SciencePG
Science Publishing Group

Numerical Solution of Linear Second Order Ordinary Differential Equations with Mixed Boundary Conditions by Galerkin Method

Akalu Abriham Anulo[1, *], Alemayehu Shiferaw Kibret[2], Genanew Gofe Gonfa[2], Ayana Deressa Negassa[2]

[1]Department of Mathematics, Institute of Technology, Dire Dawa University, Dire Dawa, Ethiopia

[2]Department of Mathematics, Jimma University, Jimma, Ethiopia

Email address:

ake.abrish@gmail.com (A. A. Anulo), kehali1k@gmail.com (A. S. Kibret), genagofa@gmail.com (G. G. Gonfa), ayanaderesa@gmail.com (A. D. Negassa)

[*]Corresponding author

Abstract: In this paper, the Galerkin method is applied to second order ordinary differential equation with mixed boundary after converting the given linear second order ordinary differential equation into equivalent boundary value problem by considering a valid assumption for the independent variable and also converting mixed boundary condition in to Neumann type by using secant and Runge-Kutta methods. The resulting system of equation is solved by direct method. In order to check to what extent the method approximates the exact solution, a test example with known exact solution is solved and compared with the exact solution graphically as well as numerically.

Keywords: Second Order Ordinary Differential Equation, Mixed Boundary Conditions, Runge-Kutta, Secant Method, Galerkin Method, Chebyshev Polynomials

1. Introduction

The goal of numerical analysis is to find the approximate numerical solution to some real physical problems by using different numerical techniques, especially when analytical solutions are not available or very difficult to obtain. Since most of mathematical models of physical phenomena are expressed in terms of ordinary differential equations, and these equations due to their nature and further applications to use computers, it needs to establish appropriate numerical methods corresponding to the type of the differential equation and conditions that govern the mathematical model of the physical phenomena. The conditions may be specified as an initial Value (IVP) or at the boundaries of the system, Boundary Value (BVP) [1].

Many problems in engineering and science can be formulated as two-point BVPs, like mechanical vibration analysis, vibration of spring, electric circuit analysis and many others. This shows that the numerical methods used to approximate the solutions of two-point boundary value problems play a vital role in all branches of sciences and engineering [2].

Among different numerical methods used to approximate two-point boundary value problems in terms of differential equations are shooting method, finite difference methods, finite element methods (FEM), Variational methods (Weighted residual methods, Ritz method) and others have been used to solve the two-point boundary value problems [3]. Both in FEM and Variational methods the main attempts were to look an approximation solution in the form of a linear combination of suitable approximation function and undetermined coefficients [4]. For a vector space of functions V, if $S = \left\{ \phi_i(x) \right\}_{i=1}^{\infty}$ be basis of V, a set of linearly independent functions, any function $f(x) \in V$ could be uniquely written as a linear combination of the basis as:

$$f(x) = \sum_{j=1}^{\infty} c_j \phi_j(x) \qquad (1)$$

The weighted residual methods use a finite number of linearly independent functions $\{\phi_i(x)\}_{i=1}^{n}$ as trial function.

Suppose that the approximation solution of the differential equation, $D(u) = L(u(x)) + f(x) = 0$, on the boundary $B(u) = [a, b]$ is in the form:

$$u(x) \approx U_N(x) = \sum_{j=1}^{N} c_j \phi_j(x) + \phi_0(x) \qquad (2)$$

Where $U_N(x)$ is the approximate solution, $u(x)$ is the exact solution " L " is a differential operator, " f " is a given function, $\phi_j(x)'s$ are finite number of basis functions and c_j unknown coefficients for $j = 1, 2, ..., N$.

The residual $R(x, c_j)$ is defined as: $R(x, c_j) = D(U_N(x)) - (L(U_N(x)) + f(x))$. Now determine c_j by requiring R to vanish in a "weighted-residual" sense:

$$\int_a^b w_i(x) R(x, c_j) dx = 0 \quad (i = 1, 2, ..., N) \qquad (3)$$

Where $w_i(x)$ are a set of linearly independent functions, called weight functions, which in general can be different from the approximation functions $\phi_j(x)$, this method is known as the weighted-residual method.

2. Galerkin Method

If $\phi_j(x) = w_i(x)$ in equation (3), then the special name of the weighted-residual method is known as the Galerkin method. Thus Galerkin method is one of the weighted residual methods in which the approximation function is the same as the weight function and hence it is also used to find the approximate solution of two-point boundary value problems [4].

The Galerkin method was invented in 1915 by Russian mathematician Boris Grigoryevich Galerkin and the origin of the method is generally associated with a paper published by Galerkin in 1915 on the elastic equilibrium of rods and thin plates. He published his finite element method in 1915. The Galerkin method can be used to approximate the solution to ordinary differential equations, partial differential equations and integral equations [5].

Many authors have been used the Galerkin method to find approximate solution of ordinary differential equations with boundary condition. Among this, a spline solution of two point boundary value problems introduced in [6], a method for solutions of nonlinear second order multi-point boundary value problems produced in [7], in [8] linear and non-linear

differential equations were solved numerically by Galerkin method using a Bernstein polynomials basis, in [9] a numerical method is established to solved second order ordinary differential equation with Neumann and Cauchy boundary conditions using Hermite polynomials, in [10] a parametric cubic spline solution of two point boundary value problems were obtained, a second-order Neumann boundary value problem with singular nonlinearity for exact three positive solutions were solved [11], a Numerical solution of a singular boundary-value problem in non-Newtonian fluid mechanics were established [12], a Fourth Order Boundary Value Problems by Galerkin Method with Cubic B-splines were solved by considering different cases on the boundary condition [13] and a special successive approximations method for solving boundary value problems including ordinary differential equations were proposed.[14]

In this paper Galerkin method will be applied to the linear second order ordinary differential equation of the form

$$\alpha(x^*) \frac{d^2 y}{dx^{*2}} + \beta(x^*) \frac{dy}{dx^*} + \delta(x^*) y = g(x^*) \; ; \; a \le x^* \le b$$

with boundary condition $\begin{array}{l} y(a) = \mu_1 \\ y'(b) = \mu_2 \end{array}$ and

$$\alpha(x^*) \frac{d^2 y}{dx^{*2}} + \beta(x^*) \frac{dy}{dx^*} + \delta(x^*) y = g(x^*) \; ; \; a \le x^* \le b \quad \text{with}$$

boundary condition $\begin{array}{l} y'(a) = \beta_0 \\ y'(b) = \beta_1 \end{array}$

3. Chebyshev Polynomial

The polynomials whose properties and applications are discussed in this paper were 'discovered' almost a century ago by the Russian mathematician Chebyshev. It is a function defined using trigonometric functions $\cos\theta$ and $\sin\theta$ for $x \in [-1, 1]$. Chebyshev polynomials of first kind with degree n for $x \in [-1, 1]$ defined as:

$T_n(x) = \cos n\theta$, such that $\cos\theta = x$, for $-1 \le x \le 1$ and $n \ge 0$

Thus $T_n(x) = \cos(n \cos^{-1} x)$,

$$\Rightarrow T_0(x) = \cos(0) = 1 \text{ and } T_1(x) = \cos(\cos^{-1} x) = x$$

From the trigonometric identity,

$$\cos(n+l)\theta + \cos(n-l)\theta = $$
$$2 \cos n\theta \cos\theta - \sin(n\theta)\sin(\theta) + \sin(n\theta)\sin(\theta)$$

$$\Rightarrow \cos(n+l)\theta + \cos(n-l)\theta = 2 \cos n\theta \cos\theta$$

$$\Rightarrow T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x).$$ Thus using the recursive relation above for n=1, 2… there is a series of Chebyshev polynomial

$$T_1(x) = x$$

$$T_2(x) = 2x^2 - 1$$

$$T_3(x) = 4x^3 - 3x$$

$$T_4(x) = 8x^4 - 8x^2 + 1$$

$$T_5(x) = 16x^5 - 20x^3 + 5x \text{ etc.}$$

Here the coefficient of x^n in $T_n(x)$ is 2^{n-1}.

The figure 1 below shows the graph of the first seven Chebyshev polynomials for $x \in [-1,1]$.
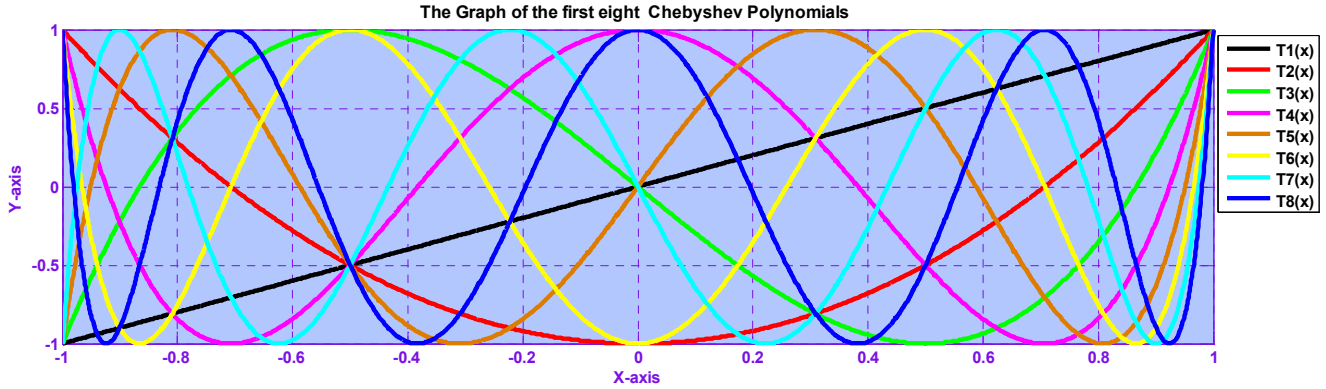

Figure 1. *The graph of the first eight Chebyshev polynomials for $x \in [-1,1]$.*

4. Runge-Kutta Method for Second Order ODE

Runge-Kutta method is a numerical method used to find approximate solution for initial value problems. In order to use Runge-Kutta method to find an approximate solution of second order ODE, it needs to convert in to a system of two first order ODEs. For two evaluation of f the method is given by

$$y_{j+1} = y_j + hy'_j + \frac{1}{2}(K_1 + K_2)$$

$$y'_{j+1} = y_j + hy'_j + \frac{1}{2h}(K_1 + 3K_2)$$

Where

$$K_1 = \frac{h^2}{2}f(x_j, y_j)$$

$$K_2 = \frac{h^2}{2}f(x_j + \frac{2}{3}h, y_j + \frac{2}{3}hy'_j + \frac{4}{9}K_1)$$

5. Secant Method

This method approximates the graph of the function $y = f(x)$ in the neighborhood of the root by a straight line (secant) passing through the points $(x_{k-1}, f(x_{k-1}))$ and $(x_k, f(x_k))$, where $f_k = f(x_k)$ and take the point of intersection of this line with the *x*-axis as the next iterate. Hence

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f_k - f_{k-1}} f_k, \text{ k=1,2, ...}$$

Or

$$x_{k+1} = \frac{x_{k-1}f_k - x_k f_{k-1}}{f_k - f_{k-1}}, \text{ k=1,2, ...}$$

Where x_{k-1} and x_k are two consecutive iterates. In this method there are two initial approximations x_0 and x_1. This method is also called Secant method.

6. Mathematical Formulation of the Method

Consider a general linear second order differential equation with two type boundary conditions

Type I: $\alpha(x^*)\dfrac{d^2y}{dx^{*2}} + \beta(x^*)\dfrac{dy}{dx^*} + \delta(x^*)y = g(x^*)$; $a \le x^* \le b$

with boundary condition

$$\begin{matrix} y(a) = \mu_1 \\ y'(b) = \mu_2 \end{matrix} \text{ (Mixed type)} \qquad (4)$$

Type II: $\alpha(x^*)\dfrac{d^2y}{dx^{*2}} + \beta(x^*)\dfrac{dy}{dx^*} + \delta(x^*)y = g(x^*)$; $a \le x^* \le b$

with boundary condition

$$\begin{matrix} y'(a) = \beta_0 \\ y'(b) = \beta_1 \end{matrix} \text{ (Neumann type)} \quad (5)$$

where $\alpha(x^*), \beta(x^*), \delta(x^*)$, $g(x^*)$ are given continuous functions for $a \le x^* \le b$ where β_0, β_1, μ_0 and μ_1 are given

constants and $y(x^*)$ is unknown function or exact solution of the boundary value problem which is to be determined.

In a BVP with mixed boundary condition, the solution is required to satisfy a Dirichlet or a Neumann boundary condition in a mutually exclusive way on disjoint parts of the boundary.

Now Consider the BVP of type II (Neumann type). To use an approximating polynomial defined for $x \in [-1,1]$ the given BVP defined on arbitrary interval [a, b] must be converted into an equivalent BVP defined on [-1, 1]. So that the approximating polynomial should be defined on [-1, 1]. Since Chebyshev polynomial is defined on [-1, 1], it is possible to use Chebyshev polynomial after converting the BVP defined on arbitrary interval [a, b] into an equivalent BVP defined on [-1, 1].

6.1. Conversion of the Domain of the BVP

The differential equation in (5) together with the Neumann boundary condition can be converted to an equivalent problem on [-1, 1] by letting

$$x^* = \frac{b-a}{2}x + \frac{b+a}{2}, \text{ for } -1 \leq x \leq 1 \text{ and } a \leq x^* \leq b$$

Then equation (4) with boundary condition is equivalent to the BVP given by

$$\tilde{\alpha}(x)\frac{d^2y}{dx^2} + \tilde{\beta}(x)\frac{dy}{dx} + \tilde{\delta}(x)y = \tilde{g}(x); \ -1 \leq x \leq 1 \quad \text{Subject to}$$

the boundary condition,

$$\begin{aligned} y'(-1) &= d_0 \\ y'(1) &= d_1 \end{aligned} \tag{6}$$

where

$$\tilde{\alpha}(x) = \frac{4}{(b-a)^2}\alpha\left(\frac{b-a}{2}x + \frac{b+a}{2}\right) \text{ since for } x^* = \frac{b-a}{2}x + \frac{b+a}{2},$$

This implies that

$$\frac{dy}{dx^*} = \frac{2}{(b-a)}\frac{dy}{dx} \text{ and } \frac{d^2y}{dx^{*2}} = \frac{4}{(b-a)^2}\frac{d^2y}{dx^2} \tag{7}$$

Now equating the D. E in (4) with (6),

$$\alpha(x^*)\frac{d^2y}{dx^{*2}} + \beta(x^*)\frac{dy}{dx^*} + \delta(x^*)y - g(x^*) =$$

$$\tilde{\alpha}(x)\frac{d^2y}{dx^2} + \tilde{\beta}(x)\frac{dy}{dx} + \tilde{\delta}(x)y - \tilde{g}(x)$$

$$\Rightarrow \tilde{\alpha}(x)\frac{d^2y}{dx^2} = \alpha(x^*)\frac{d^2y}{dx^{*2}},$$

$$\tilde{\beta}(x)\frac{dy}{dx} = \beta(x^*)\frac{dy}{dx^*}, \ \tilde{\delta}(x) = \delta(x^*) \qquad \text{and } \tilde{g}(x) = g(x^*) \tag{8}$$

Therefore, the DE in (5) with Neumann boundary condition is an equivalent BVP with the BVP in (6).

Up on substitution of (7) into (4), equation (4) yields

$$\frac{4}{(b-a)^2}\alpha(x^*)\frac{d^2y}{dx^2} = \tilde{\alpha}(x)\frac{d^2y}{dx^2},$$

$$\frac{2}{(b-a)}\beta(x^*)\frac{dy}{dx} = \tilde{\beta}(x)\frac{dy}{dx},$$

$$\tilde{\delta}(x)y = \delta(x^*)y \text{ and } \tilde{g}(x) = g(x^*)$$

$$\Rightarrow \tilde{\alpha}(x) = \frac{4}{(b-a)^2}\alpha(\frac{b-a}{2}x + \frac{b+a}{2}),$$

$$\tilde{\beta}(x) = \frac{2}{(b-a)}\beta(\frac{b-a}{2}x + \frac{b+a}{2}),$$

$$\tilde{\delta}(x) = \delta(\frac{b-a}{2}x + \frac{b+a}{2}) \text{ and}$$

$$\tilde{g}(x) = g(\frac{b-a}{2}x + \frac{b+a}{2})$$

6.2. Applying Galerkin Method

To apply the technique of Galerkin method to find an approximate solution of (4), say $\overline{y}(x)$, written as a linear combination of base functions and unknown constants. That is;

$$\overline{y}(x) = \sum_{i=0}^{n} c_i T_i(x) \tag{9}$$

where $T_i(x)$ are piecewise polynomial, namely Chebyshev polynomials of degree i and c_i's are unknown parameters, to be determined.

Now applying Galerkin method with the basis function $T_i(x)$ gives,

$$\int_{-1}^{1}[\tilde{\alpha}(x)\frac{d^2\overline{y}}{dx^2} + \tilde{\beta}(x)\frac{d\overline{y}}{dx} + \tilde{\delta}(x)\overline{y}]T_j(x)dx = \int_{-1}^{1}\tilde{g}(x)T_j(x)dx \tag{10}$$

Integrating the first term by parts on the left hand side of (10), that is

$$\int_{-1}^{1}T_j(x)\tilde{\alpha}(x)\frac{d^2\overline{y}}{dx^2}dx, \ u = T_j(x)\tilde{\alpha}(x) \text{ and } dv = \frac{d^2\overline{y}}{dx^2}dx$$

$$\Rightarrow du = \frac{d}{dx}\left[T_j(x)\tilde{\alpha}(x)\right]dx \text{ and } v = \frac{d\overline{y}}{dx}$$

$$\Rightarrow \int_{-1}^{1}\tilde{\alpha}(x)\frac{d^2\overline{y}}{dx^2}dx = uv - \int vdu$$

$$= \frac{d\overline{y}}{dx}T_j(x)\tilde{\alpha}(x)|_{-1}^{1} - \int_{-1}^{1}\frac{d\overline{y}}{dx}\frac{d}{dx}\left[T_j(x)\tilde{\alpha}(x)\right]dx \tag{11}$$

Upon substitution of (11) into (10), yields

$$\int_{-1}^{1}[-\frac{d\overline{y}}{dx}\frac{d}{dx}\left(\tilde{\alpha}(x)T_j(x)\right)+\tilde{\beta}(x)\frac{d\overline{y}}{dx}T_j(x)$$

$$+\tilde{\delta}(x)\overline{y}(x)T_j(x)]dx$$

$$=\int_{-1}^{1}\tilde{g}(x)T_j(x)dx+\tilde{\alpha}(-1)\overline{y}\,'(-1)T_j(-1)$$

$$-\tilde{\alpha}(1)\overline{y}\,'(1)T_j(1)$$

(12)

But from equation (9) the approximate solution is given by

$$\overline{y}(x)=\sum_{i=0}^{n}c_iT_i(x)$$

Substituting this into equation (12) yields,

$$\int_{-1}^{1}[-\sum_{i=0}^{n}c_iT_i'(x)\frac{d}{dx}\left(\tilde{\alpha}(x)T_j(x)\right)$$

$$+\tilde{\beta}(x)\sum_{i=0}^{n}c_iT_i'(x)T_j(x)+\tilde{\delta}(x)\sum_{i=0}^{n}c_iT_i(x)T_j(x)]dx$$

$$=\int_{-1}^{1}\tilde{g}(x)T_j(x)dx+\tilde{\alpha}(-1)\tilde{u}'(-1)T_j(-1)$$

$$-\tilde{\alpha}(1)\tilde{u}'(1)T_j(1)$$

$$\Rightarrow\sum_{i=0}^{n}c_i\int_{-1}^{1}[-T_i'(x)\frac{d}{dx}\left(\tilde{\alpha}(x)T_j(x)\right)+\tilde{\beta}(x)T_i'(x)T_j(x)$$

$$+\tilde{\delta}(x)T_i(x)T_j(x)]dx$$

$$=\int_{-1}^{1}\tilde{g}(x)T_j(x)dx+\tilde{\alpha}(-1)\overline{y}\,'(-1)T_j(-1)$$

$$-\tilde{\alpha}(1)\overline{y}\,'(1)T_j(1)$$

(13)

In the left hand side of equation (13) above it needs to know the values of $\overline{y}'(-1)$ and $\overline{y}'(1)$ which approximately equal to $y'(-1)$ and $y'(1)$ respectively, where y is the exact solution of the BVP in equation (6).

6.3. The Resulting System of Equation

Since the values of $y'(-1)$ and $y'(1)$ are known from the boundary condition, substituting these values into (13), and equation (13) gives a system of $n\times n$ equations to solve the parameters c_i's thus equation (13) in matrix form becomes:

$$\sum_{i=1}^{n}c_iK_{ij}=F_i$$

(14)

Where $K_{ij}=k_{ij}^{(1)}+k_{ij}^{(2)}+k_{ij}^{(3)}$ and $F_i=f_i^{(1)}+f_i^{(2)}$ such that

$$k_{ij}^{(1)}=\int_{-1}^{1}[-T_i'(x)\frac{d}{dx}\left(\tilde{\alpha}(x)T_j(x)\right)dx \quad k_{ij}^{(2)}=\int_{-1}^{1}\tilde{\beta}(x)T_i'(x)T_j(x)dx$$

$$k_{ij}^{(3)}=\int_{-1}^{1}\tilde{\delta}(x)T_i(x)T_j(x)dx \quad f_i^{(1)}=\int_{-1}^{1}\tilde{g}(x)T_j(x)dx$$

$$f_i^{(2)}=\tilde{\alpha}(-1)\overline{y}'(-1)T_j(-1)-\tilde{\alpha}(1)\overline{y}'(1)T_j(1)$$

Now, the unknown parameters c_i's are determined by solving the system of equation in (14) by direct method and substituting these values into (9) yields the approximate solution $\overline{y}(x)$ of the DE in (4) satisfying the given boundary conditions in (6).

Consider a BVP of type I. In this case it is impossible to use the above method directly; since $y'(a)$ is not given and hence instead it needs to convert the BVP in to type II. The conversion is made by using different numerical methods.

Consider to solve the following boundary-value problem:

$$\alpha(x)\frac{d^2y}{dx^2}+\beta(x)\frac{dy}{dx}+\delta(x)y=g(x)\ ;\ a\le x\le b$$

(15)

With boundary condition
$$\begin{aligned}y(a)&=\mu_1\\y'(b)&=\mu_2\end{aligned}$$

The idea of shooting method for (15) is to solve for $y'(a)$ hoping that $y'(b)=\mu_2$. In order to find $y'(a)$ such that $y'(b)=\mu_2$, guess $y'(a)=z$ and solve for $y'(b)$ using Runge-Kutta method for second order ODE, after having a value using the guess, denote this approximate solution y_z and hope $y_z'(b)=\mu_2$. If not, use another guess for $y'(a)$, and try to solve using the Runge-Kutta method. This process is repeated and can be done systematically until this choice satisfy $y'(b)$.

To do this, follow the steps below.

Step1:- select z_0 so that $y_z'(b)=\mu_2$, let $\psi(z)=y_z'(b)-\mu_2$. The guess for z_0

Step 2:- Now the objective is simply to solve for $\psi(z)=0$, hence secant method can be used.

Step 3:- How to compute z

Suppose that the solutions $y_{z_0}'(b)$ and $y_{z_1}'(b)$ obtained from guesses z_0 and z_1 respectively.

Step 4:- Now using secant method to find z_2 given by;

$$z_{k+1}=z_k-\frac{z_k-z_{k-1}}{y_{z_k}-y_{z_{k-1}}}y_{z_k},\ \text{k=1,2, ...}$$

Following this sequence of iteration there exists z such that

$$y_z'(b)=y'(b)$$

Thus, the Neumann boundary condition for the DE in (15)

is given by

$$y'(a) = z$$
$$y'(b) = \mu_2 \qquad (16)$$

Now to solve the DE with boundary condition in (16) it is convenient to use equation (13).

Example: - Consider the linear boundary value problem $\dfrac{d^2 y}{dx^2} + y = x^2 e^{-x}; \ 0 \leq x \leq 10$, subject to the boundary condition $y(0) = -0.2678$, $y'(10) = 0$

Whose exact solution is:-

$$y = -11/5000 \sin(x)(349\sin(10)e^{(10)}$$
$$-22500)/\cos(10)/e^{(10)}$$
$$-3839/5000\cos(x) + 1/2(x+1)^2 e^{(-x)}$$

Solution: -The above problem is a mixed boundary condition or (type II); to apply the above method it needs to convert the given boundary condition in to Neumann boundary condition. Now assume a guess depending on the value of $y'(10) = 0$, let $y'(0) = -1$ be the first guess and hoping that $y'(10) = 0$. The next step is using Ruge-Kutta method for second order differential equation, where $y''(x) = f(x, y, y')$. But for this problem, $y''(x) = f(x, y)$ since f is independent of y'

$x_0 = 0$, and $x_{end} = 10$ and take step size h=0.5, $y = y(x) = -0.2678 \ for \ x = 0 \Rightarrow y_0 = -0.2678$

$$y'(x) = -1 \ for \ x = 0 \Rightarrow y'_0 = -1$$

$$y_{j+1} = y_j + hy'_j + \frac{1}{2}(K_1 + K_2)$$

$$y'_{j+1} = y_j + hy'_j + \frac{1}{2h}(K_1 + 3K_2)$$

$$K_1 = \frac{h^2}{2}f(x_j, y_j)$$

$$K_2 = \frac{h^2}{2}f(x_j + \frac{2}{3}h, y_j + \frac{2}{3}hy'_j + \frac{4}{9}K_1) \ for \ j = 0, 1, 2 ... 20$$

This gives the result in table 1 for the first iteration, where in the ith step $x = x_i$, $y = y(x_i)$ and $y' = y'(x_i)$

Referring to table 1, take $y'_z = 0.5818$. But, $y'_z(10) \neq y'(10)$, thus it needs to guess another value for $y'(0)$. Let $y'(0) = 1$ hoping that $y'(10) = 0$. Using Runge-Kutta method for $x = 0$ to $x = 10$ and taking $h = 0.5$, $y = y(x) = -0.2678 \ for \ x = 0$ and $y' = y'(x) = 1 \ for \ x = 0$, this yields the following result, where in the ith step $x = x_i$, $y = y(x_i)$ and $y' = y'(x_i)$

Table 1. Shows the result of y and y' on the first iteration.

	x	y	y'
1st iteration	0.00	−0.2670	−1.0000
	0.50	−0.6741	−0.7212
	1.00	−0.8563	−0.1822
	1.50	−0.7336	0.4641
	2.00	−0.3198	1.0127
	2.50	0.2808	1.2968
	3.00	0.9088	1.2345
	3.50	1.3967	0.8435
	4.00	1.6149	0.2289
	4.50	1.5043	−0.4500
	5.00	1.0893	−1.0227
	5.50	0.4691	−1.3505
	6.00	−0.2095	−1.3596
	6.50	−0.7891	−1.0566
	7.00	−1.1407	−0.5233
	7.50	−1.1939	0.1056
	8.00	−0.9514	0.6776
	8.50	−0.4858	1.0594
	9.00	0.0803	1.1685
	9.50	0.6059	0.9909
	10.00	0.9659	0.5818

Table 2. Shows the result of y and y' on second iteration.

	x	y	y'
2nd iteration	0.00	-0.2678	1.0000
	0.50	0.2445	1.0344
	1.00	0.7231	0.9181
	1.50	1.1004	0.6723
	2.00	1.3183	0.3161
	2.50	1.3383	-0.1000
	3.00	1.1561	-0.4984
	3.50	0.8081	-0.7958
	4.00	0.3664	-0.9276
	4.50	-0.0772	-0.8663
	5.00	-0.4321	-0.6297
	5.50	-0.6297	-0.2761
	6.00	-0.6397	0.1101
	6.50	-0.4754	0.4395
	7.00	-0.1892	0.6393
	7.50	0.1407	0.6697
	8.00	0.4302	0.5327
	8.50	0.6095	0.2697
	9.00	0.6385	-0.0497
	9.50	0.5154	-0.3458
	10.00	0.2754	-0.5480

Now $y'_{z_1}(10) = -0.5480 \ where \ z_1 = 1$ and since z_0 and z_1 given then find z_2

$$z_2 = z_1 - \frac{z_1 - z_0}{y_{z_1} - y_{z_0}} y_{z_1} = 1 - \frac{1 - (-1)}{-0.5480 - 0.5818}(-0.5480) = 0.0299$$

Using $z_2 = 0.0299$ applying Runge-Kutta method where h=0.5, $y = y(x) = -0.2678 \ for \ x = 0$ and $y' = y'(x) = 0.0299 \ for \ x = 0$

gives the following result for the 3rd iteration.

Table 3. *Shows the result of y and y' on the third iteration.*

	x	y	y'
	0.00	-0.2678	0.0299
	0.50	-0.2014	0.1830
	1.00	-0.0432	0.3847
	1.50	0.2108	0.5717
	2.00	0.5239	0.6543
	2.50	0.8256	0.5778
	3.00	1.0365	0.3422
	3.50	1.0939	-0.0008
	4.00	0.9722	-0.3669
3rd iteration	4.50	0.6899	-0.6647
	5.00	0.3058	-0.8206
	5.50	-0.0970	-0.7975
	6.00	-0.4313	-0.6029
	6.50	-0.6278	-0.2861
	7.00	-0.6510	0.0755
	7.50	-0.5068	0.3963
	8.00	-0.2399	0.6032
	8.50	0.0784	0.6530
	9.00	0.3680	0.5413
	9.50	0.5596	0.3025
	10.00	0.6105	-0.0002

Now calculate the next guess

$$z_3 = z_2 - \frac{z_2 - z_1}{y_{z_2} - y_{z_1}} y_{z_2} = 0.0299 - \frac{0.0299 - 1}{-0.0002 + 0.5480}(0.0299) = 0.031$$

Runge-Kutta method for $y = 0.031$ yields the following result for the 4th iteration.

Table 4. *Shows the result of y and y' on the fourth iteration.*

	x	y	y'
	0.00	-0.2670	0.0301
	0.50	-0.2006	0.1828
	1.00	-0.0427	0.3842
	1.50	0.2110	0.5710
	2.00	0.5237	0.6536
	2.50	0.8252	0.5772
	3.00	1.0358	0.3419
	3.50	1.0933	-0.0007
	4.0	0.9716	-0.3665
4th iteration	4.50	0.6897	-0.6642
	5.00	0.3058	-0.8200
	5.50	-0.0967	-0.7969
	6.00	-0.4308	-0.6026
	6.50	-0.6272	-0.2861
	7.00	-0.6504	0.0753
	7.50	-0.5064	0.3959
	8.00	-0.2398	0.6027
	8.50	0.0782	0.6524
	9.00	0.3676	0.5409
	9.50	0.5591	0.3024
	10	0.6101	0.0000

As table 4 shows the guess for $y'(0) \approx 0.030$. Thus, the

Neumann boundary value problem given by

$$\frac{d^2 y}{dx^2} + y = x^2 e^{-x}; \quad 0 \le x \le 10$$

$y'(0) = 0.030$, and $y'(10) = 0$

The next step is converting the BVP into equivalent BVP defined for $-1 \le x \le 1$ by letting

$$x = \frac{(b-a)}{2}x + \frac{(b+a)}{2} = 5x + 5, \text{ since } a = 0 \text{ and } b = 10.$$

The equivalent BVP for the above problem on $-1 \le x \le 1$ becomes,

$$\frac{1}{25}\frac{d^2 y}{dx^2} + y = (5x+5)^2 e^{-(5x+5)}, \text{ for } -1 \le x \le 1$$

with boundary condition (17)

$$\begin{aligned} y'(-1) &= 0.030 \\ y'(1) &= 0 \end{aligned} \tag{17}$$

Now, suppose that $\bar{y}$ be the approximate solution of (17), given by a linear combination of constants c_i's and an approximating polynomial, called Chebyshev polynomial, thus

$$\bar{y} = \sum_{i=1}^{n} c_i T_i(x) \tag{18}$$

Upon substitution of $\bar{y}$, the approximate solution, into the differential equation in (17) gives an equation called residue given by:

$$R(x, c_i) = \frac{1}{25}\frac{d^2 \bar{y}}{dx^2} + \bar{y} - (5x+5)^2 e^{-(5x+5)} \approx 0, \text{ for } -1 \le x \le 1 \tag{19}$$

Applying Galerkin method;

$$\int_{-1}^{1} R(c_i, x) T_j(x) dx = 0$$

$$\Rightarrow \int_{-1}^{1}[\frac{1}{25}\frac{d^2 \bar{y}}{dx^2} + \bar{y} - (5x+5)^2 e^{-(5x+5)}] T_j(x) dx \approx 0$$

$$\Rightarrow \int_{-1}^{1}[\frac{1}{25}\bar{y}'' T_j(x) + \bar{y} T_j(x)] dx = \int_{-1}^{1}(5x+5)^2 e^{-(5x+5)} T_j(x) dx \tag{20}$$

Using integration by parts to simplify the first term in the right hand side, let $u = T_j(x) \Rightarrow du = T_j'(x)dx$ and $dv = \bar{y}''(x)dx \Rightarrow v = \bar{y}'(x)$, hence it gives

$$\int_{-1}^{1}\frac{1}{25}\bar{y}'' T_j(x) dx = \frac{1}{25}\left(\bar{y}'(x) T_j(x) \mid_{-1}^{1} - \int_{-1}^{1}\bar{y}' T_j'(x) dx\right)$$

Therefore equation (20) becomes

$$\int_{-1}^{1} \overline{y}T_j(x)dx - \frac{1}{25}\int_{-1}^{1} \overline{y}\,'T_j\,'(x)dx = -\frac{1}{25}\overline{y}\,'(x)T_j(x)\,|_{-1}^{1} + \int_{-1}^{1}(5x+5)^2\,e^{-(5x+5)}T_j(x)dx \tag{21}$$

Substituting the approximate solution $\overline{y} = \sum_{i=1}^{n} c_i T_i(x)$ into (21) yields,

$$\sum_{i=1}^{n} c_i \left(\int_{-1}^{1} T_i(x)T_j(x)dx - \frac{1}{25}\int_{-1}^{1} T_i\,'(x)T_j\,'(x)dx \right) = -\frac{1}{25}\overline{y}\,'(x)T_j(x)\,|_{-1}^{1} + \int_{-1}^{1}(5x+5)^2\,e^{-(5x+5)}T_j(x)dx$$

$$\sum_{i=1}^{n} c_i \left(\int_{-1}^{1} T_i(x)T_j(x)dx - \frac{1}{25}\int_{-1}^{1} T_i'(x)T_j'(x)dx \right) = \frac{1}{25}\overline{y}'(-1)T_j(-1) - \frac{1}{25}\overline{y}'(1)T_j(1)$$

$$+ \int_{-1}^{1}(5x+5)^2\,e^{-(5x+5)}T_j(x)dx \tag{22}$$

Now, using the given boundary condition in to (22), equation (22) becomes

$$\sum_{i=1}^{n} c_i \left(\int_{-1}^{1} T_i(x)T_j(x)dx - \frac{1}{25}\int_{-1}^{1} T_i'(x)T_j'(x)dx \right) = \int_{-1}^{1}(5x+5)^2\,e^{-(5x+5)}T_j(x)dx + \frac{1}{25}(0.030)T_j(-1) \tag{23}$$

For equation (23) there is a system of equation given by:

$$\sum_{i=1}^{n} c_i K_{ij} = F_i \tag{24}$$

where $K_{ij} = k_{ij}^{(1)} + k_{ij}^{(2)}$ and $F_i = f_i^{(1)} + f_i^{(2)}$

such that $k_{ij}^{(1)} = \int_{-1}^{1} T_i(x)T_j(x)dx$

$$k_{ij}^{(2)} = -\frac{1}{25}\int_{-1}^{1} T_i'(x)T_j'(x)dx$$

$$f_i^{(1)} = \frac{1}{25}(0.030)T_j(-1)$$

$$f_i^{(2)} = \int_{-1}^{1}(5x+5)^2\,e^{-(5x+5)}T_j(x)dx$$

In order to find the value of c_i's take n trial functions defined for $x \in [-1,1]$, using Chebyshev polynomials as trial function. For $n = 6$:

$$T' = \begin{bmatrix} x \\ 2x^2 - 1 \\ 4x^3 - 3x \\ 8x^4 - 8x^2 + 1 \\ 16x^5 - 20x^3 + 5x \\ 32x^6 - 48x^4 + 18x^2 - 1 \end{bmatrix}, \text{ where } T = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6] \text{ and } T' \text{ is the transpose of T.}$$

$$k_{ij}^{(1)} = \int_{-1}^{1} T_i(x)T_j(x)dx = \begin{bmatrix} 2/3 & 0 & -2/5 & 0 & -2/21 & 0 \\ 0 & 14/15 & 0 & -38/105 & 0 & -26/315 \\ -2/5 & 0 & 34/35 & 0 & -22/63 & 0 \\ 0 & -38/105 & 0 & 62/63 & 0 & -34/99 \\ -2/21 & 0 & -22/63 & 0 & 98/99 & 0 \\ 0 & -26/315 & 0 & -34/99 & 0 & 142/143 \end{bmatrix}$$

$$k_{ij}^{(2)} = -\frac{1}{25}\int_{-1}^{1} T_i'(x)T_j'(x)dx = \begin{bmatrix} 2/25 & 0 & 2/25 & 0 & 2/25 & 0 \\ 0 & 32/75 & 0 & 128/375 & 0 & 288/875 \\ 2/25 & 0 & 138/125 & 0 & 142/175 & 0 \\ 0 & 128/375 & 0 & 5632/2625 & 0 & 3968/2625 \\ 2/25 & 0 & 142/175 & 0 & 1126/315 & 0 \\ 0 & 288/875 & 0 & 3968/2625 & 0 & 52064/9625 \end{bmatrix}$$

To find the value of the coefficient matrix in equation (24) use

$$K_{ij} = k_{ij}^{(1)} + k_{ij}^{(2)}$$

So, by expressing the coefficient matrix $K_{i,j}$ and the unknown coefficient c_i's in a system of equation in matrix form:

$$K_{ij} = k_{ij}^{(1)} + k_{ij}^{(2)} \begin{bmatrix} 44/75 & 0 & -12/25 & 0 & -92/525 & 0 \\ 0 & 38/75 & 0 & -1846/2625 & 0 & -3242/7875 \\ -12/25 & 0 & -116/875 & 0 & -1828/1575 & 0 \\ 0 & -1846/2625 & 0 & -9146/7875 & 0 & -160694/86625 \\ -92/525 & 0 & -1828/1575 & 0 & -8956/3465 & 0 \\ 0 & -3242/7875 & 0 & -160694/86625 & 0 & -552582/125125 \end{bmatrix}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = [f_i] \quad (25)$$

Where f_i is 6×1 a column vector given by:-

$$f_i = \begin{bmatrix} -4/25 - 756/25e^{(-10)} \\ -22/125 - 6658/125e^{(-10)} \\ 28/125 - 14708/125e^{(-10)} \\ -38/625 - 189682/625e^{(-10)} \\ 28/3125 - 2774708/3125e^{(-10)} \\ 2818/15625 - 45459898/15625e^{(-10)} \end{bmatrix}$$

Now a 6×6 coefficient matrix which is symmetric, 6×1 unknown column vector that represent c_i's and 6×1 column vector that represents f_i. So there are six equations with six unknowns. Using $c_i = (K_{i,j})^{-1} f_i$ to solve (25), the values of the six unknowns are:

$$c_1 = 0.361550$$

$$c_2 = -0.002596$$

$$c_3 = 0.939452$$

$$c_4 = 0.360917$$

$$c_5 = -0.449830$$

$$c_6 = -0.192200$$

Now it is possible to express the approximate solution as a linear combination of constants c_i's and an approximating polynomial. So substituting c_i's and Chebyshev polynomials for n=6 the approximate solution is:

$$\bar{y} = 0.361550x - 0.002596(2x^2 - 1) + 0.939452(4x^3 - 3x) + 0.360917(8x^4 - 8x^2 + 1)$$
$$- 0.449830(16x^5 - 20x^3 + 5x) - 0.192200(32x^6 - 48x^4 + 18x^2 - 1)$$

The graph of the exact and approximate solution, to look the convergence of the approximate solution to the graph of the exact solution, looks like figure 2 below.
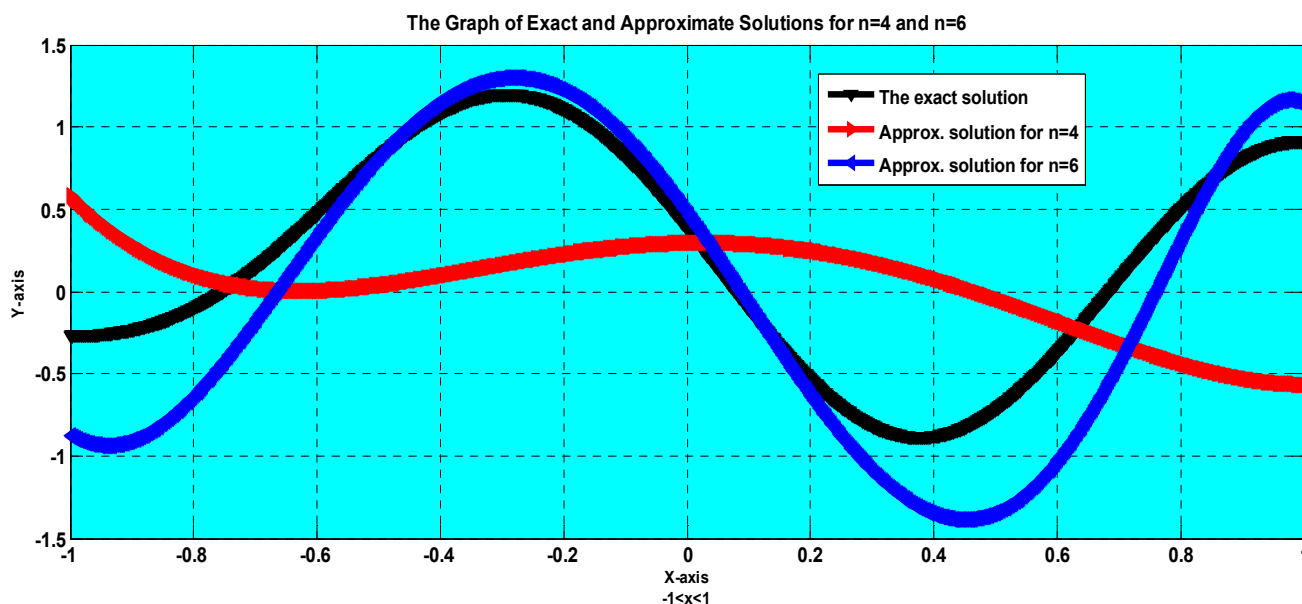
The Graph of Exact and Approximate Solutions for n=4 and n=6

Figure 2. The graph of the exact and approximate solution for n=4 and n=6.

The above graph shows that the graph of the approximate solution approaches the graph of the exact solution for the differential equation with boundary condition in problem 1.

Considering n=8, then the approximate solution and the Chebyshev polynomials are given by:

$$\overline{y} = \sum_{i=1}^{8} c_i T_i(x) \qquad (26)$$

$$T' = \begin{bmatrix} x \\ 2x^2 - 1 \\ 4x^3 - 3x \\ 8x^4 - 8x^2 + 1 \\ 16x^5 - 20x^3 + 5x \\ 32x^6 - 48x^4 + 18x^2 - 1 \\ 64x^7 - 112x^5 + 56x^3 - 7x \\ 1 - 32x^2 + 160x^4 - 256x^6 + 128x^8 \end{bmatrix}$$, where $T = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6 \ T_7 \ T_8]$ and T' is the transpose of T.

Using MATLAB code;

$$K^{(1)}{}_{ij} = \begin{bmatrix} 2/3 & 0 & -2/5 & 0 & -2/21 & 0 & -2/45 & 0 \\ 0 & 14/15 & 0 & -38/105 & 0 & -26/315 & 0 & -134/3465 \\ -2/5 & 0 & 34/35 & 0 & -22/63 & 0 & -38/495 & 0] \\ 0 & -38/105 & 0 & 62/63 & 0 & -34/99 & 0 & -158/2145 \\ -2/21 & 0 & -22/63 & 0 & 98/99 & 0 & -146/429 & 0 \\ 0 & -26/315 & 0 & -34/99 & 0 & 142/143 & 0 & -22/65 \\ -2/45 & 0 & -38/495 & 0 & -146/429 & 0 & 194/195 & 0 \\ 0 & -134/3465 & 0 & -158/2145 & 0 & -22/65 & 0 & 254/255 \end{bmatrix}$$

$$K_{ij}^{(2)} = \begin{bmatrix} 2/25 & 0 & 2/25 & 0 & 2/25 & 0 & 2/25 & 0 \\ 0 & 32/75 & 0 & 128/375 & 0 & 288/875 & 0 & 512/1575 \\ 2/25 & 0 & 138/125 & 0 & 142/175 & 0 & 286/375 & 0 \\ 0 & 128/375 & 0 & 5632/2625 & 0 & 3968/2625 & 0 & 120832/86625 \\ 2/25 & 0 & 142/175 & 0 & 1126/315 & 0 & 6086/2475 & 0 \\ 0 & 288/875 & 0 & 3968/2625 & 0 & 52064/9625 & 0 & 1376768/375375 \\ 2/25 & 0 & 286/375 & 0 & 6086/2475 & 0 & 1232966/160875 & 0 \\ 0 & 512/1575 & 0 & 120832/86625 & 0 & 1376768/375375 & 0 & 11657216/1126125 \end{bmatrix}$$

$$f_i = \begin{bmatrix} -4/25 - 756/25e^{(-10)} \\ -22/125 - 6658/125e^{(-10)} \\ 28/125 - 14708/125e^{(-10)} \\ -38/625 - 189682/625e^{(-10)} \\ 28/3125 - 2774708/3125e^{(-10)} \\ 2818/15625 - 45459898/15625e^{(-10)} \\ 35404/78125 - 827348644/78125e^{(-10)} \\ 151442/78125 - 3321109962/78125e^{(-10)} \end{bmatrix}$$

Thus the unknown parameters c_i's are

$$c_1 = 0.375249$$

$$c_2 = -0.069868$$

$$c_3 = 0.959056$$

$$c_4 = 0.286293$$

$$c_5 = -0.454061$$

$$c_6 = 0.117209$$

$$c_7 = -0.005079$$

$$c_8 = -0.299613$$

Now substituting the unknown parameters and eight Chebyshev polynomials in to (26), the approximate solution is:

$$\bar{y} = 0.375249x - 0.069868(2x^2 - 1) + 0.959056(4x^3 - 3x) + 0.286293(8x^4 - 8x^2 + 1)$$
$$-0.454061(16x^5 - 20x^3 + 5x) + 0.117209(32x^6 - 48x^4 + 18x^2 - 1)$$
$$-0.005079(64x^7 - 112x^5 + 56x^3 - 7x) - 0.299613(1 - 32x^2 + 160x^4 - 256x^6 + 128x^8)$$

The graph of the exact and approximate solution, to look the convergence of the approximate solution to the graph of the exact solution, looks like figure 3 below.
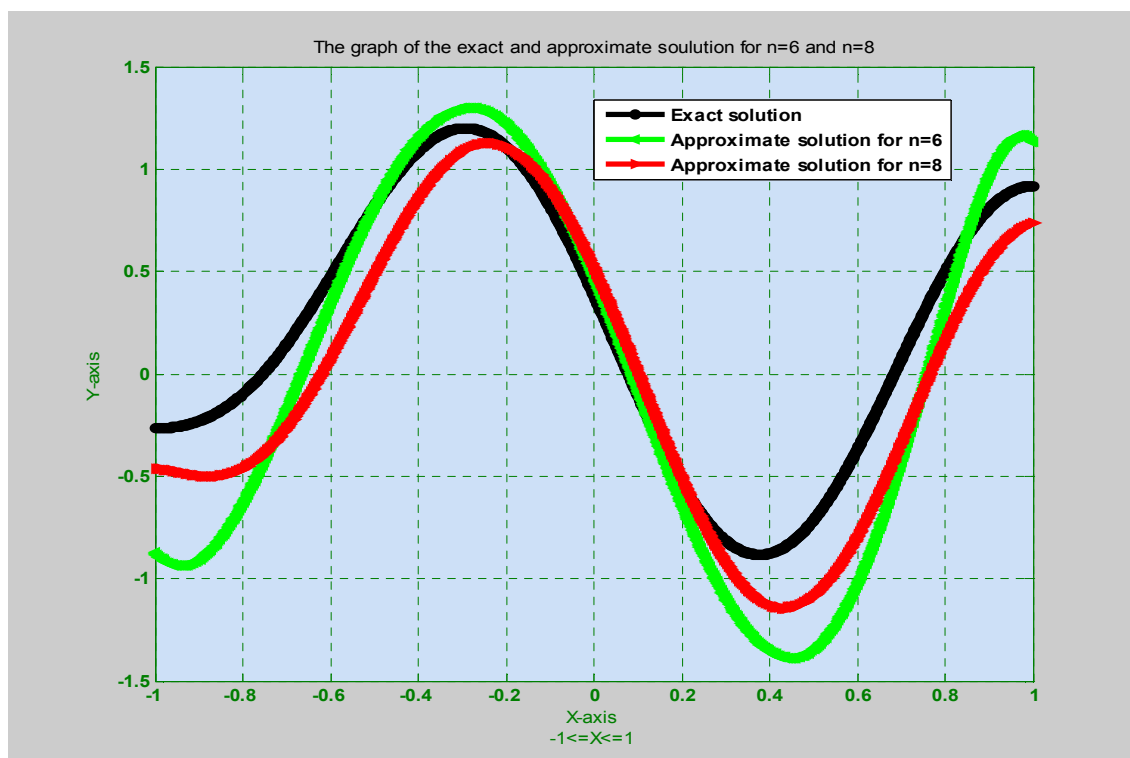
Figure 3. The graph of the exact solution and approximate solution for n=6 and n=8.

From the graph above the approximate solution approaches the graph of the exact solution when the number of the trial functions increases from 6 to 8.

Now compare the absolute error which is given by

$$error = \left| y_{exact} - y_{approx} \right|$$

Table 5. Shows the computed the absolute error for problem 1 when n=6 and n=8.

x	Exact solution	Approximate solution for n=6	Approximate solution for n=8	Absolute error for n=6	Absolute error for n=8
-1.0	-0.2670	-0.8745	-0.4640	0.6074	0.1969
-0.9	-0.2305	-0.9073	-0.4970	0.6768	0.2664
-0.8	-0.0994	-0.6369	-0.4619	0.5375	0.3625
-0.7	0.1443	-0.1813	-0.2646	0.3256	0.2089
-0.6	0.4736	0.3385	0.0794	0.1350	0.0942
-0.5	0.8180	0.8124	0.4890	0.3057	0.0291
-0.4	1.0871	1.1515	0.8573	0.0644	0.0298
-0.3	1.1994	1.2961	1.0859	0.0967	0.0136
-0.2	1.1087	1.2205	1.1089	0.1118	0.0001
-0.1	0.8185	0.9337	0.9077	0.1153	0.0892
0.0	0.3832	0.4786	0.5149	0.0954	0.0317
0.1	-0.1045	-0.0735	0.0075	0.0310	0.0120
0.2	-0.5361	-0.6303	-0.5083	0.0942	0.0277
0.3	-0.8144	-1.0896	-0.9190	0.2752	0.1046
0.4	-0.8776	-1.3536	-1.1289	0.4760	0.2513
0.5	-0.7149	-1.3457	-1.0847	0.6308	0.3698
0.6	-0.3695	-1.0312	-0.7942	0.6617	0.4247
0.7	0.0717	-0.4403	-0.3329	0.5120	0.4046
0.8	0.4990	0.3050	0.1671	0.4194	0.3319
0.9	0.8066	0.9616	0.5557	0.3155	0.2509
1.0	0.9184	1.1323	0.7374	0.2139	0.1810

As observed from table 5 the approximate solution is approaching to the exact value as the value of n increases. Take n=10, the graph of the corresponding approximate solution together with the graph of the approximate solution for n=6 and n=8 in the same plane with the exact solution is shown below.
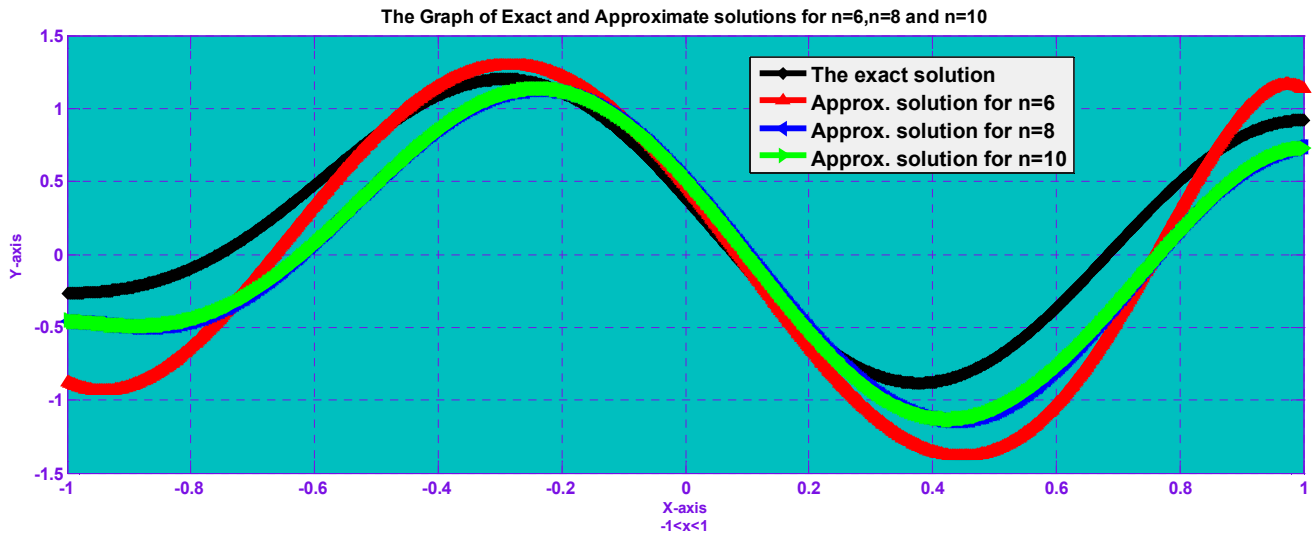
Figure 4. The graph of the exact solution and approximate solution for n=6, n=8 and n=10.

As the number of Chebyshev polynomial increases the corresponding approximate solution of the differential equation with mixed boundary condition in problem 1 approaches the graph of the exact solution.

7. Conclusion

This study introduces that, by applying Galerkin method to linear second order ordinary differential equations with mixed and Neumann boundary conditions, it is possible to find their approximate solutions. The numerical results that are obtained using this method converges to the exact solution as the number of Chebyshev polynomial increases, that will be used as a trial function; and also using small step size h, while converting the given linear second order ordinary differential equation from mixed type to Neumann boundary condition, increases the accuracy of the approximate solution. So that using this method better results will be obtained as the number of Chebyshev polynomial increases and using small step size while using Rung-Kutta method.

8. Future Scope

This study has led to an attentiveness of several topics that require further investigation, for instance linear ordinary differential equations with different order, non-linear ordinary differential equations. In order to fill the gap in terms of accuracy it is important to analyze the error of the method. Thus analyzing the error and increasing the accuracy of this method is left for future investigation.

References

[1]   Yattender Rishi Dubey: An approximate solution to buckling of plates by the Galerkin method, (August 2005).

[2]   Tai-Ran Hsu: Mechanical Engineering 130 Applied Engineering analysis, San Jose State University, (Sept 2009).

[3]   E. Suli: Numerical Solution of Ordinary Differential Equations, (April 2013).

[4]   J. N. Reddy: An Introduction to the finite element method, 3rd edition, McGraw-Hill, (Jan 2011) 58-98.

[5]   Marcos Cesar Ruggeri: Theory of Galerkin method and explanation of MATLAB code, (2006).

[6]   Jalil Rashidinia and Reza Jalilian: Spline solution of two point boundary value problems, Appl. Comput. Math 9 (2010) 258-266.

[7]   S. Das, Sunil Kumar and O. P. Singh: Solutions of nonlinear second order multipoint boundary value problems by Homotopy perturbation method, Appl. Appl. Math. 05 (2010) 1592-1600.

[8]   M. Idress Bhatti and P. Bracken: Solutions of differential equations in a Bernstein polynomials basis, J. Comput. Appl. Math. 205 (2007) 272-280.

[9]   M. M. Rahman. et.al: Numerical Solutions of Second Order Boundary Value Problems by Galerkin Method with Hermite Polynomials, (2012).

[10] Arshad Khan: Parametric cubic spline solution of two point boundary value problems, Appl. Math. Comput. 154 (2004) 175-182.

[11] Yuqiang Feng and Guangjun Li: Exact three positive solutions to a second-order Neumann boundary value problem with singular nonlinearity, Arabian J. Sci. Eng. 35 (2010) 189-195.

[12] P. M. Lima and M. Carpentier: Numerical solution of a singular boundary-value problem in non-Newtonian fluid mechanics, Computer Phys. Communica. 126(2000) 114-120.

[13] K. N. S. Kasi Viswanadham and Sreenivasulu Ballem: Fourth Order Boundary Value Problems by Galerkin Method with Cubic B-splines, (May 2013).

[14] Jahanshahi et al.: A special successive approximations method for solving boundary value problems including ordinary differential equations, (August 2013).

[15] L. Fox and I. B. Parker: Chebyshev Polynomials in Numerical Analysis, Oxford University Press, (1 May, 1967).