Proceedings of the Institute of Mathematics and Mechanics,
National Academy of Sciences of Azerbaijan
Volume 48, Number 2, 2022, Pages 249–258
https://doi.org/10.30546/2409-4994.48.2.2022.249

ON A SOLUTION TO A NONLOCAL INVERSE COEFFICIENT PROBLEM USING FEED-FORWARD NEURAL NETWORKS

REFET POLAT

Abstract. This study gives a determination of the diffusion coefficient $D(x)$ from the equation $u_t = (D(x)u_x)_x + \nu(C(x)u(x))_x + f(x,t)$ using Neumann type boundary measurements. The nonlocal condition enables us to reduce the parabolic problem to a boundary-value problem for ODE. The flux data can be used for the initial condition of the Cauchy problem obtained from the reduced problem. The feed-forward neural network is used to find the solution to the corresponding inverse problem for $D(x)$. The presented approach is based on the solution of a nonlinear optimization problem using Particle Swarm Optimization. The efficiency and applicability of the method is demonstrated using various numerical examples with noisy free and noisy data.

1. Introduction

The determination of the leading unknown coefficient in ordinary and partial differential equations is one of key current problems in inverse problem theory and practice (see [1, 2, 4, 9, 7, 8, 14] and references therein). The mathematical model of sludge particles settling in a water treatment plant (settler) is given by the transport-diffusion equation $u_t = (D(x)u_x)_x + \nu(C(x)u(x,t))_x + f(x,t)$. Here the coefficients $D(x)$ and $C(x)$ are the diffusion and "sludge concentration" functions, respectively. In the case of the residence time of sludge particles in the settler the model leads to a nonlinear age-dependent transport-diffusion equation with a nonlocal additional condition. The determination of the diffusion coefficient $D(x)$ is considered. For the case of constant ("average") velocity ν, the problem can be reduced to a boundary-value problem for the second order nonlinear ordinary differential equation in $C(x)$ [13]. Flux data is used to define the Cauchy problem in $D(x)$ which is obtained from the boundary-value problem changing the homogeneous boundary conditions with an initial condition. The feed-forward neural network is used to find the solution of the corresponding Cauchy problem for $D(x)$. Determination of $D(x)$ is obtained by introducing a nonlinear optimization problem. Particle swarm optimization (PSO) methodology gives the solution to the nonlinear optimization problem.

2010 *Mathematics Subject Classification.* 26B40, 39B22.
Key words and phrases. Inverse coefficient problem, sludge concentration, Neural networks, Particle Swarm Optimization.

2. Problem formulation

The following parabolic problem is considered.
Find a pair of $\langle u(x,t), D(x)\rangle$ $(x,t) \in \Omega_T := (-\ell,\ell)\times(0,\infty)$, *which satisfies the parabolic problem*

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(D(x)\frac{\partial u(x,t)}{\partial x}\right) - \nu\frac{\partial}{\partial x}\left(C(x)u(x,t)\right) + f(x,t), \quad \forall (x,t)\in\Omega_t\ , \quad (2.1)$$

$$u(x,0) = \varphi(x), \quad \forall x\in(-\ell,\ell), \quad (2.2)$$

$$u(-\ell,t) = u(\ell,t) = 0, \qquad \forall t\in R_+\ ,, \quad (2.3)$$

$$\int_0^\infty u(x,t)dt = C(x), \quad \forall x\in(-\ell,\ell). \quad (2.4)$$

and the flux information

$$-D(-\ell)u_x(-\ell,t) = g_1(t),\ t>0\ (\text{or}, -D(\ell)u_x(\ell,t) = g_2(t),\ t>0). \qquad (2.5)$$

It is assumed that the condition (2.5) is an additional condition. It is also assumed that the initial data and coefficients are continuous functions, i.e. $\varphi(x), D(x), C(x)\in C[-\ell,\ell]$, and

$$0 < D_* \le D(x) \le D^*, \quad 0\le C(x)\le C^*,\ \forall x\in[-\ell,\ell]. \qquad (2.6)$$

For a given coefficient $C(x), D(x)\in C[-\ell,\ell]$ are denoted by $u = u(x,t;D,C)$, the unique classical solution of the parabolic initial value problem (2.1)-(2.4). Here the coefficients $C(x), D(x)$ are assumed to be unknown and the nonlocal measured data (2.4) can be treated as observations for determination of the coefficient $D(x)$. For this reason, the problem (2.1) to (2.5) is defined as a nonlocal optimal control or identification problem, and the couple $\langle u, D\rangle$ is called a solution of the inverse coefficient problem (2.1)-(2.5). In this case the problem (2.1)-(2.4) is called a direct problem corresponding to the inverse problem (2.1)-(2.5).

3. Necessary conditions for optimality

Assume that $\langle u, D\rangle$ is a solution of the problem (2.1)-(2.5). Then integrating equation (2.1) on $[0,\infty)$ with respect to the variable $t>0$, we obtain

$$u(x,\infty) - u(x,0) =$$

$$\int_0^\infty \frac{\partial}{\partial x}\left(D(x)\frac{\partial u(x,t)}{\partial x}\right)dt - \nu\int_0^\infty \frac{\partial}{\partial x}\left(C(x)u(x,t)\right)dt + \int_0^\infty f(x,t)dt, \qquad (3.1)$$

$$\forall x\in(-\ell,\ell).$$

By using the nonlocal condition (2.4) and assuming differentiability of the function $u(x,t)$ under the integrals, we may write

$$\begin{aligned} C^2(x) &= \int_0^\infty C(x)u(x,t)dt,\ \forall x \in (-\ell,\ell), \\ C'(x) &= \int_0^\infty \frac{\partial u(x,t)}{\partial x}dt,\ \forall x \in (-\ell,\ell), \\ (C^2(x))' &= \int_0^\infty \frac{\partial}{\partial x}(C(x)u(x,t))\,dt,\ \forall x \in (-\ell,\ell), \\ (D(x)C'(x))' &= \int_0^\infty \frac{\partial}{\partial x}\left(D(x)\frac{\partial u(x,t)}{\partial x}\right)dt,\ \forall x \in (-\ell,\ell), \\ F(x) &:= \varphi(x) + \int_0^\infty f(x,t)dt. \end{aligned}$$

Taking into account the initial condition (2.2), we can rewrite the integro-differential equation (3.1) in the following reduced form:

$$\begin{cases} -\frac{d}{dx}\left(D(x)\frac{dC(x)}{dx}\right) + \nu\frac{d}{dx}(C^2(x)) = F(x),\ \forall x \in (-\ell,\ell), \\ C(-\ell) = 0, \quad C(\ell) = 0. \end{cases} \tag{3.2}$$

Hence, we prove the following propositions:

Proposition 3.1. *If $\langle u,D\rangle$ is a solution of the identification problem (2.1)-(2.5), then the function $C(x)$ satisfies the Cauchy problem* (3.2) *for the second order nonlinear ordinary differential equation.*

According to Proposition 3.1, we can now consider the *reduced problem* (3.2). Rewriting the equation in (3.2) with respect to the unknown function $D = D(x)$ and using the reduced measured data ψ which is obtained from nonlocal condition (2.4) and measured data (2.5) for the initial condition, we have the following Cauchy problem:

$$\begin{cases} D'(x) + \frac{C''(x)}{C'(x)}D(x) = G(x), \quad G(x) := \frac{\nu(C^2(x))' - F(x)}{C'(x)},\ x \in (-\ell,\ell), \\ D(-\ell) = \frac{\psi}{C'(-\ell)}, \quad \psi := -\int_0^\infty g_1(t)dt. \end{cases} \tag{3.3}$$

Proposition 3.2. *The identification problem (2.1)-(2.5) is equivalent to the following problem:*

Find the function $D(x)$, which satisfies the Cauchy problem (3.3)*.*

It is clear that if the function $C = C(x)$ is known, the solution of the Cauchy problem (3.3) has the following integral representation.

$$D(x) = \frac{1}{C'(x)}\left\{\psi + \nu C^2(x) - \int_{-\ell}^{x} G(\xi)d\xi\right\} \quad x \in [-\ell,\ell], \quad C'(x) \neq 0. \tag{3.4}$$

Even if this representation gives the analytical solution of the problem (3.3) the function $D(x)$ is undefined at the points where the derivative of the sludge concentration $C(x)$ vanishes. So, an Artificial Neural Network (ANN) approach is proposed for solving the Cauchy problem (3.3) numerically in cases where $C'(x)$ vanishes at some points in $[-\ell,\ell]$.

4. Feed-forward Neural Networks as a solution to the reduced problem

Artificial neural networks are modeled from the human brain and neural systems, which are suitable tools for solving large-scale problems. There are many references to neural networks in theory and applications, modeling, algorithms, design, architecture and mathematics [12].

We denote the ANN-solution by $N(x)$ at the point x. According to the Kolmogorov and Cybenko theorems[3, 11], we can establish a trial approximate solution given in Eq. 4.1 for the reduced problem (3.2).

$$D(x) := \frac{\psi}{C'(-\ell)} + (x+\ell)N(x;\vec{p}), \quad x \in [-\ell, \ell], \tag{4.1}$$

where $\vec{p} := \vec{p}(\vec{\alpha}, \vec{\eta}, \vec{\beta})$ is an unknown parameter vector to be determined such that $\vec{\alpha}$, $\vec{\eta}$ and $\vec{\beta} \in R^m$ where m is the total number of neurons in the hidden layer of the neural network.

As seen in Figure 1, for the input values x, the output of the network is the function $N(x;\vec{p})$ defined as follows:

$$N(x;\vec{p}) := \sum_{k=1}^{m} \alpha_k \tau(\eta_k x + \beta_k) \tag{4.2}$$

where α_k is the synaptic weight of the kth hidden neuron to the output, η_k is the synaptic coefficient from the spatial input x to the kth hidden neuron, and β_k is the bias value of the kth hidden neuron. Here, $\tau(z) = 1/(1+\exp(-z))$ is the logistic activation function.

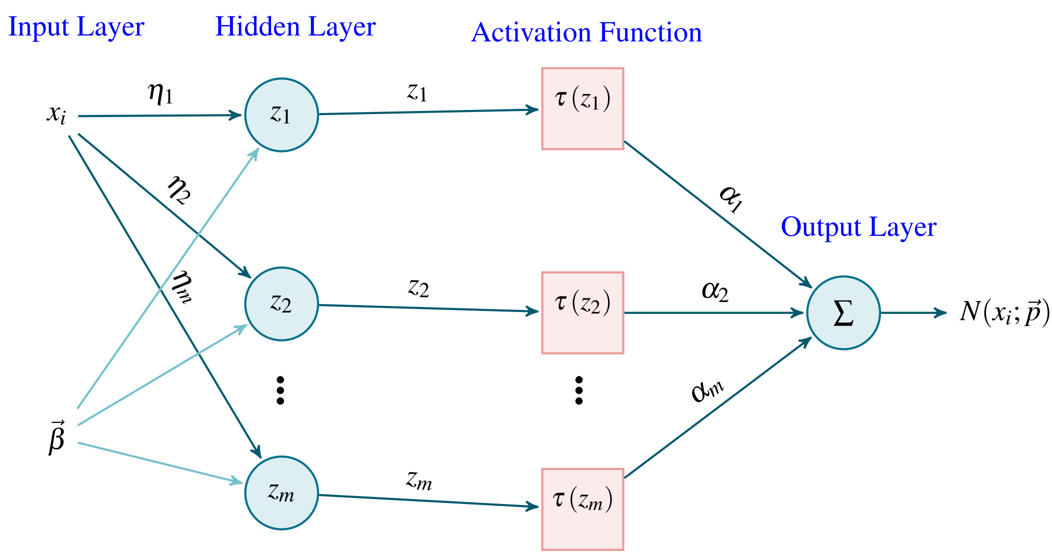

FIGURE 1. Structure of the general feed-forward single hidden layer perceptron.

As seen in Figure 1, a feed forward neural network model including a single hidden layer, which takes inputs from the input layer and produces the weighted sum of inputs added onto some bias values as outputs, is preferred to solve the problem effectively. Each neuron in the hidden layer produces the corresponding values defined as follows:

$$z_1 = \eta_1.x_i + \beta_1, \; z_2 = \eta_2.x_i + \beta_2, \; \ldots, \; z_m = \eta_m.x_i + \beta_m$$

where x_i denotes the network inputs. Then, the activation function $\tau(z)$ takes the outputs of the hidden layer and transforms them to the inputs of the output layer. Finally, the sum of the weighted outputs of the activation function then generates the output $N(x_i;\vec{p})$ of the neural network.

It is possible to obtain proper values for the adjustable parameters $\vec{p}$ such that $N(x)$ gives a good approximation to $D(x)$. It is at this stage that tools such as error norm in the Hilbert space $L^2(\Omega_T)$ can assist with a minimization problem. The unknown parameter vector $\vec{p}$ is determined by considering the minimization problem defined with the functional as follows:

$$\text{Find } \vec{p}^* \text{ such that } J(x;\vec{p}^*) = \min_{\vec{p}\in R^{3m}} J(x;\vec{p}).$$

Here, the cost function $J(x,\vec{p})$ is defined as:

$$J(x;\vec{p}) = \frac{1}{2}\sum_{i=1}^{N_x} e_i^2 \tag{4.3}$$

where

$$e_i = -D'(x_i)C'(x_i) - D(x_i)C''(x_i) + 2\nu C(x_i)C'(x_i) - F(x_i). \tag{4.4}$$

For the numerical solution of the minimization problem defined above, ”Particle Swarm Optimization (PSO)” is considered.

5. Numerical implementation

Particle swarm optimization (PSO) is a swarm intelligence technique (a search method based on a natural system), which was introduced by Kennedy and Eberhart in 1995 [10, 5]. This method performs the search of the optimal solution through agents, referred to as particles, whose trajectories are adjusted by a stochastic and a deterministic component. Each particle is influenced by its best achieved position and the group best position, but tends to move randomly. A particle i is defined by its position vector, $\vec{x}_i$, and its velocity vector, $\vec{v}_i$. Every iteration, each particle, changes its position as given in Eq. 5.2 according to the new velocity given in Eq. 5.1

$$\vec{v}_i[n+1] = w\vec{v}_i[n] + c_1 r_1(\vec{x}_{Best_i}[n] - \vec{x}_i[n]) + c_2 r_2(\vec{g}_{Best}[n] - \vec{x}_i[n]), \tag{5.1}$$

$$\vec{x}_i[n+1] = \vec{x}_i[n] + \vec{v}_i[n] \tag{5.2}$$

where $\vec{x}_{Best_i}$ denotes the best position of i^{th} particle, $\vec{g}_{Best}$ represents the best group position, the parameters w is inertia weight, c_1 and c_2 are the cognitive acceleration coefficient and social acceleration coefficient respectively, and r_1 and r_2 are two random parameters within $[0,1]$ [6].

For the numerical solution to the Cauchy problem 3.3, we constructed a uniform grid of mesh points x_i with $x_i = -\ell + ih_x, j = 0,1,2,\ldots,N_x$ where $h_x = 2\ell/N_x$. In order to do performance analysis, various measurements were used: Mean Absolute Error (*MAE*), Mean Squared Error (*MSE*) and Mean Squared Relative Error (*MSRE*). The MAE, MSE and MRSE are defined below.

$$\|D - D^{app}\|_{MAE} := \frac{1}{N_x}\sum_{i=1}^{N_x} |D_i - D_i^{app}|$$

$$\|D - D^{app}\|_{MSE} := \left(\frac{1}{N_x} \sum_{i=1}^{N_x} |D_i - D_i^{app}|^2 \right)^{(1/2)}$$

$$MSRE = \frac{\|D - D^{app}\|_{MSE}}{\|D\|_{MSE}}.$$

Now we consider some of the series of examples to demonstrate the efficiency of the proposed method.

Example 5.1. Let

$$D(x) = \frac{1}{x^2 + 1}$$

be the exact solution of the Cauchy problem (3.3) for the input data.

$$C(x) = \exp(-x) \quad G(x) := \frac{\nu (C^2(x))' - F(x)}{C'(x)}, \quad x \in (-\ell, \ell).$$

Where $\nu = -1.0$, $\ell = 1.0$ and the value $\psi = D(-\ell)C'(-\ell)$ is assumed to be synthetic noise free data given on the left boundary of the interval $(-\ell, \ell)$. Figure 2 illustrates the exact solution $D(x)$ and its approximation $D^{app}(x)$ of the Cauchy problem on the interval $[-\ell, \ell]$. The differences between the numerical solutions obtained using PSO variants by means of MAE, MSE and $MSRE$ are given in Table 1. All the results were attained with $m = 10$ neurons in feed-forward neural networks. To train the network, the quadrature nodes were firstly specified with the discretization of the domain $[-\ell, \ell]$ by taking $N_x = 21$, and a mesh was generated. The quadrature nodes were used as inputs of the neural net. Then, the optimization procedure was executed to train the network in the meaning of unsupervised learning using the cost functional given in Eq. 4.3. The maximum number of iterations were selected as 250 in the first experiments. In this study, the cognitive and social acceleration coefficients were selected as $c_1 = c_2 = 2$. The inertia coefficient was determined as a decreasing function with the initial value $w = 1$ and the damping ratio 0.99. The lower and upper boundary of arbitrary network parameters represented by $\vec{p}$ were determined as -10 and 10. The total number of individuals in the PSO population was 20 in this experiment.

TABLE 1. Errors for Example 5.1

Type of Errors		Errors for Noise Free Data	Errors for Noisy Data
	Min	$3.585e-03$	$9.118e-03$
MAE	Worst	$5.411e+00$	$2.074e+00$
	Mean	$2.307e-01 \pm 3.708e-01$	$2.392e-01 \pm 2.488e-01$
	Min	$1.926e-03$	$45.455e-03$
MSE	Worst	$2.352e+01$	$9.702e-01$
	Mean	$2.160e-01 \pm 1.508e+00$	$1.130e-01 \pm 1.259e-01$
	Min	$4.203e-04$	$1.190e-03$
$MSRE$	Worst	$5.133e+00$	$2.117e-01$
	Mean	$4.714e-02 \pm 3.291e-01$	$2.465e-02 \pm 2.748e-02$

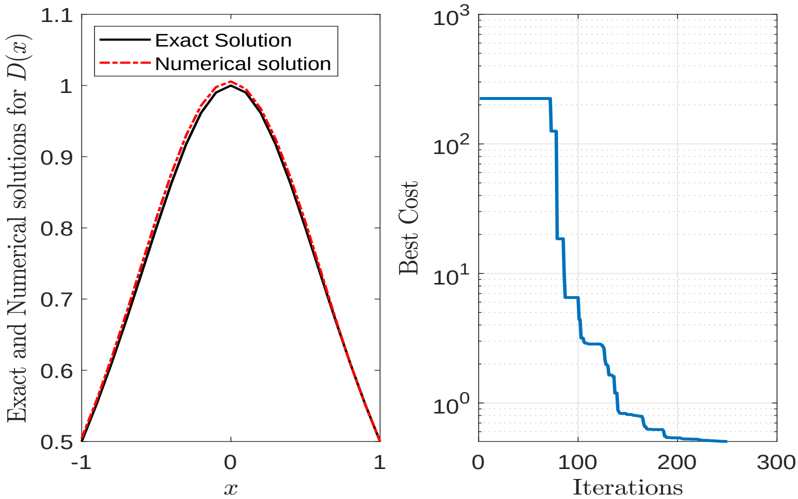

FIGURE 2. (Left) Exact and noise free approximate solution of $D(x)$ obtained from ANN initializied uniformly. (Right) Cost function profile in PSO

Consider now this example in the case of noisy data corresponding to the synthetic noisy Neumann data $D(-L)C'(-L) = \psi_\delta$, $\psi_\delta = \psi \pm \delta\psi$ with noise level $\delta = 0.05$. The errors corresponding to the noisy data are given in the last column of Table 1. The exact and noisy approximate solution is plotted in Figure 3.

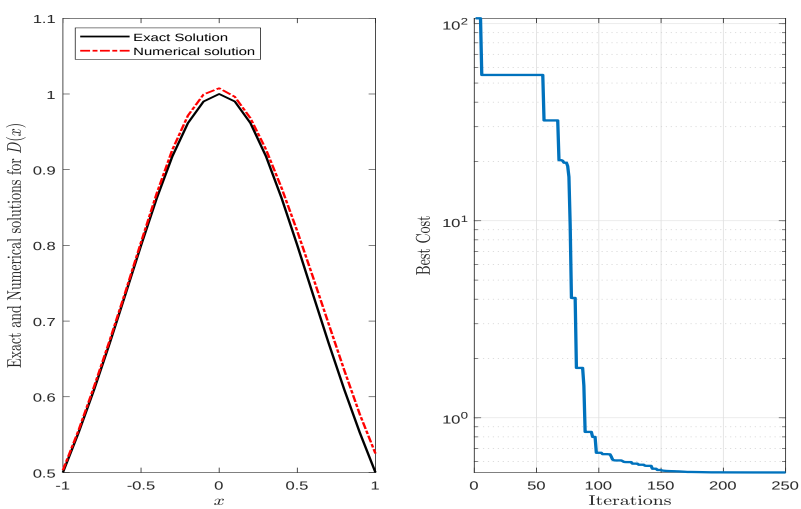

FIGURE 3. (Left) Exact and noisy approximate solution of $D(x)$ obtained from ANN initializied uniformly for noise level $\delta = 0.05$. (Right) Cost function profile in PSO

Example 5.2. Let the non-smooth function

$$D(x) = 1 + \sqrt{|x|}$$

be an exact solution of the Cauchy problem (3.3) for the input data.

$$C(x) = \exp(-x) \quad G(x) := \frac{\nu(C^2(x))' - F(x)}{C'(x)}, \quad x \in (-\ell, \ell).$$

Where $\nu = 0.5$, $\ell = 1.0$ and the value $\psi_\delta = \psi \pm \delta\psi$ is assumed to be synthetic noisy data for noise level $\delta > 0$. Figure 4 illustrates the exact solution $D(x)$ and its approximation $D^{app}(x)$ of the Cauchy problem on the interval $[-\ell, \ell]$. The errors *MAE*, *MSE* and *MSRE* are given in Table 2. All the results were attained with $m = 10$ neurons in feed-forward neural networks taking $N_x = 22$ nodes. The maximum number of iterations were selected as 250 in the second experiments. In this test example, the cognitive and social acceleration coefficients were selected as $c_1 = c_2 = 2$. The inertia coefficient was determined as a decreasing function with the initial value $w = 1$ and the damping ratio 0.99. The lower and upper boundary of arbitrary network parameters represented by $\vec{p}$ were determined as -1 and 1. The total number of individuals in the PSO population was 10 in this experiment.

TABLE 2. Errors for Example 5.2

Type of Errors		**Errors for Noise Free Data**	**Errors for Noisy Data**
	Min	$1.349e-01$	$2.876e-01$
MAE	Worst	$3.824e+00$	$6.146e+00$
	Mean	$1.498e+00 \pm 1.206e+00$	$1.756e+00 \pm 1.616e+00$
	Min	$1.152e-01$	$2.132e-01$
MSE	Worst	$2.490e+00$	$3.838e+00$
	Mean	$8.424e-01 \pm 7.081e-01$	$9.941e-01 \pm 1.067e+00$
	Min	$1.228e-03$	$2.273e-03$
MSRE	Worst	$2.654e-02$	$4.091e-02$
	Mean	$8.980e-03 \pm 7.548e-03$	$1.060e-02 \pm 1.137e-02$

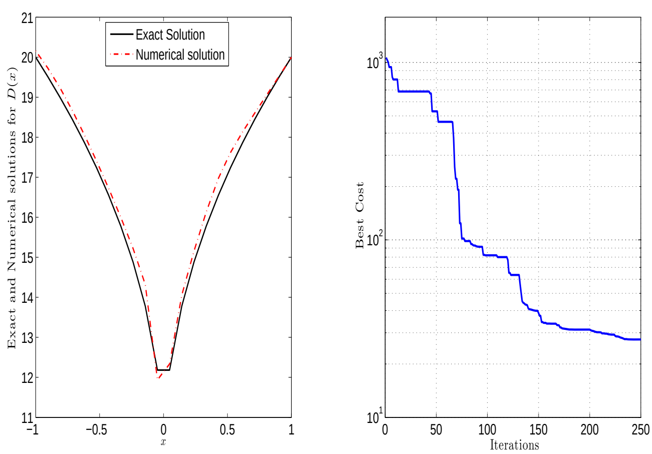

FIGURE 4. (Left) Exact and noise free approximate solution of $D(x)$ obtained from ANN initializied uniformly. (Right) Cost function profile in PSO

Consider the case where noisy data corresponds to the synthetic noisy Neumann data $D(-L)C'(-L) = \psi_\delta$, $\psi_\delta = \psi \pm \delta\psi$ with noise level $\delta = 0.05$. The errors corresponding to the noisy data are given in the last column of Table 2. The exact and noisy approximate solution is plotted in Figure 5.

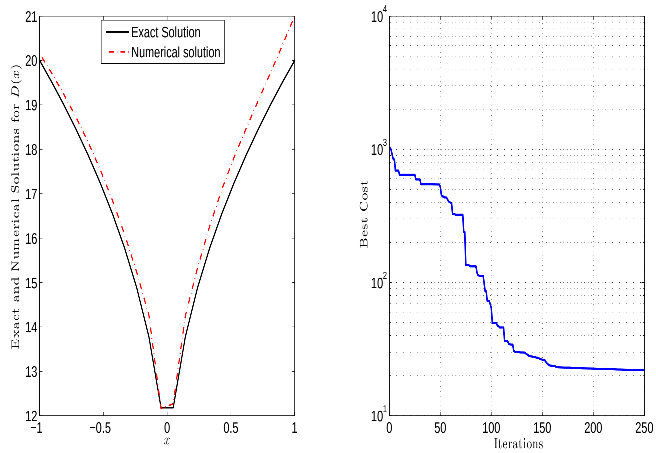

FIGURE 5. (Left) Exact and noisy approximate solution of $D(x)$ obtained from ANN initializied uniformly for noise level $\delta = 0.05$. (Right) Cost function profile in PSO

6. Conclusions

We have considered an inverse coefficient problem (2.1)-(2.5) with a nonlocal condition for a parabolic equation and reduced it to an inverse problem in ordinary differential equations. We have applied a feed-forward neural network approach for a numerical solution of the inverse problem, which is based on nonlinear optimization. Efficiency and applicability of the method has been tested using various numerical examples with noisy free and noisy data.

7. Acknowledgments

The author wishes to thank Burhan Pektas for the problem formulation and fruitful suggestions on the topic of ANN approximation. Thanks are also given to Ian Stewart Collins, the Vice Director of School of Foreign Languages at Yasar University, for his contribution in proof-reading.

References

[1] J. R. Cannon, P. Duchateau, Determining of unknown coefficients in a nonlinear heat conduction problem, *SIAM J. Appl. Math.*, **24**, (1973), no.3, 298–314.

[2] M. Choulli, P . Stefanov, Reconstruction of the coefficients of the stationary transport equation from boundary measurements, *Inverse Problems*, **12**, (1996), no.5, 19–23.

[3] G. Cybenko, Approximation by superpositions of a sigmoidal function, *Mathematics of Control, Signals and Systems*, **2**(4), (1989), 303–314.

[4] P. Duchateau, *Introduction to inverse problems in partial differential equations for engineers, physicists and mathematicians*, Kliver Academic Publishers, Netherland, 1996.
[5] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, *Sixth International Symposium on Micro Machine and Human Science*, (1995).
[6] S. Eskiizmirliler, K. Gunel, R. Polat, On the solution of the Black-Scholes equation using feed-forward neural networks, *Computational Economics*, **58**, (2021), 915–941.
[7] A. Hasanov, P. Duchateau, B. Pektaş, An adjoint problem approach and coarse-fine mesh method for identification of the diffusion coefficient in a linear parabolic equation, *Inverse and ill-posed Problems*, **14**, (2006), no.5, 435–463.
[8] A. Hasanov, B. Pektaş, Simulation of ill-conditioned situations of inverse coefficient problem for the Sturm-Liouville operator based on boundary measurements, *Mathematics and Computers in Simulation*, **61**, (2001), 47–52.
[9] A. Hasanov, T. S. Shores, Solution of an inverse coefficient problem for an ordinary differential equation, *Applicable Analysis*, **67**, (1997), no.1-2, 11–20.
[10] J. Kennedy, R. Eberhart, Particle swarm optimization, *Proceedings of International Conference on Neural Networks*, **4**, (1995), 1942–1948.
[11] A. N. Kolmogorov, On the representation of continuous functions of several variables by superpositions of continuous functions of one variable and addition. (Russian); translated from *Dokl. Akad. Nauk SSSR,1957, 114, 953–956*, *American Mathematical Society Translation*, **28**, (1963), no.2, 55–59.
[12] B. Muller, J. Reinhardt, M.T. Strickland, *Neural Networks: An Introduction second ed.*, Springer-Verlag, Berlin, 2002.
[13] B. Pektaş, Determination of unknown coefficients in a nonlocal parabolic problem from Neumann type boundary measured data, *Advances in Dynamical Systems and Applications*, **5**, no.2, (2010), 215–230.
[14] G. R. Richter, An inverse problem for the steady state diffusion equation, *SIAM J. Appl. Math.*, **41**, (1981), no.2, 210–221.

Refet Polat
Yasar University, Izmir, Turkey,
Department of Mathematics, Yasar University, Bornova, Izmir, Turkey
E-mail address: `refet.polat@yasar.edu.tr`

Received: August 31, 2022; Accepted: November 1, 2022