Optimal Control Approach for Solving Linear Volterra Integral Equations

Sohrab Effati
Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran
E-mail: s-effati@um.ac.ir

Mohammad Hadi Noori Skandari
Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran
E-mail: hadinoori344@yahoo.com

Abstract—In this paper we present a new approach for linear Volterra integral equations that is based on optimal control theory. Some optimal control problems corresponding Volterra integral equation be introduced which we solve these problems by discretization methods and linear programming approaches. Finally, some examples are given to show the efficiency of approach.

Index Terms — Volterra integral equations, Optimal control, Linear programming.

I. INTRODUCTION

Volterra integral equations arise in many physical applications, e.g., potential theory and Dirichlet problems and electostatics. Also, Volterra integral equations are applied in the biology, chemistry, engineering, mathematical problems of radiation equilibrium, the particle transport problems of astrophysics and reactor theory, and radiation heat transfer problems [1,2,3,4,5].

There exist some valid approximate and numerical methods for solving Volterra integral equation such as Adomian decomposition method [6], Walsh functions method and multigrid approach [7, 8], Bernstein Polynomials method [9], collocation method [10,11,12,13,14,15], Langrange interpolation method [16], Taylor-series expansion method [17,18,19], classical Neumann-series method [20], spectral methods [21,22,23,24,25,26,27], finite element method [28], Sinc method [29,30], Galerkin method [31,32], wavelet approach [33], block-by-block method [34] and homotopy analytic method [35,36,37,38,39].

In this paper, we present a different approach from above methods for solving linear Volterra integral equations which is based on the optimal control theory [40,41,42]. Consider the following linear Volterra integral equations of second kind:

\[ y(x) = \varphi(x) + \int_a^x g(x,t) y(t) \, dt, \quad x \in [a,b], \quad (1) \]

where \(a\) and \(b\) are constant, functions \(g(\cdot,\cdot)\) and \(\varphi(\cdot)\) are continuously differentiable with respect to \(x\). In equation (1), functions \(\varphi(\cdot)\) and \(g(\cdot,\cdot)\) are known and \(y(\cdot)\) is an unknown function. We assume that equation (1) have a solution.

The structure of this paper is as follows: Section 2 shows that solving Volterra integral equation (1) is equivalent to solve several optimal control problems. In section 3, a discretization method is applied to convert the problem to the corresponding linear programming problem. In section 4, the applicability of the approach is illustrated in several examples in which the computed results are compared with the exact solution. Section 5, gives the conclusion of this paper.

II. Optimal control problems

In this section, we are going to introduce some optimal control problems corresponding Volterra integral equation (1). Let \(p = \frac{\partial g}{\partial x}\). By using Leibnitz rule for derivatives we have:

\[ \frac{d}{dx} \int_a^x g(x,t) y(t) \, dt = g(x,x) y(x) + \int_a^x p(x,t) y(t) \, dt. \quad (2) \]

Also from equation (2) and differentiating both sides of (1) respect to \(x\) we have:

\[ y'(x) = \varphi'(x) + g(x,x) y(x) + \int_a^x p(x,t) y(t) \, dt, \quad x \in [a,b], \quad (3) \]

Now let \(x \in [a,b]\) be an arbitrary given number. We define the following problem:

\[ \begin{align*}
\left\{ \begin{array}{l}
\int_a^z (v'(t) - p(z,t) y(t)) \, dt = 0, \quad z \in [a,x] \\
y(a) = \varphi(a), \quad v(a) = 0.
\end{array} \right. \\
\end{align*} \quad \text{(4)} \]

Theorem III1: Let \(x \in [a,b]\) be an arbitrary number and \(\textbf{(}y^\ast(\cdot),v^\ast(\cdot)\text{)}\) be solution of problem (4)-(6). Then we have:

\[ y^\ast(x) = \varphi(x) + \int_a^x g(x,t) y^\ast(t) \, dt. \quad (7) \]

Proof: Let \(x \in [a,b]\). By initial conditions (6) and equation (5), we obtain

\[ v^\ast(z) = \int_a^z p(z,t) y^\ast(t) \, dt, \quad z \in [a,x] \quad (8). \]

So, by using equation (8) and (4), we take

Copyright © 2012 MECS
\[ y^v(z) = \phi'(z) + g(z, z) y^v(z) \]
\[ + \int_a^x p(z, t) y^v(t) dt, \quad z \in [a, x] \]

Hence
\[ y^v(z) = \phi'(z) + \frac{d}{dz} \int_a^x g(z, t) y^v(t) dt, \quad z \in [a, x]. \]

By integrating both sides of above equation on \([a, x]\) and conditions (6), we obtain the equation (7).

Now, let \( x \in [a, b] \) be an arbitrary given number and define the following optimal control problem:

**Minimize**
\[ \int_a^x |p'(t) + g(t, t) y(t) + v(t) - u_1(t)| dt \]

**subject to**
\[ y'(t) = u_1(t), \quad t \in [a, x] \]
\[ v'(t) = u_2(t), \quad t \in [a, x] \]
\[ r(t, y, u_2) = 0, \quad t \in [a, x] \]
\[ y(a) = \phi(a), \quad v(a) = 0, \]
where \((y(\cdot), v(\cdot))\) and \((u_1(\cdot), u_2(\cdot))\) are state and control variables, respectively, and for \( t \in [a, x] \)
\[ r(t, y, u_2) = \int_a^x (u_2(w) - p(w, w) y(w)) dw. \]

**Theorem II.2:** Let \( x \in [a, b] \) be an arbitrary given number and pairs \((y^v(\cdot), v^v(\cdot))\) and \((u_1^v(\cdot), u_2^v(\cdot))\) be optimal state and control of the problem (9), respectively. Then function \( y^v(\cdot)\) satisfies equation (7).

**Proof:** Assume that \( x \in [a, b] \) is an arbitrary given number. By theorem II.1, it is sufficient that we show pair \((y^v(\cdot),\nu^v(\cdot))\) is the solution of problem (4)-(6). Define the following problem corresponding to the problem (9):

**Minimize**
\[ J(y, v) = \int_a^x |p'(t) + g(t, t) y(t) + v(t) - y'(t)| dt \]

**subject to**
\[ \int_a^x (v(w) - p(t, w) y(w)) dw = 0, \quad t \in [a, x] \]
\[ y(a) = \phi(a), \quad v(a) = 0. \]

Now, let \( y(\cdot) = \bar{y}(\cdot) \) be the solution of the equation (1), and \( v(\cdot) = \bar{v}(\cdot) \) satisfies equation (12) where \( y(\cdot) = \bar{y}(\cdot) \). It is trivial that pair \((y^v(\cdot), v^v(\cdot))\) satisfies initial conditions (13) and \( J(\bar{y}, \bar{v}) = 0 \). On other hand, since \((y^v(\cdot), v^v(\cdot))\) and \((u_1^v(\cdot), u_2^v(\cdot))\) are optimal solutions of problem (9), pair \((y^v(\cdot), v^v(\cdot))\) is the optimal solution of the problem (11)-(13) and we have \( J(y^v, v^v) = 0 \). Thus

\[ \int_a^x |p'(t) + g(z, z) y^v(z) + v^v(z) - y'(z)| dt = 0. \]

Hence
\[ y^v(t) = \phi'(t) + g(t, t) y^v(t) + v^v(t). \]

Moreover, equation (5) and initial conditions (6) hold for pair \((y^v(\cdot), v^v(\cdot))\). Thus, pair \((y^v(\cdot), v^v(\cdot))\) is the solution of the problem (4)-(6) and this completes proof. \( \Box \)

Now let \( n \in \mathbb{N} \) be a given large number. Set \( x_0 = a \),
\[ h = \frac{b - a}{n} \]
and \( x_j = a + jh \) for \( j = 1, 2, ..., n \). The corresponding optimal control problem (9) for \( x = x_j \), \( j = 1, 2, ..., n \) is as follows:

**Minimize**
\[ \int_a^x |p'(t) + g(t, t) y(t) + v(t) - u_1(t)| dt \]

**subject to**
\[ y'(t) = u_1(t), \quad t \in [a, x] \]
\[ v'(t) = u_2(t), \quad t \in [a, x] \]
\[ r(t, y, u_2) = 0, \quad t \in [a, x] \]
\[ y(a) = \phi(a), \quad v(a) = 0, \]
where \( r(\cdot, \cdot) \) is defined by (10). We rewrite problem (14) for \( j = 1, 2, ..., n \) as follows:

**Minimize**
\[ \sum_{k=1}^{j} \int_{x_{k-1}}^{x_k} |p'(t) + g(t, t) y(t) + v(t) - u_1(t)| dt \]

**subject to**
\[ y'(t) = u_1(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, ..., j \]
\[ v'(t) = u_2(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, ..., j \]
\[ r(t, y, u_2) = 0, \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, ..., j \]
\[ y(x_0) = \phi(x_0), \quad v(x_0) = 0. \]

By assumption \( \mu(t) = \phi'(t) + g(t, t) y(t) + v(t) - u_1(t) \), \( t \in [a, b] \) the problems (15) for all \( j = 1, 2, ..., n \) is equivalent to the following multi-objective programming problem:

**Minimize**
\[ \{ \int_{x_{k-1}}^{x_k} |\mu(z)| dz : \int_{x_{k-1}}^{x_k} |\mu(z)| dz + \int_{x_{k}}^{x_{k+1}} |\mu(z)| dz, \]
\[ ... \int_{x_{n-2}}^{x_{n-1}} |\mu(z)| dz + \int_{x_{n-1}}^{x_n} |\mu(z)| dz \}

**subject to**
\[ y'(t) = u_1(t), \quad t \in [x_{k-1}, x_k], \]
\[ v'(t) = u_2(t), \quad t \in [x_{k-1}, x_k], \]
\[ r(t, y, u_2) = 0, \quad t \in [x_{k-1}, x_k], \]
\[ y(x_0) = \phi(x_0), \quad v(x_0) = 0, \]
\[ k = 1, 2, ..., j, \quad j = 1, 2, ..., n. \]

By multi-objective programming methods (see [43]), we
can use the following problem instead of problem (16):

\[ \text{Minimize} \quad J = \sum_{k=1}^{n} (n-k+1) \int_{x_{k-1}}^{x_k} |\mu(z)|dz \tag{17} \]

subject to

\[ y'(t) = u_i(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{18} \]
\[ v'(t) = u_i(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{19} \]
\[ r_i(t, y, u_s) = 0, \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{20} \]
\[ y(x_0) = \phi(x_0), \quad v(x_0) = 0. \tag{21} \]

Define the following problem for \( k = 1, 2, \ldots, n \):

\[ \text{Minimize} \quad J_k = (n-k+1) \int_{x_{k-1}}^{x_k} |\mu(z)|dz \tag{22} \]

subject to

\[ y'(t) = u_i(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{23} \]
\[ v'(t) = u_i(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{24} \]
\[ r_i(t, y, u_s) = 0, \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{25} \]
\[ y(x_{k-1}) = y^{[k-1]}, \quad v(x_{k-1}) = v^{[k-1]}, \quad k = 1, 2, \ldots, n \tag{26} \]

where \( y^{[k-1]} \) and \( v^{[k-1]} \) are known numbers which we can obtain these numbers by solving problem (22)-(26) on interval \([x_{k-1}, x_k]\). Moreover, for \( t \in [x_{k-1}, x_k] \)

\[ r_{i-1}(t, y, u_s) = \int_{x_{k-1}}^{t} (u_s(w) - p(t, w))v(w)dw. \tag{27} \]

Now we have the following theorem:

**Theorem III:** Let \( (y^{[k-1]}(\cdot), v^{[k-1]}(\cdot)) \) and \( (u^{[k-1]}(\cdot), u^{[k-1]}(\cdot)) \) for any \( k = 1, 2, \ldots, n \) be optimal state and control of problem (22)-(26), respectively. Then optimal solutions of problem (17)-(21) are as follows:

\[ y^{*}(t) = y^{[k-1]}(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{28} \]
\[ v^{*}(t) = v^{[k-1]}(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{29} \]
\[ u^{*}(t) = u^{[k-1]}(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{30} \]
\[ u^{*}(t) = u^{[k-1]}(t), \quad t \in [x_{k-1}, x_k], \quad k = 1, 2, \ldots, n \tag{31} \]

**Proof:** Consider the following sets:

\[ D = \{ (y, v, u_i, u_s) : (y, v, u_i, u_s) \text{ satisfies in equations} \ (18)-(20) \text{ on} [x_0, x_n] \} \tag{32} \]
\[ D_k = \{ (y, v, u_i, u_s) : (y, v, u_i, u_s) \text{ satisfies in equations} \ (23)-(26) \text{ on} [x_{k-1}, x_k] \}, \quad k = 1, 2, \ldots, n \tag{33} \]

It is trivial that \( D = \bigcup_{k=1}^{n} D_k \). In addition, the value of objective functional in problems (17)-(21) and (22)-(26) are nonnegative. Thus, we have:

\[ \text{Minimize} \quad J = \sum_{k=1}^{n} \left( \text{Minimize} \quad J_k \right) \tag{34} \]

The system (27) is immediate from latter relation and optimality principle in continuous-time dynamic programming which initiated by Bellman (see section 8 of [44]). □

**Remark II-4:** Note that \( y^{[0]} = \phi(x_0) \) and \( v^{[0]} = 0 \).

### III. Linear programming problems

In this section, we convert the continuous problem (22)-(28) to an equivalent discrete problem and solve the obtained problem by linear programming method (see [45]). For this purpose, consider the following approximations in numerical differentiation and integration for \( k = 1, 2, \ldots, n \) and \( t = x_{k-1}, x_k \):

\[ \int_{x_{k-1}}^{x_k} |\mu(z)|dz = \frac{h}{2} \left[ |\mu(x_{k-1})| + |\mu(x_k)| \right] \]
\[ y'(t) = y_k - y_{k-1} - hu_{k-1} = 0, \quad v'(t) = v_k - v_{k-1} - hu_{k-1} = 0 \]
\[ u_{k-1} - p(x_k, x_{k-1})y_{k-1} + u_{k-2} - p(x_k, x_{k-1})y_{k-1} = 0 \]
\[ y_k = y^{[k-1]}, \quad v_k = v^{[k-1]}, \quad k = 1, 2, \ldots, n \tag{35} \]

Using the above approximation we can transform problem (22)-(28) to the following corresponding problem for \( k = 1, 2, \ldots, n \):

\[ \text{Minimize} \quad J_k = \frac{h}{2} (n-k+1) \left( |\mu_{k-1}| + |\mu_k| \right) \tag{36} \]

subject to

\[ y_k - y_{k-1} - hu_{k-1} = 0, \quad v_k - v_{k-1} - hu_{k-1} = 0 \]
\[ u_{k-1} - p(x_k, x_{k-1})y_{k-1} + u_{k-2} - p(x_k, x_{k-1})y_{k-1} = 0 \]
\[ y_k = y^{[k-1]}, \quad v_k = v^{[k-1]}, \quad k = 1, 2, \ldots, n \tag{37} \]

where for \( l = k - 1, k \) we have:

\[ u_{l} = u_{l}(x_{k-1}), u_{l+1} = u_{l+1}(x_{k-1}) \]
\[ \mu_l = \mu(x_{k-1}), \quad y_l = y(x_{k-1}), \quad v_l = v(x_{k-1}) \tag{38} \]

Now by applying the techniques of linear and nonlinear programming [45,46], and relation

\[ \mu(t) = \varphi(t) + g(t, t) y(t) + v(t) - u_i(t), \quad t \in [a,b] \tag{39} \]

the nonlinear problem (28) be converted to the following corresponding linear programming problem for \( k = 1, 2, \ldots, n \):

\[ \text{Minimize} \quad J_k = \frac{h}{2} (n-k+1) \left( z_{k-1} + z_k \right) \tag{40} \]

subject to

\[ \mu_{k-1} + z_{k-1} \leq 0, \quad -\mu_{k+1} + z_{k+1} \leq 0 \]
\[ \mu_k + z_k \leq 0, \quad -\mu_k + z_k \leq 0 \]

\[ g(x_{k-1}, x_k) y_k + v_k - u_i - \mu_k = -\varphi(x_k), \]
\[ g(x_k, x_{k-1}) y_k + v_k - u_i - \mu_k = -\varphi(x_k), \]
\[ y_k - y_{k-1} - hu_{k-1} = 0, \quad y_k - y_{k-1} - hu_{k-1} = 0 \tag{41} \]
where decision variables of this problem are $z_i, \mu_i, y_i, v_i, u_i$, and $u_{2i}$ for $l = k - 1, k$. Not that by solving problems (29), we obtain the approximations $y^*_k$, $k = 1, 2, \ldots, n$.

Remark III.1: Note that, in this approach, we solve the linear programming problem (29) and use theorem 2-3 to approximate the solution of equation (1). In next section, we illustrative the efficiency of our approach in some numerical examples.

IV. Simulation results

In this section, we use our approach to solve two linear Volterra integral equations. Here we apply the MATLAB software and simplex method [45] for solving linear programming problem (29).

Example IV.1: Consider the following Volterra integral equation of second kind:

$$
\begin{align*}
(2 \cos(x)) (\ln(3) - \ln(2 + \cos(x))),
\end{align*}
$$

Here for $n = 10, 20, 50$ and $n = 100$, we solve corresponding problem (29) for equation (30). In Figures 1-4, the connected line indicates the graph of exact solutions. The comparison of obtained results and exact solution for equation (29) are showed in Table 1.

Example IV.2: Consider the following Volterra integral equation of second kind:

$$
\begin{align*}
(2 \cos(x)) (\ln(3) - \ln(2 + \cos(x))),
\end{align*}
$$

Here for $n = 10, 20, 30$ and $n = 40$, the corresponding problem (29) for equation (31) is solved. The connected line indicates the graph of exact solutions. We can compare the obtained results and exact solution of equation (31) in Table 2.

V. Conclusions

In this paper, we obtained optimal control problems corresponding Volterra integral equation. By discretization method these optimal control problems converted to the corresponding linear programming problems. Thus we can solve linear programming problems instead of Volterra integral equations. By this approach, we can obtain a good approximation for the solution of linear Volterra integral equation.
Optimal Control Approach for Solving Linear Volterra Integral Equations

Fig. 4: The approximate solution for $n = 100$

Table 1: Comparison of exact and approximate solutions for Ex. IV.1.

<table>
<thead>
<tr>
<th>x</th>
<th>Approximate solution for $n=10$</th>
<th>Approximate solution for $n=20$</th>
<th>Approximate solution for $n=50$</th>
<th>Approximate solution for $n=100$</th>
<th>Exact solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>x=0.2</td>
<td>0.23098159</td>
<td>0.25026868</td>
<td>0.25929638</td>
<td>0.26309086</td>
<td>0.26692094</td>
</tr>
<tr>
<td>x=0.5</td>
<td>0.86113061</td>
<td>0.92262860</td>
<td>0.96197727</td>
<td>0.97551724</td>
<td>0.98800616</td>
</tr>
<tr>
<td>x=0.8</td>
<td>1.96419901</td>
<td>2.09890818</td>
<td>2.18542479</td>
<td>2.21524555</td>
<td>2.21524555</td>
</tr>
</tbody>
</table>

Fig. 5: The approximate solution for $n = 10$.

Fig. 6: The approximate solution for $n = 20$.

Fig. 7: The approximate solution for $n = 30$.

Fig. 8: The approximate solution for $n = 40$. 
Table 2: Comparison of exact and approximate solutions for Ex. IV.2.

<table>
<thead>
<tr>
<th>( n )</th>
<th>( \lambda )</th>
<th>Approximate solution for ( n=10 )</th>
<th>Approximate solution for ( n=20 )</th>
<th>Approximate solution for ( n=30 )</th>
<th>Approximate solution for ( n=40 )</th>
<th>Exact solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>0.29158041</td>
<td>0.23111194</td>
<td>0.23534042</td>
<td>0.23753096</td>
<td>0.24425855</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.75296658</td>
<td>0.77435737</td>
<td>0.79243764</td>
<td>0.80190046</td>
<td>0.82450683</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>1.54942048</td>
<td>1.68180277</td>
<td>1.73246743</td>
<td>1.75923588</td>
<td>1.78043274</td>
<td></td>
</tr>
</tbody>
</table>

REFERENCES


