Journal of Education and Science (ISSN 1812-125X), Vol: 32, No: 01, 2023 (81-90)

Solving Fredholm Integral Equation Based On Padé Approximants Using Particle Swarm Optimization

Israa I. M. Sheekhoo[1*] **, Azzam S. Y. Aladool** [2]

[1*,2] Department of Mathematics, College of Education for Pure Sciences, University of Mosul Mosul , Iraq.

E-mails : [1*] israa.21esp31@student.uomosul.edu.iq, [2]azzam.aladool@uomosul.edu.iq

(Received January 01, 2022; Accepted February 15, 2023; Available online March 01, 2023)

DOI: 10.33899/edusj.2023.137531.1309,

Abstract

Several scientific and engineering applications are usually described as integral equations. A new approach for solving the type of linear and nonlinear Fredholm integral equation of the second kind is proposed. Although many methods provide an analytic solution, there are different types of integral equations are difficult to solve. Therefore, the numerical approach for solving integral equations is used. Fredholm integral equations of the second kind have been converted to unconstrained optimization problems to find their approximate solutions. This work employs particle swarm optimization combined with padé expansion to find an approximate solution of the Fredholm integral equation. This is applied by minimizing the fitness function value. The fitness function is calculated using the discrete least squares weighted function. The proposed algorithm is applied to solve linear and non-linear FIE. The results are compared to exact solutions. The stability of the proposed algorithm is also presented. The results are promising in terms of convergence , stability and accuracy of the approximate solution.

Keywords: Fredholm integral equations, particle swarm optimization, padé approximation

حل معادلة فريدهولم التكاملية بالاعتماد على تقريبات بادي و باستخدام امثلية سرب الجسيمات

إسراء ابراهيم محمد شيخو[1*] **، عزام صلاح الدين يونس العدول**[2]

[1*, 2] قسم الرياضيات، كلية التربية للعلوم الصرفة، جامعة الموصل، الموصل، العراق

المستخلص:

يتم وصف العديد من التطبيقات العلمية والهندسية على شكل معادلات تكاملية. حيث تم اقتراح نهج جديد لحل معادلة فريدهولم التكاملية الخطية والغير الخطية من النوع الثاني. على الرغم من وجود العديد من الطرائق لحل معادلة فريدهولم التكاملية تحليليًا، إلا أن هناك انواع مختلفة من المعادلات التكاملية و التي من الصعب ايجاد حلا لها. لذلك يتم استخدام الطرائق العددية لحل المعادلة التكاملية. تم تحويل معادلات فريدهولم التكاملية من النوع الثاني إلى مسائل امثلية غيرمقيدة للحصول على حلول تقريبية لها. في هذا البحث تم استخدام امثلية سرب الجسيمات مع توسيع بادي لإيجاد حل تقريبي لمعادلة فريدهولم التكاملية. يتم تطبيق ذلك عن طريق تقليل قيمة دالة اللياقة و يتم حساب دالة اللياقة باستخدام دالة ترجيح المربعات الصغرى المتقطعة. تم تطبيق الخوارزمية المقترحة لحل معادلة فريدهولم التكاملية

الخطية وغير الخطية. تم مقارنة النتائج بالحلول الدقيقة. كما تم ايضا عرض استقرار الخوارزمية المقترحة. اظهرت هذه الدراسة نتائج جيدة من حيث تقارب الحل التقريبي واستقراره ودقته.

الكلمات المفتاحية: معادلات فريدهولم التكاملية ، امثلية سرب الجسيمات ، تقريب بادي

1.INTRODUCTION

Equations with an unknown function that have the integral sign are known as integral equations [1]. The literature on integral equations and their applications is extensive since integral equations are one of the key instruments in many fields of applied mathematics, physics, and engineering [2]. Additionally, these equations can be reformulated from other mathematical issues such as ordinary differential equations and partial differential equations, As a result understanding integral equations and how to solve them is extremely useful in applications [3].

It is increasingly common for engineers and mathematicians to use numerical simulations to model phenomena, especially when analytical solutions are unavailable[4]. One of the oldest problems in applied mathematics is to find numerical solutions to Fredholm integral equations of the second kind[5]. There are many methods to solve Fredholm integral equations of the second kind including[6] homotopy perturbation method(HPM)[7]. Modified Neumann series[8]. Adomian Decomposition method[9]. Taylor series method[1], Monte Carlo method[10].Space kernel approximation methods [11].Quadrature method[12]. q-homotopy analysis method[5]. Numerical solutions of integral equations are often complicated and require extensive arithmetic Operations[13].

Recently, researchers are interested to solve some mathematical problems by using intelligence algorithms such as Genetic Algorithm(GA), Neural Networks (NN), particle swarm optimization (PSO)[14]. [15]. Optimizing refers to the process of determining the most optimal solution for a given problem [16]. It appears that converting some mathematical problems into optimization problems offered interesting results for finding approximate solutions to some problems[17]. Algorithms that use intelligent processes have been successfully applied in solving ordinary differential equations[18] , systems of ordinary differential equations[19].

Many optimization techniques are improved to solve mathematical problems iteratively based on a given measurement of quality [20]. Ordinary differential equations (ODEs) and systems of ordinary differential equations(SODEs) are solved by applying intelligent algorithms based on function expansions [14]. Evolution Strategies and even expansions of the Fourier series are utilized to solve initial and boundary value problems [21]. A particle Swarms Optimization(PSO) based on the Fourier series is used to solve nonlinear ordinary differential equations and integral equations [14]. Accurate results are obtained by converting ordinary differential equations into constrained optimization problems [22]. In this study Fredholm integral equation of the second kind (FIE) is solved using Padé expansion. it is possible to use [-1,1] as a search space by Padé expansion which is fraction expansion. To obtain a highly accurate Padé approximation, a small number of variables is typically required[19]. This paper aims to demonstrate that particle swarm optimization(PSO) algorithm can also be adapted to find an approximate solution of the Fredholm integral equation of the second kind based on the Padé expansion.

In this work, different examples of linear and nonlinear Fredholm integral equations are discussed. After this introduction work principles are presented in the next section, followed by the methodology for solving ODEs. The examples are then presented along with the settings, and the results are shown. Then conclusions are drawn.

2.PRELIMINARY

In this section, an explanation of the Fredholm integral equation of the second kind is presented. The fundamentals of particle swarm optimization algorithms are also discussed. Afterward, the PSO algorithm is described.

2.1Fredholm integral equations of the second kind(FIEs)

The Fredholm integral equations (FIEs) of the second kind are considered:

$$u(x) = f(x) + \lambda \int_a^b k\big(x,t,u(t)\big)\, dt \quad a \le x \le b \tag{1}$$

Where $\lambda \in R$, k is the kernel , f is the known function and u is the unknown function [12].

2.2Particle swarm optimization

Kennedy and Eberhart presented particle swarm optimization (PSO) in 1995[23]. A particle swarm optimization (PSO) algorithm is a population-based stochastic optimization algorithm inspired by the collective behavior of animals such as birds and fish [24]. There is a velocity and a position for each member of the swarm (called particle), particles adjust their positions in the search space based on time by two equations as in the references [23][25].

$$\delta_i(t+1) = \omega \times \delta_i(t) + c_1 \times rand_1 \times \big(\sigma_i(t) - \gamma_i(t)\big) + c_2 \times rand_2 \times \big(\lambda(t) - \gamma_i(t)\big) \tag{2}$$

$$\gamma_i(t+1) = \gamma_i(t) + \delta_i(t+1) \tag{3}$$

In this case, γ_i represents the position of the i^{th}particle, and δ_i represents its velocity. There is a previous best particle referred to as $\sigma_i(t)$ and a global best particle referred to as $\lambda(t)$, within the interval $[0,1]$, rand1 and rand2 represent random vectors, ω represents inertia, it is important to note that c_1 and c_2 are positive constants, referred to as ‘’acceleration coefficients”, each velocity vector is clamped within the range $[varmax\,, varmin]$ decrease the probability that a particle leaves search space, and $t = 1,2,3 \dots . Maxit$, represents the number of iterations, and ($Maxit$) is the max iteration. The number of populations in a swarm is called $nPop$.

A particle swarm optimization (PSO) algorithm can search in N dimensions depending on the number of variables $nVar$, the values of c_1 and c_2 and ω are determined as follows:[22]

$$\chi = \begin{cases} \dfrac{2k}{\left|2-\vartheta-\sqrt{\vartheta^2-4\vartheta}\right|} & if\ \vartheta > 4 \\ k & if\ \vartheta \le 4 \end{cases} \tag{4}$$

and

$$\vartheta = \vartheta_1 + \vartheta_2\,, \quad c_1 = \vartheta_1 \times \chi \quad , c_2 = \vartheta_2 \times \chi \quad ,w = \chi \tag{5}$$

Then $k \in [0,1]$, and the values ϑ_1 , ϑ_2 are chosen randomly to achieve $\vartheta > 4$.

Using the following parameter ε, the non-converging behavior of PSO can be avoided by continuously damping the velocity in each iteration.

$$w(t+1) = \varepsilon \times w(t) \tag{6}$$

We can obtain the velocity space by following these steps:

$$Max\ Velocity = \alpha \times (VarMax - VarMin)\,,$$
$$Min\ Velocity = -Max\ Velocit\ y \tag{7}$$

and α is a positive parameter.

2.3Algorithms of PSO

PSO consists the following steps:[26][27]

i) Parameters and inputs are initialized, $\vartheta_1, \vartheta_2, k\ , \varepsilon\ \ , \alpha\ , VarMax\ , VarMin\ , nVar\ , Maxit\ and\ nPop$
ii) Let the initial velocity is equal to zero.
iii) Initializing an arbitrary particles' position in the search space.
iv) The fitness function of the swarm particles is evaluated
v) Find the global optimal position (gbest)of the particle swarm, and the particles' optimal position (pbest_i),where $i = 1,2, \dots N$.
vi) The position and velocity of particles are updated according to Eq. (2) and Eq. (3).
vii) Go to step (iv) if the number of iterations is less than $Maxit$.

3.METHODOLOGY

An approach for finding the approximate solution of FIE is presented. This work starts by defining the expansion function. FIE is also converted into unconstrained optimization problems. In addition, the fitness function and the discrete least squares weighted function. Lastly, the algorithm for obtaining an approximate FIE solution is presented.

3.1.The Padé expansion approximation

A systematic method for determining the optimal Padé Degree for a given problem that may be not studied [28]. Consider the following expression of Padé expansion to approximate the solution of FIE:[29].

$$u(x) \cong U_{apprx}(x) = \frac{\mathrm{f}(x)}{g(x)} = \frac{\sum_{m=0}^{n_1} p_m x^m}{\sum_{m=0}^{n_2} q_m x^m} \qquad (8\ )$$

Where $x \in I = [x_0, x_n]$, $n_1 + n_2 = nVar$ and p_m , q_m are real coefficients which belong to the search space$[VarMin, Var\ Max]$, $U_{apprx}(x)$,$u(x)$ are the approximate solution and the exact solution respectively, $g(x) \neq 0\ \forall\ x \in I$.

3.2.Convert FIE into an unconstrained optimization problem

To convert Fredholm integral equations of the second kind into an unconstrained optimization problems, suppose that $U(x)$ is an approximate solution to (FIE)and is substituted for Eq(9) yields:

$$Er(x) = \left| u(x) - f(x) - \lambda \int_a^b k\big(x, t, u(t)\big) dt \right| \qquad (9)$$

An optimal solution to the FIE can be obtained when $E_r(x)$ near to zero. It is necessary to use a quantitative criterion that determines the accuracy of the approximate solution to reduce $Er(x)$. This can be minimized using a discrete least squares weighted function[19].

3.2.1.The Discrete Least Squares Weighted Function (DLSWF)

To compute the discrete least squares weighted function, the following steps are followed:[19][22]

Taking the interval I and dividing it into N points $\{x_0 = a\ , x_1, x_2, x_3, \dots., x_n = b\}$, where $x_k = x_0 + hk\ \ , \forall\ k = 0,1,2,3, \dots. n\ and\ h > 0$, and

$$\mathrm{DLSWF} = \sqrt{\frac{\sum_{k=1}^{N} (E_r(x_k))^2}{N}} \qquad (10)$$

3.3.FIE -PSO Algorithm

Here is an outline of the proposed algorithm for solving FIE:

Step(1): Create an array of Padé expansion coefficients $nVar$ as follows: $[p_0, p_1, \dots\dots, p_{n_1}, q_0, q_1, \dots\dots q_{n_2}]$.
Step(2): Convert the FIE into the implicit form as in Eq(9).
Step(3): The fitness function is determined as follows:

$$FITFUN = DLSWF \quad (11)$$

Step(4): To determine Padé coefficients, PSO parameters are initialized.
Step(5): A PSO algorithm should be applied to minimize fitness.
Step(6): Do step(5) until $FITFUN < TOL$ or the maximum number of iterations is reached.
An evaluation of the algorithm is performed by calculating the Mean Absolute Error (MAE) using the approximate solution $U_{app}(x)$ and exact solution u:

$$MAE = \frac{\sum_{k=1}^{N}\left|u(x_k) - U_{app}(x_k)\right|}{N} \quad (12)$$

4.NUMERICAL RESULTS:

A PSO algorithm combined with a Padé approximant is presented in this section to approximate the solution of FIE. Additionally, the algorithm's convergence and stability are explained.

4.1.Numerical Examples

In this paper , linear and non- linear (FIE) are included in **TABLE1:**[11] [30][31][5] [32]**:**

TABLE 1: Shows different examples of FIE, with their exact solutions.LFIE and NLFIE are denoted linear and non-linear Fredholm integral equations of the second kind respectively.

Examples	FIE	Exact Solutions
LFIE1	$u(x) = -\frac{2}{\pi}\cos(x) + \frac{4}{\pi}\int_0^{\frac{\pi}{2}} \cos(x-t)\, u(t)dt$	$u(x) = \sin x$
LFIE2	$u(x) = \frac{\pi}{2}x - x + x\tan^{-1}x - \int_{-1}^{1} xu(t)dt$	$u(x) = x\tan^{-1}x$
LFIE3	$u(x) = -\frac{1}{2} + sec^2x + \frac{1}{2}\int_0^{\frac{\pi}{4}} u(t)dt$	$u(x) = sec^2x$
NLFIE 4	$u(x) = e^x - \frac{(1+2e^3)x}{9} + \int_0^1 xtu^3(t)dt$	$u(x) = e^x$
NLFIE5	$u(x) = \frac{7}{8}x + \frac{1}{2}\int_0^1 xtu^2(t)dt$	$u(x) = x$
NLFIE6	$u(x) = 1 + x + \left(1 - \frac{3}{2}\ln(3) + \frac{\sqrt{3}}{6}\pi\right)x^2 + \int_0^1 2x^2\, tln\big(u(t)\big)dt$	$u(x) = x^2 + x + 1$

4.2.Setup of the program

TABLE2 shows examples of parameter values. After running the algorithm ten times, its reliability is verified. To implement the algorithm, Matlab R2020a software is used, There is a type of computer Lenovo laptop, Intel (R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz with RAM 4.00 GB, Windows 10 Pro and system type 64-bit.

TABLE2: The following parameters are used in all examples:

Parameter	ϑ_1	ϑ_2	k	ε	α	h	TOL	$Maxit$	$npop$
Value	2.05	2.05	1	0.8	0.4	0.01	1e-10	300	300

4.3.FIE Solutions

As a result of the PSO algorithm, the coefficients for approximate solutions of FIE for all examples are listed in TABLE 3. For example, the best approximated solution to equation (13) can be found by LFIE1 (u) is shown, where $nVar = 20$, and $x \epsilon [0,1]$ as follows:

$$u(x) \cong \frac{0.000561 + x + x^2 - 0.13998\, x^3 + 0.923627\, x^4 - 0.36192 x^5 - x^6 + 0.828749 x^7 - 0.1865 x^8 + x^9}{1 + x + 0.257652 x^2 + 0.218963\, x^3 + 0.098654\, x^4 + 0.112121 x^5 + 0.343289 x^6 - 0.49893 x^7 + 0.109897 x^8 + x^9} \quad (13)$$

TABLE 3: Displays variables based on their values using by FIE - PSO algorithm.
The number of variables, $nVar = 20$, $[Var\ max, Var\ \min] = [1, -1]$.

		Examples					
	$coff.$	LFIE1	LFIE2	LFIE3	NLFIE4	NLFIE5	NLFIE6
$m = 0$	p_m	0.000561	-0.00166	-0.99964	0.234767	0.000115	0.594451
	q_m	1	-0.99994	-1	0.23349	0.816768	0.595102
$m = 1$	p_m	1	0.012421	-0.35876	0.404484	0.8083	0.762303
	q_m	1	-0.78956	-0.29384	0.16751	0.609462	0.153907
$m = 2$	p_m	1	-0.94264	-1	0.213982	0.686411	0.999954
	q_m	0.257652	-0.16965	-0.55942	0.093998	0.99996	0.245754
$m = 3$	p_m	-0.13998	-0.96643	-0.95151	0.756128	0.764937	0.13353
	q_m	0.218963	-0.38035	0.999977	0.12931	0.253347	0.000274
$m = 4$	p_m	0.923627	-0.16219	0.121869	0.532422	0.501501	1
	q_m	0.098654	-0.35364	-0.76838	0.330257	1	0.380442
$m = 5$	p_m	-0.36192	0.535447	-0.38403	0.373091	1	0.80171
	q_m	0.098654	1	0.281965	0.163921	0.977373	0.016396
$m = 6$	p_m	-1	0.138245	-0.99335	0.106877	0.999994	0.233356
	q_m	0.343289	-0.65668	-0.66001	0.097573	0.966248	0.454073
$m = 7$	p_m	0.828749	0.255614	-0.99796	0.344359	0.704551	0.679155
	q_m	-0.49893	0.373171	-0.84193	0.134048	0.836949	0.473395
$m = 8$	p_m	-0.1865	-0.92336	0.01119	0.360563	0.768608	0.999999
	q_m	0.109897	-0.22828	0.898963	0.032124	0.491111	0
$m = 9$	p_m	1	-0.04433	-0.67931	0.743734	0.952863	0.972592
	q_m	1	-0.46648	0.171335	0.112907	0.2361	0.07274

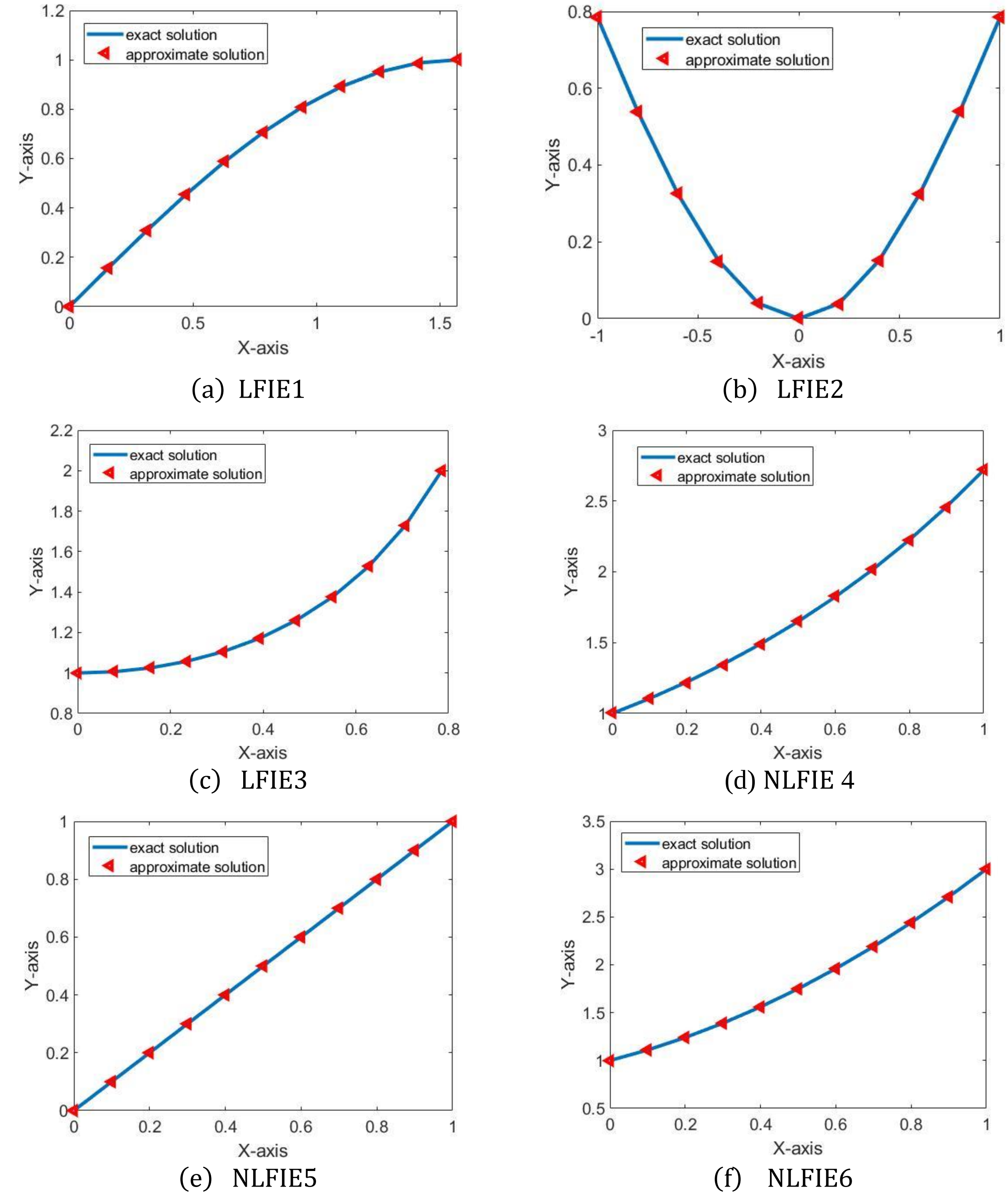

Figure 1: Exact and approximate solutions are compared

The exact solutions to several examples are shown in Figure 1. Within the defined domain, the approximate solutions in Figure 1 were consistent with the exact solution. Hence, the FIE -PSO algorithm can solve linear and non-linear FIE.

The algorithm may reach the exact solution. However, the approach frequently leaves an error rate derived from the approximation. The value of TOL and calculated by considering the value of Mean Absolute Error (MAE) shown in TABLE 4.

TABLE 4: Indicates the Mean Absolute Error between the approximate and exact solutions

Examples	MAE
LFIE1	9.43E-04
LFIE2	1.83E-03
LFIE3	6.60E-04
NLFIE4	0.00418301
NLFIE5	6.00E-05
NLFIE6	6.68E-04

As a result of this method, convergence occurs rapidly.FIGURE 2 shows the convergence of the FIE-PSO algorithm over 300 iterations for all examples, acceptable solutions are obtained with less than 100 iterations and are stable to 300 iterations (see FIGURE 2). Display the convergence of the algorithm

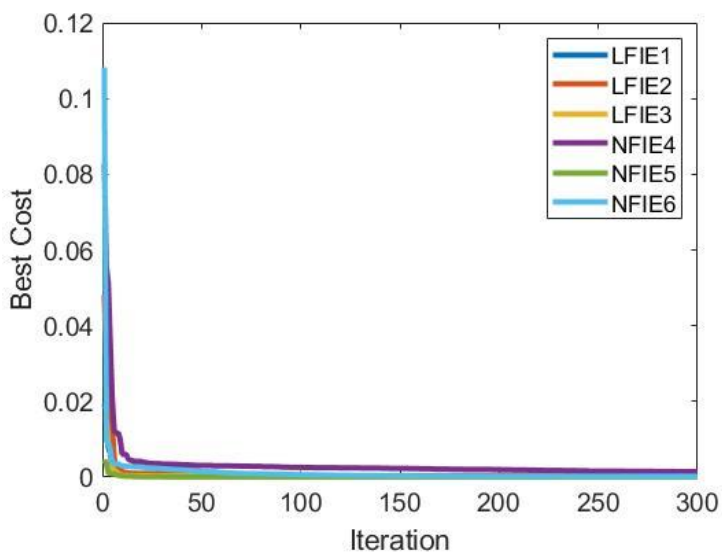

Figure 2: Display the convergence of the algorithm

5.CONCLUSIONS

The particle swarm algorithm is practical for solving Fredholm integral equations of the second kind. FIE has been converted into an unconstrained optimization problem by using approximate expansions. POS algorithm was used with fitness function to find an approximate solution to linear and non-linear Fredholm integral equations of the second kind. To describe approximate solutions, the proposed algorithm used the fractions expansion as an approximation based on the Padé approximation. By the use of the proposed FIE-PSO algorithm, approximate solutions are found using Padé approximants. Based on the results, the algorithm successfully solves linear and non-linear Fredholm integral equations of the second kind. There are many advantages to using this approach, including accurate convergence, stability, and accuracy. As a result, this method is recommended for solving Volterra integral equations.

6.ACKNOWLEDGMENT

It is the authors' pleasure to express their gratitude to both the technical and academic staff of the University of Mosul as the College of Education for pure Science. This work would not have been possible without their support.

7.REFERENCES

[1] H. H. Hameed, H. M. Abbas, and Z. A. Mohammed, "Taylor Series Method for Solving Linear Fredholm Integral Equation of Second Kind Using MATLAB," *J. Babylon Univ.*, vol. 19, no. 1, pp. 14–23, 2011.

[2] Q. Wang, K. Wang, and S. Chen, "Least squares approximation method for the solution of Volterra–Fredholm integral equations," *J. Comput. Appl. Math.*, vol. 272, pp. 141–147, Dec. 2014, doi: 10.1016/j.cam.2014.05.010.

[3] M. M. Rahman, M. A. Hakim, M. K. Hasan, M. K. Alam, and L. N. Ali, "Numerical Solutions of Volterra Integral Equations of Second kind with the help of Chebyshev Polynomials," *Ann. Pure Appl. Math.*, vol. 1, no. 2, pp. 158–167, 2012.

[4] A. A.M. and A. M. Muhammad, "Cubic Spline for SolvingVolterra Integral Equation with Delay," *J. Misan Res.*, vol. 14, no. 27–4, pp. 470–480, 2018.

[5] R. K. Jbr and A. Al-Rammahi, "Q-homotopy analysis method for solving nonlinear fredholm integral equation of the second kind," *Int. J. Nonlinear Anal. Appl.*, vol. 12, no. 2, pp. 2145–2152, 2021, doi: 10.22075/ijnaa.2021.5355.

[6] Z. Mahmoodi, J. Rashidinia, and E. Babolian, "Spline Collocation for Nonlinear Fredholm Integral Equations," *Int. J. Math. Model. Comput.*, vol. 1, no. 1, pp. 69–75, 2011.

[7] J. Biazar and H. Ghazvini, "Numerical solution for special non-linear Fredholm integral equation by HPM," *Appl. Math. Comput.*, vol. 195, no. 2, pp. 681–687, Feb. 2008, doi: 10.1016/j.amc.2007.05.015.

[8] M. Amirfakhrian and S. M. Mirzaei, "Modified Neumann series for solving Fredholm integral equation," *Appl. Math. Sci.*, vol. 3, no. 13–16, pp. 623–628, 2009.

[9] A. K. Hussain, "Approximation Solution of Fredholm Integral Equation Using Adomian Decomposition Method," *J. Madenat Alelem Univ. Coll.*, vol. 2, no. 1, pp. 13–22, 2010.

[10] H. ZhiMin, Y. ZaiZai, and C. JianRui, "Monte Carlo Method for Solving the Fredholm Integral Equations of the Second Kind," *Transp. Theory Stat. Phys.*, vol. 41, no. 7, pp. 513–528, Dec. 2012, doi: 10.1080/00411450.2012.695317.

[11] A. A. Mohammed, "The Numerical Solution of Fredholm integral Equations of the Second Kind," *Int. J. Res.*, vol. 3, no. 17, pp. 1332–1354, 2016, doi: 10.1137/0704029.

[12] R. Katani, "Numerical solution of the Fredholm integral equations with a quadrature method," *SeMA J.*, vol. 76, no. 2, pp. 271–276, 2019, doi: 10.1007/s40324-018-0175-z.

[13] M. H. A. Sathar, A. F. N. Rasedee, A. A. Ahmedov, and N. Bachok, "Numerical solution of nonlinear fredholm and volterra integrals by Newton–Kantorovich and haar wavelets methods," *Symmetry (Basel).*, vol. 12, no. 12, pp. 1–13, 2020, doi: 10.3390/sym12122034.

[14] M. Babaei, "A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization," *Appl. Soft Comput.*, vol. 13, no. 7, pp. 3354–3365, Jul. 2013, doi: 10.1016/j.asoc.2013.02.005.

[15] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Oct. 2007, doi: 10.1007/s11721-007-0002-0.

[16] S. Sivanandam, SN and Deepa, SN and Sivanandam, SN and Deepa, *Introduction to Genetic Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. doi: doi.org/10.1007/978-3-540-73190-0_2.

[17] A. H. Borzabadi and O. S. Fard, "Approximate Solution of Nonlinear Fredholm Integral Equations of the First Kind via Converting to Optimization Problems," *World Acad. Sci. Eng. Technol.*, vol. 1, no. 9, pp. 459–462, 2007.

[18] D. A. Diver, "Applications of genetic algorithms to the solution of ordinary differential equations," *J. Phys. A. Math. Gen.*, vol. 26, no. 14, p. 3503, 1993.

[19] R. H. S. Alazzawi and A. S. Y. Aladool, "Solving Systems of Ordinary Differential Equations Using Particle Swarm Optimization Based on Padé Approximant" to appear in AlP Conference Proceedings(ISSN: 0094-243X,1551-7616),2023.

[20] C. Annie, *Padé Approximants for Operators:theory and applications*, 1st ed. Springer Berlin, Heidelberg, 2006. doi: https://doi.org/10.1007/BFb0099706.

[21] A. Akyüz-Daşcıoğlu and H. Çerdik Yaslan, "An approximation method for the solution of nonlinear integral equations," *Appl. Math. Comput.*, vol. 174, no. 1, pp. 619–629, Mar. 2006, doi: 10.1016/j.amc.2005.04.108.

[22] A. T. Abed and A. S. Y. Aladool, "Applying particle swarm optimization based on Padé approximant to solve ordinary differential equation," *Numer. Algebr. Control Optim.*, vol. 12, no. 2, p. 321, 2022, doi: 10.3934/naco.2021008.

[23] A. Sahu, S. K. Panigrahi, and S. Pattnaik, "Fast Convergence Particle Swarm Optimization for Functions Optimization," *Procedia Technol.*, vol. 4, pp. 319–324, 2012, doi: 10.1016/j.protcy.2012.05.048.

[24] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm : an overview," *Soft Comput.*, vol. 22, no. 2, pp. 387–408, 2018, doi: 10.1007/s00500-016-2474-6.

[25] N. Taher, A. Babak, J. Olamaei, and Ai. Arefi, "An efficient hybrid evolutionary optimization algorithm based on PSO and SA for clustering," *J. Zhejiang Univ. Sci. A*, vol. 10, no. 4, pp. 512–519, 2009, doi: 10.1631/jzus.A0820196.

[26] D. Yong, W. Chuansheng, and G. Haimin, "Particle Swarm Optimization Algorithm with Adaptive Chaos Perturbation," *Cybern. Inf. Technol.*, vol. 15, no. 6, pp. 70–80, 2015, doi: 10.1515/cait-2015-0068.

[27] S. Shabir and R. Singla, "A Comparative Study of Particle Swarm Optimization and Genetic Algorithm," *Int. J. Electr. Eng.*, vol. 9, no. 2, pp. 215–223, Nov. 2016, doi: 10.48161/qaj.v1n1a7.

[28] H. Vazquez-Leal *et al.*, "Direct application of Padé approximant for solving nonlinear differential equations," *Springerplus*, vol. 3, no. 1–11, p. 563, Dec. 2014, doi: 10.1186/2193-1801-3-563.

[29] M. Thiagarajan and K. Senthilkumar, "DTM-Pade Approximants for MHD Flow with Suction/Blowing," *J. Appl. Fluid Mech.*, vol. 6, no. 4, pp. 537–543, Oct. 2013, doi: 10.36884/jafm.6.04.21211.

[30] A.-M. Wazwaz, *Linear and Nonlinear Integral Equations*, Edition, 1. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-21449-3.

[31] J. A. Othman and S. kareem Rabiha, "Solving Nonlinear Fredholm Integral Equation of the Second Type via Solving Nonlinear Fredholm Integral Equation of the Second Type via Nonlinear Programming Techniques," *Int J Appl Mat*, vol. 29, no. 1, pp. 1258–1262, 2014.

[32] N. Karamollahi, M. Heydari, and G. B. Loghmani, "Approximate solution of nonlinear Fredholm integral equations of the second kind using a class of Hermite interpolation polynomials," *Math. Comput. Simul.*, vol. 187, pp. 414–432, 2021, doi: 10.1016/j.matcom.2021.03.015.