Numerical Solution of Tenth Order Boundary Value Problems by Galerkin Method with Septic B-splines

K. N. S. Kasi Viswanadham* and Sreenivasulu Ballem

Department of Mathematics, National Institute of Technology, Warangal, India

Abstract: In this paper, we present a finite element method involving Galerkin method with septic B-splines as basis functions to solve a general tenth order boundary value problem. The basis functions are redefined into a new set of basis functions which vanish on the boundary where the given set of boundary conditions are prescribed. The proposed method was applied to solve several examples of tenth order linear and nonlinear boundary value problems. The solution of a nonlinear boundary value problem has been obtained as the limit of a sequence of solution of linear boundary value problems generated by quasilinearization technique. The obtained numerical results are compared with the exact solutions available in the literature.

Keywords: Absolute error; basis function; Galerkin method; septic B-spline; tenth order boundary value problem.

1. Introduction

In this paper, we consider a general tenth order linear boundary value problem given by

\[ a_0(x) y^{(10)}(x) + a_1(x) y^{(9)}(x) + a_2(x) y^{(8)}(x) + a_3(x) y^{(7)}(x) + a_4(x) y^{(6)}(x) + a_5(x) y^{(5)}(x) \\
+ a_6(x) y^{(4)}(x) + a_7(x) y^{(3)}(x) + a_8(x) y^{(2)}(x) + a_9(x) y^{(1)}(x) + a_{10}(x) y(x) = b(x), \quad c < x < d \]

subject to boundary conditions

\[ y(c) = A_0, \quad y(d) = C_0, \quad y'(c) = A_1, \quad y'(d) = C_1, \quad y''(c) = A_2, \quad y''(d) = C_2, \quad y'''(c) = A_3, \quad y'''(d) = C_3, \quad y^{(4)}(c) = A_4, \quad y^{(4)}(d) = C_4. \]

where \( A_0, C_0, A_1, C_1, A_2, C_2, A_3, C_3, A_4, C_4 \) are finite real constants and \( a_0(x), a_1(x), a_2(x), a_3(x), a_4(x), a_5(x), a_6(x), a_7(x), a_8(x), a_9(x), a_{10}(x), b(x) \) are all continuous functions defined on the interval \([c, d]\).

Generally, this type of tenth order boundary value problem arises in the study of hydrodynamics and hydro magnetic stability, mathematical modeling of the viscoelastic flows and other areas of applied mathematics, physics, engineering sciences. When an infinite horizontal layer of fluid is heated from below and is under the action of rotation, instability sets in. When this instability is an ordinary convection, the ordinary differential equation is of sixth
order. When the instability sets in as over stability, it is modeled by an eighth order ordinary differential equation. Suppose, now that a uniform magnetic field is also applied across the fluid in the same direction as gravity. When instability sets now as ordinary convection, it is modeled by a tenth order boundary value problem [1].

The existence and uniqueness of solutions of these problems have been discussed by Agarwal [2]. The boundary value problems of higher order differential equations have been investigated due to their mathematical importance and the potential for applications in diversified applied sciences. Solving these type of boundary value problems analytically is very difficult and analytical solutions are available in very rare cases. Very few authors have attempted the numerical solution of tenth order boundary value problems. Some of the numerical methods have been developed over the years to approximate the solution for these type of boundary value problems. Twizell et al. [3] developed numerical methods for eighth, tenth, twelfth order eigen value problems arising in thermal instability. Siddiqi and Twizell [4] developed the solution of special case of tenth order boundary value problems using tenth degree splines. Ghazala Akram and Siddiqi [5, 6] presented the solution of special case of tenth order boundary value problems using an eleventh degree polynomial and non-polynomial splines. Scott and Watts [7] applied a combination of superposition and orthonormalization to solve a linear boundary value problems. Scott and Watson [8] described several computer codes that were developed using superposition and orthonormalization technique and invariant imbedding.


In this paper, we try to present a simple finite element method which involves a Galerkin approach with septic B-splines as basis functions to solve a general tenth order two point boundary value problem of the type (1)-(2). This paper is organized as follows. Section 2 deals with the justification for using the Galerkin method. In Section 3, a description of the Galerkin method with septic B-splines as basis functions is explained. In particular, we first introduce the basic concept of septic B-splines and followed by the proposed method. In Section 4, the
Numerical Solution of Tenth Order Boundary Problems
by Galerkin Method with Septic B-splines

procedure to solve the nodal parameters has been presented. In Section 5, the proposed method is tested on several linear and nonlinear boundary value problems. The solution to a nonlinear problem has been obtained as the limit of a sequence of solution of linear problems generated by the quasilinearization technique [22]. Finally, in the last Section, the conclusions are presented.

2. Justification for using Galerkin method

For the few decades, the finite element method (FEM) has become very powerful, useful tool to solve the boundary value problems in the complex geometry. In FEM, the approximate solution can be written as a linear combination of basis functions which constitute a basis for the approximation space under consideration. FEM involves variational methods like Rayleigh Ritz, Galerkin, Petrov-Galerkin, Least Squares and Collocation etc.

In Galerkin method, the residual of approximation is made orthogonal to the basis functions. When one uses Galerkin method, a weak form of approximation solution for a given differential equation exists and is unique under appropriate conditions [23, 24] irrespective of properties of a given differential operator. Further, a weak solution also tends to a classical solution of given differential equation, provided sufficient attention is given to boundary conditions [25]. That means the basis functions should vanish on the boundary where the Dirichlet type of boundary conditions are prescribed. Hence in this paper we employed the use of Galerkin method with septic B-splines as basis functions to approximate the solution of tenth order boundary value problems.

3. Description of the method

Definition of septic B-spline: The cubic B-splines are defined in [26-28]. In a similar analogue, the septic B-splines can be defined. The existence of septic spline interpolate } (x) to a function in a closed interval } (c, d] for spaced knots (need not be evenly spaced) of a partition } (c = x_0 < x_1 < ... < x_{n+1} = d) is established by constructing it. The construction of } (x) is done with the help of the septic B-splines. Introduce fourteen additional knots } (x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_{n+1}, x_{n+2}, x_{n+3}, x_{n+4}, x_{n+5}, x_{n+6} and x_{n+7} in such a way that } (x_7 < x_6 < x_5 < x_4 < x_3 < x_2 < x_1 < x_0 and x_0 < x_{n+1} < x_{n+2} < x_{n+3} < x_{n+4} < x_{n+5} < x_{n+6} < x_{n+7}.

Now the septic B-splines } (B_i(x))'s are defined by

\[
B_i(x) = \begin{cases} 
\sum_{r=i-4}^{i+4} \frac{(x - x_r)^7}{\pi'(x_r)}, & x \in [x_{i-4}, x_{i+4}] \\
0, & \text{otherwise}
\end{cases}
\]

where } (x - x_r)^7 = \begin{cases} 
(x - x_r)^7, & \text{if } x_r \geq x \\
0, & \text{if } x_r \leq x
\end{cases}

and \( \pi(x) = \prod_{r=i-4}^{i+4} (x - x_r) \)

where } (B_3(x), B_2(x), B_1(x), B_0(x), ..., B_d(x), B_{n+1}(x), B_{n+2}(x), B_{n+3}(x)) forms a basis for the space } (S_7(\pi)) of septic polynomial splines. Schoenberg [28] has proved that septic B-splines are the unique nonzero splines of smallest compact support with the knots at

To solve the boundary value problem (1) and (2) by the Galerkin method with septic B-splines as basis functions, we define the approximation for $y(x)$ as

$$y(x) = \sum_{j=3}^{n+3} \alpha_j B_j(x)$$

where $\alpha_j$'s are the nodal parameters to be determined. In Galerkin method, the basis functions should vanish on the boundary where the Dirichlet type of boundary conditions are specified. In the set of septic B-splines \{ $B_3(x), B_2(x), B_1(x), \ldots, B_n(x), B_{n+1}(x), B_{n+2}(x), B_{n+3}(x)$ \}, the basis functions $B_3(x), B_2(x), B_1(x), B_0(x), \ldots, B_n(x), B_{n+1}(x), B_{n+2}(x)$ and $B_{n+3}(x)$ do not vanish at one of the boundary points. So, there is a necessity of redefining the basis functions into a new set of basis functions which vanish on the boundary where the Dirichlet type of boundary conditions are specified.

Since, we are approximating the tenth order boundary value problem by septic B-splines polynomial, we redefine the basis functions into a new set of basis functions which vanish on the boundary where the given set of boundary conditions are prescribed. The procedure for redefining of the basis functions is as follows.

Using the definition of septic B-splines and the Dirichlet boundary conditions of (2), we get the approximate solution at the boundary points as

$$A_0 = y(e) = y(x_0) = \alpha_{-3} B_{-3}(x_0) + \alpha_{-2} B_{-2}(x_0) + \alpha_{-1} B_{-1}(x_0) + \alpha_0 B_0(x_0) + \alpha_1 B_1(x_0)$$

$$C_0 = y(d) = y(x_n) = \alpha_{n+3} B_{n+3}(x_n) + \alpha_{n+2} B_{n+2}(x_n) + \alpha_{n+1} B_{n+1}(x_n) + \alpha_n B_n(x_n)$$

Eliminating $\alpha_{-3}$ and $\alpha_{n+3}$ from the equations (3), (4) and (5), we get

$$y(x) = w_i(x) + \sum_{j=-2}^{n+2} \alpha_j P_j(x)$$

where $w_i(x) = \frac{A_0}{B_{-3}(x_0)} B_{-3}(x) + \frac{C_0}{B_{n+3}(x_n)} B_{n+3}(x)$

$$P_j(x) = \begin{cases} B_j(x) \frac{B_j(x_0)}{B_{-3}(x_0)} B_{-3}(x), & j = -2, -1, 0, 1, 2, 3 \\ B_j(x), & j = 4, \ldots, n-4 \\ B_j(x) \frac{B_j(x_n)}{B_{n+3}(x_n)} B_{n+3}(x), & j = n-3, n-2, n-1, n, n+1, n+2. \end{cases}$$

Using the Neumann boundary conditions of (2) to the approximate solution $y(x)$ given by (6), we get

$$A_1 = y'(e) = y'(x_0) = w'_i(x_0) + \alpha_{-2} P'_2(x_0) + \alpha_{-1} P'_1(x_0) + \alpha_0 P'_0(x_0) + \alpha_1 P'_1(x_0) + \alpha_2 P'_2(x_0) + \alpha_3 P'_3(x_0)$$

$$C_1 = y'(d) = y'(x_n) = \alpha_{n+3} P'_0(x_n) + \alpha_{n+2} P'_1(x_n) + \alpha_{n+1} P'_2(x_n) + \alpha_n P'_3(x_n)$$

250 \ Int. J. Appl. Sci. Eng., 2015. 13, 3
Numerical Solution of Tenth Order Boundary Problems
by Galerkin Method with Septic B-splines

\[ C_1 = y'(d) = y'(x_n) = w'_1(x_n) + \alpha_{n-3}P'_{n-3}(x_n) + \alpha_{n-2}P'_{n-2}(x_n) + \alpha_{n-1}P'_{n-1}(x_n) + \alpha_nP'_n(x_n) \]  
\[ + \alpha_{n+1}P'_{n+1}(x_n) + \alpha_{n+2}P'_{n+2}(x_n) \]  
\[ (10) \]

Eliminating \( \alpha_2 \) and \( \alpha_{n+2} \) from the equations (6), (9) and (10), we get the approximation for \( y(x) \) as

\[ y(x) = w_2(x) + \sum_{j=1}^{n+1} \alpha_jQ_j(x) \]  
\[ (11) \]

where \( w_2(x) = w_1(x) + \frac{A_1 - w'_1(x_n)}{P'_2(x_n)}P_2(x) + \frac{C_1 - w'_1(x_n)}{P'_n+2(x_n)}P_{n+2}(x) \)
\[ (12) \]

and \( Q_j(x) = \begin{cases} 
    P_j(x), & j = 0, 1, 2, 3 \\
    P_j(x) - \frac{P'_j(x_n)}{P'_{n+2}(x_n)}P_{n+2}(x), & j = -1, 0, 1, 2, 3 \\
    P_j(x) - \frac{P'_j(x_n)}{P'_{n+2}(x_n)}P_{n+2}(x), & j = n-3, n-2, n-1, n+1.
\end{cases} \]  
\[ (13) \]

Using the second order derivative boundary conditions of (2) to the approximate solution \( y(x) \) given by (11), we get

\[ A_2 = y''(c) = y''(x_0) = w''_2(x_0) + \alpha_{n-2}Q''_n(x_0) + \alpha_nQ''_n(x_0) \]  
\[ (14) \]

\[ C_2 = y''(d) = y''(x_n) = w''_2(x_n) + \alpha_{n-3}Q''_{n-3}(x_n) + \alpha_{n-2}Q''_{n-2}(x_n) + \alpha_{n-1}Q''_{n-1}(x_n) \]
\[ + \alpha_nQ''_n(x_n) + \alpha_{n+1}Q''_{n+1}(x_n) \]  
\[ (15) \]

Eliminating \( \alpha_3 \) and \( \alpha_{n+1} \) from the approximations (11), (14) and (15), we get the approximation for \( y(x) \) as

\[ y(x) = w_3(x) + \sum_{j=0}^{n} \alpha_jR_j(x) \]  
\[ (16) \]

where \( w_3(x) = w_2(x) + \frac{A_2 - w''_2(x_n)}{Q''_1(x_n)}Q_1(x) + \frac{C_2 - w''_2(x_n)}{Q''_{n+1}(x_n)}Q_{n+1}(x) \)
\[ (17) \]

and \( R_j(x) = \begin{cases} 
    Q_j(x), & j = 0, 1, 2, 3 \\
    Q_j(x) - \frac{Q''_j(x_n)}{Q''_{n+1}(x_n)}Q_{n+1}(x), & j = 4, ..., n-4 \\
    Q_j(x) - \frac{Q''_j(x_n)}{Q''_{n+1}(x_n)}Q_{n+1}(x), & j = n-3, n-2, n-1, n.
\end{cases} \]  
\[ (18) \]

Using the third order derivative boundary conditions of (2) to the approximate solution \( y(x) \) given by (16), we get

\[ A_3 = y'''(c) = y'''(x_0) = w'''_2(x_0) + \alpha_{n-3}R'''_{n-3}(x_0) + \alpha_nR'''_n(x_0) + \alpha_{n+1}R'''_{n+1}(x_0) \]  
\[ (19) \]

\[ C_3 = y'''(d) = y'''(x_n) = w'''_3(x_n) + \alpha_{n-2}R'''_{n-2}(x_n) + \alpha_{n-1}R'''_{n-1}(x_n) + \alpha_nR'''_n(x_n) \]  
\[ (20) \]
Eliminating $\alpha_0$ and $\alpha_n$ from the equations (16), (19) and (20), we get the approximation for $y(x)$ as

$$y(x) = w_4(x) + \sum_{j=1}^{n-1} \alpha_j S_j(x)$$  \hspace{1cm} (21)

where $w_4(x) = w_3(x) + A_1 - w_3'''(x_0) \frac{R_0''(x_0)}{R_0'''(x_0)} R_0(x)$  \hspace{1cm} (22)

and $S_j(x) =  \begin{cases} R_j(x), & j = 1, 2, 3 \\ R_j(x) - \frac{R_j'''(x_0)}{R_0'''(x_0)} R_0(x), & j = n - 3, n - 2, n - 1 \end{cases}$  \hspace{1cm} (23)

Using the fourth order derivative boundary conditions of (2) to the approximate solution $y(x)$ given by (21), we get

$$A_4 = y^{(4)}(c) = y^{(4)}(x_0) = w_4^{(4)}(x_0) + \alpha_1 S_1^{(4)}(x_0) + \alpha_2 S_2^{(4)}(x_0) + \alpha_3 S_3^{(4)}(x_0)$$  \hspace{1cm} (24)

$$C_4 = y^{(4)}(d) = y^{(4)}(x_n) = w_4^{(4)}(x_n) + \alpha_{n-3} S_{n-3}^{(4)}(x_0) + \alpha_{n-2} S_{n-2}^{(4)}(x_n) + \alpha_{n-1} S_{n-1}^{(4)}(x_n)$$  \hspace{1cm} (25)

Eliminating $\alpha_1$ and $\alpha_{n-1}$ from the equations (21), (24) and (25), we get the approximation for $y(x)$ as

$$y(x) = w(x) + \sum_{j=2}^{n-2} \alpha_j \tilde{B}_j(x)$$  \hspace{1cm} (26)

where $w(x) = w_4(x) + \frac{A_4 - w_4^{(4)}(x_0)}{S_1^{(4)}(x_0)} S_1(x) + \frac{C_4 - w_4^{(4)}(x_n)}{S_{n-2}^{(4)}(x_n)} S_{n-2}(x)$  \hspace{1cm} (27)

and $\tilde{B}_j(x) = \begin{cases} S_j(x), & j = 2, 3 \\ S_j(x) - \frac{S_j^{(4)}(x_0)}{S_1^{(4)}(x_0)} S_1(x), & j = 2, 3 \\ S_j(x) - \frac{S_j^{(4)}(x_n)}{S_{n-1}^{(4)}(x_n)} S_{n-1}(x), & j = n - 3, n - 2 \end{cases}$  \hspace{1cm} (28)

Now the new set of basis functions for the approximation $y(x)$ is $\{\tilde{B}_j(x), j = 2, ..., n - 2\}$.

Applying the Galerkin method to (1) with a new set of basis functions, we get

$$\int_{a_0}^{a_0} \left[ a_0(x) y^{(0)}(x) + a_1(x) y^{(0)}(x) + a_2(x) y^{(0)}(x) + a_3(x) y^{(0)}(x) + a_4(x) y^{(0)}(x) + a_5(x) y^{(0)}(x) + a_6(x) y^{(0)}(x) + a_7(x) y^{(0)}(x) + a_8(x) y^{(0)}(x) \right] \tilde{B}_i(x) dx = \int_{a_0}^{a_0} b(x) \tilde{B}_i(x) dx \text{ for } i = 2, 3, ..., n - 2$$  \hspace{1cm} (29)
Integrating by parts the first four terms on the left hand side of (29) and after applying the boundary conditions prescribed in (2), we get

\[ \int_{x_0}^{x} a_0(x) \tilde{B}_i(x) y^{(10)}(x) dx = - \int_{x_0}^{x} \frac{d^5}{dx^5} \left[ a_0(x) \tilde{B}_i(x) \right] y^{(5)}(x) dx \]  
(30)

\[ \int_{x_0}^{x} a_1(x) \tilde{B}_i(x) y^{(9)}(x) dx = \int_{x_0}^{x} \frac{d^4}{dx^4} \left[ a_1(x) \tilde{B}_i(x) \right] y^{(5)}(x) dx \]  
(31)

\[ \int_{x_0}^{x} a_2(x) \tilde{B}_i(x) y^{(8)}(x) dx = - \int_{x_0}^{x} \frac{d^3}{dx^3} \left[ a_2(x) \tilde{B}_i(x) \right] y^{(5)}(x) dx \]  
(32)

\[ \int_{x_0}^{x} a_3(x) \tilde{B}_i(x) y^{(7)}(x) dx = \int_{x_0}^{x} \frac{d^2}{dx^2} \left[ a_3(x) \tilde{B}_i(x) \right] y^{(5)}(x) dx \]  
(33)

Substituting (30) to (34) in (29) and using the approximation for \( y(x) \) given in (26), and after rearranging the terms for resulting equations, we get a system of equations in the matrix form as

\[ \mathbf{A} \alpha = \mathbf{B} \]  
(34)

where \( \mathbf{A} = [a_{ij}] \); 
\( a_{ij} = \int_{x_0}^{x} \left[ a_4(x) \tilde{B}_i(x) \tilde{B}_j^{(6)}(x) + \frac{d}{dx} \left( a_0(x) \tilde{B}_i(x) \right) \tilde{B}_j^{(5)}(x) + \frac{d^4}{dx^4} \left( a_1(x) \tilde{B}_i(x) \right) \tilde{B}_j^{(4)}(x) + a_6(x) \tilde{B}_i(x) \tilde{B}_j^{(3)}(x) + a_7(x) \tilde{B}_i(x) \tilde{B}_j^{(2)}(x) + a_{10}(x) \tilde{B}_i(x) \tilde{B}_j(x) \right] dx \)

for \( i=2, 3, \ldots, n-2; j=2, 3, \ldots, n-2. \)

\( \mathbf{B} = [b_i]; \)

\( b_i = \int_{x_0}^{x} \left[ b(x) \tilde{B}_i(x) - a_4(x) \tilde{B}_i(x) w^{(6)}(x) + \frac{d^2}{dx^2} \left( a_0(x) \tilde{B}_i(x) \right) \right] dx \)

\( \frac{d^3}{dx^3} \left( a_2(x) \tilde{B}_i(x) \right) - \frac{d^2}{dx^2} \left( a_2(x) \tilde{B}_i(x) \right) - a_3(x) \tilde{B}_i(x) w^{(5)}(x) - a_6(x) \tilde{B}_i(x) w^{(4)}(x) \)

\( - a_7(x) \tilde{B}_i(x) w^{(3)}(x) - a_8(x) \tilde{B}_i(x) w^{(2)}(x) - a_9(x) \tilde{B}_i(x) w'(x) - a_{10}(x) \tilde{B}_i(x) w(x) \right] dx \)

and \( \alpha = [\alpha_2 \alpha_3 \ldots \alpha_{n-2}]^T, \) for \( i = 2, 3, \ldots, n-2. \)
4. Procedure to find a solution for nodal parameters

A typical integral element in the matrix \( A \) is
\[
\sum_{m=0}^{n-1} I_m, \quad I_m = \int_{x_m}^{x_{m+1}} r_i(x) r_j(x) Z(x) dx
\]
and \( r_i(x), r_j(x) \) are the septic B-spline basis functions or their derivatives. It may be noted that
\[
I_m = 0 \quad \text{if} \quad (x_{m+4}, x_{j+4}) \cap (x_{j+4}, x_{j+7}) = \emptyset.
\]
To evaluate each \( I_m \), we employed 8-point Gauss-Legendre quadrature formula. Thus the stiffness matrix \( A \) is a fifteen diagonal band matrix. The nodal parameter vector \( \alpha \) has been obtained from the system \( A \alpha = B \) by using a band matrix solution package. We have used the FORTRAN-90 program to solve the boundary value problems (1) - (2) by the proposed method.

5. Numerical results

To test the efficiency of the proposed method for solving the tenth order boundary value problems of the types (1) and (2), we considered three linear boundary value problems and three nonlinear boundary value problems. Numerical results for each problem are presented in tabular forms and compared with the exact solutions available in the literature.

When a differential equation is approximated by \( m^{th} \) order splines, it yields an accuracy of \((m+1)^{th}\) order [26]. The methods used by the other authors to solve the tenth order boundary value problem contain of using B-splines of order more than 7. Therefore their methods yield an accuracy of order more than 8. Whereas in our proposed method, we solved the tenth order boundary value problems by using septic B-splines which yields an accuracy of order 8. Our proposed method is an easy one to implement and solving the tenth order boundary value problem. That’s why we have not compared our results with the other methods.

Example 1
Consider the linear boundary value problem
\[
y^{(10)} + y = -10(2\sin x - 9\cos x), \quad -1 \leq x \leq 1
\] (37)
subject to \( y(-1) = y(1) = 0 \), \( y'(1) = 2\cos 1 \), \( y''(-1) = y''(1) = 2\cos 1 - 4\sin 1 \),
\( y'''(-1) = -y'''(1) = 6\cos 1 + 6\cos 1 \), \( y^{(4)}(-1) = -12\cos 1 + 8\sin 1 \), \( y^{(4)}(1) = -12\cos 1 + 8\sin 1 \).
The exact solution for the above problem is \( y = (x^2 - 1)\cos x \).

The proposed method is tested on this problem where the domain \([-1, 1]\) is divided into 10 equal subintervals. The obtained numerical results for this problem are given in Table 1. The maximum absolute error obtained by the proposed method is 2.920628x10^{-5}.

Example 2
Consider the linear boundary value problem
\[
y^{(10)} - (x^2 - 2x) y = 10\cos x - (x - 1)^3 \sin x, \quad -1 \leq x \leq 1
\] (38)
subject to \( y(-1) = 2\sin 1 \), \( y(1) = 0 \), \( y'(-1) = -2\cos 1 - \sin 1 \), \( y'(1) = \sin 1 \), \( y''(-1) = 2\cos 1 - 2\sin 1 \),\n\( y''(1) = 2\cos 1, \quad y'''(-1) = 2\cos 1 + 3\sin 1 \), \( y'''(1) = -3\sin 1 \), \( y^{(4)}(-1) = -4\cos 1 + 2\sin 1 \),\n\( y^{(4)}(1) = -4\cos 1 \). The exact solution for the above problem is \( y = (x - 1)\sin x \).
Numerical Solution of Tenth Order Boundary Problems  
by Galerkin Method with Septic B-splines 

### Table 1. Numerical results for Example 1

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.8</td>
<td>-2.508144E-01</td>
<td>5.483627E-06</td>
</tr>
<tr>
<td>-0.6</td>
<td>-5.282148E-01</td>
<td>9.536743E-07</td>
</tr>
<tr>
<td>-0.4</td>
<td>-7.736912E-01</td>
<td>8.702278E-06</td>
</tr>
<tr>
<td>-0.2</td>
<td>-9.408639E-01</td>
<td>2.980232E-06</td>
</tr>
<tr>
<td>0.0</td>
<td>-1.0000000000</td>
<td>1.955032E-05</td>
</tr>
<tr>
<td>0.2</td>
<td>-9.408639E-01</td>
<td>2.920628E-05</td>
</tr>
<tr>
<td>0.4</td>
<td>-7.736912E-01</td>
<td>2.169609E-05</td>
</tr>
<tr>
<td>0.6</td>
<td>-5.282148E-01</td>
<td>7.390761E-05</td>
</tr>
<tr>
<td>0.8</td>
<td>-2.508144E-01</td>
<td>7.450818E-07</td>
</tr>
</tbody>
</table>

The proposed method is tested on this problem where the domain [-1, 1] is divided into 10 equal subintervals. The obtained numerical results for this problem are given in Table 2. The maximum absolute error obtained by the proposed method is 2.110004x10^-5.

### Example 3
Consider the linear boundary value problem

\[
y^{(10)} - y'' + xy = (-8 + x - x^2)e^x, \quad 0 < x < 1
\]  

subject to \( y(0) = 1, \ y'(0) = 0, \ y''(0) = 0, \ y''(1) = -e, \ y'''(0) = -2e, \ y'''(1) = -3e, \ y''(0) = -3, \ y''(1) = -4e. \) The exact solution for the above problem is \( y = (1 - x)e^x. \)

The proposed method is tested on this problem where the domain [0, 1] is divided into 10 equal subintervals. The obtained numerical results for this problem are given in Table 3. The maximum absolute error obtained by the proposed method is 4.452467x10^-5.

### Table 2. Numerical results for Example 2

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.8</td>
<td>1.291241</td>
<td>4.649162E-06</td>
</tr>
<tr>
<td>-0.6</td>
<td>9.034280E-01</td>
<td>1.329184E-05</td>
</tr>
<tr>
<td>-0.4</td>
<td>5.451856E-01</td>
<td>2.050400E-05</td>
</tr>
<tr>
<td>-0.2</td>
<td>2.384032E-01</td>
<td>9.477139E-06</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0000000000</td>
<td>2.731677E-06</td>
</tr>
<tr>
<td>0.2</td>
<td>-1.589355E-01</td>
<td>1.458824E-05</td>
</tr>
<tr>
<td>0.4</td>
<td>-2.336510E-01</td>
<td>2.110004E-05</td>
</tr>
<tr>
<td>0.6</td>
<td>-2.258570E-01</td>
<td>1.908839E-05</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.434712E-01</td>
<td>1.342595E-05</td>
</tr>
</tbody>
</table>
Table 3. Numerical results for Example 3

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>9.946538E-01</td>
<td>1.537800E-05</td>
</tr>
<tr>
<td>0.2</td>
<td>9.771222E-01</td>
<td>4.452467E-05</td>
</tr>
<tr>
<td>0.3</td>
<td>9.449012E-01</td>
<td>3.331900E-05</td>
</tr>
<tr>
<td>0.4</td>
<td>8.950948E-01</td>
<td>3.552437E-05</td>
</tr>
<tr>
<td>0.5</td>
<td>8.243606E-01</td>
<td>9.477139E-06</td>
</tr>
<tr>
<td>0.6</td>
<td>7.288475E-01</td>
<td>2.586842E-05</td>
</tr>
<tr>
<td>0.7</td>
<td>6.041259E-01</td>
<td>3.975630E-05</td>
</tr>
<tr>
<td>0.8</td>
<td>4.451082E-01</td>
<td>3.531575E-05</td>
</tr>
<tr>
<td>0.9</td>
<td>2.459602E-01</td>
<td>2.214313E-05</td>
</tr>
</tbody>
</table>

Example 4
Consider the nonlinear boundary value problem

\[ y^{(10)} + e^{-x} y^2 = e^{-x} + e^{-3x}, \quad 0 < x < 1 \]  

subject to \( y(0) = 1, \ y(1) = e^{-1}, \ y'(0) = -1, \ y'(1) = -e^{-1}, \ y''(0) = 1, \ y''(1) = e^{-1}, \ y''(0) = -1, \ y''(1) = -e^{-1}, \ y^{(4)}(0) = 1, \ y^{(4)}(1) = e^{-1}. \) The exact solution for the above problem is \( y = e^{-x}. \)

The nonlinear boundary value problem (40) is converted into a sequence of linear boundary value problems generated by quasilinearization technique [22] as

\[ y^{(10)}_{(n+1)} + [2y_{(n)}e^{-x}]y^{(1)}_{(n+1)} = [y_{(n)}]^2 e^{-x} + e^{-x} + e^{-3x}, \quad n = 0, 1, 2, 3, \ldots \]  

subject to \( y^{(1)}_{(n+1)}(0) = 1, \ y^{(1)}_{(n+1)}(1) = e^{-1}, \ y^{(1)}_{(n+1)}(0) = -1, \ y^{(1)}_{(n+1)}(1) = -e^{-1}, \ y^{(2)}_{(n+1)}(0) = 1, \ y^{(2)}_{(n+1)}(1) = e^{-1}, \ y^{(2)}_{(n+1)}(0) = -1, \ y^{(2)}_{(n+1)}(1) = -e^{-1}, \ y^{(4)}_{(n+1)}(0) = 1, \ y^{(4)}_{(n+1)}(1) = e^{-1}. \)

Here \( y_{(n+1)} \) is the \((n+1)\)th approximation for \( y(x) \). The domain \([0, 1]\) is divided into 10 equal subintervals and the proposed method is applied to the sequence of linear problems (41) Numerical results for this problem are presented in Table 4. The maximum absolute error obtained by the proposed method is 4.929304x10^{-5}.

Table 4. Numerical results for Example 4

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>9.048374E-01</td>
<td>6.735325E-06</td>
</tr>
<tr>
<td>0.2</td>
<td>8.187308E-01</td>
<td>4.410744E-06</td>
</tr>
<tr>
<td>0.3</td>
<td>7.408182E-01</td>
<td>3.629923E-05</td>
</tr>
<tr>
<td>0.4</td>
<td>6.703200E-01</td>
<td>4.839897E-05</td>
</tr>
<tr>
<td>0.5</td>
<td>6.065307E-01</td>
<td>4.929304E-05</td>
</tr>
<tr>
<td>0.6</td>
<td>5.488116E-01</td>
<td>3.945827E-05</td>
</tr>
<tr>
<td>0.7</td>
<td>4.965853E-01</td>
<td>9.834766E-06</td>
</tr>
<tr>
<td>0.8</td>
<td>4.493290E-01</td>
<td>1.996756E-06</td>
</tr>
<tr>
<td>0.9</td>
<td>4.065697E-01</td>
<td>5.066395E-06</td>
</tr>
</tbody>
</table>
**Example 5**

Consider the nonlinear boundary value problem

\[ y^{(10)} - y''' = 2e^x y^2, \quad 0 < x < 1 \]  

subject to \( y(0) = 1, \ y(1) = e^{-1}, \ y'(0) = -1, \ y'(1) = -e^{-1}, \ y''(0) = 1, \ y''(1) = e^{-1}, \ y'''(0) = -1, \ y'''(1) = -e^{-1}, \) \( y^{(4)}(0) = 1, \ y^{(4)}(1) = e^{-1}. \) The exact solution for the above problem is \( y = e^{-x}. \)

The nonlinear boundary value problem (42) is converted into a sequence of linear boundary value problems generated by quasilinearization technique [22] as

\[
y_{(n+1)}^{(10)} - y_{(n+1)}''' - 4e^x y_{(n+1)} y_{(n+1)} = -2[y_{(n)}^2] e^x, \quad n = 0, 1, 2, 3, \ldots
\]

subject to \( y_{(n+1)}(0) = 1, \ y_{(n+1)}(1) = e^{-1}, \ y'_{(n+1)}(0) = -1, \ y'_{(n+1)}(1) = -e^{-1}, \ y''_{(n+1)}(0) = 1, \ y''_{(n+1)}(1) = e^{-1}, \ y'''_{(n+1)}(0) = -1, \ y'''_{(n+1)}(1) = -e^{-1}, \)

\( y^{(4)}_{(n+1)}(0) = 1, \ y^{(4)}_{(n+1)}(1) = e^{-1}. \)

Here \( y_{(n+1)} \) is the \((n+1)^{th}\) approximation for \( y(x) \). The domain \([0, 1]\) is divided into 10 equal subintervals and the proposed method is applied to the sequence of linear problems (43). Numerical results for this problem are presented in Table 5. The maximum absolute error obtained by the proposed method is \( 4.929304 \times 10^{-5} \).

**Table 5. Numerical results for Example 5**

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>9.048374E-01</td>
<td>6.735325E-06</td>
</tr>
<tr>
<td>0.2</td>
<td>8.187308E-01</td>
<td>4.410744E-06</td>
</tr>
<tr>
<td>0.3</td>
<td>7.408182E-01</td>
<td>3.629923E-05</td>
</tr>
<tr>
<td>0.4</td>
<td>6.703200E-01</td>
<td>4.839897E-05</td>
</tr>
<tr>
<td>0.5</td>
<td>6.065307E-01</td>
<td>4.92904E-05</td>
</tr>
<tr>
<td>0.6</td>
<td>5.488116E-01</td>
<td>3.945827E-05</td>
</tr>
<tr>
<td>0.7</td>
<td>4.965853E-01</td>
<td>9.834766E-06</td>
</tr>
<tr>
<td>0.8</td>
<td>4.493290E-01</td>
<td>1.996756E-06</td>
</tr>
<tr>
<td>0.9</td>
<td>4.065697E-01</td>
<td>5.066395E-06</td>
</tr>
</tbody>
</table>

**Example 6**

Consider the nonlinear boundary value problem

\[ y^{(10)} = \frac{14175}{4} (x + y + 1)^{11}, \quad 0 \leq x \leq 1 \]  

subject to \( y(0) = 1, \ y(1) = 0, \ y'(0) = -\frac{1}{2}, \ y'(1) = 1, \ y''(0) = \frac{1}{2}, \ y''(1) = 4, \ y'''(0) = \frac{3}{4}, \ y'''(1) = 12, \ y^{(4)}(0) = \frac{3}{2}, \ y^{(4)}(1) = 48. \) The exact solution for the above problem is \( y = \frac{2}{2-x} - x - 1. \)
The nonlinear boundary value problem (44) is converted into a sequence of linear boundary value problems generated by quasilinearization technique [22] as

\[ y^{(10)}_{(n+1)} - \frac{14175 \times 11}{4} (x + y_{(n)} + 1)^{10} y^{(n+1)}_{(n+1)} = \frac{14175}{4} (x + y_{(n)} + 1)^{10} (1 + x - 10 y_{(n)}), \quad n = 0, 1, 2, \ldots \quad (45) \]

subject to

\[ y_{(n+1)}(0)=1, \quad y_{(n+1)}(1)=0, \quad y'_{(n+1)}(0) = \frac{-1}{2}, \quad y''_{(n+1)}(0) = \frac{1}{2}, \quad y'''_{(n+1)}(0) = 4, \quad y''''_{(n+1)}(0) = \frac{3}{4}, \]

\[ y''''_{(n+1)}(1) = 12, \quad y^{(4)}_{(n+1)}(0) = \frac{3}{2}, \quad y^{(4)}_{(n+1)}(1) = 48. \]

Here \( y_{(n+1)} \) is the \((n+1)^{th}\) approximation for \( y(x) \). The domain \([0, 1]\) is divided into 10 equal subintervals and the proposed method is applied to the sequence of linear problems (45). Numerical results for this problem are presented in Table 6. The maximum absolute error obtained by the proposed method is \( 1.598895 \times 10^{-5} \).

### Table 6. Numerical results for Example 6

<table>
<thead>
<tr>
<th>x</th>
<th>Exact Solution</th>
<th>Absolute error by the proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>-4.736842E-02</td>
<td>2.000481E-06</td>
</tr>
<tr>
<td>0.2</td>
<td>-8.888889E-02</td>
<td>5.461276E-06</td>
</tr>
<tr>
<td>0.3</td>
<td>-1.235294E-01</td>
<td>1.087785E-06</td>
</tr>
<tr>
<td>0.4</td>
<td>-1.500000E-01</td>
<td>2.965331E-06</td>
</tr>
<tr>
<td>0.5</td>
<td>-1.666667E-01</td>
<td>1.206994E-06</td>
</tr>
<tr>
<td>0.6</td>
<td>-1.714286E-01</td>
<td>1.050532E-05</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.615385E-01</td>
<td>1.598895E-05</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.333333E-01</td>
<td>1.588464E-05</td>
</tr>
<tr>
<td>0.9</td>
<td>-8.181816E-02</td>
<td>1.063198E-05</td>
</tr>
</tbody>
</table>

### 6. Conclusions

In this paper, we have deployed a Galerkin method with septic B-splines as basis functions to solve a general tenth order boundary value problem. The septic B-splines basis set has been redefined into a new set of basis functions which vanish on the boundary where the given set of boundary conditions are prescribed. The proposed method has been tested on three linear and three nonlinear tenth order boundary value problems. The numerical results obtained by the proposed method are in good agreement with the exact solutions available in the literature. The objective of this paper is to present a simple and accurate method to solve a general tenth order boundary value problem.

### References


Numerical Solution of Tenth Order Boundary Problems
by Galerkin Method with Septic B-splines


