ISSN 2348 – 8034
Impact Factor- 3.155

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

USING ARTIFICIAL NEURAL NETWORK FOR SOLVING TWO-DIMENSIONAL INTEGRAL EQUATIONS

Aseel Hisham Khamas Al-Akaedee
Department of Clinical Laboratory Sciences, College of Pharmacy, Al-Mustansiriyah University, Baghdad, Iraq.

ABSTRACT

This paper proposed artificialneural networks to find the solution of linear two-dimensional Fredholm integral equation. The three-layer network which has been used successfully to solve this problem.The proposed neural network can get a real input vector and calculates its corresponding output vector. In this method, a back propagation is used to train the network. Where the Levenberg-Marquardt algorithm used for adjusting the connection weights.The method is illustrated by several examples with computer simulations.

Keywords- *Artificial neural networks,back propagation,Levenberg-Marquardt algorithm, Fredholm integralequation, Two-dimensional integral equation.*

I. INTRODUCTION

Since many mathematical formulations of physical phenomena containintegral equations and these equations are very useful for solving manyproblems in several applied fields like mathematical physics and engineering, therefore various approaches for solving these problems havebeen proposed. First time, Taylor expansion approach was presentedfor solution of integral equations by Kanwal and Liu in [1] and then hasbeen extended in [2], [3-6]. Also variational iteration method[7]and Adomian decomposition method [8] are effective and convenientfor solving integral equations. The homotopy analysis method (HAM) was proposed by Liao [9- 12] and then has been applied in [8], [13, 14].Hadizadeh and Asgary [15]using the bivariate Chebyshev collocation method solvedthe linear Volterra–Fredholm integral equations of thesecond kind. Alipanah and Esmaeili [16] approximated thesolution of the two-dimensional Fredholm integral equationusing Gaussian radial basis function based on Legendre–Gauss–Lobatto nodes and weights. Two-dimensionalorthogonal triangular functions are used in [17] as a newset of basis functions to approximate solutions of nonlineartwo-dimensional integral equations. Babolian et al. [18]applied two-dimensional rationalized Haar functions forfinding the numerical solution of nonlinear second kindtwo-dimensional integral equations. They reduced thepresent problem to solve a nonlinear system of algebraicequations using bivariate collocation method and Newton–Cotes nodes. Moreover, some different valid methods forsolving these kind of equations have been developed.

This paper focuses on constructing a new algorithm by the use of neural networks to reach anapproximate solution of the linear two-dimensional Fredholmintegral equation. For this purpose, first unknown twovariablefunction in the problem is replaced by a three-layerneural network. This architecture ofArtificialneural networks (ANN) can calculate the output corresponding toinput vector. Now the error function to be minimized is definedon the set points. Consequently, the suggested ANNusing a training algorithm that is based onLevenberg-Marquardt algorithmis used to adjust parameters (the weights and biases)to any desired degree of accuracy.

II. PRELIMINARIES

In this section we will introduce the basic definitions andintroductory concepts in integral equations. In addition thebasic principles of ANNapproach are presented and reviewed for solving linearsecond kind two-dimensional integral equations (2D-IEs).

2.1. Integral equations

Integral equations appear in many scientific and engineeringapplications, especially when initial value problems forboundary value problems are converted to integral equations.As stated before, we will introduce the definition of linear two-dimensional integral equations ofthe second kind.

Definition 2.2 [11]

The linear two-dimensional Fredholmintegral equation (2D-FIE) of the second kind is presentedby the form:

ISSN 2348 – 8034
Impact Factor- 3.155

$$F(x,y) = f(x,y) + \lambda \int_c^d \int_a^b k(x,y,s,t)F(s,t)\,\mathrm{d}s\mathrm{d}t, \qquad (1)$$
$$(x,y) \in [a,b] \times [c,d]$$

Where λ is a constant parameter, the kernel k and f are givenanalytic functions on $L^2([a, b]\times[c, d])$. The two-variableunknown function F that must be determined appearsinside and outside the integral signs. This is a second kind integral equation.
It is important topoint out that if the unknown function appears only insidethe integral signs, the resulting equation is of first kind.

Notice that, if the function f(x, y) in the present integralequations is identically zero, the equation is called homogeneous.Otherwise it is called inhomogeneous. These threeconcepts play a major role in the structure of the solution.

III. ARTIFICIAL NEURAL NETWORKS

Artificial neural networks (ANNs) can be considered assimplified computational structures that are inspired byobserved process in natural networks of biological neuronsin the brain. They are nonlinear mapping architecturesbased on the function of the human brain, therefore can beconsidered as powerful tools for modeling, especially whenthe underlying data relationship is unknown. In other words, in contrast to conventionalmethods, which are used to perform specific task,most ANNs are more versatile. This feature raisesa very appealing computational model which can beapplied to solve variety of problems [19].

IV. ARCHITECT OF THE STRUCTURE NETWORK

In this section we will explain how this approach can be used to find the approximate solution of the linear two-dimensional Fredholm integral equation (2D-FIE) of the second kind(equation (1)).We suggest three layer ANN: input layer consist two input nodes, hidden layer consist nine hidden nodeswith tansig. transfer function and output layer consist one output node with linear transfer function, see Figure (1). The suggested networkis doing as the following:

Input nodes:

The input nodes make no change in their inputs, so:

$O_1 = x$

$O_2 = y$

Hidden nodes:

Input into a node in hidden layer is a weighted sum ofoutputs from nodes connected to it. Each unit takes its netinput and applies antransfer function to it. The input/output relation is normally given as follows:

$O_p = g(\text{ net } (p))$;

$$net(p) = \sum_{i=1}^{2} (w_{pi}.o_i) + b_p, \quad p = 1, \ldots, N.$$

where net(p) describes the result of the net outputs $o_i$impacting on unit p. Also, $w_{pi}$are weights connectingneuron i to neuron p and b_p is a bias for neuron p. Bias termis baseline input to a node in absence of any other inputs.

Output node:

$$u_N(x,y) = \sum_{p=1}^{N} net(p).$$

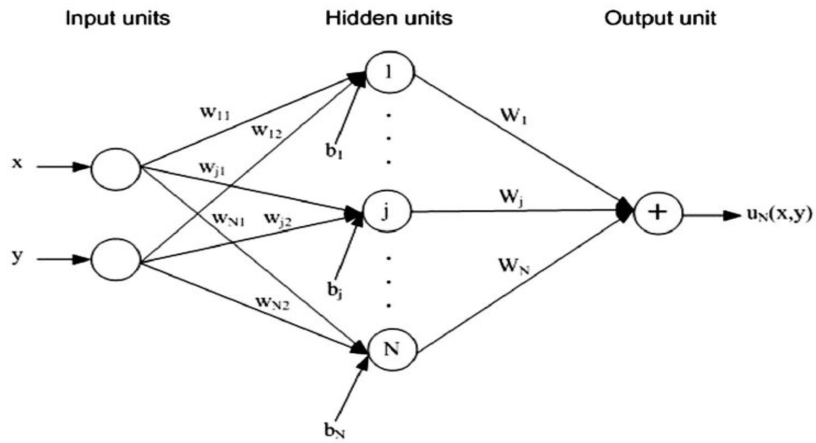

Figure 1: Architect of suggested ANN

V. TRAINING SUGGESTED NETWORK

The suggested ANN is trained by backpropagationalgorithm that is based on supervised procedure.In other words, the network is trained using asupervised training algorithm which uses the training datato adjust the network weights and biases. Now let $w_{p,q}$,$w_p$and b_p (for p = 1, . . ., N; q = 1, 2 (are initialized at smallrandom values for input signals. For parameter $w_{p,q}$adjustment rule can be written as follows**:**

Where r is the number of adjustments, η is the learning rate. Similarly thisadjustment rule can be written for other weight parameters.

Training algorithm

Step 1: Let> 0, and Emax> 0 are chosen. Then crisp quantitieswp,q, w_p, and b_p (for p = 1, ...,N; q= 1,2) are initialized at random values.

Step 2: Let r := 0 where r is the number of iterations of the training algorithm. Then the running error E is set to 0.
Step 3: Let r := r + 1. Then,

i) Forward calculation: Calculate the output vector $u_N(x_i, y_j)$ by presenting the input vectors w, x_i, and y_j.
ii) Back-propagation: Adjust crisp parameter wp,q, w_p and $b_p$using the error function
Step 4: Cumulative cycle error is computed by adding the present error to E.
Step 5: The training cycle is completed. For E <Emax terminate the training session. If E >Emax then E is set to 0 and we initiate a new training cycle by going back to Step 3.

VI. EXAMPLE

This section contain example of linear two-dimensional Fredholm integral equations of second kind. In this example, we illustrate the use of suggested ANN to approximate the solutions of the given integral equation. Where the computed values of the approximatesolution are calculated over a number of epoch and the error function is plotted. Also, to show the efficiency of thepresent method for our problem, results will be comparedwith the exact solution.
In the following simulations, we use the specifications as follows: Learning constant = 0.1, epoch = 10000, and stopping conditions Emax<0.0001.
Consider the linear 2D-FIE:

With the exact solution F(x, y) = x.cosy**.**

The suggest network trained using a grid of ten equidistant points in [0, 1]. Figure (1) display the neural solution. Figure (2) illustrate the accuracy of solution. Table (1) gave the exact and neural solution with Levenberg – Marquardt training algorithms (trainlm), Table (2) gave the performance of the train for epoch and time, Table (3) gave the initial weight and bias of the designer network. Also, to show the efficiency of the present method for our problem, results will be compared with the other method such shifted Legendre collocationmethod(ShLCM) and given in Table (1).

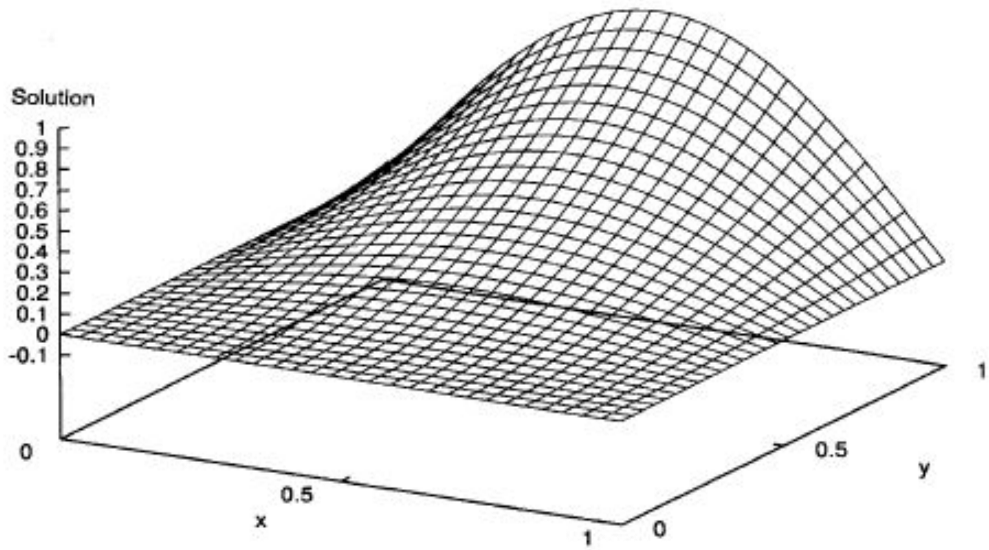

Figure 1:ANN solution for example

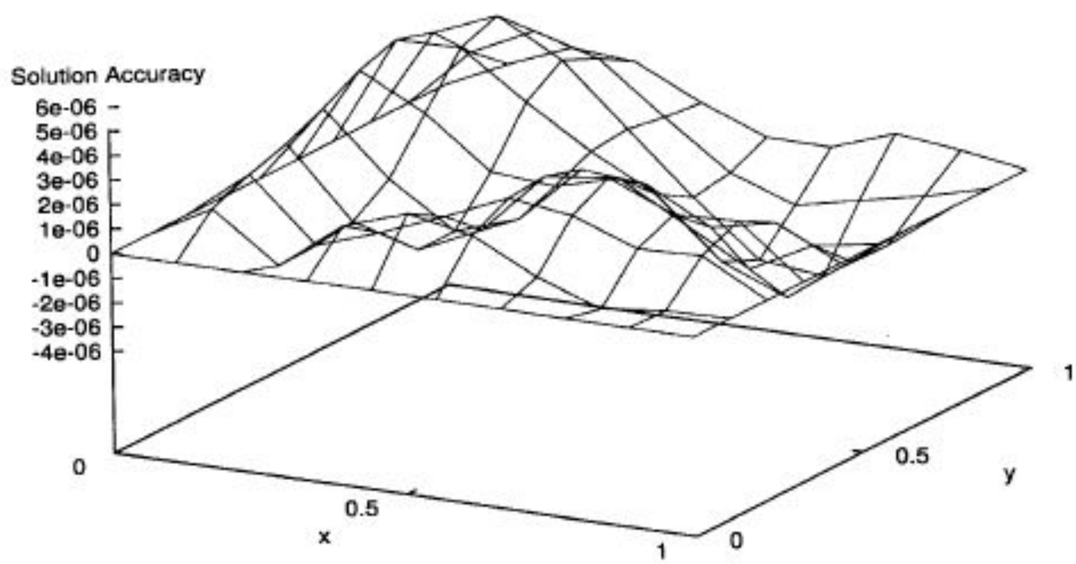

Figure 2: Accuracy of solution for example

Table 1: A comparison between Exact, ANN andShLCM solution of example

x, y	Exact solution	ANN solution	ShLCM solution	E=│exact sol. – ANNsol.│
0, 0	0	0		0
0.1, 0.1	0.09950	0.09905	0.09951	0.000444
0.2, 0.2	0.19601	0.19585	0.19575	0.000143
0.3, 0.3	0.28660	0.28654	0.28610	0.000056
0.4, 0.4	0.36842	0.36836	0.36798	0.000037
0.5, 0.5	0.43879	0.43867	0.43878	0.000023
0.6, 0.6	0.49520	0.49518	0.49589	0.000013
0.7, 0.7	0.53539	0.53528	0.53671	0.000013
0.8, 0.8	0.55737	0.55728	0.55865	0.000013
0.9, 0.9	0.55945	0.55938	0.55910	0.000013
1, 1	Cos1	0.	0.	

Table 2: The performance of the train for epoch and time

Train Function	Performance of train	Epoch	Time	Msereg.
Trainlm	8.16e-18	35341	0:04:55	5.4100e-016

Table 3: Initial weight and bias of the network for training algorithm

Initial weights and bias for trainlm		
Net.IW{1,1}	Net.LW{2,1}	Net.B{1}
0.4424	0.6834	0.4243
0.6878	0.7040	0.2703
0.3592	0.4423	0.1971
0.7363	0.0196	0.8217
0.3947	0.3309	0.4299
0.3228	0.3922	0.3428
0.2936	0.6304	0.8816
0.6271	0.4718	0.7028
0.3375	0.5108	0.4519

VII. CONCLUSIONS

In this paper, an artificial neural network has beenproposed toapproximate solution of a linear two-dimensional Fredholm integral equation. So, a three layer ANN has been proposed. Thisnetwork is able of estimating approximate solution ofassumed equation using the Levenberg – Marquardt training algorithm. The analyzed example illustrated the abilityand reliability of the present approach. The comparison between ANN and exact solution admit aremarkable accuracy. Extensions to the case of more generalof integral equations are left for future studies.

REFERENCES

[1] R. P. Kanwal and K. C. Liu, A Taylor expansion approach for solving integral equations, Int. J. Math. Educ. Sci. Technol, 20 (1989), 411–414. Neural network method for solving linear Fredholm integral equations 71

[2] M. Gulsu and M. Sezer, The approximate solution of high order linear difference equation with variable coefficients in terms of Taylor polynomials, Appl. Math.Comput, 168 (2005), 76–88.
[3] S. Nas, S. Yalcynbas and M. Sezer, A Taylor polynomial approach for solving high-order linear Fredholmintegrodifferential equations, Int. J. Math. Educ. Sci. Technol, 31 (2000), 213–225.
[4] N. Sezer, Taylor polynomial solution of Volterra integral equations, Int. J. Math. Educ. Sci. Technol, 25 (1994), 625–633.
[5] M. Sezer and M. Gulsu, A new polynomial approach for solving difference and Fredholmintegro-difference equations with mixed argument, Appl. Math. Com-put, 171 (2004), 332–344.
[6] M. Sezer, A method for approximate solution of the second order linear differential equations in terms of Taylor polynomials, Int. J. Math. Educ. Sci. Technol, 27 (1996), 821–834.
[7] X. Lan, Variational iteration method for solving integral equations, Comp. Math.withAppl, 54 (2007), 1071–1078.
[8] S. Abbasbandy, Numerical solution of integral equation: Homotopy perturbation method and Adomians decomposition method, Appl. Math. Comput, 173 (2006), 493–500.
[9] S. J. Liao, Beyond perturbation: Introduction to the Homotopy analysis method, Chapman Hall/CRC Press, Boca Raton (2003).
[10] S. J. Liao, On the homotopy analysis method for nonlinear problems, Appl. Math. Comput, 147 (2004), 499–513.
[11] S. J. Liao, Notes on the Homotopy analysis method: some definitions and theorems, Commun. Nonlinear Sci. Numer.Simul, (2008) doi:10.1016/j.cnsns.2008.04–013.
[12] S. J. Liao and Y. Tan, A general approach to obtain series solutions of nonlinear differential equations, Stud. Appl. math, 119 (2007), 297–355.
[13] M. El-Shahed, Application of Heshomotopy perturbation method to Volterras integro-differential equation, Int. J. Non. Sci. Num. Simul, 6 (2005),163–168.
[14] A. Golbabai and B. Keramati, Modified homotopy perturbation method for solving Fredholm integral equations, Chaos Soli. Frac, (2006), doi:10.1016/j.chaos.2006.10.037.
[15] Hadizadeh, M., Asgary, M., (2005), Anecient numerical approximationfor the linear class of mixed integral equations. Appl. Math.Comput., Vol. 167, pp:1090–1100.
[16] Alipanah, A., Esmaeili, Sh, (2011), Numerical solution of the two-dimensional Fredholm integral equations using Gaussian radialbasis function. J. Comput. Appl. Math., Vol. 235, pp: 5342–5347.
[17] Babolian, E., Maleknejad, K., Roodaki, M., Almasieh, H., (2010), Two-dimensionaltriangular functions and their applications to nonlinear2D Volterra-Fredholm integral equations,Comput. Math.Appl., Vol. 60, pp:1711–1722.
[18]Babolian, E., Bazm, S., Lima, P., (2011), Numerical solution of nonlineartwo-dimensional integral equations using rationalized Haarfunctions,Commun. Nonlinear Sci. Numer.Simulat, Vol. 16,pp: 1164-1175.
[19] Hristev, R. M., (1998), The ANN Book, Edition 1, GNU Public License.