Spline Collocation for Solving System of Fredholm and Volterra Integral Equations

N. Ebrahimi, J. Rashidinia

Abstract—In this paper, numerical solution of system of Fredholm and Volterra integral equations by means of the Spline collocation method is considered. This approximation reduces the system of integral equations to an explicit system of algebraic equations. The solution is collocated by cubic B-spline and the integrand is approximated by the Newton-Cotes formula. The error analysis of proposed numerical method is studied theoretically. The results are compared with the results obtained by other methods to illustrate the accuracy and the implementation of our method.

Keywords—Convergence analysis, Cubic B-spline, Newton-Cotes formula, System of Fredholm and Volterra integral equations.

I. INTRODUCTION

Consider the system of integral equations of the form:

\[ F(t) = G(t) + \int_a^b K(t,x,F(x)) \, dx, \quad t \in [a,b] \]  

(1)

where

\[ F(t) = [f_1(t), f_2(t), \ldots, f_n(t)]^T, \]

\[ G(t) = [g_1(t), g_2(t), \ldots, g_n(t)]^T, \]

\[ K(t,x,F(x)) = \sum_{i=1}^{n} k_i(t,x,f_i(x)), i = 1, \ldots, n. \]

In (1) the known kernel \( K \) is continuous, the function \( G(x) \) is given, existence and uniqueness of the solution of (1) is given by [22] and \( F(x) \) is the solution to be determined [1], [2]. We consider the \( i \)-th equation of (1):

\[ f_i(t) = g_i(t) + \sum_{j=1}^{n} k_j(t,x,f_j(x)) \, dx, \quad t \in [a,b], i = 1, \ldots, n \]

(2)

There has been considerable interest in solving integral equation (1). E. Babolian et al. [13] applied an Adomian decomposition method for solving system of linear Fredholm integral equations of the second kind. Numerical solution of the system of linear Fredholm integral equations (1), has been proposed by Maleknejad et al. in [14], [15]. Rationalized Haar functions have been used for direct numerical solutions in [14] and also Block-Pulse functions to propose solutions for the system of Fredholm integral equations have been developed in [15]. This type of equations have been solved in many papers with different methods such as Taylor’s expansion [16], [17], operational matrices method [18] homotopy perturbation method [19]-[22], Sinc collocation method [23], [24] and Adomian’s decomposition method [25]-[28]. Using a global approximation to the solution of Fredholm and Volterra integral equation of the second kind is constructed by means of the cubic spline quadrature in [9]-[12].

In this paper, we use cubic B-spline collocation for approximation unknown function and use of the Newton-Cotes rules for approximating integrand.

II. CUBIC B-SPLINE

We introduce the cubic B-spline space and basis functions to construct an interpolant \( S_N = [s_1, s_2, \ldots, s_n]^T \), to be used in the formulation of the cubic B-spline collocation method.

Let \( \pi(a = t_0 < t_1 < \cdots < t_N = b) \), be a uniform partition of the interval \([a,b]\) with step size \( h = \frac{b-a}{N} \). The cubic spline space is denoted by

\[ S_3(a) = \{ s \in C^2[a,b] ; s|_{[t_k, t_{k+1}]} \in P_3, i = 1, \ldots, n, k = 0, \ldots, N \}, \]

where \( P_3 \) is the class of cubic polynomials. The construction of the cubic B-spline interpolate \( S_N \) to the analytical solution \( F(t) = [f_1(t), f_2(t), \ldots, f_n(t)]^T \), for (1) can be performed with the help of the four additional knots such that

\[ t_{-2} < t_{-1} < t_0 < t_{N+1} < t_{N+2}. \]

Following [9] we can define a cubic B-spline \( s_i(t) \) of the form

\[ s_i(t) = \sum_{k=-1}^{N+1} c_{ik} B_k^3(t) \]

(3)

where

\[ B_k^3(t) = \frac{1}{6h^3} \begin{cases} 
(t-t_{k-1})^3, & t \in [t_{k-2}, t_{k-1}] \\
0, & t \notin [t_{k-2}, t_{k-1}] \end{cases} \]

(4)

satisfying the following interpolator conditions:

\[ s_i(t_k) = f_i(t_k), \quad 1 \leq i \leq n, 0 \leq k \leq N, \]

and the end conditions

\[ s_i'(t_0) = f_i(t_0), \quad s_i'(t_N) = f_i'(t_N), \quad i = 1, \ldots, n, \] or

\[ \text{and} \]

\[ s_i''(t_0) = D s_i(t_0), \quad s_i''(t_N) = D s_i'(t_N), \quad j = 1, 2, \] or

\[ (iii) s_i'(t_0) = 0, \quad s_i'(t_N) = 0. \]
III. THE COLLOCATION METHOD

A. Nonlinear Fredholm Integral Equation

In the given nonlinear Fredholm integral equation (2), we can approximate the unknown function by cubic B-spline (3),

\[ s(t) = g(t) + \sum_{i=1}^{n} \int_a^b k_i(t,x,s(x)) \, dx, \quad t \in [a, b], \quad i = 1, \ldots, n. \tag{5} \]

We now collocate (5) at collocation points \( t_k = a + kh, \quad k = 0,1, \ldots, N \), and we obtain

\[ s(t_k) = g(t_k) + \sum_{i=1}^{n} \int_a^b k_i(t_k,x,s(x)) \, dx, \quad i = 1, \ldots, n, \quad k = 0,1, \ldots, N. \tag{6} \]

To approximate the integral equation (6), we can use the Newton-Cotes formula [2], when \( n \) is even then the Simpson rule can be used and when \( n \) is multiple of 3, we have to use the three-eighth rule, then we get the following nonlinear system:

\[ s(t_k) = g(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 0,1, \ldots, N. \tag{7} \]

where \( w_{kr} \) represents the weights for a quadrature rule of Newton-Cotes type. By solving the above nonlinear system for each \( i \), we can determine the coefficients in (3), by setting coefficients in (3), we obtain the approximate solution for (1).

To approximate the nonlinear system (11), we need more equations to obtain the unique solution for (11). Hence by associating (11) with (4) for the \( i \)-th equation, we have the following nonlinear system

\[ \begin{align*}
  s(t_k) &= g(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 1, \ldots, N, \\
  D_s(t_i) &= D_s(t_N), \quad j = 1,2, \\
  s(t_0) &= g(t_0). \tag{12}
\end{align*} \]

B. Nonlinear Volterra Integral Equation

Now we consider the system of nonlinear Volterra integral equations

\[ F(t) = G(t) + \int_a^t K(t,x,F(x)) \, dx, \quad t \in [a, b] \tag{9} \]

the solutions of (9) can be replaced with cubic B-spline and so we collocate (9) at collocation points \( t_k = a + kh, \quad k = 0,1, \ldots, N \), and we obtain

\[ s(t_k) = g(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 0,1, \ldots, N. \tag{10} \]

where \( w_{kr} \) represents the weights for a quadrature rule of Newton-Cotes type. By solving the above nonlinear system for each \( i \), we can determine the coefficients in (3), by setting coefficients in (3), we obtain the approximate solution for (1).

To approximate the integral equation (10), we can use the Newton-Cotes formula, when \( n \) is even then the Simpson rule can be used and when \( n \) is multiple of 3, we have to use the three-eighth rule, then we get the following nonlinear system:

\[ \begin{align*}
  s(t_k) &= g(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 1, \ldots, N, \\
  s(t_0) &= g(t_0). \tag{11}
\end{align*} \]

IV. ANALYSIS: CONVERGENCE OF THE APPROXIMATE SOLUTION

In this section, we consider the error analysis of the system of nonlinear Volterra integral equations of the second kind.

We assume that \( S_N = [s_1, s_2, \ldots, s_N]^T \), is approximated solution for \( F(t) = [f_1(t), f_2(t), \ldots, f_n(t)]^T \). We need to recall the following definition in [9].

Definition: The most immediate error analysis for spline approximates \( s \) to a given function \( f \) defined on an interval \([a,b]\) follows from the second integral relations.

\[ \| f - S \| \leq \gamma h^{j+1}, \quad j = 0,1,2,3,4, \]

where \( \gamma = \max_{t \in [a,b]} |f(t)| \), and \( f^{(j)} \) the \( j \)-th derivative.

Theorem: The approximate method

\[ s(t_k) = g(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 1, \ldots, N. \tag{13} \]

for solution of the system of nonlinear Volterra integral equation (9) is converge and the error bounded is

\[ |E_{SN}(h,t_k)| \leq h W L \left[ \frac{1}{1 - h W L} + \frac{1}{1 - h W L} \right] |E(f,h,t_k)|. \]

Proof: We know that at \( t_k = a + kh, h = \frac{t-a}{N} \), \( k = 0,1, \ldots, N \), the corresponding approximation method for the system of nonlinear Volterra integral equation (9) is

\[ S_N(t_k) = G(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,S(x_r)), \quad k = 1, \ldots, N. \tag{14} \]

By discretizing (9) and approximating the integrand by the Newton-Cotes formula, we obtain

\[ F(t_k) = G(t_k) + h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,F(x_r)) + E(h,t_k), \quad k = 1, \ldots, N. \tag{15} \]

where

\[ E(h,t_k) = \int_a^t K(t,x,F(x)) \, dx - h \sum_{r=0}^{N} w_{kr} K(t_k,x_r,F(x_r)). \]

By subtracting (15) from (14) and using interpolatory conditions of cubic B-spline, we get
Fredholm integral equation with exact solution, \[ f(t) = (t - 1, t^2)^T, \]
where \( f(t) = t - t^2 + \int_0^1 \left( t x f_1(x) + x^2 f_2(x) \right) dx, \]
and
\[ f_1(t) = \frac{t}{2} - \frac{t^2}{3} + \int_0^1 \left( t x^2 f_1(x) + x^2 f_2(x) \right) dx. \]

The absolute errors at the particular grid points are tabulated in Table IV and compared with the absolute errors obtained by with [3], [5], [19]. This table verified that our results are more accurate in comparison.

Example 4. Consider the system of following nonlinear Volterra integral equation with exact solution
\[ (f_1(t), f_2(t))^T = (t; t^2)^T, \]
\[ f_1(t) = t - t^2 + \int_0^1 (f_1(x) + f_2(x)) dx, \]
\[ f_2(t) = t - \frac{t^2}{2} - \frac{t^3}{3} + \int_0^1 (t x f_1(x) + f_2(x)) dx. \]

The absolute errors at the particular grid points are tabulated in Table IV and compared with the absolute errors obtained by with [3], [5], [19]. This table verified that our results are more accurate in comparison.

### TABLE I

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.71E – 05</td>
<td>2.11E – 07</td>
<td>1.32E – 08</td>
<td>1.13E – 03</td>
<td>1.2E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>1.78E – 05</td>
<td>2.19E – 07</td>
<td>1.37E – 08</td>
<td>1.15E – 03</td>
<td>1.15E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>1.83E – 05</td>
<td>2.26E – 07</td>
<td>1.41E – 08</td>
<td>1.19E – 03</td>
<td>1.19E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>1.87E – 05</td>
<td>2.31E – 07</td>
<td>1.44E – 08</td>
<td>1.29E – 03</td>
<td>1.28E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>1.87E – 05</td>
<td>2.32E – 07</td>
<td>1.45E – 08</td>
<td>1.44E – 03</td>
<td>1.44E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1.84E – 05</td>
<td>2.29E – 07</td>
<td>1.43E – 08</td>
<td>1.67E – 03</td>
<td>1.69E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>1.77E – 05</td>
<td>2.19E – 07</td>
<td>1.37E – 08</td>
<td>1.72E – 03</td>
<td>1.70E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>1.63E – 05</td>
<td>2.02E – 07</td>
<td>1.26E – 08</td>
<td>1.77E – 03</td>
<td>1.77E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>1.41E – 05</td>
<td>1.75E – 07</td>
<td>1.09E – 08</td>
<td>1.93E – 03</td>
<td>1.93E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>1.09E – 05</td>
<td>1.35E – 07</td>
<td>8.46E – 09</td>
<td>2.19E – 03</td>
<td>2.20E – 03</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6.51E – 06</td>
<td>8.01E – 08</td>
<td>5.01E – 09</td>
<td>2.61E – 02</td>
<td>4.21E – 02</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This system has been solved by our method with \( N = 10, 20, 30, 60, \) the absolute errors at the particular grid points are tabulated in Table I, and compared with the absolute errors obtained by with [7]. This table verified that our results are more accurate in comparison with [7].
### Table II

<table>
<thead>
<tr>
<th>$t$</th>
<th>Our method $N=10$</th>
<th>Our method $N=20$</th>
<th>Our method $N=30$</th>
<th>Method in [7]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.25E – 06</td>
<td>7.78E – 08</td>
<td>1.53E – 08</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>1.03E – 06</td>
<td>6.39E – 08</td>
<td>1.26E – 08</td>
<td>1.37E – 04</td>
</tr>
<tr>
<td>0.2</td>
<td>7.78E – 07</td>
<td>4.83E – 08</td>
<td>9.53E – 09</td>
<td>9.27E – 04</td>
</tr>
<tr>
<td>0.3</td>
<td>5.01E – 07</td>
<td>3.10E – 08</td>
<td>6.12E – 09</td>
<td>2.67E – 03</td>
</tr>
<tr>
<td>0.4</td>
<td>1.99E – 07</td>
<td>1.22E – 08</td>
<td>2.41E – 09</td>
<td>5.45E – 03</td>
</tr>
<tr>
<td>0.5</td>
<td>1.22E – 07</td>
<td>7.80E – 09</td>
<td>1.54E – 09</td>
<td>9.23E – 03</td>
</tr>
<tr>
<td>0.6</td>
<td>6.41E – 07</td>
<td>2.89E – 09</td>
<td>5.72E – 09</td>
<td>1.38E – 02</td>
</tr>
<tr>
<td>0.7</td>
<td>8.15E – 07</td>
<td>5.10E – 08</td>
<td>1.01E – 08</td>
<td>1.93E – 02</td>
</tr>
<tr>
<td>0.8</td>
<td>1.17E – 06</td>
<td>7.37E – 08</td>
<td>1.45E – 08</td>
<td>2.56E – 02</td>
</tr>
<tr>
<td>0.9</td>
<td>1.55E – 06</td>
<td>9.68E – 08</td>
<td>1.91E – 08</td>
<td>3.32E – 02</td>
</tr>
<tr>
<td>1</td>
<td>1.92E – 06</td>
<td>1.20E – 07</td>
<td>2.37E – 08</td>
<td>4.19E – 02</td>
</tr>
</tbody>
</table>

### Table III

<table>
<thead>
<tr>
<th>$t$</th>
<th>Results $E(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.94E – 04</td>
</tr>
<tr>
<td>0.1</td>
<td>1.88E – 04</td>
</tr>
<tr>
<td>0.2</td>
<td>1.82E – 04</td>
</tr>
<tr>
<td>0.3</td>
<td>1.77E – 04</td>
</tr>
<tr>
<td>0.4</td>
<td>1.71E – 04</td>
</tr>
<tr>
<td>0.5</td>
<td>1.65E – 04</td>
</tr>
<tr>
<td>0.6</td>
<td>1.60E – 04</td>
</tr>
<tr>
<td>0.7</td>
<td>1.55E – 04</td>
</tr>
<tr>
<td>0.8</td>
<td>1.49E – 04</td>
</tr>
<tr>
<td>1</td>
<td>1.38E – 04</td>
</tr>
</tbody>
</table>

### Table IV

<table>
<thead>
<tr>
<th>$t$</th>
<th>Results $E(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5.62E – 05</td>
</tr>
<tr>
<td>0.1</td>
<td>5.28E – 05</td>
</tr>
<tr>
<td>0.2</td>
<td>4.94E – 05</td>
</tr>
<tr>
<td>0.3</td>
<td>4.60E – 05</td>
</tr>
<tr>
<td>0.4</td>
<td>4.26E – 05</td>
</tr>
<tr>
<td>0.5</td>
<td>3.92E – 05</td>
</tr>
<tr>
<td>0.6</td>
<td>3.58E – 05</td>
</tr>
<tr>
<td>0.7</td>
<td>3.25E – 05</td>
</tr>
<tr>
<td>0.8</td>
<td>2.91E – 05</td>
</tr>
<tr>
<td>0.9</td>
<td>2.57E – 05</td>
</tr>
<tr>
<td>1</td>
<td>2.23E – 05</td>
</tr>
</tbody>
</table>

### References