The Collocation Method for Solving the Linear Fredholm
Integral Equation of the Second Kind Using Bernstein
Polynomials

Dr. Hussain Ali Al -Juboury * & Shymaa Hussain Salih**

Received on:8/4/2009
Accepted on:5/11/2009

Abstract

Integral equation of the second kind which has, extensively, been solved by
different ways, but not this one that deals with the collocation method using Bernstein
polynomials together with some useful examples to declare the method.

Keywords: Collocation method Fredholm integral equation of the second kind,
Bernstein polynomials, least square error.

Introduction

The well-known Fredholm integral equation of the second kind has been
discussed thoroughly in the recent time using various kinds of approximation
methods, in this paper; we submit another known method that is the
collocation method to solve this integral equation of Fredholm. Also
has been given some illustrative examples which has evaluated by a
computer program.

Recall Fredholm of the second kind in the form:

\[ U(x) = f(x) + \lambda \int_{a}^{b} K(x, y)U(y)dy \quad (1) \]

where the integral limits \( a \) and \( b \) are constants and \( K(x,y) \) is the kernel
function and \( f(x) \) is any continuous
function. Without loss of generality, let \( a=0 \) and \( b=1 \).

Numerical Solution: The discrete form
for the exact solution \( U(x) \) for equation
(1) can be written in the form:

\[ U(x) = U_N(x) \]

where \( N \) is a positive integer. This
paper pivoted to implement Bernstein
polynomials (B-spline) as a discrete
function (polynomial) of \( U_N(x) \) i.e.

\[ (2) U_N(x) = \sum_{i=0}^{N} a_i B_{i,N}(x) \quad x \in R \]

\[ B_{i,N}(x) = \binom{N}{i} (1-x)^{N-i} \]

https://doi.org/10.30684/etj.28.3.19
2412-0758/University of Technology-Iraq, Baghdad, Iraq
This is an open access article under the CC BY 4.0 license http://creativecommons.org/licenses/by/4.0
for \( i=0,1,2,...,n \), where the combination
\[
\binom{N}{i} = \frac{N!}{i!(N-i)!}
\]
There are \( N+1 \) \( N \) th degree Bernstein polynomials [1], [2].

The Bernstein polynomials of degree 1 are:
\[
B_{0,1}(x) = 1 - x \\
B_{1,1}(x) = x
\]

The Bernstein polynomials of deg. 2 are
\[
B_{0,2}(x) = (1-x)^2 \\
B_{1,2}(x) = 2x(1-x) \\
B_{2,2}(x) = x^2
\]

Some necessary characters of Bernstein polynomials: (for the proof see [5, 6])

i) A recursive definition of the Bernstein polynomials of degree \( N \) can be written as:
\[
B_{i,N}(x) = (1-x)B_{i,N-1}(x) + xB_{i-1,N-1}(x)
\]
\( i = 1,2,...,N \)
i) The Bernstein Polynomials are all non-negative for \( 0 \leq x \leq 1 \) (see[5]).

iii) The Bernstein polynomials form a partition of unity.

iv) Converting from Bernstein basis to power basis (proof see 5) as:
\[
B_{i,N}(x) = \sum_{k=i}^{N} (-1)^{k-i} \binom{N}{k} \binom{k}{i} x^k
\]
\( i=1,2,...,N \).

Therefore we could write the function
\[
U(x) = U_N(x) = a_0 B_{0,0}(x) + a_1 B_{1,0}(x) + \ldots + a_N B_{N,0}(x)
\]
\[
= a_0 \sum_{k=0}^{N} (-1)^k \binom{N}{k} x^k + \ldots +
\]
\[
+ a_N \sum_{k=N}^{k} (-1)^{k-N} \binom{N}{k} x^k
\]
\[
= a_0 + a_1 \sum_{k=1}^{N} (-1)^{k-1} \binom{N}{k} \binom{1}{1} x^k + \ldots +
\]
\[
+ a_N \sum_{k=0}^{N} (-1)^{k-N} \binom{N}{k} x^N
\]

This system can be written in a matrix form as follows
\[
U_N(x) = \begin{bmatrix} 1 & x & x^2 \end{bmatrix} \begin{bmatrix} a_{0,0} & 0 & L \\ a_{0,1} & a_{1,1} & L \\ a_{0,N} & a_{1,N} & L \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_N \end{bmatrix} = \begin{bmatrix} b \end{bmatrix}
\]

Using operator forms, equation (3) can be written as
\[
L[U] = f(x)
\]
where the operator \( L \) is defined as
\[
L[U] = U(x) - \int_{a}^{b} k(x, y) U(y) dy
\]
The unknown function \( U(x) \) is approximated in the form
\[
(5) U_N(x) = \sum_{i=0}^{N} a_i B_i(x)
\]
Substituting equation (5) in (4) to obtain
\[
L[U_N] = f(x) + E_N(x)
\]
where
\[
L[U_N(x)] = U_N(x) - \int_{a}^{b} k(x, y) U_N(y) dy
\]
For which we have the residue equation
\[ E_N(x) = L[U_N(x)] - f(x) \]  
Substituting Eq.(5) in Eq.(6) to get
\[ E_N(x) = L\left(\sum_{i=0}^{N} a_i B_i(x)\right) - f(x) \]
\[ = \sum_{i=1}^{N} a_i L(B_i(x)) - f(x) \]
Obviously, the weighted function sets its weighted integral equals to zero i.e.
\[ \int w_j E_N(x) d(x) = 0 \]
Inserting Eq.(7) into Eq.(8) to obtain
\[ \int w_j \left[ \sum_{i=0}^{N} a_i L(B_i(x)) - f(x) \right] = 0 \]
Whence
\[ \sum_{i=0}^{N} a_i \int w_j L(B_i(x)) = \int w_j f(x) \]
where
\[ L(B_j(x)) = B_j(x) - \int_{a}^{b} k(x,y) B_j(y) dy \]
\[ (9) \]
\[ (8) \]
\[ \int w_j E_N(x) d(x) = 0 \]
\[ (8) \]
\[ \int w_j \sum_{i=0}^{N} a_i L(B_i(x)) - f(x) \]
\[ (7) \]
\[ E_N(x) = L\left(\sum_{i=0}^{N} a_i B_i(x)\right) - f(x) \]
\[ (6) \]
\[ E_N(x) = L[U_N(x)] - f(x) \]
\[ \int_{a}^{b} \delta(x-x_j) \delta(x-x_j) d(x) \]
\[ = E(x_j) \int_{x_j}^{x_j} \delta(x-x_j) d(x) = 0, \quad j = 0, 1, \ldots, N \]
Therefore,
\[ E_M(x_j) = 0, \quad \text{where} \quad x_j = \frac{1}{i+1} \]
This implies that
\[ \sum_{j=0}^{N} L(B_j(x_j)) a_j = f(x_i) \quad i = 0, 1, \ldots, N \]
\[ (10) \]
Equation (10) is merely a system of \( N+1 \) equations with \( N+1 \) unknowns \( a_i \), \( i = 0, 1, \ldots, N \) as follows:
\[ \begin{bmatrix} L(B_0(x_0)) & L(B_1(x_1)) & \cdots & L(B_N(x_N)) \\ L(B_0(x_1)) & L(B_1(x_1)) & \cdots & L(B_N(x_N)) \\ \vdots & \vdots & \ddots & \vdots \\ L(B_0(x_N)) & L(B_1(x_N)) & \cdots & L(B_N(x_N)) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_N) \end{bmatrix} \]
Put
\[ K= \begin{bmatrix} L(B_0(x_0)) & L(B_1(x_1)) & \cdots & L(B_N(x_N)) \\ L(B_0(x_1)) & L(B_1(x_1)) & \cdots & L(B_N(x_N)) \\ \vdots & \vdots & \ddots & \vdots \\ L(B_0(x_N)) & L(B_1(x_N)) & \cdots & L(B_N(x_N)) \end{bmatrix} \]
\[ A= \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} \text{ and } H= \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_N) \end{bmatrix} \]
Then using Gauss elimination method to solve the obtained linear system
Conclusions

Bernstein Polynomial is introduced to find the approximate solution of Fredholm integral equation of the second kind. Two numerical examples were submitted to illustrate the given idea with good approximate results were achieved. We conclude that:

1. The use of Bernstein polynomials give, as it is expected, like the other polynomials, an accurate numerical solution for the simple continuous functions.

2. An advantage of using the Bernstein polynomials lies in their dependence upon a free parameter \( n \); this dependence gives the smallest least square error.

References


Table (1)

<table>
<thead>
<tr>
<th>x</th>
<th>Exact sol.</th>
<th>Bernstein Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>0.1</td>
<td>0.1000</td>
<td>0.1000</td>
</tr>
<tr>
<td>0.2</td>
<td>0.2000</td>
<td>0.2000</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3000</td>
<td>0.3000</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4000</td>
<td>0.4000</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5000</td>
<td>0.5000</td>
</tr>
<tr>
<td>0.6</td>
<td>0.6000</td>
<td>0.6000</td>
</tr>
<tr>
<td>0.7</td>
<td>0.7000</td>
<td>0.7000</td>
</tr>
<tr>
<td>0.8</td>
<td>0.8000</td>
<td>0.8000</td>
</tr>
<tr>
<td>0.9</td>
<td>0.9000</td>
<td>0.9000</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

L.Sq.E 0.0000

Table (2)

<table>
<thead>
<tr>
<th>x</th>
<th>Exact solution</th>
<th>Bernstein Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.000</td>
<td>-3.000</td>
<td>-3.000</td>
</tr>
<tr>
<td>-0.900</td>
<td>-2.700</td>
<td>-2.700</td>
</tr>
<tr>
<td>-0.800</td>
<td>-2.400</td>
<td>-2.400</td>
</tr>
<tr>
<td>-0.700</td>
<td>-2.100</td>
<td>-2.100</td>
</tr>
<tr>
<td>-0.600</td>
<td>-1.800</td>
<td>-1.800</td>
</tr>
<tr>
<td>-0.500</td>
<td>-1.500</td>
<td>-1.500</td>
</tr>
<tr>
<td>-0.400</td>
<td>-1.200</td>
<td>-1.200</td>
</tr>
<tr>
<td>-0.300</td>
<td>-0.900</td>
<td>-0.900</td>
</tr>
<tr>
<td>-0.200</td>
<td>-0.600</td>
<td>-0.600</td>
</tr>
<tr>
<td>-0.100</td>
<td>-0.300</td>
<td>-0.300</td>
</tr>
<tr>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>0.100</td>
<td>0.300</td>
<td>0.300</td>
</tr>
<tr>
<td>0.200</td>
<td>0.600</td>
<td>0.600</td>
</tr>
<tr>
<td>0.300</td>
<td>0.900</td>
<td>0.900</td>
</tr>
<tr>
<td>0.400</td>
<td>1.200</td>
<td>1.200</td>
</tr>
<tr>
<td>0.500</td>
<td>1.500</td>
<td>1.500</td>
</tr>
<tr>
<td>0.600</td>
<td>1.800</td>
<td>1.800</td>
</tr>
<tr>
<td>0.700</td>
<td>2.100</td>
<td>2.100</td>
</tr>
<tr>
<td>0.800</td>
<td>2.400</td>
<td>2.400</td>
</tr>
<tr>
<td>0.900</td>
<td>2.700</td>
<td>2.700</td>
</tr>
<tr>
<td>1.000</td>
<td>3.000</td>
<td>3.000</td>
</tr>
</tbody>
</table>

L.Sq.Error 0.0000