Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 682537, 5 pages
http://dx.doi.org/10.1155/2013/682537

Research Article

New Perturbation Iteration Solutions for Fredholm and Volterra Integral Equations

İhsan Timuçin Dolapçı,[1] Mehmet Şenol,[2] and Mehmet Pakdemirli[1]

[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

Academic Editor: Livija Cveticanin

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

$$\begin{gathered} y(t) = x(t) + \int_0^1 k(t,s)\, y(s)\, ds, \\ y(t) = x(t) + \int_0^t k(t,s)\, y(s)\, ds \end{gathered} \tag{1}$$

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

$$y(t) = x(t) + \int_0^t k(t,s)\, y(s)\, ds \tag{2}$$

that has the form of

$$F\left(u, \int u, \varepsilon\right) = 0, \tag{3}$$

where

$$F = y(t) - x(t) - \varepsilon \int_0^t k(t,s)\, y(s)\, ds \tag{4}$$

and ε is the artificially introduced perturbation parameter. In this method, we use only one correction term in the perturbation expansion:

$$u_{n+1} = u_n + \varepsilon (u_c)_n. \tag{5}$$

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

$$\begin{aligned} &F\left(u_n, \int u_n, 0\right) + F_u\left(u_n, \int u_n, 0\right)\varepsilon (u_c)_n \\ &\quad + F_\varepsilon\left(u_n, \int u_n, 0\right)\varepsilon \\ &\quad + F_{\int u}\left(u_n, \int u_n, 0\right)\varepsilon \int (u_c)_n = 0 \end{aligned} \tag{6}$$

or

$$(u_c)_n \frac{\partial F}{\partial u} + \left(\int (u_c)_n\right)\frac{\partial F}{\partial\left(\int u\right)} + \frac{\partial F}{\partial \varepsilon} + \frac{F}{\varepsilon} = 0. \tag{7}$$

All derivatives are evaluated at $\varepsilon = 0$.

Starting with the initial condition u_0, first $(u_c)_0$ has been calculated by the help of (7). Then we substitute $(u_c)_0$ 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

$$u(x) = \int_0^1 \left(-\frac{1}{3}e^{2x-(5/3)t}\right) u(t)\, dt + e^{2x+(1/3)} \tag{8}$$

with exact solution

$$u(x) = e^{2x}. \tag{9}$$

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

$$F(u,\varepsilon) = u(x) - e^{2x+(1/3)} - \varepsilon \int_0^1 \left(-\frac{1}{3}e^{2x-(5/3)t}\right) u(t)\, dt, \tag{10}$$

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

$$\begin{aligned} &F = u_n(x) - e^{2x+(1/3)}, \qquad F_u = 1, \\ &F_\varepsilon = -\int_0^1 \left(-\frac{1}{3}e^{2x-(5t/3)} u_n(t)\right) dt, \qquad F_{\int u} = 0. \end{aligned} \tag{11}$$

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

$$e^{1/3+2x} + \int_0^1 -\frac{1}{3}e^{-5t/3+2x} u_n(t)\, dt = (u_c)_n(x) + u_n(x). \tag{12}$$

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

$$u_0 = 1 \tag{13}$$

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

$$\begin{aligned} u_1 &= -\frac{e^{2x}}{5} + \frac{1}{5}e^{-5/3+2x} + e^{1/3+2x}, \\ u_2 &= -\frac{1}{5}e^{-5/3+2x}\left(-1 + e^{1/3} + e^{5/3} - 11e^2 + 5e^{7/3}\right), \\ u_3 &= \frac{1}{5}e^{-(5/3)+2x} \\ &\quad \times \left(1 + e^{1/3}\left(-2 + e^{1/3}\right.\right. \\ &\quad \left.\left.\times\left(1 + e\left(-1 + 17e^{1/3} - 16e^{2/3} + 5e\right)\right)\right)\right). \end{aligned} \tag{14}$$

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:

$$u(x) = \cos x - \int_0^x (x-t)\cos(x-t)\, u(t)\, dt. \tag{15}$$

The exact solution of the problem is

$$u(x) = \frac{1}{3}\left(2\cos\sqrt{3}x + 1\right). \tag{16}$$

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

$$F(u,\varepsilon) = u(x) - \cos x - \varepsilon \int_0^x (x-t)\cos(x-t)\, u(t)\, dt, \tag{17}$$

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

$$\begin{aligned} &F = u_n(x) - \cos x, \qquad F_u = 1, \\ &F_\varepsilon = \int_0^x (x-t)\cos(x-t)\, u_n(t)\, dt, \qquad F_{\int u} = 0. \end{aligned} \tag{18}$$

Equation (7) reduces to

$$\int_0^x (x-t)\cos(t-x)\, u_n dt + (u_c)_n + u_n = \cos x. \tag{19}$$

Table 1: Numerical result of Example 1.

x	u_5	u_{10}	u_{15}	u_{20}	Exact solution	Error (u_{20})
0.0	1.00572	0.99995	1.00000	1.00000	1.00000	5.20782×10^{-9}
0.1	1.22839	1.22134	1.22140	1.22140	1.22140	6.36084×10^{-9}
0.2	1.50035	1.49174	1.49183	1.49182	1.49182	7.76915×10^{-9}
0.3	1.83254	1.82202	1.82212	1.82212	1.82212	9.48926×10^{-9}
0.4	2.23826	2.22542	2.22554	2.22554	2.22554	1.15902×10^{-8}
0.5	2.73382	2.71813	2.71828	2.71828	2.71828	1.41563×10^{-8}
0.6	3.33910	3.31993	3.32012	3.32012	3.32012	1.72906×10^{-8}
0.7	4.07838	4.05498	4.05520	4.05520	4.05520	2.11187×10^{-8}
0.8	4.98135	4.95276	4.95304	4.95303	4.95303	2.57945×10^{-8}
0.9	6.08423	6.04931	6.04965	6.04965	6.04965	3.15055×10^{-8}
1.0	7.43130	7.38865	7.38906	7.38906	7.38906	3.84809×10^{-8}

Table 2: Numerical result of Example 2.

x	u_5	u_{10}	Exact solution	Error (u_{10})
0.0	1	1	1	0
0.1	0.990025	0.990025	0.990025	0
0.2	0.960398	0.960398	0.960398	1.11022×10^{-16}
0.3	0.912007	0.912007	0.912007	0
0.4	0.846298	0.846298	0.846298	1.11022×10^{-16}
0.5	0.765240	0.765240	0.765240	0
0.6	0.671256	0.671256	0.671256	2.22045×10^{-16}
0.7	0.567160	0.567160	0.567160	0
0.8	0.456066	0.456066	0.456066	1.11022×10^{-16}
0.9	0.341300	0.341300	0.341300	5.55112×10^{-17}
1.0	0.226296	0.226296	0.226296	5.55112×10^{-17}

Choosing the initial guess

$$u_0 = 1 \tag{20}$$

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

$$\begin{aligned} u_1 &= 1 - x\sin x, \\ u_2 &= 1 + \frac{1}{12}x\left(-12 + x^2\right)\sin x, \\ u_3 &= 1 + \frac{1}{480}x\left(15x\cos x - \left(495 - 45x^2 + x^4\right)\sin x\right). \end{aligned} \tag{21}$$

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

$$u(x) = e^{3x} - \frac{1}{9}\left(2e^3 + 1\right)x + \int_0^1 xtu(t)\,dt \tag{22}$$

with the exact solution

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

Equation (22) is rewritten in the following form:

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

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

$$\begin{gathered} F = u_n(x) - e^{3x} + \frac{1}{9}\left(2e^3 + 1\right)x, \qquad F_u = 1, \\ F_\varepsilon = -\int_0^1 xtu_n(t)\,dt, \qquad F_{\int u} = 0. \end{gathered} \tag{25}$$

Equation (7) reduces to

$$9\left(e^{3x} + \int_0^1 txu_n dt\right) = x + 2e^3x + 9(u_c)_n + 9u_n. \tag{26}$$

Choosing the initial guess

$$u_0 = 1 \tag{27}$$

Table 3: Numerical result of Example 3.

x	u_5	u_{10}	u_{15}	u_{20}	Exact solution	Error (u_{20})
0	1	1	1	1	1	0
0.1	1.34483	1.34984	1.34986	1.34986	1.34986	2.45400×10^{-9}
0.2	1.81206	1.82208	1.82212	1.82212	1.82212	4.90801×10^{-9}
0.3	2.44451	2.45954	2.45960	2.45960	2.45960	7.36201×10^{-9}
0.4	3.299996	3.32003	3.32011	3.32012	3.32012	9.81602×10^{-9}
0.5	4.45654	4.48159	4.48169	4.48169	4.48169	1.22700×10^{-8}
0.6	6.01947	6.04952	6.04964	6.04965	6.04965	1.47240×10^{-8}
0.7	8.13096	8.16603	8.16617	8.16617	8.16617	1.71780×10^{-8}
0.8	10.98293	11.02301	11.02317	11.02318	11.02318	1.96320×10^{-8}
0.9	14.83446	14.87955	14.87973	14.87973	14.87973	2.20860×10^{-8}
1.0	20.03523	20.08533	20.08553	20.08554	20.08554	2.45400×10^{-8}

Table 4: Numerical result of Example 4.

x	u_5	u_{10}	u_{15}	u_{20}	Exact solution	Error (u_{20})
0.0	0	0	0	0	0	0
0.1	0.0099999	0.00999	0.01000	0.01000	0.01000	0
0.2	0.0399996	0.03999	0.04000	0.04000	0.04000	0
0.3	0.0899991	0.08999	0.09000	0.09000	0.09000	1.38778×10^{-17}
0.4	0.159998	0.15999	0.16000	0.16000	0.16000	0
0.5	0.249998	0.24999	0.25000	0.25	0.25	0
0.6	0.359996	0.35999	0.36000	0.36000	0.36000	5.55112×10^{-17}
0.7	0.489995	0.48999	0.49000	0.49000	0.49000	5.55112×10^{-17}
0.8	0.639994	0.63999	0.64000	0.64000	0.64000	0
0.9	0.809992	0.80999	0.81000	0.81000	0.81	1.11022×10^{-16}
1.0	0.99999	0.99999	1.00000	1.0	1.0	0

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

$$\begin{aligned} u_1 &= e^{3x} + \frac{7x}{18} - \frac{2e^3 x}{9}, \\ u_2 &= e^{3x} - \frac{1}{54}\left(-7 + 4e^3\right)x, \\ u_3 &= e^{3x} - \frac{1}{162}\left(-7 + 4e^3\right)x. \end{aligned} \tag{28}$$

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:

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

The exact solution of the problem is

$$u(x) = x^2. \tag{30}$$

Equation (29) is rewritten in the following form:

$$F(u,\varepsilon) = u(x) - \frac{9}{10}x^2 - \varepsilon \int_0^1 \frac{1}{2}x^2 t^2 u(t)\,dt, \tag{31}$$

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

$$(u_c)_n + u_n = \frac{9x^2}{10} + \int_0^1 \frac{1}{2}t^2 x^2 u_n dt. \tag{32}$$

One may select the initial guess as $u_0 = 0$. The successive approximations are

$$\begin{aligned} u_0 &= 0, \\ u_1 &= \frac{9x^2}{10}, \\ u_2 &= \frac{99x^2}{100}, \\ u_3 &= \frac{999x^2}{1000}. \end{aligned} \tag{33}$$

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 and Volterra type integral equations for the first time. Numerical results show that method PIA$(1, 1)$ is an effective perturbation-iteration technique producing successful analytical results for integral equations.

References

[1] A. H. Nayfeh, *Perturbation Methods*, Pure and Applied Mathematics, John Wiley & Sons, New York, NY, USA, 1973.

[2] D. W. Jordan and P. Smith, *Nonlinear Ordinary Differential Equations*, Oxford Applied Mathematics and Computing Science Series, The Clarendon Press, New York, NY, USA, 2nd edition, 1987.

[3] A. V. Skorokhod, F. C. Hoppensteadt, and H. Salehi, *Random Perturbation Methods with Applications in Science and Engineering*, vol. 150 of *Applied Mathematical Sciences*, Springer, New York, NY, USA, 2002.

[4] J.-H. He, "Iteration perturbation method for strongly nonlinear oscillations," *Journal of Vibration and Control*, vol. 7, no. 5, pp. 631–642, 2001.

[5] R. E. Mickens, "Iteration procedure for determining approximate solutions to nonlinear oscillator equations," *Journal of Sound and Vibration*, vol. 116, no. 1, pp. 185–187, 1987.

[6] R. E. Mickens, "A generalized iteration procedure for calculating approximations to periodic solutions of 'truly nonlinear oscillators'," *Journal of Sound and Vibration*, vol. 287, no. 4-5, pp. 1045–1051, 2005.

[7] R. E. Mickens, "Iteration method solutions for conservative and limit-cycle $x^{1/3}$ force oscillators," *Journal of Sound and Vibration*, vol. 292, no. 3–5, pp. 964–968, 2006.

[8] K. Cooper and R. E. Mickens, "Generalized harmonic balance/numerical method for determining analytical approximations to the periodic solutions of the $x^{4/3}$ potential," *Journal of Sound and Vibration*, vol. 250, no. 5, pp. 951–954, 2002.

[9] H. Hu and Z.-G. Xiong, "Oscillations in an $x^{(2m+2)/(2n+1)}$ potential," *Journal of Sound and Vibration*, vol. 259, no. 4, pp. 977–980, 2003.

[10] J.-H. He, "Homotopy perturbation method with an auxiliary term," *Abstract and Applied Analysis*, Article ID 857612, 7 pages, 2012.

[11] S. Q. Wang and J. H. He, "Nonlinear oscillator with discontinuity by parameter-expansion method," *Chaos, Solitons and Fractals*, vol. 35, no. 4, pp. 688–691, 2008.

[12] S. Iqbal and A. Javed, "Application of optimal homotopy asymptotic method for the analytic solution of singular Lane-Emden type equation," *Applied Mathematics and Computation*, vol. 217, no. 19, pp. 7753–7761, 2011.

[13] S. Iqbal, M. Idrees, A. M. Siddiqui, and A. R. Ansari, "Some solutions of the linear and nonlinear Klein-Gordon equations using the optimal homotopy asymptotic method," *Applied Mathematics and Computation*, vol. 216, no. 10, pp. 2898–2909, 2010.

[14] Y. Aksoy and M. Pakdemirli, "New perturbation-iteration solutions for Bratu-type equations," *Computers & Mathematics with Applications*, vol. 59, no. 8, pp. 2802–2808, 2010.

[15] M. Pakdemirli, Y. Aksoy, and H. Boyacı, "A new perturbation-iteration approach for first order differential equations," *Mathematical & Computational Applications*, vol. 16, no. 4, pp. 890–899, 2011.

[16] Y. Aksoy, M. Pakdemirli, S. Abbasbandy, and H. Boyacı, "New perturbation-iteration solutions for nonlinear heat tranfer equations," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 22, no. 7, 2012.

[17] M. Pakdemirli and H. Boyacı, "Generation of root finding algorithms via perturbation theory and some formulas," *Applied Mathematics and Computation*, vol. 184, no. 2, pp. 783–788, 2007.

[18] M. Pakdemirli, H. Boyacı, and H. A. Yurtsever, "Perturbative derivation and comparisons of root-finding algorithms with fourth order derivatives," *Mathematical & Computational Applications*, vol. 12, no. 2, pp. 117–124, 2007.

[19] M. Pakdemirli, H. Boyacı, and H. A. Yurtsever, "A root-finding algorithm with fifth order derivatives," *Mathematical & Computational Applications*, vol. 13, no. 2, pp. 123–128, 2008.