spline method (ECBM). Optimization of the free parameters and calculations on the truncation error will follow. Three examples will be presented and comparisons with other methods will be made.

Extended cubic B-spline method

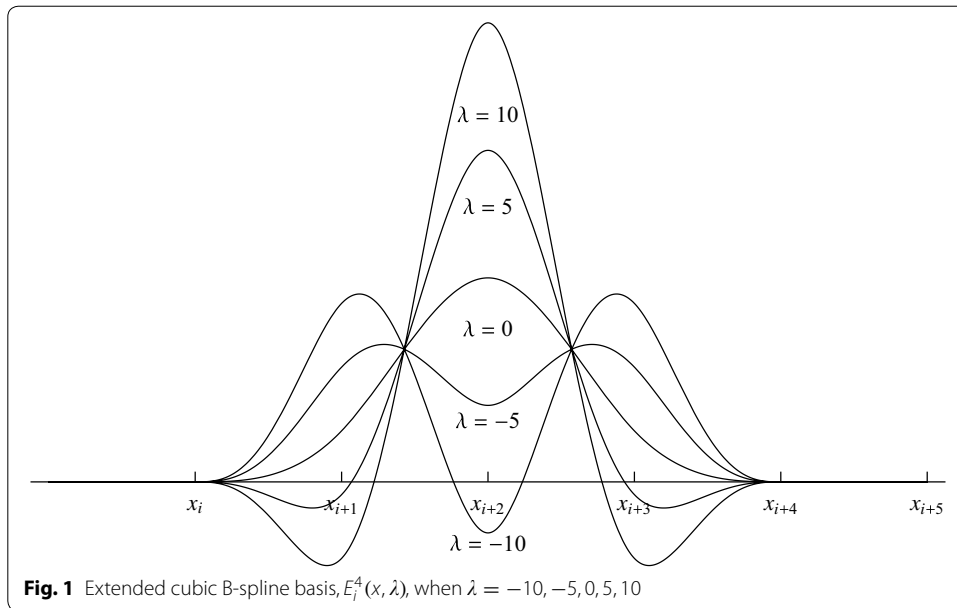
Extended cubic B-spline is an extension of B-spline Gang and Guo-Zhao (2008). One free parameter, λ , is introduced within the basis function where this parameter can be used to alter the shape of the generated curve. The value of λ can be varied to obtain different numerical results. In this study, this value is optimized to produce approximate solutions with the least error.

Extended cubic B-spline

Suppose that $\{x_i\}_{i=0}^n$ is a uniform partition of a finite interval $[a, b]$ with $n \in \mathbb{Z}^+$ such that $a = x_0 < x_1 < \dots < x_n = b$. The partition can be extended using $h = \frac{b-a}{n}$, $x_0 = a$, $x_i = x_0 + ih$, and $i \in \mathbb{Z}$. Extended cubic B-spline basis function is established from a linear combination of the cubic B-spline basis function (Gang and Guo-Zhao 2008). Here, the blending function of degree four, E_i^4 , as shown in (2), is used.

$$E_i^4(x, \lambda) = \frac{1}{24h^4} \begin{cases} 4h(1 - \lambda)(x - x_i)^3 + 3\lambda(x - x_i)^4, & x \in [x_i, x_{i+1}], \\ (4 - \lambda)h^4 + 12h^3(x - x_{i+1}) + 6h^2(2 + \lambda)(x - x_{i+1})^2 - 12h(x - x_{i+1})^3 - 3\lambda(x - x_{i+1})^4, & x \in [x_{i+1}, x_{i+2}], \\ (4 - \lambda)h^4 + 12h^3(x_{i+3} - x) + 6h^2(2 + \lambda)(x_{i+3} - x)^2 - 12h(x_{i+3} - x)^3 - 3\lambda(x_{i+3} - x)^4, & x \in [x_{i+2}, x_{i+3}], \\ 4h(1 - \lambda)(x_{i+4} - x)^3 + 3\lambda(x_{i+4} - x)^4, & x \in [x_{i+3}, x_{i+4}], \end{cases} \tag{2}$$

Extended cubic B-spline basis will degenerate into cubic B-spline basis when $\lambda = 0$. For $-8 \leq \lambda \leq 1$, cubic B-spline and extended cubic B-spline share the same properties: partition of unity, non-negativity, C^2 continuity, and local support Hamid (2010). Figure 1 displays a family of extended cubic B-spline bases with different values of λ .



From the basis function, an arbitrary spline curve can be generated by the following formula:

$$U(x, \lambda) = \sum_{i=-3}^{n-1} C_i E_i^4(x, \lambda), x \in [x_0, x_n], \quad C_i \in \mathbb{R}.$$

As a result, $U(x, \lambda)$ is a piecewise polynomial functions of degree 4. Similarly, for $-8 \leq \lambda \leq 1$, cubic B-spline and extended cubic B-spline curves have the same properties: symmetry, geometric invariability, and convex hull Goh et al. (2011). The values of E_i and its derivatives E_i', E_i'' at the nodal points are tabulated in Table 1.

Extended cubic B-spline interpolation

Suppose that the spline curves $U(x, \lambda_1)$ and $V(x, \lambda_2)$ are the approximation to the exact solutions, $u(x)$ and $v(x)$, respectively, defined as follows:

$$\begin{cases} U(x, \lambda_1) = \sum_{i=-3}^{n-1} C_i E_i^4(x, \lambda_1), & x \in [x_0, x_n], \quad C_i \in \mathbb{R} \\ V(x, \lambda_2) = \sum_{i=-3}^{n-1} D_i E_i^4(x, \lambda_2), & x \in [x_0, x_n], \quad D_i \in \mathbb{R} \end{cases} \quad (3)$$

Therefore, from Table 1, the values of $U(x, \lambda_1), U'(x, \lambda_1), U''(x, \lambda_1), V(x, \lambda_2), V'(x, \lambda_2),$ and $V''(x, \lambda_2)$ at knot x_i can be simplified into Eqs. (4) and (5).

Table 1 Coefficient of $E_i, E_i',$ and E_i''

x	x_i	x_{i+1}	x_{i+2}	x_{i+3}	x_{i+4}
E_i	0	$\frac{4-\lambda}{24}$	$\frac{8+\lambda}{12}$	$\frac{4-\lambda}{24}$	0
E_i'	0	$\frac{-1}{2h}$	$\frac{0}{h}$	$\frac{1}{2h}$	0
E_i''	0	$\frac{2+\lambda}{2h^2}$	$\frac{-2-\lambda}{h^2}$	$\frac{2+\lambda}{2h^2}$	0

$$\begin{cases} U(x_i, \lambda_1) = C_{i-3} \left(\frac{4-\lambda_1}{24} \right) + C_{i-2} \left(\frac{8+\lambda_1}{12} \right) + C_{i-1} \left(\frac{4-\lambda_1}{24} \right) \\ U'(x_i, \lambda_1) = C_{i-3} \left(-\frac{1}{2h} \right) + C_{i-2} \left(\frac{0}{h} \right) + C_{i-1} \left(\frac{1}{2h} \right) \\ U''(x_i, \lambda_1) = C_{i-3} \left(\frac{2+\lambda_1}{2h^2} \right) + C_{i-2} \left(\frac{-2-\lambda_1}{h^2} \right) + C_{i-1} \left(\frac{2+\lambda_1}{2h^2} \right) \end{cases} \tag{4}$$

$$\begin{cases} V(x_i, \lambda_2) = D_{i-3} \left(\frac{4-\lambda_2}{24} \right) + D_{i-2} \left(\frac{8+\lambda_2}{12} \right) + D_{i-1} \left(\frac{4-\lambda_2}{24} \right) \\ V'(x_i, \lambda_2) = D_{i-3} \left(-\frac{1}{2h} \right) + D_{i-2} \left(\frac{0}{h} \right) + D_{i-1} \left(\frac{1}{2h} \right) \\ V''(x_i, \lambda_2) = D_{i-3} \left(\frac{2+\lambda_2}{2h^2} \right) + D_{i-2} \left(\frac{-2-\lambda_2}{h^2} \right) + D_{i-1} \left(\frac{2+\lambda_2}{2h^2} \right) \end{cases} \tag{5}$$

Equations (4) and (5) will be used in simplifying the terms in the system of boundary value problems.

Solution of system of second order boundary value problem

In this part, a collocation approach based on extended cubic B-spline basis functions is used to obtain the numerical solutions of a class of systems of linear second order boundary value problems (1). The approximate solution (3) should satisfy the differential equation at points x_i . This can be done by putting (3) into (1) resulting in Eqs. (6)–(9).

$$\begin{aligned} U''(x_i, \lambda_1) + a_1(x_i)U'(x_i, \lambda_1) + a_2(x_i)U(x_i, \lambda_1) \\ + a_3(x_i)V''(x_i, \lambda_2) + a_4(x_i)V'(x_i, \lambda_2) + a_5(x_i)V(x_i, \lambda_2) = f_1(x_i), \quad i = 0, 1, 2, \dots, n \end{aligned} \tag{6}$$

$$\begin{aligned} V''(x_i, \lambda_2) + b_1(x_i)V'(x_i, \lambda_2) + b_2(x_i)V(x_i, \lambda_2) \\ + b_3(x_i)U''(x_i, \lambda_1) + b_4(x_i)U'(x_i, \lambda_1) + b_5(x_i)U(x_i, \lambda_1) = f_2(x_i), \quad i = 0, 1, 2, \dots, n \end{aligned} \tag{7}$$

$$U(x_i, \lambda_1) = 0, \quad x = 0, n \tag{8}$$

$$V(x_i, \lambda_2) = 0, \quad x = 0, n \tag{9}$$

Equations (4)–(5) are substituted into Eqs. (6)–(9) resulting in a linear system of $2(n + 3)$ equations with $2(n + 3)$ unknowns, $C_{-3}, C_{-2}, \dots, C_{n-1}, D_{-3}, D_{-2}, \dots, D_{n-1}$. This system can be written in the matrix-vector

$$XY = Z, \tag{10}$$

where $Y = [C_{-3}, C_{-2}, \dots, C_{n-1}, D_{-3}, D_{-2}, \dots, D_{n-1}]^T$, $Z = [0, f_1(x_0), \dots, f_1(x_n), 0, 0, f_2(x_0), \dots, f_2(x_n), 0]^T$, and X is a $2(n + 3) \times 2(n + 3)$ matrix given by

$$X = \begin{pmatrix} M_1 & | & M_2 \\ \cdots & \cdots & \cdots \\ M_4 & | & M_3 \end{pmatrix}$$

The four sub-matrices M_1, M_2, M_3 , and M_4 are calculated as follows:

$$\begin{aligned}
 M_1 &= \begin{pmatrix} \frac{4-\lambda_1}{24} & \frac{8+\lambda_1}{12} & \frac{4-\lambda_1}{24} & 0 & \dots & 0 & 0 \\ \alpha_1(x_0) & \beta_1(x_0) & \gamma_1(x_0) & 0 & \dots & 0 & 0 \\ 0 & \alpha_1(x_1) & \beta_1(x_1) & \gamma_1(x_1) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \alpha_1(x_n) & \beta_1(x_n) & \gamma_1(x_n) \\ \cdot & \cdot & \cdot & \cdot & \frac{4-\lambda_1}{24} & \frac{8+\lambda_1}{12} & \frac{4-\lambda_1}{24} \end{pmatrix}_{(n+3) \times (n+3)} \\
 M_2 &= \begin{pmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \alpha_2(x_0) & \beta_2(x_0) & \gamma_2(x_0) & 0 & \dots & 0 & 0 \\ 0 & \alpha_2(x_1) & \beta_2(x_1) & \gamma_2(x_1) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \alpha_2(x_n) & \beta_2(x_n) & \gamma_2(x_n) \\ \cdot & \cdot & \cdot & \cdot & 0 & 0 & 0 \end{pmatrix}_{(n+3) \times (n+3)} \\
 M_3 &= \begin{pmatrix} \frac{4-\lambda_2}{24} & \frac{8+\lambda_2}{12} & \frac{4-\lambda_2}{24} & 0 & \dots & 0 & 0 \\ \alpha_3(x_0) & \beta_3(x_0) & \gamma_3(x_0) & 0 & \dots & 0 & 0 \\ 0 & \alpha_3(x_1) & \beta_3(x_1) & \gamma_3(x_1) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \alpha_3(x_n) & \beta_3(x_n) & \gamma_3(x_n) \\ \cdot & \cdot & \cdot & \cdot & \frac{4-\lambda_2}{24} & \frac{8+\lambda_2}{12} & \frac{4-\lambda_2}{24} \end{pmatrix}_{(n+3) \times (n+3)} \\
 M_4 &= \begin{pmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \alpha_4(x_0) & \beta_4(x_0) & \gamma_4(x_0) & 0 & \dots & 0 & 0 \\ 0 & \alpha_4(x_1) & \beta_4(x_1) & \gamma_4(x_1) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \alpha_4(x_n) & \beta_4(x_n) & \gamma_4(x_n) \\ \cdot & \cdot & \cdot & \cdot & 0 & 0 & 0 \end{pmatrix}_{(n+3) \times (n+3)}
 \end{aligned}$$

The elements of these sub-matrices are specified below for $i = 0, 1, \dots, n$.

$$\begin{aligned}
 \alpha_1(x_i) &= \frac{2 + \lambda_1}{2h^2} - a_1(x_i) \frac{1}{2h} + a_2(x_i) \frac{4 - \lambda_1}{24} \\
 \alpha_2(x_i) &= a_3(x_i) \frac{2 + \lambda_2}{2h^2} - a_4(x_i) \frac{1}{2h} + a_5(x_i) \frac{4 - \lambda_2}{24} \\
 \alpha_3(x_i) &= \frac{2 + \lambda_2}{2h^2} - b_1(x_i) \frac{1}{2h} + b_2(x_i) \frac{4 - \lambda_2}{24} \\
 \alpha_4(x_i) &= b_3(x_i) \frac{2 + \lambda_1}{2h^2} - b_4(x_i) \frac{1}{2h} + b_5(x_i) \frac{4 - \lambda_1}{24} \\
 \beta_1(x_i) &= \frac{-2 - \lambda_1}{h^2} + a_1(x_i) \frac{0}{h} + a_2(x_i) \frac{8 + \lambda_1}{12} \\
 \beta_2(x_i) &= a_3(x_i) \frac{-2 - \lambda_2}{h^2} + a_4(x_i) \frac{0}{h} + a_5(x_i) \frac{8 + \lambda_2}{12} \\
 \beta_3(x_i) &= \frac{-2 - \lambda_2}{h^2} + b_1(x_i) \frac{0}{h} + b_2(x_i) \frac{8 + \lambda_2}{12} \\
 \beta_4(x_i) &= b_3(x_i) \frac{-2 - \lambda_1}{h^2} + b_4(x_i) \frac{0}{h} + b_5(x_i) \frac{8 + \lambda_1}{12} \\
 \gamma_1(x_i) &= \frac{2 + \lambda_1}{2h^2} + a_1(x_i) \frac{1}{2h} + a_2(x_i) \frac{4 - \lambda_1}{24} \\
 \gamma_2(x_i) &= a_3(x_i) \frac{2 + \lambda_2}{2h^2} + a_4(x_i) \frac{1}{2h} + a_5(x_i) \frac{4 - \lambda_2}{24} \\
 \gamma_3(x_i) &= \frac{2 + \lambda_2}{2h^2} + b_1(x_i) \frac{1}{2h} + b_2(x_i) \frac{4 - \lambda_2}{24} \\
 \gamma_4(x_i) &= b_3(x_i) \frac{2 + \lambda_1}{2h^2} + b_4(x_i) \frac{1}{2h} + b_5(x_i) \frac{4 - \lambda_1}{24}
 \end{aligned}$$

$$\begin{aligned}
 D_1(x, \lambda_1, \lambda_2) &= U''(x, \lambda_1, \lambda_2) + a_1(x)U'(x, \lambda_1, \lambda_2) + a_2(x)U(x, \lambda_1, \lambda_2) \\
 &\quad + a_3(x)V''(x, \lambda_1, \lambda_2) \\
 &\quad + a_4(x)V'(x, \lambda_1, \lambda_2) + a_5(x)V(x, \lambda_1, \lambda_2) - f_1(x), \quad x \in [x_0, x_n], \\
 D_2(x, \lambda_1, \lambda_2) &= V''(x, \lambda_1, \lambda_2) + b_1(x)V'(x, \lambda_1, \lambda_2) + b_2(x)V(x, \lambda_1, \lambda_2) + b_3(x)U''(x, \lambda_1, \lambda_2) \\
 &\quad + b_4(x)U'(x, \lambda_1, \lambda_2) + b_5(x)U(x, \lambda_1, \lambda_2) - f_2(x), \quad x \in [x_0, x_n],
 \end{aligned}$$

which can be expanded into Eqs. (16) and (17).

$$D_1(x, \lambda_1, \lambda_2) = \begin{cases} U_1''(x, \lambda_1, \lambda_2) + a_1(x)U_1'(x, \lambda_1, \lambda_2) \\ \quad + a_2(x)U_1(x, \lambda_1, \lambda_2) + a_3(x)V_1''(x, \lambda_1, \lambda_2) \\ \quad + a_4(x)V_1'(x, \lambda_1, \lambda_2) + a_5(x)V_1(x, \lambda_1, \lambda_2) - f_1(x), & x \in [x_0, x_1], \\ U_2''(x, \lambda_1, \lambda_2) + a_1(x)U_2'(x, \lambda_1, \lambda_2) \\ \quad + a_2(x)U_2(x, \lambda_1, \lambda_2) + a_3(x)V_2''(x, \lambda_1, \lambda_2) \\ \quad + a_4(x)V_2'(x, \lambda_1, \lambda_2) + a_5(x)V_2(x, \lambda_1, \lambda_2) - f_1(x), & x \in [x_1, x_2], \\ \vdots \\ U_n''(x, \lambda_1, \lambda_2) + a_1(x)U_n'(x, \lambda_1, \lambda_2) \\ \quad + a_2(x)U_n(x, \lambda_1, \lambda_2) + a_3(x)V_n''(x, \lambda_1, \lambda_2) \\ \quad + a_4(x)V_n'(x, \lambda_1, \lambda_2) + a_5(x)V_n(x, \lambda_1, \lambda_2) - f_1(x), & x \in [x_{n-1}, x_n]. \end{cases} \tag{16}$$

$$D_2(x, \lambda_1, \lambda_2) = \begin{cases} V_1''(x, \lambda_1, \lambda_2) + b_1(x)V_1'(x, \lambda_1, \lambda_2) \\ \quad + b_2(x)V_1(x, \lambda_1, \lambda_2) + b_3(x)U_1''(x, \lambda_1, \lambda_2) \\ \quad + b_4(x)U_1'(x, \lambda_1, \lambda_2) + b_5(x)U_1(x, \lambda_1, \lambda_2) - f_2(x), & x \in [x_0, x_1], \\ V_2''(x, \lambda_1, \lambda_2) + b_1(x)V_2'(x, \lambda_1, \lambda_2) \\ \quad + b_2(x)V_2(x, \lambda_1, \lambda_2) + b_3(x)U_2''(x, \lambda_1, \lambda_2) \\ \quad + b_4(x)U_2'(x, \lambda_1, \lambda_2) + b_5(x)U_2(x, \lambda_1, \lambda_2) - f_2(x), & x \in [x_1, x_2], \\ \vdots \\ V_n''(x, \lambda_1, \lambda_2) + b_1(x)V_n'(x, \lambda_1, \lambda_2) \\ \quad + b_2(x)V_n(x, \lambda_1, \lambda_2) + b_3(x)U_n''(x, \lambda_1, \lambda_2) \\ \quad + b_4(x)U_n'(x, \lambda_1, \lambda_2) + b_5(x)U_n(x, \lambda_1, \lambda_2) - f_2(x), & x \in [x_{n-1}, x_n]. \end{cases} \tag{17}$$

Since $D_1(x, \lambda_1, \lambda_2)$ and $D_2(x, \lambda_1, \lambda_2)$ are piecewise functions with n equations, it is wise to have some representatives from every sub-interval. The representative is taken to be the midpoint of every sub-interval. Therefore, $x_i^* = \frac{x_i+x_{i+1}}{2}$, for $i = 0, 1, \dots, n - 1$. Evaluating $D_1(x, \lambda_1, \lambda_2)$ and $D_2(x, \lambda_1, \lambda_2)$ at $\{x_i^*\}_{i=0}^{n-1}$ would produce a sequence of $2n$ expressions containing λ_1 and λ_2 ,

$$\begin{cases} D_1(x_0^*, \lambda_1, \lambda_2), \\ D_1(x_1^*, \lambda_1, \lambda_2), \\ \vdots \\ D_1(x_{n-1}^*, \lambda_1, \lambda_2), \end{cases} \tag{18}$$

$$\begin{cases} D_2(x_0^*, \lambda_1, \lambda_2), \\ D_2(x_1^*, \lambda_1, \lambda_2), \\ \vdots \\ D_2(x_{n-1}^*, \lambda_1, \lambda_2), \end{cases} \tag{19}$$

By handling Eqs. (18) and (19) as the error at collocation points, the expressions are combined using the two-norm formula resulting equation (20). This equation measures the accuracy of the approximated solution, $U(x, \lambda_1, \lambda_2)$ and $V(x, \lambda_1, \lambda_2)$ without including the exact solution.

$$d_1(\lambda_1, \lambda_2) = \sqrt{\sum_{i=0}^{n-1} [D_1(x_i^*, \lambda_1, \lambda_2)]^2 + \sum_{i=0}^{n-1} [D_2(x_i^*, \lambda_1, \lambda_2)]^2} \tag{20}$$

Also, from Eq. (20) we can obtain $d_2(\lambda_1, \lambda_2)$ which is assumed to be easier to calculate than the former.

$$d_2(\lambda_1, \lambda_2) = \sum_{i=0}^{n-1} [D_1(x_i^*, \lambda_1, \lambda_2)]^2 + \sum_{i=0}^{n-1} [D_2(x_i^*, \lambda_1, \lambda_2)]^2 \tag{21}$$

On the other hand, we can combine the expressions using one-norm formula, as in (22).

$$d_3(\lambda_1, \lambda_2) = \sum_{i=0}^{n-1} |D_1(x_i^*, \lambda_1, \lambda_2)| + \sum_{i=0}^{n-1} |D_2(x_i^*, \lambda_1, \lambda_2)| \tag{22}$$

This is done to make comparisons between results of $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ in terms of computational time and accuracy. $d_3(\lambda_1, \lambda_2)$ is significantly more simplified than the other two. Finally, we can substitute the optimized value of λ_1 and λ_2 in the approximate solution for the problems.

Error estimation

The technique for finding the error estimate as in Kadalbajoo and Kumar (2007) is extended to the system of linear second order differential equations. In this part, a truncation error for the present method in the interval [0, 1] is presented. Suppose that $u(x)$ and $v(x)$ are functions with continuous derivatives in [0, 1]. By using the formulas of $u(x)$ in (4), the following relationship can be obtained.

$$h \left[\left(\frac{4 - \lambda_1}{24} \right) U'(x_{i-1}, \lambda_1) + \left(\frac{8 + \lambda_1}{12} \right) U'(x_i, \lambda_1) + \left(\frac{4 - \lambda_1}{24} \right) U'(x_{i+1}, \lambda_1) \right] = \frac{1}{2} [U(x_{i+1}, \lambda_1) - U(x_{i-1}, \lambda_1)] \tag{23}$$

Similarly, Eqs. (24)–(26) can be derived, where $U'''(x_{i+}, \lambda_1)$ and $U''(x_{i-}, \lambda_1)$ represent $U'''(x_i, \lambda_1)$ in (x_i, x_{i+1}) and (x_{i-1}, x_i) , respectively.

$$h^2 U''(x_i, \lambda_1) = 6[U(x_{i+1}, \lambda_1) - U(x_i, \lambda_1)] - 2h \left[\left(\frac{8 + \lambda_1}{4} \right) U'(x_i, \lambda_1) + \left(\frac{4 - \lambda_1}{4} \right) U'(x_{i+1}, \lambda_1) \right] \tag{24}$$

$$h^3 U'''(x_{i+}, \lambda_1) = 12[U(x_i, \lambda_1) - U(x_{i+1}, \lambda_1)] + 6h[U'(x_i, \lambda_1) + (U'(x_{i+1}, \lambda_1))] \tag{25}$$

$$h^3 U'''(x_{i-}, \lambda_1) = 12[U(x_{i-1}, \lambda_1) - U(x_i, \lambda_1)] + 6h[U'(x_{i-1}, \lambda_1) + U'(x_i, \lambda_1)] \tag{26}$$

By using the operator notation $E(U(x_i)) = U(x_{i+1})$, Eq. (23) can be written as Sastry (2012)

$$h \left[\left(\frac{4 - \lambda_1}{24} \right) E^{-1} + \left(\frac{8 + \lambda_1}{12} \right) + \left(\frac{4 - \lambda_1}{24} \right) E \right] U'(x_i, \lambda_1) = \frac{1}{2} (E - E^{-1}) u(x_i).$$

By expanding $E = e^{hD}$ in powers of hD , we get

$$\begin{aligned} & h \left[\left(\frac{8 + \lambda_1}{12} \right) + \left(\frac{4 - \lambda_1}{12} \right) \left(1 + \frac{h^2 D^2}{2!} + \frac{h^4 D^4}{4!} + \dots \right) \right] U'(x_i, \lambda_1) \\ &= \left(hD + \frac{h^3 D^3}{3!} + \frac{h^5 D^5}{5!} + \dots \right) u(x_i) \end{aligned}$$

Upon simplification, we have

$$\begin{aligned} U'(x_i, \lambda_1) &= \left(D + \frac{h^2 D^3}{3!} + \frac{h^4 D^5}{5!} + \dots \right) \left[1 + \left(\frac{4 - \lambda_1}{12} \right) \left(\frac{h^2 D^2}{2!} + \frac{h^4 D^4}{4!} + \frac{h^6 D^6}{6!} \right) \right]^{-1} u(x_i) \\ &= \left(D + \frac{h^2 D^3}{3!} + \dots \right) \left[1 - \left(\frac{4 - \lambda_1}{12} \right) \left(\frac{h^2 D^2}{2!} + \dots \right) + \left(\frac{4 - \lambda_1}{12} \right)^2 \left(\frac{h^2 D^2}{2!} + \dots \right)^2 \right] u(x_i) \\ &= \left(D + \frac{h^2 D^3}{3!} + \dots \right) \left[1 - \left(\frac{4 - \lambda_1}{24} \right) h^2 D^2 - \left(\frac{4 - \lambda_1}{288} \right) h^4 D^4 - \dots + \left(\frac{(4 - \lambda_1)^2}{576} \right) h^4 D^4 + \dots \right] u(x_i) \\ &= \left(D + \frac{h^2 D^3}{3!} + \frac{h^4 D^5}{5!} + \dots \right) \left[1 - \left(\frac{4 - \lambda_1}{24} \right) h^2 D^2 + \left(\frac{\lambda_1^2 - 6\lambda_1 + 8}{576} \right) h^4 D^4 + \dots \right] u(x_i) \\ &= \left[D - \left(\frac{4 - \lambda_1}{24} \right) h^2 D^3 + \left(\frac{\lambda_1^2 - 6\lambda_1 + 8}{576} \right) h^4 D^5 + \dots + \frac{1}{6} h^2 D^3 - \left(\frac{4 - \lambda_1}{144} \right) h^4 D^5 + \dots \right] u(x_i) \\ &= \left[D + \left(\frac{\lambda_1}{24} \right) h^2 D^3 + \left(\frac{5\lambda_1^2 - 10\lambda_1 - 16}{2280} \right) h^4 D^5 + \dots \right] u(x_i). \end{aligned}$$

Therefore,

$$U'(x_i, \lambda_1) = u'(x_i) + \left(\frac{\lambda_1}{24} \right) h^2 u'''(x_i) + \left(\frac{5\lambda_1^2 - 10\lambda_1 - 16}{2280} \right) h^4 u^{(5)}(x_i) + O(h^6). \tag{27}$$

Similar approach is applied on Eqs. (24)–(26) that results in relations (28)–(30).

$$U''(x_i, \lambda_1) = \left(1 + \frac{\lambda_1}{2} \right) u''(x_i) + \left(\frac{\lambda_1^2 - 4}{48} \right) h^2 u^{(4)}(x_i) + O(h^4) \tag{28}$$

$$U'''(x_i, \lambda_1) = \left(1 + \frac{\lambda_1}{2} \right) u'''(x_i) + \left(\frac{\lambda_1^2 + 4\lambda_1 + 4}{48} \right) h^2 u^{(5)}(x_i) + O(h^4) \tag{29}$$

$$U^{(4)}(x_i, \lambda_1) = \left(1 + \frac{\lambda_1}{2} \right) u^{(4)}(x_i) + \left(\frac{\lambda_1^2 + 2\lambda_1}{48} \right) h^2 u^{(6)}(x_i) + O(h^4) \tag{30}$$

By using $e_1(x) = U(x, \lambda_1) - u(x)$ and substituting relations (27)–(30) in the Taylor series expansion of $e_1(x_i + \theta h)$, we obtain

$$e_1(x_i + \theta h) = \left(\frac{\theta^2 \lambda_1}{24}\right) h^2 u''(x_i) + \left(\frac{1 + 2\theta^2}{24}\right) \theta \lambda_1 h^3 u'''(x_i) + \left(\frac{\lambda_1^2 + 2\theta^2 \lambda_1 - 4}{96}\right) \theta^2 h^4 u^{(4)}(x_i) + O(h^5).$$

Similarly, we can use the definition $e_2(x) = V(x, \lambda_2) - v(x)$ to have

$$e_2(x_i + \theta h) = \left(\frac{\theta^2 \lambda_2}{24}\right) h^2 v''(x_i) + \left(\frac{1 + 2\theta^2}{24}\right) \theta \lambda_2 h^3 v'''(x_i) + \left(\frac{\lambda_2^2 + 2\theta^2 \lambda_2 - 4}{96}\right) \theta^2 h^4 v^{(4)}(x_i) + O(h^5).$$

Therefore, the extended cubic B-spline has a truncation error of order h^2 . Apparently, the value of λ_1 and λ_2 have influences on the order.

Results and discussions

Several examples are discussed to demonstrate the efficiency of the proposed method. The results are compared with that of variational iteration, analytical approximation, sinc-collocation, reproducing kernel, He’s homotopy perturbation, Laplace homotopy analysis, and B-spline methods (Lu 2007; Geng and Cui 2007; Dehghan and Saadatmandi 2007; Saadatmandi et al. 2009; Ogunlaran and Ademola 2015; Caglar and Caglar 2009). The results are also presented with different values of n . Calculations were carried out using Wolfram Mathematica 10 with Intel(R) Core(TM) i5 CPU 3GHz processor, 4.00 GB RAM. The optimization can only be done for $n \leq 5$ due to the computational limit of the computer. Numerical errors are calculated using infinite and two norms, as respectively follows:

$$L_\infty = \max_i |u(x_i) - U(x_i)| \quad \text{or} \quad L_\infty = \max_i |v(x_i) - V(x_i)|$$

$$L_2 = \sqrt{\sum_{i=1}^n (u(x_i) - U(x_i))^2} \quad \text{or} \quad L_2 = \sqrt{\sum_{i=1}^n (v(x_i) - V(x_i))^2}$$

Example 1 Consider the following system Lu (2007),

$$\begin{cases} u''(x) + (2x - 1)u'(x) + \cos(\pi x)v'(x) = f_1(x) \\ v''(x) + xu(x) = f_2(x) \\ u(0) = u(1) = 0, v(0) = v(1) = 0, \end{cases} \tag{31}$$

where $0 < x < 1$, $f_1(x) = -\pi^2 \sin(\pi x) + (2x - 1)\pi \cos(\pi x) + (2x - 1) \cos(\pi x)$, and $f_2(x) = 2 + x \sin(\pi x)$. The exact solutions are $u(x) = \sin(\pi x)$ and $v(x) = x^2 - x$.

Table 2 displays the values of λ_1 and λ_2 when $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ are minimized for $n = 5$. The L_∞ and L_2 for each pair are also presented. From the table, it can be deduced that minimizing $d_3(\lambda_1, \lambda_2)$ is the best option because the results

are comparable and the computational time is significantly less than that of $d_1(\lambda_1, \lambda_2)$ and $d_2(\lambda_1, \lambda_2)$. Therefore, the chosen values of λ_1 and λ_2 are $-6.639145E-02$ and $1.161882E-06$, respectively. Also, it can be observed that minimizing $d_2(\lambda_1, \lambda_2)$ gives similar results with minimizing $d_1(\lambda_1, \lambda_2)$ with significantly less computational time.

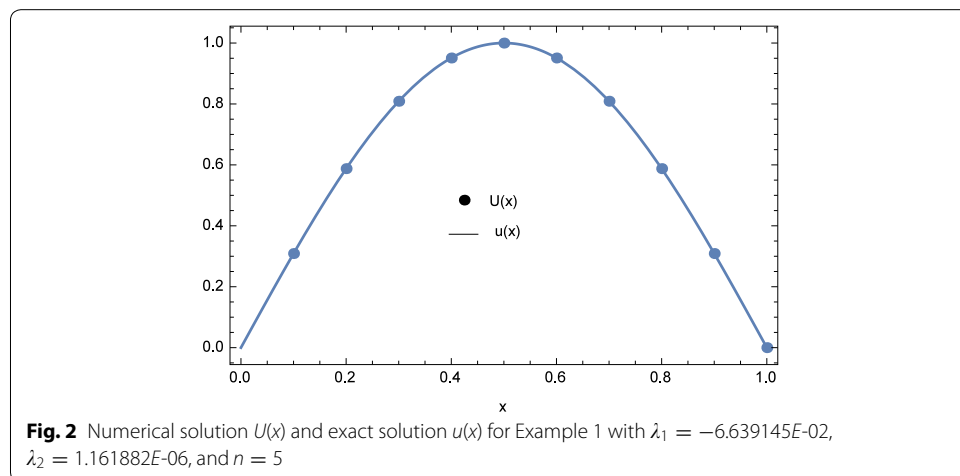
The approximate and exact solutions at the nodal points are displayed in Table 3. From the table, the approximate solutions agree with the exact solutions. Hence, for this example, the results are acceptable and accurate. The plots of the numerical results are shown in Figs. 2 and 3. Comparisons between the L_∞ of ECBM, He’s homotopy perturbation method (Saadatmandi et al. 2009), and Laplace homotopy analysis method (Ogunlaran and Ademola 2015) were made in Table 4. ECBM produced more accurate results than both methods except for the results of $u(x)$ generated by the Laplace homotopy analysis method (Ogunlaran and Ademola 2015). Moreover, the numerical results for Example

Table 2 Computational time and norms for different optimization equations $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ with $n = 5$

Minimization values	$d_1(\lambda_1, \lambda_2)$	$d_2(\lambda_1, \lambda_2)$	$d_3(\lambda_1, \lambda_2)$
λ_1	$-6.639979E-02$	$-6.639979E-02$	$-6.639145E-02$
λ_2	$-1.230437E-06$	$-1.230522E-06$	$1.161882E-06$
Computational time (s)	$1.306340E+04$	$2.728410E+03$	$2.230830E+00$
L_∞	$1.377934E-04$	$1.377934E-04$	$1.413576E-04$
L_2	$2.306527E-04$	$2.306527E-04$	$2.364995E-04$

Table 3 Comparison of ECBM results with the exact solution for Example 1 when $\lambda_1 = -6.639145E-02$, $\lambda_2 = 1.161882E-06$, and $n = 5$

x	Exact solution $u(x)$	Approx. solution $U(x)$	Absolute error $ U(x) - u(x) $	Exact solution $v(x)$	Approx. solution $V(x)$	Absolute error $ V(x) - v(x) $
0.2	0.587785	0.587696	$8.897274E-05$	-0.160000	-0.160004	$3.641560E-06$
0.4	0.951057	0.950915	$1.413501E-04$	-0.240000	-0.240006	$6.478141E-06$
0.6	0.951057	0.950915	$1.413576E-04$	-0.240000	-0.240007	$7.169404E-06$
0.8	0.587785	0.587696	$8.891932E-05$	-0.160000	-0.160005	$4.793718E-06$



1 when $\lambda_1 = -1.0E-03$, $\lambda_2 = 0$, and $n = 41$ are shown in Tables 5 and 6. In this case, the values of λ_1 and λ_2 were obtained by trial and error. From the table, the ECBM produced more accurate results than variational iteration method (VIM) and cubic B-spline method (CBM). The norms for both n are shown in Table 7. It can be observed that ECBM improves the accuracy of CBM significantly.

Example 2 Consider the following equations Khuri and Sayfy (2009),

$$\begin{cases} u''(x) + u'(x) + xu(x) + v'(x) + 2xv(x) = f_1(x) \\ v''(x) + v(x) + 2u'(x) + x^2u(x) = f_2(x) \\ u(0) = u(1) = 0, v(0) = v(1) = 0 \end{cases} \quad (32)$$

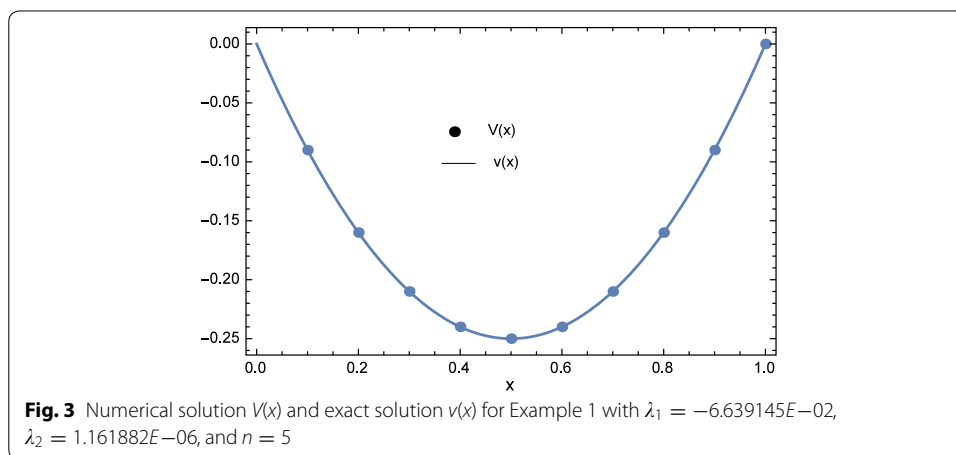


Table 4 L_∞ of He’s homotopy perturbation method Saadatmandi et al. (2009), Laplace homotopy analysis method Ogunlaran and Ademola (2015), and ECBM for Example 1 when $n = 5$

	He’s homotopy perturbation method	Laplace homotopy analysis method	ECBM ($\lambda_1 = -6.639145E-02$, $\lambda_2 = 1.161882E-06$)
$U(x)$	2.1E-04	2.2E-05	1.4E-04
$V(x)$	3.2E-04	1.1E-05	7.2E-06

Table 5 Absolute errors of VIM Lu (2007), CBM Caglar and Caglar (2009), and ECBM results for Example 1 with $n = 41$ for $u(x)$

x	VIM	CBM	ECBM ($\lambda_1 = \lambda_2 = 0$)	ECBM ($\lambda_1 = -1.0E-03, \lambda_2 = 0$)
0.1	3.30E-04	1.40E-04	1.30E-04	2.83E-06
0.2	2.51E-03	2.80E-04	2.56E-04	5.55E-06
0.3	7.84E-03	3.90E-04	3.60E-04	7.81E-06
0.4	1.66E-02	4.60E-04	4.28E-04	9.30E-06
0.5	2.77E-02	4.80E-04	4.52E-04	9.82E-06
0.6	3.87E-02	4.60E-04	4.28E-04	9.30E-06
0.7	4.59E-02	3.90E-04	3.60E-04	7.81E-06
0.8	4.49E-02	2.80E-04	2.56E-04	5.56E-06
0.9	3.09E-02	1.50E-04	1.30E-04	2.83E-06

Table 6 Absolute errors of CBM Caglar and Caglar (2009) and ECBM results for Example 1 with $n = 41$ for $v(x)$

x	CBM	ECBM ($\lambda_1 = \lambda_2 = 0$)	ECBM ($\lambda_1 = -1.0E-03, \lambda_2 = 0$)
0.1	5.74E-06	5.74E-06	1.25E-07
0.2	1.13E-05	1.13E-05	2.46E-07
0.3	1.64E-05	1.64E-05	3.56E-07
0.4	2.03E-05	2.03E-05	4.42E-07
0.5	2.26E-05	2.26E-05	4.91E-07
0.6	2.26E-05	2.26E-05	4.92E-07
0.7	2.01E-05	2.01E-05	4.37E-07
0.8	1.51E-05	1.51E-05	3.29E-07
0.9	8.14E-06	8.14E-06	1.76E-07

Table 7 L_∞ and L_2 of ECBM results for Example 1

n	5	5	41	41
λ_1	0.000000	-6.639145E-02	0.000000	-1.000000E-03
λ_2	0.000000	1.161882E-06	0.000000	0.000000
L_∞ of $U(x)$	2.791929E-02	1.413576E-04	4.518529E-04	9.817274E-06
L_∞ of $V(x)$	1.423849E-03	7.169404E-06	2.263578E-05	4.917602E-07
L_2 of $U(x)$	4.600584E-02	2.362253E-04	9.969665E-04	2.165970E-05
L_2 of $V(x)$	2.262625E-03	1.138452E-05	5.066609E-05	1.100638E-06

Table 8 Computational time and norms for different optimization equations $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ with $n = 5$

Minimization values	$d_1(\lambda_1, \lambda_2)$	$d_2(\lambda_1, \lambda_2)$	$d_3(\lambda_1, \lambda_2)$
λ_1	-1.273122E-02	-1.273121E-02	-1.269208E-02
λ_2	-6.634562E-02	-6.634562E-02	-6.634523E-02
Computational time (s)	5.517106E+02	5.196057E+02	2.959325E+01
L_∞	1.750978E-04	1.750978E-04	1.750618E-04
L_2	2.913261E-04	2.913260E-04	2.926986E-04

where $0 \leq x \leq 1$, $f_1(x) = -2(x + 1) \cos(x) + \pi \cos(\pi x) + 2x \sin(\pi x) + (4x - 2x^2 - 4) \sin(x)$, and $f_2(x) = -4(x - 1) \cos(x) - 2(2 - x^2 + x^3) \sin(x) - (\pi^2 - 1) \sin(\pi x)$. The exact solutions are $u(x) = 2(1 - x) \sin(x)$, and $v(x) = \sin(\pi x)$.

Table 8 displays the values of λ_1 and λ_2 when $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ are minimized for $n = 5$, with their respective L_∞ and L_2 . Again, minimizing $d_3(\lambda_1, \lambda_2)$ is the best option because the results are comparable and the computational time is significantly less than that of $d_1(\lambda_1, \lambda_2)$ and $d_2(\lambda_1, \lambda_2)$. Therefore, the chosen values of λ_1 and λ_2 are $-1.269208E-02$ and $-6.634523E-02$, respectively. For this example, minimizing $d_2(\lambda_1, \lambda_2)$ gives similar results with minimizing $d_1(\lambda_1, \lambda_2)$ with almost similar computational time.

The approximate and exact solutions at the nodal points are displayed in Table 9. Again, from the table, the approximate solutions agree with the exact solutions. The plots of the

numerical results are shown in Figs. 4 and 5. The numerical results for $\lambda_1 = \lambda_2 = -1.0E-03$ and $n = 25$ are shown in Tables 10 and 11 and compared with reproducing kernel and sinc methods (Geng and Cui 2007; Dehghan and Saadatmandi 2007). The values of λ_1 and λ_2 were obtained by trial and error. It can be seen that ECBM produced results with significantly higher accuracy than the other two. The infinite and two norms are shown in Table 12. For this example, ECBM improves the accuracy of CBM for $u(x)$ and gives out similar results for $v(x)$.

Table 9 Comparison of ECBM results with the exact solution for Example 2 when $\lambda_1 = -0.012692$, $\lambda_2 = -0.066345$, and $n = 5$

x	Exact solution $u(x)$	Approx. solution $U(x)$	Absolute error $ U(x) - u(x) $	Exact solution $v(x)$	Approx. solution $V(x)$	Absolute error $ V(x) - v(x) $
0.2	0.317871	0.317853	1.769288E-05	0.587785	0.587676	1.093618E-04
0.4	0.467302	0.467284	1.800318E-05	0.951057	0.950881	1.750618E-04
0.6	0.451714	0.451696	1.804713E-05	0.951057	0.950882	1.744319E-04
0.8	0.286942	0.286926	1.603373E-05	0.587785	0.587678	1.068617E-04

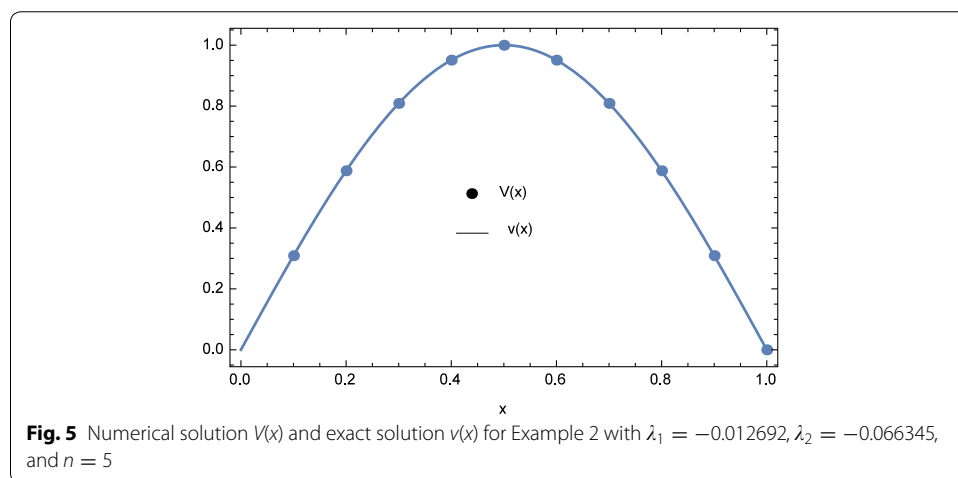
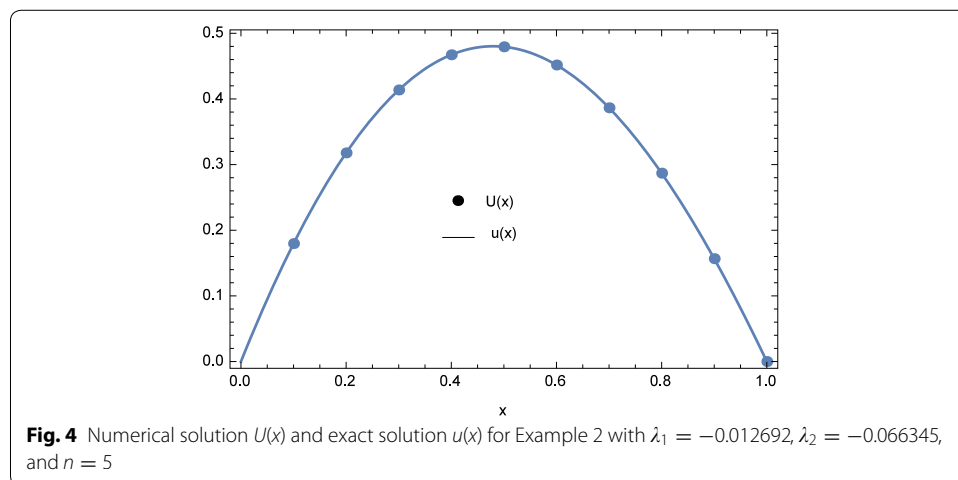


Table 10 Maximum errors of reproducing kernel Geng and Cui (2007), Sinc method Dehghan and Saadatmandi (2007), and ECBM results for Example 2 with $n = 25$ for $u(x)$

x	Reproducing kernel	Sinc method	ECBM ($\lambda_1 = \lambda_2 = 0$)	ECBM ($\lambda_1 = \lambda_2 = -1.0E-03$)
0.08	3.3E-03	3.2E-03	1.3E-04	1.4E-05
0.24	7.7E-03	9.4E-04	2.7E-04	1.1E-05
0.40	9.7E-03	2.0E-03	2.7E-04	2.1E-05
0.56	9.5E-03	2.2E-04	2.0E-04	5.9E-05
0.72	7.3E-03	4.1E-03	9.4E-05	7.8E-05
0.88	3.4E-03	1.0E-02	1.6E-05	5.6E-05
0.96	1.1E-03	2.1E-03	3.6E-08	2.3E-05

Table 11 Maximum errors of reproducing kernel Geng and Cui (2007), Sinc method Dehghan and Saadatmandi (2007), and ECBM results for Example 2 with $n = 25$ for $v(x)$

x	Reproducing kernel	Sinc method	ECBM ($\lambda_1 = \lambda_2 = 0$)	ECBM ($\lambda_1 = \lambda_2 = -1.0E-03$)
0.08	7.7E-03	1.5E-03	3.8E-04	2.2E-04
0.24	2.2E-02	7.0E-03	9.9E-04	6.0E-04
0.40	2.7E-02	7.4E-03	1.3E-03	8.3E-04
0.56	2.7E-02	1.0E-02	1.4E-03	8.6E-04
0.72	2.0E-02	4.4E-03	1.1E-03	6.8E-04
0.88	9.4E-03	2.1E-02	5.0E-04	3.3E-04
0.96	3.1E-03	6.9E-03	1.7E-04	1.1E-04

Example 3 Finally, we consider the system Caglar and Caglar (2009),

$$\begin{cases} u''(x) + xu(x) + xv(x) = 2 \\ v''(x) + 2xv(x) + 2xu(x) = -2 \\ u(0) = u(1) = 0, v(0) = v(1) = 0 \end{cases} \quad (33)$$

where $0 < x < 1$. The exact solutions are $u(x) = x^2 - x$ and $v(x) = x - x^2$.

Table 13 displays the values of λ_1 and λ_2 when $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ are minimized for $n = 5$ together with the values of L_∞ and L_2 . Minimizing $d_3(\lambda_1, \lambda_2)$ is the best option because the computational time is significantly less than that of $d_1(\lambda_1, \lambda_2)$ and $d_2(\lambda_1, \lambda_2)$. However, the minimizing values of λ_1 and λ_2 are equivalent to CBM. It

Table 12 L_∞ and L_2 of ECBM results for Example 2

n	5	5	25	25
λ_1	0.000000	-1.269208E-02	0.000000	-1.000000E-03
λ_2	0.000000	-6.634523E-02	0.000000	-1.000000E-03
L_∞ of $U(x)$	2.086834E-03	1.804713E-05	2.720423E-04	7.798961E-05
L_∞ of $V(x)$	1.750618E-04	1.750618E-04	1.364287E-03	8.604698E-04
L_2 of $U(x)$	2.087051E-03	3.492752E-05	4.590374E-04	1.179224E-04
L_2 of $V(x)$	2.906072E-04	2.906072E-04	2.491362E-03	1.556034E-03

can also be observed that minimizing $d_2(\lambda_1, \lambda_2)$ gives similar results with minimizing $d_1(\lambda_1, \lambda_2)$ with a little less computational time.

The approximate and exact solutions at the nodal points are displayed in Table 14. The plots of the numerical results are shown in Figs. 6 and 7. The numerical results for $n = 21$ and $\lambda_1 = \lambda_2 = 1.25E-14$ are shown in Table 15 and compared with CBM Caglar and Caglar (2009). The values of λ_1 and λ_2 were obtained by trial and error. It can be seen that ECBM produced slightly more accurate results than CBM. The infinite and two norms are shown in Table 16.

Conclusions

In this research, a new method for finding approximate solutions for a system of second order boundary value problems based on extended cubic B-spline was proposed. This method is called extended cubic B-spline method. The error estimation was carried

Table 13 Computational time and norms for different optimization equations $d_1(\lambda_1, \lambda_2)$, $d_2(\lambda_1, \lambda_2)$, and $d_3(\lambda_1, \lambda_2)$ with $n = 5$

Minimization values	$d_1(\lambda_1, \lambda_2)$	$d_2(\lambda_1, \lambda_2)$	$d_3(\lambda_1, \lambda_2)$
λ_1	0.000000	0.000000	0.000000
λ_2	0.000000	0.000000	0.000000
Computational time (s)	7.314018E+01	6.738385E+01	4.973284E+00
L_∞	3.691492E-15	3.691492E-15	3.691492E-15
L_2	6.058413E-15	6.058413E-15	6.058413E-15

Table 14 Comparison of ECBM results with the exact solution for Example 3 when $\lambda_1 = 0.000000$, $\lambda_2 = 0.000000$, and $n = 5$

x	Exact solution $u(x)$	Approx. solution $U(x)$	Absolute error $ U(x) - u(x) $	Exact solution $v(x)$	Approx. solution $V(x)$	Absolute error $ V(x) - v(x) $
0.2	-0.160000	-0.160000	4.163336E-16	0.160000	0.160000	4.718448E-16
0.4	-0.240000	-0.240000	2.775558E-17	0.240000	0.240000	6.106227E-16
0.6	-0.240000	-0.240000	9.992007E-16	0.240000	0.240000	2.775558E-16
0.8	-0.160000	-0.160000	3.469447E-15	0.160000	0.160000	3.691492E-15

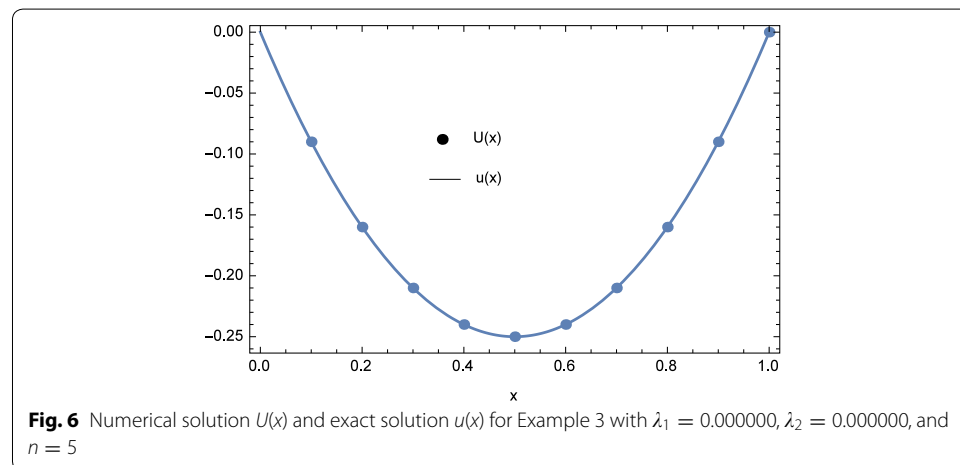


Fig. 6 Numerical solution $U(x)$ and exact solution $u(x)$ for Example 3 with $\lambda_1 = 0.000000$, $\lambda_2 = 0.000000$, and $n = 5$

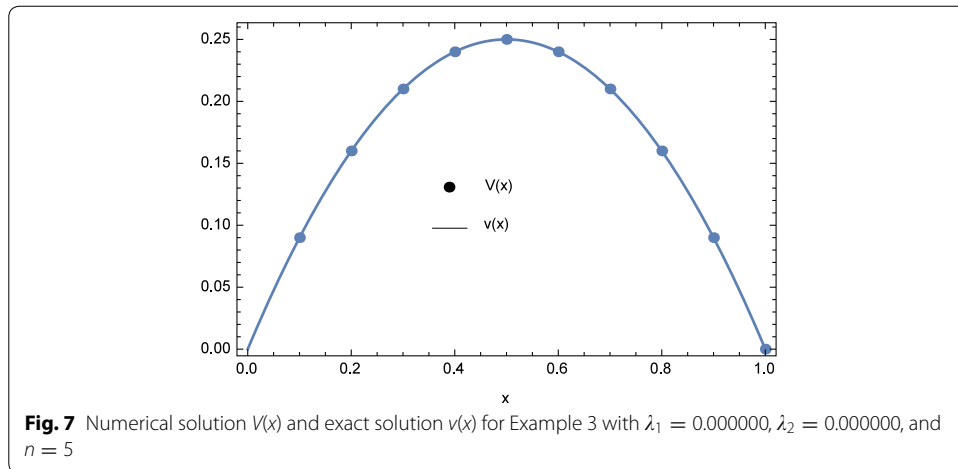


Table 15 Comparison of norms of CBM and ECBM for Example 3 when $n = 21$ for $u(x)$ and $v(x)$

Errors	ECBM ($\lambda_1 = \lambda_2 = 0$)		ECBM ($\lambda_1 = \lambda_2 = 1.25E-14$)	
	$u(x)$	$v(x)$	$u(x)$	$v(x)$
L_∞	3.720357E-13	2.531308E-13	1.725009E-13	1.668943E-13
L_2	4.367056E-13	4.365110E-13	2.930975E-13	2.223093E-13

Table 16 L_∞ and L_2 of ECBM results for Example 3

n	5	21	21
λ_1	0.000000	0.000000	1.250000E-14
λ_2	0.000000	0.000000	1.250000E-14
L_∞ of $U(x)$	3.469447E-15	3.720357E-13	1.725009E-13
L_∞ of $V(x)$	3.691492E-15	2.530308E-13	1.668943E-13
L_2 of $U(x)$	3.634497E-15	4.367056E-13	2.930975E-13
L_2 of $V(x)$	3.781487E-15	4.365110E-13	2.223093E-13

out and the truncation error was found to be of order h^2 , whereby the values of the free parameters λ_1 and λ_2 have influence on the order. This method improved the accuracy of its predecessor, CBM, and produced more accurate results than some other numerical methods. It is also found that minimizing the one-norm term, $d_3(\lambda_1, \lambda_2)$ is sufficient to obtain the optimized values of λ_1 and λ_2 . More work can be done in the optimizing technique to improve the computational time.

Authors' contributions

ASH suggested the method and the problem and wrote the first version of the paper and carried out the works to generate results using Mathematica. NNAH checked the paper and the Mathematica program, gave constructive comments and suggestions to ASH to improve the quality of the paper. AIMI did the final checking and reviewing. All authors read and approved the final manuscript.

Acknowledgements

We thank Dr. Abedel-Karrem Alomari and Maher Hailat for their help in this study. Also the authors would like to thank the Editor-in-Chief and the reviewers for their valuable suggestions to improve this work.

Competing interests

The authors declare that they have no competing interests.

Received: 9 May 2016 Accepted: 27 July 2016

Published online: 09 August 2016

References

- Arqub OA, Abo-Hammour Z (2014) Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm. *Inf Sci* 279:396–415
- Bataineh AS, Noorani MSM, Hashim I (2009) Modified homotopy analysis method for solving systems of second-order BVPs. *Commun Nonlinear Sci Numer Simul* 14(2):430–442
- Caglar N, Caglar H (2009) B-spline method for solving linear system of second-order boundary value problems. *Comput Math Appl* 57(5):757–762
- Chen S, Hu J, Chen L, Wang C (2005) Existence results for n-point boundary value problem of second order ordinary differential equations. *J Comput Appl Math* 180(2):425–432
- Cheng X, Zhong C (2005) Existence of positive solutions for a second-order ordinary differential system. *J Math Anal Appl* 312(1):14–23
- Dehghan M, Saadatmandi A (2007) The numerical solution of a nonlinear system of second-order boundary value problems using the sinc-collocation method. *Math Comput Model* 46(11):1434–1441
- Dehghan M, Lakestani M (2008) Numerical solution of nonlinear system of second-order boundary value problems using cubic B-spline scaling functions. *Int J Comput Math* 85(9):1455–1461
- Dehghan M, Nikpour A (2013) Numerical solution of the system of second-order boundary value problems using the local radial basis functions based differential quadrature collocation method. *Appl Math Model* 37(18):8578–8599
- El-Gamel M (2012) Sinc-collocation method for solving linear and nonlinear system of second-order boundary value problems. *Appl Math* 3(11):1627
- Gang X, Guo-Zhao W (2008) Extended cubic uniform B-spline and α -B-spline. *Acta Autom Sin* 34(8):980–984
- Geng F, Cui M (2007) Solving a nonlinear system of second order boundary value problems. *J Math Anal Appl* 327(2):1167–1181
- Geng F, Cui M (2011) Homotopy perturbation-reproducing kernel method for nonlinear systems of second order boundary value problems. *J Comput Appl Math* 235(8):2405–2411
- Goh J, Majid AA, Ismail AIM (2011) Extended cubic uniform B-spline for a class of singular boundary value problems. *Sci Asia* 37(1):79–82
- Hamid NNA (2010) Splines for linear two-point boundary value problems. MSc thesis, Universiti Sains Malaysia
- Hamid NNA, Majid AA, Ismail AIM (2011) Extended cubic B-spline method for linear two-point boundary value problems. *Sains Malays* 40(11):1285–1290
- Han XL, Liu SJ (2003) An extension of the cubic uniform B-spline curves. *J Comput Aided Des Comput Graph* 15(5):576–578
- Kadalbajoo MK, Kumar V (2007) B-spline method for a class of singular two-point boundary value problems using optimal grid. *Appl Math Comput* 188(2):1856–1869
- Khuri SA, Sayfy A (2009) Spline collocation approach for the numerical solution of a generalized system of second-order boundary-value problems. *Appl Math Sci* 3(45):2227–2239
- Lu J (2007) Variational iteration method for solving a nonlinear system of second-order boundary value problems. *Comput Math Appl* 54(7):1133–1138
- Ogunlaran OM, Ademola AT (2015) On the Laplace homotopy analysis method for a nonlinear system of second-order boundary value problems. *Gen* 26(2):11–22
- Saadatmandi A, Dehghan M, Eftekhari A (2009) Application of He's homotopy perturbation method for non-linear system of second-order boundary value problems. *Nonlinear Anal Real World Appl* 10(3):1912–1922
- Sastry SS (2012) Introductory methods of numerical analysis. PHI Learning Pvt, Ltd
- Thompson HB, Tisdell C (2002) Boundary value problems for systems of difference equations associated with systems of second-order ordinary differential equations. *Appl Math Lett* 15(6):761–766

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
