Research Article

New Perturbation Iteration Solutions for Fredholm and Volterra Integral Equations

İhsan Timuçin Dolapcı,¹ Mehmet Şenol,² and Mehmet Pakdemirli¹

¹ Department of Mechanical Engineering, Celal Bayar University, Muradiye, 45140 Manisa, Turkey
² Department of Mathematics, Faculty of Science, Nevşehir University, 50300 Nevşehir, Turkey

Correspondence should be addressed to Mehmet Pakdemirli; mpak@cbu.edu.tr

Received 29 December 2012; Accepted 22 April 2013

In this paper, recently developed perturbation iteration method is used to solve Fredholm and Volterra integral equations. The study shows that the new method can be applied to both types of integral equations. Some numerical examples are given, and results are compared with other studies to illustrate the efficiency of the method.

1. Introduction

As one of the most important subjects of mathematics, differential and integral equations are widely used to model a variety of physical problems. Perturbation methods have been used in search of approximate analytical solutions for over a century [1–3]. Algebraic equations, integral-differential equations, and difference equations could be solved by these techniques approximately.

However, a major difficulty in the implementation of perturbation methods is the requirement of a small parameter or inserting a small artificial parameter in the equation. Solutions obtained by these methods are therefore restricted by a validity range of physical parameters. To eliminate the small parameter assumption in regular perturbation analysis, iteration techniques are incorporated with perturbations. Many attempts in this issue appear in the literature recently [4–13].

Recently, a new perturbation-iteration algorithm has been developed by Pakdemirli and his coworkers [14–16]. A preliminary study of developing root finding algorithms systematically [17–19] finally led to generalization of the method to differential equations also [14–16]. An iterative scheme is constituted over the perturbation expansion in the new technique. The method has been successfully implemented to first-order equations [15] and Bratu-type second-order equations [14].

In this paper, this new technique is applied to integral equations for the first time. Fredholm and Volterra integral equations

\[ y(t) = x(t) + \int_0^1 k(t, s)y(s)\, ds, \]

\[ y(t) = x(t) + \int_0^t k(t, s)y(s)\, ds \]  

are considered, where \( x(t) \in L^2[0, 1] \), \( k(t, s) \in L^2[0, 1] \times [0, 1] \), and \( y(t) \) is the unknown function to be determined. Results are compared with some other studies.

2. Overview of the Method

In the present paper, the simplest perturbation-iteration algorithm PIA(1, 1) is used by taking one correction term in the perturbation expansion and correction terms of only first derivatives in the Taylor series expansion, that is, \( n = 1, m = 1 \) [14–16]. Consider the Volterra integral equation

\[ y(t) = x(t) + \int_0^t k(t, s)y(s)\, ds \]  

(2)
that has the form of

\[ F(u, \int u, \varepsilon) = 0, \]

(3)

where

\[ F = y(t) - x(t) - \varepsilon \int_0^t k(t, s) y(s) \, ds \]

(4)

and \( \varepsilon \) is the artificially introduced perturbation parameter.

In this method, we use only one correction term in the perturbation expansion:

\[ u_{n+1} = u_n + \varepsilon \langle u_\zeta \rangle_n. \]

(5)

Substituting (5) into (3) and expanding in a Taylor series with first-order derivatives only yield

\[ F(u_n, \int u_n, 0) + F_u(u_n, \int u_n, 0) \varepsilon + F_\varepsilon(u_n, \int u_n, 0) \varepsilon \int (u_\zeta)_n = 0 \]

(6)

or

\[ \langle u_\zeta \rangle_n \frac{\partial F}{\partial u} + \int (u_\zeta)_n \frac{\partial F}{\partial \int u} + \frac{\partial F}{\partial \varepsilon} + F = 0. \]

(7)

All derivatives are evaluated at \( \varepsilon = 0 \).

Starting with the initial condition \( u_0 \), first \( \langle u_\zeta \rangle_0 \) has been calculated by the help of (7). Then we substitute \( \langle u_\zeta \rangle_0 \) into (5) to find \( u_1 \). Iteration process is repeated using (7) and (5) until we obtain a satisfactory result.

### 3. Application

**Example 1.** Consider the Fredholm integral equation of the second kind

\[ u(x) = \int_0^1 \left( \frac{1}{3} e^{2x-(5/3)y} \right) u(t) \, dt + e^{2x+1/3} \]

(8)

with exact solution

\[ u(x) = e^{2x}. \]

(9)

Equation (8) can be rewritten in the following form:

\[ F(u, \varepsilon) = u(x) - e^{2x+1/3} - \varepsilon \int_0^1 \left( \frac{1}{3} e^{2x-(5/3)y} \right) u(t) \, dt, \]

(10)

where \( \varepsilon \) is a small parameter. The terms in (7) are

\[ F = u_n(x) - e^{2x+1/3}, \quad F_u = 1, \]

\[ F_\varepsilon = -\varepsilon \int_0^1 \left( \frac{1}{3} e^{2x-(5/3)y} \right) u_n(t) \, dt, \quad F_{\int u} = 0. \]

(11)

Note that introducing the small parameter \( \varepsilon \) as a coefficient of the integral term simplifies (7) and makes it solvable. For this specific example (7) reads

\[ e^{1/3+2x} + \int_0^1 \frac{1}{3} e^{-5t/3+2x} u_n(t) \, dt = \langle u_\zeta \rangle_n(x) + u_n(x). \]

(12)

When applying the iteration formula (5), we select an initial guess appropriate to the boundary condition and at each step we determine coefficients from the boundary condition. Starting with the initial function

\[ u_0 = 1 \]

(13)

and using the formula, the approximate solutions at each step are

\[ u_1 = -e^{2x} + \frac{1}{5} e^{-5/3+2x} + e^{1/3+2x}, \]

\[ u_2 = -\frac{1}{5} e^{-5/3+2x} \left( -1 + e^{1/3} + e^{5/3} - 11e^2 + 5e^{7/3} \right), \]

\[ u_3 = \frac{1}{5} e^{-(5/3)+2x} \times \left( 1 + e^{1/3} \left( -2 + e^{1/3} \times \left( 1 + e \left( -1 + 17e^{1/3} - 16e^{2/3} + 5e \right) \right) \right) \right). \]

(14)

Higher iterations are not given here for brevity. Using a symbolic manipulation software, iterations could be calculated up to any order. In Table 1, some of our iterations are compared with the exact solution and the error between the exact solution, and \( u_{20} \) are given which are of order \( 10^{-8} \).

**Example 2.** Consider the following integral equation:

\[ u(x) = \cos x - \int_0^x (x-t) \cos (x-t) \, u(t) \, dt. \]

(15)

The exact solution of the problem is

\[ u(x) = \frac{1}{3} \left( 2 \cos \sqrt{3}x + 1 \right). \]

(16)

Equation (15) can be rewritten in the following form:

\[ F(u, \varepsilon) = u(x) - \cos x - \varepsilon \int_0^x (x-t) \cos (x-t) \, u(t) \, dt, \]

(17)

where \( \varepsilon \) is a small artificial parameter. The terms in (7) are

\[ F = u_n(x) - \cos x, \quad F_u = 1, \]

\[ F_\varepsilon = \int_0^x (x-t) \cos (x-t) u_n(t) \, dt, \quad F_{\int u} = 0. \]

(18)

Equation (7) reduces to

\[ \int_0^x (x-t) \cos (t-x) u_n \, dt + \langle u_\zeta \rangle_n + u_n = \cos x. \]

(19)
Table 1: Numerical result of Example 1.

<table>
<thead>
<tr>
<th>x</th>
<th>( u_5 )</th>
<th>( u_{10} )</th>
<th>( u_{15} )</th>
<th>( u_{20} )</th>
<th>Exact solution</th>
<th>Error (( u_{10} ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>1.00072</td>
<td>0.99995</td>
<td>1.00000</td>
<td>1.00000</td>
<td>1.00000</td>
<td>5.20782 \times 10^{-9}</td>
</tr>
<tr>
<td>0.1</td>
<td>1.22839</td>
<td>1.22134</td>
<td>1.22140</td>
<td>1.22140</td>
<td>1.22140</td>
<td>6.38084 \times 10^{-9}</td>
</tr>
<tr>
<td>0.2</td>
<td>1.50035</td>
<td>1.49174</td>
<td>1.49183</td>
<td>1.49182</td>
<td>1.49182</td>
<td>7.76915 \times 10^{-9}</td>
</tr>
<tr>
<td>0.3</td>
<td>1.83254</td>
<td>1.82202</td>
<td>1.82212</td>
<td>1.82212</td>
<td>1.82212</td>
<td>9.48926 \times 10^{-9}</td>
</tr>
<tr>
<td>0.4</td>
<td>2.23826</td>
<td>2.22542</td>
<td>2.22554</td>
<td>2.22554</td>
<td>2.22554</td>
<td>1.15902 \times 10^{-8}</td>
</tr>
<tr>
<td>0.5</td>
<td>2.73382</td>
<td>2.71813</td>
<td>2.71828</td>
<td>2.71828</td>
<td>2.71828</td>
<td>1.41563 \times 10^{-8}</td>
</tr>
<tr>
<td>0.6</td>
<td>3.33910</td>
<td>3.31993</td>
<td>3.32012</td>
<td>3.32012</td>
<td>3.32012</td>
<td>1.72906 \times 10^{-8}</td>
</tr>
<tr>
<td>0.7</td>
<td>4.07838</td>
<td>4.05498</td>
<td>4.05520</td>
<td>4.05520</td>
<td>4.05520</td>
<td>2.11187 \times 10^{-8}</td>
</tr>
<tr>
<td>0.8</td>
<td>4.98135</td>
<td>4.95276</td>
<td>4.95304</td>
<td>4.95304</td>
<td>4.95304</td>
<td>2.57945 \times 10^{-8}</td>
</tr>
<tr>
<td>0.9</td>
<td>6.08423</td>
<td>6.04931</td>
<td>6.04965</td>
<td>6.04965</td>
<td>6.04965</td>
<td>3.15055 \times 10^{-8}</td>
</tr>
<tr>
<td>1.0</td>
<td>7.43130</td>
<td>7.38865</td>
<td>7.38906</td>
<td>7.38906</td>
<td>7.38906</td>
<td>3.84809 \times 10^{-8}</td>
</tr>
</tbody>
</table>

Table 2: Numerical result of Example 2.

<table>
<thead>
<tr>
<th>x</th>
<th>( u_5 )</th>
<th>( u_{10} )</th>
<th>Exact solution</th>
<th>Error (( u_{10} ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>0.990025</td>
<td>0.990025</td>
<td>0.990025</td>
<td>0.11022 \times 10^{-16}</td>
</tr>
<tr>
<td>0.2</td>
<td>0.960398</td>
<td>0.960398</td>
<td>0.960398</td>
<td>0</td>
</tr>
<tr>
<td>0.3</td>
<td>0.912007</td>
<td>0.912007</td>
<td>0.912007</td>
<td>0</td>
</tr>
<tr>
<td>0.4</td>
<td>0.846298</td>
<td>0.846298</td>
<td>0.846298</td>
<td>1.11022 \times 10^{-16}</td>
</tr>
<tr>
<td>0.5</td>
<td>0.765240</td>
<td>0.765240</td>
<td>0.765240</td>
<td>0</td>
</tr>
<tr>
<td>0.6</td>
<td>0.671256</td>
<td>0.671256</td>
<td>0.671256</td>
<td>2.22045 \times 10^{-16}</td>
</tr>
<tr>
<td>0.7</td>
<td>0.567160</td>
<td>0.567160</td>
<td>0.567160</td>
<td>0</td>
</tr>
<tr>
<td>0.8</td>
<td>0.456066</td>
<td>0.456066</td>
<td>0.456066</td>
<td>1.11022 \times 10^{-16}</td>
</tr>
<tr>
<td>0.9</td>
<td>0.341300</td>
<td>0.341300</td>
<td>0.341300</td>
<td>5.55112 \times 10^{-17}</td>
</tr>
<tr>
<td>1.0</td>
<td>0.226296</td>
<td>0.226296</td>
<td>0.226296</td>
<td>5.55112 \times 10^{-17}</td>
</tr>
</tbody>
</table>

Choosing the initial guess \( u_0 = 1 \) and using the formula, the approximate solutions at each step are

\[
u_1 = 1 - x \sin x,
\]

\[
u_2 = 1 + \frac{1}{12} x (-12 + x^2) \sin x,
\]

\[
u_3 = 1 + \frac{1}{480} x (15 x \cos x - (495 - 45 x^2 + x^4) \sin x).
\]

Higher iterations are not given for brevity. In Table 2, some of our iterations are compared with the exact solution, and the errors between the exact solution and \( u_{10} \) are given which are of order \( 10^{-16} \).

**Example 3.** Consider the equation

\[
u(x) = e^{3x} - \frac{1}{9} (2e^3 + 1) x + \int_0^1 x t u(t) dt \tag{22}
\]

with the exact solution

\[
u(x) = e^{3x}. \tag{23}
\]

Equation (22) is rewritten in the following form:

\[
F(u, \varepsilon) = u(x) - e^{3x} + \frac{1}{9} (2e^3 + 1) x - \varepsilon \int_0^1 x t u(t) dt, \tag{24}
\]

where \( \varepsilon \) is an artificially introduced small parameter. The terms in (7) are

\[
F = u_n(x) - e^{3x} + \frac{1}{9} (2e^3 + 1) x, \quad F_u = 1, \tag{25}
\]

\[
F_\varepsilon = - \int_0^1 x t u_n(t) dt, \quad F_{\int u} = 0.
\]

Equation (7) reduces to

\[
9 \left( e^{3x} + \int_0^1 t x u_n dt \right) = x + 2 e^3 x + 9 (u_n)_n + 9 u_n. \tag{26}
\]

Choosing the initial guess

\[
u_0 = 1 \tag{27}
\]
Table 3: Numerical result of Example 3.

<table>
<thead>
<tr>
<th>𝑥</th>
<th>𝑢_5</th>
<th>𝑢_10</th>
<th>𝑢_15</th>
<th>𝑢_20</th>
<th>Exact solution</th>
<th>Error (𝑢_20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>1.34483</td>
<td>1.34984</td>
<td>1.34986</td>
<td>1.34986</td>
<td>1.34986</td>
<td>2.45400 × 10^{-9}</td>
</tr>
<tr>
<td>0.2</td>
<td>1.81206</td>
<td>1.82212</td>
<td>1.82212</td>
<td>1.82212</td>
<td>1.82212</td>
<td>4.90801 × 10^{-9}</td>
</tr>
<tr>
<td>0.3</td>
<td>2.44451</td>
<td>2.45960</td>
<td>2.45960</td>
<td>2.45960</td>
<td>2.45960</td>
<td>7.36201 × 10^{-9}</td>
</tr>
<tr>
<td>0.4</td>
<td>3.29996</td>
<td>3.32011</td>
<td>3.32011</td>
<td>3.32012</td>
<td>3.32012</td>
<td>9.81602 × 10^{-9}</td>
</tr>
<tr>
<td>0.5</td>
<td>4.45654</td>
<td>4.48169</td>
<td>4.48169</td>
<td>4.48169</td>
<td>4.48169</td>
<td>1.22700 × 10^{-8}</td>
</tr>
<tr>
<td>0.6</td>
<td>6.01947</td>
<td>6.04965</td>
<td>6.04965</td>
<td>6.04965</td>
<td>6.04965</td>
<td>1.47240 × 10^{-8}</td>
</tr>
<tr>
<td>0.7</td>
<td>8.13096</td>
<td>8.16617</td>
<td>8.16617</td>
<td>8.16617</td>
<td>8.16617</td>
<td>1.71780 × 10^{-8}</td>
</tr>
<tr>
<td>0.8</td>
<td>10.98293</td>
<td>11.02318</td>
<td>11.02318</td>
<td>11.02318</td>
<td>11.02318</td>
<td>1.96320 × 10^{-8}</td>
</tr>
<tr>
<td>0.9</td>
<td>14.83446</td>
<td>14.87973</td>
<td>14.87973</td>
<td>14.87973</td>
<td>14.87973</td>
<td>2.20860 × 10^{-8}</td>
</tr>
<tr>
<td>1.0</td>
<td>20.03523</td>
<td>20.08554</td>
<td>20.08554</td>
<td>20.08554</td>
<td>20.08554</td>
<td>2.45400 × 10^{-8}</td>
</tr>
</tbody>
</table>

Table 4: Numerical result of Example 4.

<table>
<thead>
<tr>
<th>𝑥</th>
<th>𝑢_5</th>
<th>𝑢_10</th>
<th>𝑢_15</th>
<th>𝑢_20</th>
<th>Exact solution</th>
<th>Error (𝑢_20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>0.0099999</td>
<td>0.00999</td>
<td>0.01000</td>
<td>0.01000</td>
<td>0.01000</td>
<td>0.00999</td>
</tr>
<tr>
<td>0.2</td>
<td>0.0399996</td>
<td>0.03999</td>
<td>0.04000</td>
<td>0.04000</td>
<td>0.04000</td>
<td>0.03999</td>
</tr>
<tr>
<td>0.3</td>
<td>0.0899991</td>
<td>0.08999</td>
<td>0.09000</td>
<td>0.09000</td>
<td>0.09000</td>
<td>0.08999</td>
</tr>
<tr>
<td>0.4</td>
<td>0.159998</td>
<td>0.15999</td>
<td>0.16000</td>
<td>0.16000</td>
<td>0.16000</td>
<td>0.15999</td>
</tr>
<tr>
<td>0.5</td>
<td>0.249998</td>
<td>0.24999</td>
<td>0.25000</td>
<td>0.25000</td>
<td>0.25000</td>
<td>0.24999</td>
</tr>
<tr>
<td>0.6</td>
<td>0.359996</td>
<td>0.35999</td>
<td>0.36000</td>
<td>0.36000</td>
<td>0.36000</td>
<td>0.35999</td>
</tr>
<tr>
<td>0.7</td>
<td>0.489995</td>
<td>0.48999</td>
<td>0.49000</td>
<td>0.49000</td>
<td>0.49000</td>
<td>0.48999</td>
</tr>
<tr>
<td>0.8</td>
<td>0.639994</td>
<td>0.63999</td>
<td>0.64000</td>
<td>0.64000</td>
<td>0.64000</td>
<td>0.63999</td>
</tr>
<tr>
<td>0.9</td>
<td>0.809992</td>
<td>0.80999</td>
<td>0.81000</td>
<td>0.81000</td>
<td>0.81000</td>
<td>0.80999</td>
</tr>
<tr>
<td>1.0</td>
<td>0.999999</td>
<td>0.99999</td>
<td>1.00000</td>
<td>1.00000</td>
<td>1.00000</td>
<td>0.99999</td>
</tr>
</tbody>
</table>

and using the formula, the approximate solutions at each step are

\[ u_1 = e^{3x} + \frac{7x}{18} - \frac{2e^{3x}x}{9}, \]
\[ u_2 = e^{3x} - \frac{1}{54} (7 + 4e^{3x})x, \] (28)
\[ u_3 = e^{3x} - \frac{1}{162} (7 + 4e^{3x})x. \]

Higher iterations are not given for brevity. In Table 3, some of our iterations are compared with the exact solution, and the errors between the exact solution and \( u_{20} \) are given which are of order \( 10^{-8} \).

Example 4. Consider the following integral equation:

\[ u(x) = \frac{9}{10} x^2 + \frac{1}{2} \int_0^x t^2 u(t) \, dt. \] (29)

The exact solution of the problem is \( u(x) = x^2 \). (30)

Equation (29) is rewritten in the following form:

\[ F(u, \epsilon) = u(x) - \frac{9}{10} x^2 - \frac{1}{2} \int_0^x t^2 u(t) \, dt, \] (31)

and proceeding in a similar way yields the following iteration algorithm:

\[ (u_n(x))_n + u_n = \frac{9x^2}{10} + \int_0^x \frac{1}{2} t^2 x^2 u_n(t) \, dt. \] (32)

One may select the initial guess as \( u_0 = 0 \). The successive approximations are

\[ u_0 = 0, \]
\[ u_1 = \frac{9x^2}{10}, \]
\[ u_2 = \frac{99x^2}{100}, \]
\[ u_3 = \frac{999x^2}{1000}. \] (33)

Higher iterations are not given for brevity. In Table 4, some of our iterations are compared with the exact solution, and the errors between the exact solution and \( u_{20} \) are given which are of order \( 10^{-17} \).

4. Conclusion

In this paper, we have applied the newly developed Perturbation Iteration Algorithm PIA(1,1) to some Fredholm
Numerical results show that method PIA(1, 1) is an effective perturbation-iteration technique producing successful analytical results for integral equations.

References


