A new cubic B-spline method for approximating the solution of a class of nonlinear second-order boundary value problem with two dependent variables

Feng-Gong Lang[*], **Xiao-Ping Xu**

School of Mathematical Sciences, Ocean University of China, Qingdao, Shandong, 266100, China

[*]Corresponding author, e-mail: fenggonglang@sina.com

ABSTRACT: In this paper, we will apply cubic B-splines on a uniform mesh to explore the numerical solutions and numerical derivatives of a class of nonlinear second-order boundary value problems with two dependent variables. Our new method is based on the cubic spline interpolation. The analytical solutions and any-order derivatives can be well approximated with 4th order accuracy. Furthermore, our new method is also able to solve general nonlinear 4th-order two-point boundary value problems. Numerical results show that our method is very practical and effective.

KEYWORDS: numerical solution, numerical derivative, 4th-order two-point boundary value problem

INTRODUCTION

Nonlinear systems of second-order two-point boundary value problems (2pBVPs) often arise in many fields, such as physics, engineering, and biology. The existence of solutions to these problems has been studied[1,2]. However, it is not easy to obtain their analytical solutions. In fact, there are very few effective numerical methods with high order accuracy for these problems. There are only a few numerical methods for solving the special nonlinear system of second-order 2pBVPs

$$
\left.
\begin{aligned}
& a_0(x)u'' + a_1(x)u' + a_2(x)u + a_3(x)v'' \\
& + a_4(x)v' + a_5(x)v + G_1(x,u,v) = f_1(x), \\
& b_0(x)v'' + b_1(x)v' + b_2(x)v + b_3(x)u'' \\
& + b_4(x)u' + b_5(x)u + G_2(x,u,v) = f_2(x), \\
& u(0) = u(1) = 0, \quad v(0) = v(1) = 0, \quad x \in [0,1],
\end{aligned}
\right\}
\tag{1}
$$

where $a_i(x)$ and $b_i(x)(i = 0,1,\dots,5)$ are continuous, G_1 and G_2 are nonlinear functions of x, u, and v. For example, the iterative reproducing kernel method[3], the variational iteration method[4], the Chebyshev finite difference method[5], the sinc-collocation method[6], the homotopy perturbation method[7], and the non-polynomial spline method[8] have been studied for (1). In addition, a combined homotopy perturbation and reproducing kernel method was also given in Ref. 9 for a simpler nonlinear system, where $a_0(x) = b_0(x) = 1$, $a_3(x) = a_4(x) = a_5(x) = 0$, and $b_3(x) = b_4(x) = b_5(x) = 0$. Some others proposed

cubic B-spline methods. The method in Ref. 10 was applicable to (1); the method in Ref. 11 was only applicable to the linear case of (1), i.e., $G_1(x,u,v) = G_2(x,u,v) = 0$; and the method in Ref. 12 was applicable to a small generalized case of (1), i.e., only including $a_6(x)u'v'$ and $b_6(x)u'v'$ as additional terms in the above system (1). We conclude that the applications of these methods are all limited. On the one hand, these methods are only applicable to a class of simple nonlinear system of second-order 2pBVPs (1); on the other hand, these methods do not provide the approximation to the derivatives of $u(x)$ and $v(x)$.

In this paper, we will study an effective numerical method for solving the general nonlinear system

$$
\left.
\begin{aligned}
u'' &= F(x,u,u',v,v'), \quad x \in [a,b] \\
v'' &= G(x,u,u',v,v'), \quad x \in [a,b] \\
u(a) &= u_0, \quad u(b) = u_1, \\
v(a) &= v_0, \quad v(b) = v_1,
\end{aligned}
\right\}
\tag{2}
$$

where F and G are two nonlinear functions of x, u, u', v, and v'. We use cubic B-splines on uniform partitions over $[a,b]$ to explore the numerical solutions and their derivatives for (2). Our new cubic spline method is an improvement over the above-mentioned methods not only because it is valid for the general case (2) but also because it is able to approximate the analytic solutions and their any-order derivatives $u^{(k)}(x)$ and $v^{(k)}(x)(k = 0,1,2,\dots)$ with $O(h^4)$ errors.

Our method is also able to solve general nonlinear 4th-order 2pBVPs numerically since these problems

Table 1 The values of $B_i(x)$, $B_i'(x)$, and $B_i''(x)$ at the knots.

	x_{i-1}	x_i	x_{i+1}	else
$B_i(x)$	$\frac{1}{6}$	$\frac{4}{6}$	$\frac{1}{6}$	0
$B_i'(x)$	$1/2h$	0	$-1/2h$	0
$B_i''(x)$	$1/h^2$	$-2/h^2$	$1/h^2$	0

can be transformed into a nonlinear system of second-order 2pBVPs. It is another important application of our method because numerical analysis literature contains little on the numerical methods for the general nonlinear 4th-order 2pBVPs. 4th-order 2pBVPs often arise in the mathematical modelling of viscoelastic and inelastic flows, deformation of beams and plate deflection theory (see Refs. 13, 14 and references cited therein).

The method has the following main advantages. (a) The method is applicable to general nonlinear systems of second-order boundary value problems. It is also powerful in solving general nonlinear 4th-order boundary value problems. (b) The method is able to provide any-order numerical derivatives in addition to the numerical solution with 4th order accuracy.

The remainder of this paper is organized as follows. In the 2nd section, we give some preliminary results of cubic B-spline interpolation; in the 3rd section, we construct the $O(h^4)$ approximation $\widetilde{M}_j$ for $y''(x_j)$ by using the proper linear combinations of M_j. The new numerical method is described in the 4th section, where a nonlinear system with the coefficients of the B-splines as unknowns is studied. The 5th section is devoted to numerical comparisons and tests. Numerical results show that our method is very powerful and effective in approximating the numerical solutions and numerical derivatives of (2). Finally, we conclude our paper in the last section.

PRELIMINARIES OF CUBIC SPLINE INTERPOLATION

Divide $I = [a, b]$ into n subintervals $I_i = [x_i, x_{i+1}] (i = 0, 1, \ldots, n-1)$ by the equidistant knots $x_i = a + ih (i = 0, 1, \ldots, n)$, where $h = (b-a)/n$. The cubic spline space over the uniform partition is defined as follows:

$$S_3(I) = \{s(x) \in C^2(I) \,|\, s(x)I_i| \in P_3, i = 0, \ldots, n-1\},$$

where $s(x)_{|I_i}$ denotes the restriction of $s(x)$ over I_i, and P_3 denotes the set of univariate cubic polynomials. $S_3(I)$ is a linear space and its dimension is $n + 3$. Extend $I = [a, b]$ to $\widetilde{I} = [a - 3h, b + 3h]$ with the equidistant knots $x_i = a + ih (i = -3, -2, \ldots, n+3)$.

The typical cubic B-spline $B_i(x)(i = -1, 0, \ldots, n+1)$ is defined as follows[15]:

$$B_i(x) = \frac{1}{6h^3} \begin{cases} (x - x_{i-2})^3, & x \in [x_{i-2}, x_{i-1}] \\ h^3 + 3h^2(x - x_{i-1}) \\ +3h(x - x_{i-1})^2 \\ -3(x - x_{i-1})^3, & x \in [x_{i-1}, x_i] \\ h^3 + 3h^2(x_{i+1} - x) \\ +3h(x_{i+1} - x)^2 \\ -3(x_{i+1} - x)^3, & x \in [x_i, x_{i+1}] \\ (x_{i+2} - x)^3, & x \in [x_{i+1}, x_{i+2}] \\ 0, & \text{otherwise.} \end{cases}$$

$B_i(x)(i = -1, 0, 1, \ldots, n+1)$ are linearly independent. They form the basis splines of $S_3(I)$. The values of $B_i^{(k)}(x)(i = -1, 0, 1, \ldots, n+1; k = 0, 1, 2)$ at the knots are given in Table 1.

For a given function $y(x)$ (assuming it is sufficiently smooth), there exists a unique cubic spline $s(x) = \sum_{i=-1}^{n+1} c_i B_i(x) \in S_3(I)$ satisfying the interpolation conditions

$$s(x_i) = y(x_i), \quad (i = 0, 1, \ldots, n)$$
$$s'(a) = y'(a), \quad s'(b) = y'(b).$$

For $j = 0, 1, \ldots, n$, let $s_j = s(x_j)$, $m_j = s'(x_j)$, and $M_j = s''(x_j)$. Since $s(x) = \sum_{i=-1}^{n+1} c_i B_i(x)$, by Table 1, we have

$$y_j = s(x_j) = \sum_{i=-1}^{n+1} c_i B_i(x_j) = \frac{c_{j-1} + 4c_j + c_{j+1}}{6}, \tag{3}$$

$$m_j = s'(x_j) = \sum_{i=-1}^{n+1} c_i B_i'(x_j) = \frac{c_{j+1} - c_{j-1}}{2h}, \tag{4}$$

$$M_j = s''(x_j) = \sum_{i=-1}^{n+1} c_i B_i''(x_j) = \frac{c_{j-1} - 2c_j + c_{j+1}}{h^2}. \tag{5}$$

Moreover, in Ref. 16, we have

$$m_j = s'(x_j) = y'(x_j) - \tfrac{1}{180} h^4 y^{(5)}(x_j) + O(h^6), \tag{6}$$

$$\begin{aligned} M_j &= s''(x_j) \\ &= y''(x_j) - \tfrac{1}{12} h^2 y^{(4)}(x_j) + \tfrac{1}{360} h^4 y^{(6)}(x_j) \\ &\quad + O(h^6). \end{aligned} \tag{7}$$

Hence, from (6) and (7), we have

$$\|m_j - y'(x_j)\|_\infty = \max_{0 \leqslant j \leqslant n} |m_j - y'(x_j)| = O(h^4), \tag{8}$$

while $\|M_j - y''(x_j)\|_\infty = O(h^2)$.

THE $O(h^4)$ APPROXIMATION TO $y''(x)$ AT THE KNOTS

For $j = 1, 2, \ldots, n - 1$, we will use $M_j(j = 0, 1, \ldots, n)$ to get a new $O(h^4)$ approximation to $y''(x_j)$. By (7), expanding M_{j-1} at x_j, we have

$$
\begin{aligned}
M_{j-1} &= y''(x_{j-1}) - \tfrac{1}{12}h^2 y^{(4)}(x_{j-1}) \\
&\quad + \tfrac{1}{360}h^4 y^{(6)}(x_{j-1}) + O(h^6) \\
&= y''(x_j) - hy'''(x_j) + \tfrac{5}{12}h^2 y^{(4)}(x_j) \\
&\quad - \tfrac{1}{12}h^3 y^{(5)}(x_j) + O(h^4).
\end{aligned}
$$

Similarly, expanding M_{j+1} at x_j, we have

$$
\begin{aligned}
M_{j+1} &= y''(x_j) + hy'''(x_j) + \tfrac{5}{12}h^2 y^{(4)}(x_j) \\
&\quad + \tfrac{1}{12}h^3 y^{(5)}(x_j) + O(h^4).
\end{aligned}
$$

Choosing three parameters A, B and C so that the error order of $\widetilde{M}_j = AM_j + BM_{j-1} + CM_{j+1}$ is as high as possible, we have

$$
\begin{aligned}
A + B + C &= 1, \\
-B + C &= 0, \\
-A + 5B + 5C &= 0.
\end{aligned}
$$

Hence $A = \tfrac{10}{12}$, $B = C = \tfrac{1}{12}$. So $\widetilde{M}_j = y''(x_j) + O(h^4)$. By (5), we have

$$
\begin{aligned}
\widetilde{M}_j &= \frac{M_{j-1} + 10M_j + M_{j+1}}{12} \\
&= \frac{c_{j-2} + 8c_{j-1} - 18c_j + 8c_{j+1} + c_{j+2}}{12h^2}. \quad (9)
\end{aligned}
$$

Next we will give the $O(h^4)$ approximation to $y''(x_0)$ and $y''(x_n)$. Unlike the inner knots case, we must use four neighbouring values, i.e., M_0, M_1, M_2 and M_3, to approximate $y''(x_0)$. If not, the error will be lower than $O(h^4)$. Similarly, expanding M_1, M_2 and M_3 at x_0, we have

$$
\begin{aligned}
M_1 &= y''(x_0) + hy'''(x_0) + \tfrac{5}{12}h^2 y^{(4)}(x_0) \\
&\quad + \tfrac{1}{12}h^3 y^{(5)}(x_0) + O(h^4), \\
M_2 &= y''(x_0) + 2hy'''(x_0) + \tfrac{23}{12}h^2 y^{(4)}(x_0) \\
&\quad + \tfrac{7}{6}h^3 y^{(5)}(x_0) + O(h^4), \\
M_3 &= y''(x_0) + 3hy'''(x_0) + \tfrac{53}{12}h^2 y^{(4)}(x_0) \\
&\quad + \tfrac{17}{4}h^3 y^{(5)}(x_0) + O(h^4).
\end{aligned}
$$

Choosing four parameters A, B, C, and D so that the error order of $\widetilde{M}_0 = AM_0 + BM_1 + CM_2 + DM_3$ is

as high as possible, we have

$$
\begin{aligned}
A + B + C + D &= 1, \\
B + 2C + 3D &= 0, \\
-A + 5B + 23C + 53D &= 0, \\
B + 14C + 51D &= 0,
\end{aligned}
$$

and we get $A = \tfrac{7}{6}$, $B = -\tfrac{5}{12}$, $C = \tfrac{1}{3}$, and $D = -\tfrac{1}{12}$. By (5), we get

$$
\begin{aligned}
\widetilde{M}_0 &= \frac{14M_0 - 5M_1 + 4M_2 - M_3}{12} \\
&= \frac{14c_{-1} - 33c_0 + 28c_1 - 14c_2 + 6c_3 - c_4}{12h^2}
\end{aligned}
$$

is an $O(h^4)$ approximation to $y''(x_0)$. By symmetry, we have

$$
\begin{aligned}
\widetilde{M}_n &= \frac{14M_n - 5M_{n-1} + 4M_{n-2} - M_{n-3}}{12} \\
&= \frac{1}{12h^2}(-c_{n-4} + 6c_{n-3} - 14c_{n-2} \\
&\quad + 28c_{n-1} - 33c_n + 14c_{n+1})
\end{aligned}
$$

is an $O(h^4)$ approximation to $y''(x_n)$.

THE NUMERICAL METHOD

In this section, we give the cubic B-spline method for (2). Let $u(x) = \sum_{i=-1}^{n+1} c_i B_i(x)$ and $v(x) = \sum_{i=-1}^{n+1} d_i B_i(x)$ be the cubic spline solution of (2). Discretizing (2) at the inner knots ($j = 1, 2, \ldots, n - 1$), we get

$$
\begin{aligned}
u''(x_j) &= F(x_j, u(x_j), u'(x_j), v(x_j), v'(x_j)), \\
v''(x_j) &= G(x_j, u(x_j), u'(x_j), v(x_j), v'(x_j)).
\end{aligned}
$$

By (3), (4) and (9), we have

$$
\begin{aligned}
&\frac{c_{j-2} + 8c_{j-1} - 18c_j + 8c_{j+1} + c_{j+2}}{12h^2} \\
&= F\left(x_j, \frac{c_{j-1} + 4c_j + c_{j+1}}{6}, \frac{c_{j+1} - c_{j-1}}{2h}, \right. \\
&\quad \left. \frac{d_{j-1} + 4d_j + d_{j+1}}{6}, \frac{d_{j+1} - d_{j-1}}{2h}\right), \quad (10)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{d_{j-2} + 8d_{j-1} - 18d_j + 8d_{j+1} + d_{j+2}}{12h^2} \\
&= G\left(x_j, \frac{c_{j-1} + 4c_j + c_{j+1}}{6}, \frac{c_{j+1} - c_{j-1}}{2h}, \right. \\
&\quad \left. \frac{d_{j-1} + 4d_j + d_{j+1}}{6}, \frac{d_{j+1} - d_{j-1}}{2h}\right), \quad (11)
\end{aligned}
$$

where $O(h^4)$ terms are all dropped. We still need 8 equations, which are obtained from the boundary conditions

$$u(a) = u_0, \qquad v(a) = v_0,$$
$$u''(a) = F(a, u_0, u'(a), v_0, v'(a)),$$
$$v''(a) = G(a, u_0, u'(a), v_0, v'(a)),$$
$$u''(b) = F(b, u_1, u'(b), v_1, v'(b)),$$
$$v''(b) = G(b, u_1, u'(b), v_1, v'(b)),$$
$$u(b) = u_1, \qquad v(b) = v_1.$$

We have

$$\frac{c_{-1} + 4c_0 + c_1}{6} = u_0, \tag{12}$$

$$\frac{d_{-1} + 4d_0 + d_1}{6} = v_0,$$

$$\frac{c_{n-1} + 4c_n + c_{n+1}}{6} = u_1,$$

$$\frac{d_{n-1} + 4d_n + d_{n+1}}{6} = v_1,$$

and

$$\frac{14c_{-1} - 33c_0 + 28c_1 - 14c_2 + 6c_3 - c_4}{12h^2}$$
$$= F\left(a, u_0, \frac{-c_{-1} + c_1}{2h}, v_0, \frac{-d_{-1} + d_1}{2h}\right), \tag{13}$$

$$\frac{14d_{-1} - 33d_0 + 28d_1 - 14d_2 + 6d_3 - d_4}{12h^2}$$
$$= G\left(a, u_0, \frac{-c_{-1} + c_1}{2h}, v_0, \frac{-d_{-1} + d_1}{2h}\right), \tag{14}$$

$$\frac{1}{12h^2}(-c_{n-4} + 6c_{n-3} - 14c_{n-2}$$
$$+ 28c_{n-1} - 33c_n + 14c_{n+1})$$
$$= F\left(b, u_1, \frac{-c_{n-1} + c_{n+1}}{2h},\right.$$
$$\left. v_1, \frac{-d_{n-1} + d_{n+1}}{2h}\right), \tag{15}$$

$$\frac{1}{12h^2}(-d_{n-4} + 6d_{n-3} - 14d_{n-2}$$
$$+ 28d_{n-1} - 33d_n + 14d_{n+1})$$
$$= G\left(b, u_1, \frac{-c_{n-1} + c_{n+1}}{2h},\right.$$
$$\left. v_1, \frac{-d_{n-1} + d_{n+1}}{2h}\right). \tag{16}$$

(10), (11) and (12)–(16) give us $2n + 6$ nonlinear equations with c_j and d_j ($j = -1, 0, \ldots, n + 1$) as unknowns. After solving the nonlinear system, we get the coefficients of $u(x) = \sum_{i=-1}^{n+1} c_i B_i(x)$ and $v(x) = \sum_{i=-1}^{n+1} d_i B_i(x)$.

These techniques are also valid for

$$u'' = F(x, u, u', v, v'), \qquad x \in [a, b]$$
$$v'' = G(x, u, u', v, v'), \qquad x \in [a, b]$$
$$\alpha_1 u(a) + \beta_1 u'(a) + \gamma_1 v(a) + \delta_1 v'(a) = \theta_1$$
$$\alpha_2 u(a) + \beta_2 u'(a) + \gamma_2 v(a) + \delta_2 v'(a) = \theta_2$$
$$\alpha_3 u(b) + \beta_3 u'(b) + \gamma_3 v(b) + \delta_3 v'(b) = \theta_3$$
$$\alpha_4 u(b) + \beta_4 u'(b) + \gamma_4 v(b) + \delta_4 v'(b) = \theta_4.$$

The process is similar. Only the last eight boundary-condition equations are different.

In the following, we give some remarks on the numerical treatments of (2). By (3), we use $u_j = \frac{1}{6}(c_{j-1} + 4c_j + c_{j+1})$ and $v_j = \frac{1}{6}(d_{j-1} + 4d_j + d_{j+1})$ to approximate $u(x_j)$ and $v(x_j)$, where $j = 0, 1, \ldots, n$. The errors are $O(h^4)$.

By (4), (6) and (8), we use $u'_j = (c_{j+1} - c_{j-1})/2h$ and $v'_j = (d_{j+1} - d_{j-1})/2h$ to approximate $u'(x_j)$ and $v'(x_j)$, and the errors are $O(h^4)$, where $j = 0, 1, \ldots, n$.

For $j = 1, 2, \ldots, n - 1$, by (2), (10) and (11), we use $u''_j = (c_{j-2} + 8c_{j-1} - 18c_j + 8c_{j+1} + c_{j+2})/12h^2$ or $u''_j = F(x_j, u_j, u'_j, v_j, v'_j)$ and $v''_j = (d_{j-2} + 8d_{j-1} - 18d_j + 8d_{j+1} + d_{j+2})/12h^2$ or $v''_j = G(x_j, u_j, u'_j, v_j, v'_j)$ to approximate $u''(x_j)$ and $v''(x_j)$. Similarly, we use the representations in the left side or the right side of (13)–(16) to approximate $u''(a), v''(a)$ and $u''(b), v''(b)$. Based on the previous analysis, all the errors are $O(h^4)$.

From (2), differentiating $u'' = F$, we have

$$u''' = F'_1 + F'_2 u' + F'_3 u'' + F'_4 v' + F'_5 v''$$
$$= F'_1 + F'_2 u' + F'_3 F + F'_4 v' + F'_5 G$$
$$:= \Phi(x, u, u', v, v'),$$

which shows that u''' is a multivariate function of x, u, u', v, and v'. We can use $u'''_j = \Phi(x_j, u_j, u'_j, v_j, v'_j)$ to approximate $u'''(x_j) = \Phi(x_j, u(x_j), u'(x_j), v(x_j), v'(x_j))$. Since u_j, u'_j, v_j and v'_j possess $O(h^4)$ errors to $u(x_j), u'(x_j), v(x_j)$ and $v'(x_j)$, we see that $u'''_j = \Phi(x_j, u_j, u'_j, v_j, v'_j)$ also possesses $O(h^4)$ errors to $u'''(x_j)$ by the multivariate Taylor's formula. The approximation to $v'''(x_j)$ is similar.

In fact, for any integer $k \geqslant 4$, by repeatedly differentiating $u'''(x), v'''(x)$ and using (2), it is easy to observe that $u^{(k)}(x)$ and $v^{(k)}(x)$ are two nonlinear

functions of x, u, u' and v, v'. In a similar manner, we can get the approximation to $u^{(k)}(x_j)$ and $v^{(k)}(x_j)$ with $O(h^4)$ errors.

The above items are all related to numerical approximation at the knots. Finally, we point out how to approximate $u^{(k)}(\bar{x})$, where $\bar{x} \in (x_j, x_{j+1})$ is not a knot. By Taylor's formula, we have

$$
\begin{aligned}
u^{(k)}(\bar{x}) = {} & u^{(k)}(x_j) + u^{(k+1)}(x_j)(\bar{x} - x_j) \\
& + \tfrac{1}{2} u^{(k+2)}(x_j)(\bar{x} - x_j)^2 \\
& + \tfrac{1}{6} u^{(k+3)}(x_j)(\bar{x} - x_j)^3 \\
& + \tfrac{1}{24} u^{(k+2)}(\xi_j)(\bar{x} - x_j)^4.
\end{aligned}
$$

To get an $O(h^4)$ error, we use

$$
\begin{aligned}
u_{\bar{x}}^{(k)} = {} & u_j^{(k)} + u_j^{(k+1)}(\bar{x} - x_j) \\
& + \tfrac{1}{2} u_j^{(k+2)}(\bar{x} - x_j)^2 + \tfrac{1}{6} u_j^{(k+3)}(\bar{x} - x_j)^3
\end{aligned}
$$

to approximate $u^{(k)}(\bar{x})$. $v^{(k)}(\bar{x})$ can be approximated similarly. These techniques can also be applied similarly to the nonlinear system of second-order 2pBVPs which involve three or more unknown functions.

There are many methods for the solving the simple linear 4th-order 2pBVP

$$
\begin{aligned}
y^{(4)}(x) &= p(x)y(x) + q(x) \\
y(a) &= y_a, \quad y(b) = y_b, \\
y''(a) &= y_a'', \quad y''(b) = y_b''.
\end{aligned}
$$

However, there are very few effective methods for the following nonlinear 4th-order 2pBVP:

$$
\left.
\begin{aligned}
y^{(4)}(x) &= G(x, y(x), y'(x), y''(x), y'''(x)) \\
y(a) &= y_a, \quad y(b) = y_b, \\
y''(a) &= y_a'', \quad y''(b) = y_b''.
\end{aligned}
\right\} \quad (17)
$$

In order to solve (17) by using the above method, we first transform it into a nonlinear system of second-order 2pBVPs,

$$
\begin{aligned}
u'' &= v, \quad v'' = G(x, u, u', v, v') \\
u(a) &= u_0 = y_a, \quad u(b) = u_1 = y_b \\
v(a) &= v_0 = y_a'', \quad v(b) = v_1 = y_b'',
\end{aligned}
$$

by $u = y(x)$ and $v = y''(x)$. Let $u(x) = y(x) = \sum_{i=-1}^{n+1} c_i B_i(x)$ and $v(x) = y''(x) = \sum_{i=-1}^{n+1} d_i B_i(x)$. Then we get $2n + 6$ equations by (10), (11), and (12)–(16). Fortunately, (10), (13) and (15) can be simplified to

$$
\frac{c_{j-2} + 8c_{j-1} - 18c_j + 8c_{j+1} + c_{j+2}}{12h^2}
$$
$$
= \frac{d_{j-1} + 4d_j + d_{j+1}}{6},
$$

$$
\frac{14c_{-1} - 33c_0 + 28c_1 - 14c_2 + 6c_3 - c_4}{12h^2}
$$
$$
= \frac{d_{-1} + 4d_0 + d_1}{6},
$$

and

$$
\frac{1}{12h^2}(-c_{n-4} + 6c_{n-3} - 14c_{n-2} \\
+ 28c_{n-1} - 33c_n + 14c_{n+1})
$$
$$
= \frac{d_{n-1} + 4d_n + d_{n+1}}{6},
$$

respectively.

Similarly, our method can also work well for nonlinear sixth-order 2pBVPs, which are often arise in hydromagnetics, hydrodynamics, stellar convection dynamics[17]. We point out that many published methods are only valid for simple and linear sixth-order 2pBVPs.

NUMERICAL TESTS

First, we compare our method with other methods with two examples in the form of (1). Numerical comparisons are performed by MATLAB.

Example 1 Consider the nonlinear system of second-order 2pBVPs

$$
\begin{aligned}
u'' + xu' + 2xv + xu^2 &= -2 + x^2 + x^5 - 2x^4 \\
&\quad + 2x \sin \pi x, \\
v' + v + x^2 u + \sin x v^2 &= x^3 - x^4 + \pi \cos \pi x \\
&\quad + \sin \pi x + \sin x (\sin \pi x)^2, \\
u(0) = u(1) = 0, &\quad v(0) = v(1) = 0,
\end{aligned}
$$

where the exact solutions are $u(x) = x - x^2$ and $v(x) = \sin \pi x$, $x \in [0, 1]$. This example has been studied by the iterative reproducing kernel method[3], the sinc-collocation method[6], and the combined homotopy perturbation and reproducing kernel method[9].

See Table 2 and Table 3 for the respective absolute errors at the same selected points. Our errors are obtained by the new cubic spline method with $n = 25$. Our results are clearly better than the others.

Example 2 Consider the nonlinear system of second-order 2pBVPs

$$
\begin{aligned}
u'' + (2x - 1)u' + \cos \pi x v' &= f(x), \\
v'' + xu &= 2 + x \sin \pi x, \\
u(0) = u(1) = 0, &\quad v(0) = v(1) = 0,
\end{aligned}
$$

where $f(x) = (2x - 1)(\pi + 1) \cos \pi x - \pi^2 \sin \pi x$ and the exact solutions are $u(x) = \sin \pi x$ and $v(x) = x^2 - x$, $x \in [0, 1]$.

Table 2 The absolute errors of $u(x)$ for Example 1.

x_i	$E(u)$	$E(u)$ Ref. 3	$E(u)$ Ref. 6	$E(u)$ Ref. 9
0.08	2.2×10^{-8}	5.0×10^{-4}	1.4×10^{-4}	2.0×10^{-5}
0.24	6.2×10^{-8}	1.4×10^{-3}	4.4×10^{-5}	5.7×10^{-5}
0.40	8.9×10^{-8}	2.1×10^{-3}	6.7×10^{-5}	8.6×10^{-5}
0.56	9.1×10^{-8}	2.2×10^{-3}	9.3×10^{-5}	9.8×10^{-5}
0.72	6.5×10^{-8}	1.8×10^{-3}	4.9×10^{-5}	9.4×10^{-5}
0.88	2.5×10^{-8}	9.0×10^{-4}	8.6×10^{-5}	6.5×10^{-5}
0.96	6.4×10^{-9}	3.0×10^{-4}	7.1×10^{-5}	1.4×10^{-5}

Table 3 The absolute errors of $v(x)$ for Example 1.

x_i	$E(v)$	$E(v)$ Ref. 3	$E(v)$ Ref. 6	$E(v)$ Ref. 9
0.08	2.9×10^{-7}	2.0×10^{-3}	2.4×10^{-4}	1.1×10^{-4}
0.24	7.0×10^{-7}	5.6×10^{-3}	2.3×10^{-3}	3.3×10^{-4}
0.40	7.4×10^{-7}	7.9×10^{-3}	8.9×10^{-4}	4.6×10^{-4}
0.56	4.0×10^{-7}	8.2×10^{-3}	1.4×10^{-3}	4.8×10^{-4}
0.72	1.7×10^{-7}	6.5×10^{-3}	3.1×10^{-3}	3.8×10^{-4}
0.88	8.3×10^{-7}	3.1×10^{-3}	1.6×10^{-3}	2.9×10^{-4}
0.96	1.1×10^{-6}	1.0×10^{-3}	9.8×10^{-4}	6.2×10^{-5}

We compare our results with the cubic spline method in Ref. 11 by this example. See Table 4 for the respective maximum absolute errors of $u(x)$ and $v(x)$, where

$$E[u, n] = \max_{0 \leqslant i \leqslant n} |u(x_i) - u_i|,$$

and $E[v, n]$ is similarly defined. With the same step h, our errors are less than 0.1% of the errors in Ref. 11. Clearly, the new cubic spline method is more powerful.

Next, we study two examples to show our new cubic spline method is also very effective for the general nonlinear system of 2nd-order 2pBVPs and nonlinear 4th-order 2pBVPs, not only in approximating the analytic solutions but also in approximating their any-order derivatives. All the errors are $O(h^4)$ theoretically. To the best of our knowledge, currently, it is rare to find another method to produce the same numerical data with $O(h^4)$ errors.

Example 3 Consider the nonlinear system of second-order 2pBVPs

$$u'' = u'v + 2\ln v + (\sin x)v + u - 2\cos x + 2x - 1,$$
$$v'' = u'(v')^2 + (\sin x)v^2 + uv - e^{-x}\cos x - 1,$$
$$u(0) - 2u'(0) + v(0) + v'(0) = 0,$$
$$-u(0) + u'(0) - v(0) + 3v'(0) = -5,$$
$$u(1) - u'(1) + 2v(1) + 2v'(1) = \cos 1 + \sin 1,$$
$$u(1) - 3u'(1) + 2e^2 v(1) = \cos 1 + 3\sin 1,$$

where the exact solutions are $u(x) = e^x + \cos x$ and $v(x) = e^{-x}$.

Table 4 The maximum absolute errors of $u(x)$ and $v(x)$ for Example 2.

n	$E[u, n]$	$E[u, n]$ [11]	$E[v, n]$	$E[v, n]$ [11]
20	1.3×10^{-7}	1.9×10^{-3}	6.2×10^{-8}	9.6×10^{-5}
40	8.0×10^{-8}	4.7×10^{-4}	3.9×10^{-9}	2.4×10^{-5}
60	1.6×10^{-8}	2.1×10^{-4}	7.8×10^{-10}	1.1×10^{-5}

Table 5 The maximum absolute errors and the error orders of $u(x)$ and $v(x)$ for Example 3.

n	5	10	20	40
$E[u, n]$	1.3×10^{-5}	6.1×10^{-7}	3.2×10^{-8}	1.9×10^{-9}
$O_0(u)$	4.51	4.42	4.25	4.10
$E[v, n]$	6.1×10^{-6}	3.7×10^{-7}	2.2×10^{-8}	1.1×10^{-9}
$O_0(v)$	4.03	4.06	4.06	4.35
$E[u', n]$	8.2×10^{-6}	3.3×10^{-7}	3.0×10^{-8}	4.1×10^{-9}
$O_1(u)$	4.88	4.63	3.46	2.89
$E[v', n]$	4.0×10^{-6}	1.6×10^{-7}	1.0×10^{-8}	9.6×10^{-10}
$O_1(v)$	5.25	4.67	3.90	3.44
$E[u'', n]$	1.5×10^{-5}	5.4×10^{-7}	3.3×10^{-8}	7.6×10^{-9}
$O_2(u)$	4.77	4.82	4.00	2.13
$E[v'', n]$	5.3×10^{-6}	3.6×10^{-7}	2.5×10^{-8}	2.2×10^{-9}
$O_2(v)$	3.99	3.89	3.87	3.53
$E[u''', n]$	5.9×10^{-5}	2.0×10^{-6}	1.3×10^{-7}	1.9×10^{-8}
$O_3(u)$	5.11	4.88	3.90	2.83
$E[v''', n]$	4.7×10^{-5}	2.6×10^{-6}	1.4×10^{-7}	7.0×10^{-9}
$O_3(v)$	3.97	4.17	4.25	4.28

See Table 5 for the maximum absolute errors $E[u^{(\mu)}, n]$, $E[v^{(\mu)}, n]$ and error orders $O_\mu(u)$, $O_\mu(v)$, where

$$E[u^{(\mu)}, n] = \max_{0 \leqslant i \leqslant n} \left| u^{(\mu)}(x_i) - u_i^{(\mu)} \right|,$$

$$O_\mu(u) := O_\mu(u, n_1 \to n_2)$$
$$= \frac{\ln(E[u^{(\mu)}, n_1]/E[u^{(\mu)}, n_2])}{\ln(n_2/n_1)}.$$

$E[v^{(\mu)}, n]$ and $O_\mu(v)$ are similarly defined. It is easy to show that all the maximum absolute errors decrease by about $\frac{1}{16}$ when the partition is refined by $\frac{1}{2}$ step by step. Theoretically, O_μ should be 4. Here, they are influenced more or less by the unavoidable computer round-off errors.

Example 4 Consider the nonlinear 4th-order 2pBVP

$$y^{(4)}(x) = e^{-x}(y'(x))^2 - x^2 y(x)$$
$$+ e^x(12 + 8x - 3x^2 - 4x^3),$$
$$y(0) = 0, \quad y(1) = e,$$
$$y''(0) = 2, \quad y''(1) = 7e$$

Table 6 The maximum absolute errors and the error orders of $y^{(\mu)}(x)$ for Example 4.

n	20	40	80	160
$E[y, n]$	4.6×10^{-8}	3.2×10^{-9}	2.0×10^{-10}	9.9×10^{-12}
$O_0(y)$	3.37	3.87	3.96	4.36
$E[y', n]$	3.1×10^{-6}	2.0×10^{-7}	1.3×10^{-8}	7.9×10^{-10}
$O_1(y)$	3.93	3.96	3.98	4.01
$E[y'', n]$	8.3×10^{-7}	5.3×10^{-8}	3.3×10^{-9}	2.4×10^{-10}
$O_2(y)$	3.95	3.99	4.00	3.79
$E[y''', n]$	9.6×10^{-6}	6.3×10^{-7}	4.1×10^{-8}	2.6×10^{-9}
$O_3(y)$	3.84	3.92	3.96	3.96
$E[y^{(4)}, n]$	1.8×10^{-5}	1.2×10^{-6}	7.5×10^{-8}	4.7×10^{-9}
$O_4(y)$	3.83	3.91	3.95	4.00
$E[y^{(5)}, n]$	2.0×10^{-5}	1.4×10^{-6}	8.7×10^{-8}	5.5×10^{-9}
$O_5(y)$	3.85	3.91	3.95	4.00

where the exact solution is $y(x) = x^2 e^x$.

See Table 6 for the maximum absolute errors and the error orders of $y^{(\mu)}(x)$. The numerical results are very good. Hence our method is also effective for a nonlinear 4th-order 2pBVP (17).

REFERENCES

1. 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**, 761–6.
2. Cheng X, Zhong C (2005) Existence of positive solutions for a second order ordinary differential system. *J Math Anal Appl* **312**, 14–23.
3. Geng FZ, Cui MG (2007) Solving a nonlinear system of second order boundary value problems. *J Math Anal Appl* **327**, 1167–81.
4. Lu JF (2007) Variational iteration method for solving a nonlinear system of second-order boundary value problems. *Comput Math Appl* **54**, 1133–8.
5. Saadatmandi A, Farsangi JA (2007) Chebyshev finite difference method for a nonlinear system of second-order boundary value problems. *Appl Math Comput* **192**, 586–91.
6. 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**, 1434–41.
7. Saadatmandi A, Dehghan M, Eftekharia A (2009) Application of He's homotopy perturbation method for non-linear system of second-order boundary value problems. *Nonlinear Anal R World Appl* **10**, 1912–22.
8. Caglar H, Caglar N, Akkoyunlu C (2010) Non-polynomial spline method of a non-linear system of second-order boundary value problems. *J Comput Anal Appl* **12**, 544–59.
9. Geng FZ, Cui MG (2011) Homotopy perturbation-reproducing kernel method for nonlinear systems of second order boundary value problems. *J Comput Appl Math* **235**, 2405–11.
10. 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**, 1455–61.
11. Caglar N, Caglar H (2009) B-spline method for solving linear system of second-order boundary value problems. *Comput Math Appl* **57**, 757–62.
12. 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**, 2227–39.
13. Gupta Y, Srivastava PK (2011) A computational method for solving two point boundary value problems of order four. *Int J Comput Tech Appl* **2**, 1426–31.
14. Gupta Y, Kumar M (2012) B-spline based numerical algorithm for singularly perturbed problem of fourth order. *Am J Comput Appl Math* **2**, 29–32.
15. Schoenberg IJ (1946) Contribution to the problem of approximation of equidistant data by analytic functions. *Q Appl Math* **4**, 45–99, 112–41.
16. Fyfe DJ (1969) The use of cubic splines in the solution of two point boundary value problems. *Comput J* **12**, 188–92.
17. Islam S, Tirmizi IA, Haq F, Khan M (2008) Non-polynomial splines approach to the solution of sixth-order boundary-value problems. *Appl Math Comput* **195**, 270–84.