Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

An efficient parallel algorithm for the numerical solution for Singularly Perturbed Delay Differential Equations with Layer Behavior

Rana T. Shwayyea
Department of Physics, College of Education / University of AL-Qadisiyah
Email : rana.alrubai@qu.edu.iq

ARTICLE INFO

Article history:
Received: // 2020/2/12
Rrevised form: //
Accepted : 13/05/2020
Available online: 18/07/2020

Keywords:

Singularly perturbed boundary value problems; delay term; boundary layer; neural network.

ABSTRACT

The numerical solution of a Singularly Perturbed Delay Differential Equations (SPDDE) is defined as a very charged problematic of computational becouse to the non-local nature of this type of differential Equations. Prove that parallelism can be used to overawed these problems for this purpose we suggest the application of parallel processors as the best solution to overcome the difficulties of the perturbed that occurs in Perturbed Delay Differential Equations on a matching processer. Allowing to several latest publications, this process has been effectively applied to a big number of SPDDE rising from a change of application fields. The exact quality of the conception of parallelism is argued in fact and several examples are presented to demonstrate the feasibility of our approach.

https://doi.org/10.29304/ jqcm.2020.12.2.700

1 . Introduction

A SPDDE is an ordenary differntial equation in which the hieghest erivative is increased by a minor parameter and having delay term. Interest in studying this important topic has increased in recent times due to its great importance in numerical solutions . There are many uses for these important topics, for example, the mathematical modeling used in [1], in the study of bistabl plans [2], and variable problemes in controle theory [3] Which are the only realistic simulations . Lange and Miura [4] Those who studied in this paper studied the problems of pertuerbed boundary values.Kadalbajoo and Sharma [5, 6], They studied the numerical solutions of pertuerbed differential equations, which are of great importance to this topic .In [7], they authors studied the numerical solutions of pertuerbed differential equations of the second degree, with significant delay in the term interaction through the method of integrated characters using exponential functions and quadratic rules with weight and duration remaining in an integrated form. In [8], the authors Juegal Mohapeatra, Srinivasan Natesan Create a numerical method for a class of individually disturbed

Corresponding author: Rana T. Shwayyea
Email: rana.alrubai@qu.edu.iq
Communicated by Alaa Hussein Hamadi

pertuerbed differential equations with minimal delay. The numerical method consists of a finite difference factor in the direction of the wind on an adaptive network,
Which is formed by distributing the observation function along the arc . In[9], the authors MK. Kiadalbajoo, Devndra Kumiar Treat the spread with a small delay parameter by presenting a numerical method to a pertuerbed differential equation .

The goal of this paper is to project feed for ward nueral network (FFNN) to solve SPDDE. Using a multe-layer with 7 hiden units (neurones) and one linear output unit, the sigmoed activatione function of each unit in hiden layer is tansign function, where the Levenberge – Marquardit treining algorethm is used to treining the network .

1.Introduction:

A SPDDE is an ordenary differntial equation in which the hieghest erivative is increased by a minor parameter and having delay term. Interest in studying this important topic has increased in recent times due to its great importance in numerical solutions . There are many uses for these important topics, for example, the mathematical modeling used in [1], in the study of bistabl plans [2], and variable problemes in controle theory [3] Which are the only realistic simulations . Lange and Miura [4] Those who studied in this paper studied the problems of pertuerbed boundary values.Kadalbajoo and Sharma [5, 6], They studied the numerical solutions of pertuerbed differential equations, which are of great importance to this topic .In [7], they authors studied the numerical solutions of pertuerbed differential equations of the second degree, with significant delay in the term interaction through the method of integrated characters using exponential functions and quadratic rules with weight and duration remaining in an integrated form. In [8], the authors Juegal Mohapeatra, Srinivasan Natesan Create a numerical method for a class of individually disturbed pertuerbed differential equations with minimal delay. The numerical method consists of a finite difference factor in the direction of the wind on an adaptive network,
Which is formed by distributing the observation function along the arc . In[9], the authors MK. Kiadalbajoo, Devndra Kumiar Treat the spread with a small delay parameter by presenting a numerical method to a pertuerbed differential equation .

The goal of this paper is to project feed for ward nueral network (FFNN) to solve SPDDE. Using a multe-layer with 7 hiden units (neurones) and one linear output unit, the sigmoed activatione function of each unit in hiden layer is tansign function, where the Levenberge – Marquardit treining algorethm is used to treining the network .

2. Layer on the left side

Consider SPDDE of the form

$$\varepsilon\gamma''(\times) + \hbar(\times)\gamma'(\times - \tau) + g(\times)\gamma(\times) = \mathcal{F}(\times) \quad , 0 \leq \times \leq 1 \qquad \textbf{(1)}$$

with boundary conditions

$$\gamma(0) = \alpha \;\; , -\tau \leq \times \leq 0 \qquad \textbf{2(a)}$$
$$\gamma(1) = \beta \qquad \textbf{2(b)}$$

where ε is small paramter, $0 < \varepsilon << 1$ and τ is also small shifting paramter, $0 < \tau << 1$; $\hbar(\times)$, $g(\times)$, $\mathcal{F}(\times)$ are bc functions in (0, 1) and α, β are finite constants. More, we accept that $\hbar(\times) \geq \aleph >$

0 in [0, 1],where $\aleph$ is positive constant. This statement just suggests that the baoundary layer will be in the neighborhood of $\times = 0$.

3. Right - end boundary layer problem

We now accept that $\hbar(\times) \leq \aleph < 1$ in [0, 1],where $\aleph$is negative constant. This supposition merely suggests that the baoundary layer will be in the neighborhood of $\times = 1$.

4. Neural Network [10]

Neural network of this starting It can be run by each and is distributed parallel It is a system that handles a large number of contacts and information with performance numbers with neurological and biological networks. Ann is regularly considered as only or multidimensional. In defining the total of covers, input Units are not totaled as a class, for they do not complete any design. Evenly, number Layers in a network can be definite as the total of covers of weighted links among neuronal plates. In a coated neural network, neurons are prepared into coats. We need at tiniest two covers: input and output. Covers between input and output cover (if any) They are called concealed covers, which in turn are called concealed bulges concealed units. Further concealed neurons upsurge the network's capability to excerpt high-level numbers from (Data Entry. Training is the process of adjusting contact weights and biases b. In the first step,Network output and the difference between actual (obtained) and required (target) outputs Outputs (ie, error) are intended for the weights and biases formed (arbitrary values). Through The second step, the initial weights are adjusted in all connectors and biases in all neurons Reduce the error by propagating the error backwards

5. Description of the method

In this section illustrate how our approach can be used to the approximation solution of the SPDDE is defined in (1) with above boundary layers .The approximate solution can be defined according to the technique described in this paper

$$\boldsymbol{\gamma}_t(\times_i, \rho) = \Lambda(\times) + \Gamma(\times, \Theta(\times, \rho)) \qquad (3)$$

where $\Theta(\times, \rho)$ is a singl-output FFNN with paramters ρ and n input units feed with the input vector $\times$. The term $\Lambda(\times)$ contians no adjust able paramters and contents the BCs. The next term Γ is created so as not to contribut to the BCs .And in general can define solution of SPDDE using FFNN.To illustrat the technique, we will reflect the 2nd-order SPDDE:

$$\frac{d^2\boldsymbol{\gamma}}{d\times^2} = \Omega(\times, \boldsymbol{\gamma}, \boldsymbol{\gamma}', \boldsymbol{\varepsilon}, \tau) \qquad (4)$$

where $\times \in [a,b]$ and the BC: $\boldsymbol{\gamma}\,(a) = \boldsymbol{\alpha}$, $\boldsymbol{\gamma}\,(b) = \boldsymbol{\beta}$; an approximation solution can be written as

$$\boldsymbol{\gamma}_t(\times_i, \rho) = \frac{(b\boldsymbol{\alpha} - a\boldsymbol{\beta})}{(b-a)} + \frac{(\boldsymbol{\beta} - \boldsymbol{\alpha})}{b-a}\times + (\times - a)(\times - b)\Theta(\times, \rho) \qquad (5)$$

The our goal in this paper is to design a FFNN $\Theta(\times, \rho)$such that $\boldsymbol{\gamma}_t$ Fit Informatics unknown function $\boldsymbol{\gamma}(\times)$ in any accuracy .Now rewrite (5) to be as following:-

$$\Theta(\times, \rho) = \frac{\boldsymbol{\gamma}_t(\times) - \frac{\times - a}{b-a}\boldsymbol{\beta} - \frac{b-\times}{b-a}\boldsymbol{\alpha}}{(\times - \mathrm{a})(\times - \mathrm{b})} \quad \times \neq a, b \qquad (6)$$

The error quantety to be minimizede is given by

$$E(\rho) = \sum_{i=1}^{n}\left\{\frac{d^2\boldsymbol{\gamma}_t(\times_i,\rho)}{d\times^2} - \Omega(\times_i, \boldsymbol{\gamma}_t(\times_i,\rho), \frac{dy_t(\times_i,\rho)}{d\times}, \boldsymbol{\varepsilon}, \tau)\right\}^2 \quad \text{where the } \times i \in [a,\ b] \qquad (7)$$

Since

$$\frac{d\boldsymbol{\gamma}_t(\times,\rho)}{d\times} = \frac{(\boldsymbol{\beta}-\boldsymbol{\alpha})}{(b-a)} + \{(\times - a) + (\times - b)\}\,\Theta(\times,\rho) + (\times - a)(\times - b)\frac{d\Theta(\times,\rho)}{d\times}$$

$$\frac{d^2\boldsymbol{\gamma}_t(\times,\rho)}{d\times^2} = 2\Theta(\times,\rho) + 2\{(\times - a) + (\times - b)\}\,\frac{d\Theta(\times,\rho)}{d\times} + (\times - a)(\times - b)\frac{d^2\Theta(\times,\rho)}{d\times^2} \qquad (8)$$

6. Numerical Results and Discussion

To determine the effectiveness of the technique, In this section of the research, we examined four examples of all the different cases with respect to the border layers and different values of the parameters of the perturbed and delay. Some examples did not contain an analytical solution and compared with some numerical solutions and analytical solutions with numbers and graphs, which shows the superiority is clear in relation to the proposed method.

Example 1

Study an example of SPDDE with" left layer":

$\varepsilon\Upsilon''(\times) + \Upsilon'(\times - \tau) - \Upsilon(\times) = 0$, $0 \le \times \le 1$

with boundary conditions

$$\Upsilon(0) = 1$$
$$\Upsilon(1) = 1$$

"The exact solution is given by" $\gamma(\times) = \frac{((1-e^{\xi_2})e^{\xi_1\times} + (e^{\xi_2}-1)e^{\xi_2\times})}{(e^{\xi_1} - e^{\xi_2})}$

Where $\xi_1 = \frac{(-1-\sqrt{1+4(\varepsilon-\tau)})}{2(\varepsilon-\tau)}$, $\xi_2 = \frac{(-1+\sqrt{1+4(\varepsilon-\tau)})}{2(\varepsilon-\tau)}$

compared with results of paper [9].

Table 1: Compare between exact and proposed method forexample 1for $\varepsilon = 0.001$,τ = 0.0001

Input x	Exact $y_a(x)$	suggested method $y_t(x)$	Numerical solution [9]	$E(x) = \mid y_t(x) - y_a(x) \mid$
0	1.000000000000	1.000000000000	1.0000000	0.000000000000E+00
0.01	0.371914090003	0.371914090605	0.3724909	6.016098730299E-10
0.02	0.375641680331	0.375641680353	0.3753635	2.189354253446E-11
0.03	0.379413533243	0.379413533619	0.3791343	3.764627498626E-10
0.04	0.383223259752	0.383223259742	0.3829441	1.038746866300E-11
0.06	0.390957858335	0.390957858387	0.3906790	5.210187836724E-11
0.08	0.398848564392	0.398848564926	0.3985702	5.341356801303E-10
0.20	0.449652021910	0.449652021615	0.4493791	2.954539390920E-10
0.50	0.606803174487	0.606803174803	0.6065730	3.157875072546E-10
0.60	0.670560975535	0.670560975574	0.6703575	3.899758294068E-11
0.90	0.904918712281	0.904918712244	0.9048500	3.727873565396E-11
1	1.000000000000	1.000000000000	1.0000000	0.000000000000E+00

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

Train Function	Performance of train	Epoch	Time	MSE
Trainlm	2.39-31	22	0:00:01	7.32962E-20

Table 3 : Initial Weights and bias for trainlm

Weights and bias for trainlm			
Net.IW{1,1}	**Net.IU{1,1}**	**Net.LW{2,1}**	**Net.B{1}**
0.3548	0.6509	0.5206	0.2609
0.9674	0.9643	0.9823	0.3598
0.0963	0.6539	0.8326	0.4764
0.5234	0.8053	0.6898	0.9635
0.8710	0.3326	0.1895	0.6529

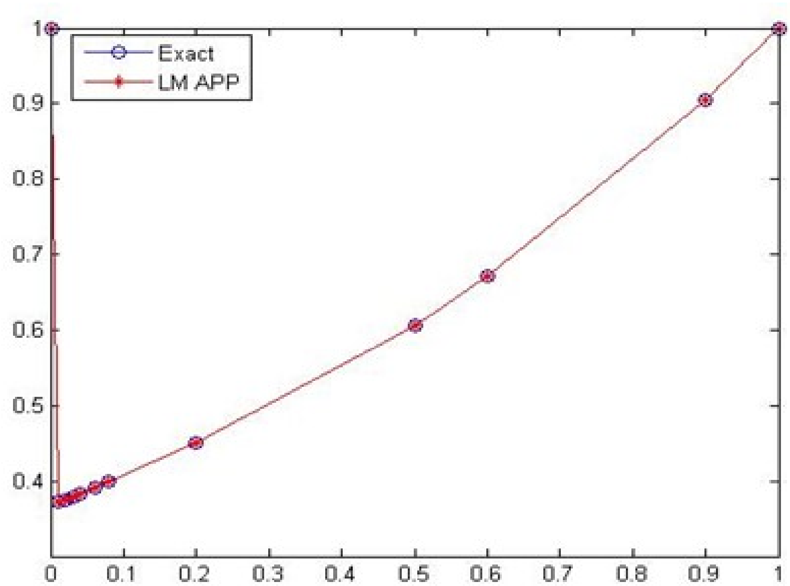

Figure 1. Exact and Approximate solution of example1 for $\varepsilon = 0.001$,$\tau = 0.0001$

Tal

Table 4: : Compare between exact and proposed method forexample 1for 1for $\varepsilon = 0.0001$,$\tau = 0.00001$

Input x	Exact $y_a(x)$	suggested method $y_t(x)$	$E(x) = \mid y_t(x) - y_a(x) \mid$
Input	Exact $y_a(x)$	suggested method	$E(x) = \mid y_t(x) - y_a(x) \mid$
0.1	0.406899055265	0.406899055496	2.310537317030E-10
0.2	0.449652539205	0.449652587774	4.856917357188E-08
0.3	0.496898194767	0.496898194766	5.044853423897E-13
0.4	0.549108021049	0.549108021045	3.535061132709E-12
0.5	0.606803610791	0.606803618671	7.880012309336E-09
0.6	0.670561361253	0.670561361756	5.034846983776E-10
0.7	0.741018232602	0.741018232998	3.960383132551E-10
0.8	0.818878111353	0.818878111362	9.330314298950E-12
0.9	0.904918842412	0.904918842798	3.857814068198E-10
1.0	1.000000000000	1.000000000000	0.000000000000E+00

Table 5:The performance of the train with epoch and time

Train Function	Performance of train	Epoch	Time	MSE
Trainlm	4.6534	50	0:00:02	2.20124E-16

Table 6 : Initial weight and bias of the network for different training algorithm

Weights and bias for trainlm			
Net.IW{1,1}	Net.IU{1,1}	Net.LW{2,1}	Net.B{1}
0.5997	0.2964	0.9329	0.4496
0.7838	0.8987	0.5983	0.6112
0.0834	0.2331	0.7001	0.7889
0.9804	0.6435	0.6492	0.3297
0.0388	0.6932	0.3445	0.1165

Example 2

Study an example of SPDDE with right-end boundary layer [12]

$\varepsilon\gamma''(x) - \gamma'(x-\tau) - \gamma(x) = 0$, $0 \le x \le 1$, $-\tau \le x \le 0$
with b.c $\gamma(0) = 1$, $\gamma(1) = -1$

"The exact solution is given by" $\gamma(x) = \frac{((1+e^{\xi_2})e^{\xi_1 x} - (e^{\xi_1}+1)e^{\xi_2 x})}{(e^{\xi_2} - e^{\xi_1})}$

Where $\xi_1 = \frac{(1-\sqrt{1+4(\varepsilon+\tau)})}{2(\varepsilon+\tau)}$, $\xi_2 = \frac{(1+\sqrt{1+4(\varepsilon+\tau)})}{2(\varepsilon+\tau)}$

Table 7: : Compare between exact and proposed method for example for $\varepsilon = 0.01$, $\tau = 0.001$

Input x	Exact $y_a(x)$	suggested method $y_t(x)$	$E(x) = \mid y_t(x) - y_a(x) \mid$
0	0.000000000000	0	0.000000000000E+00
0.1	-0.100000000000	-0.100000000220	2.199999904473E-10
0.2	-0.200000000000	-0.200000000983	9.829999869648E-10
0.3	-0.300000000000	-0.300000000084	8.400002915465E-11
0.4	-0.400000000000	-0.400000000567	5.670000025049E-10
0.5	-0.500000000000	-0.500000000934	9.339999884617E-10
0.6	-0.600000000000	-0.600000000034	3.400002501763E-11
0.7	-0.700000000000	-0.700000000972	9.720000360147E-10
0.8	-0.800000000000	-0.800000000884	8.839999843246E-10
0.9	-0.900000000000	-0.900000000000	0.000000000000E+00
1	-1.000000000000	-0.010000000000	9.900000000000E-01

Table 8:The performance of the train with epoch and time

Train Function	Performance of train	Epoch	Time	MSE
Trainlm	2.7689	33	0:00:01	2.63461E-19

Table 9 : Initial weight and bias of the network for different training algorithm

Weights and bias for trainlm			
Net.IW{1,1}	Net.IU{1,1}	Net.LW{2,1}	Net.B{1}
0.8046	0.7959	0.4665	0.4888
0.9375	0.8574	0.9898	0.1444
0.9402	0.2363	0.8788	0.8995
0.6870	0.4858	0.5768	0.4858
0.3848	0.6566	0.5477	0.4758

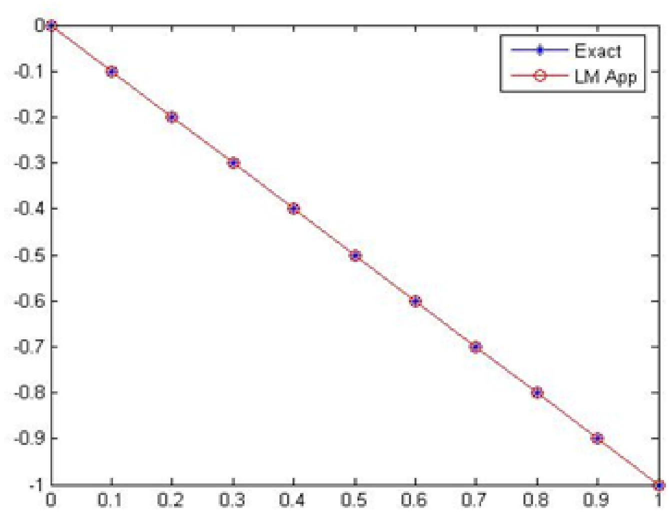

Figure 2. Exact and Approximate solution of example1 for $\varepsilon = 0.01$, $\tau = 0.001$

Example 3

Study an example of nonlinear SPDDE [11]

$\varepsilon\gamma'' + \gamma(\times)\gamma'(\times - \tau) - \gamma(\times) = 0$

under the interval with B.C $\gamma(\times) = 1$, $-\tau \leq \times \leq 0$, $\gamma(1) = 1$
The exact solution is not known.

Table 10: Numerical results of Example 3 for ε= 0.001 and different values of τ

	Numerical results		
x	$\tau_1 = 0.0002$	$\tau_2 = 0.0006$	$\tau_3 = 0.00009$
0	1.000000000000	1.000000000000	1.000000000000
0.1	0.394764663542	0.394764666396	0.394764663553
0.2	0.449379653098	0.449379653054	0.449379658923
0.3	0.411296485522	0.411296485296	0.411296485526
0.4	0.544886359863	0.544886359174	0.544886359811
0.5	0.526599731324	0.526599739362	0.526599731358
0.6	0.671276386097	0.671276386032	0.671276386087
0.7	0.634824163797	0.634824163732	0.634824155231
0.8	0.823957234665	0.823957234669	0.823957234880
0.9	0.915851135426	0.915851132865	0.915851135451
1	1.000000000000	1.000000000000	1.000000000000

Table 11:The performance of the train with epoch and time

Train Function	Performance of train	Epoch	Time
Trainlm τ_1	1.4327	20	0:00:01
Trainlm τ_2	2.9832	55	0:00:02
Trainlm τ_3	2.1232	12	0:00:00

Example 4

Study an example of nonlinear SPDDE [11]

$\varepsilon\gamma'' + 2\gamma'(\times - \tau) + e^{\gamma(\times)} = 0$ under the interval with b.c $\gamma(\times) = 0$, $-\tau \leq \times \leq 0$, $\gamma(1) = 0$

"The exact solution is not known".

Table 12: Numerical results of Example 3 for ε= 0.0001 and different values of τ

Numerical results			
x	$\tau_1 = 0.0001$	$\tau_2 = 0.0004$	$\tau_3 = 0.00008$
0	0.000000000000	0.000000000000	0.000000000000
0.1	−0.189388733852	−0.189388799527	−0.189388732597
0.2	−0.048362098365	−0.048362098366	−0.048362097759
0.3	−0.139655972443	−0.139655972034	−0.139655972482
0.4	−0.049812338768	−0.049812995632	−0.049812338777
0.5	−0.039655972443	−0.039655972086	−0.039655972491
0.6	−0.073784286498	−0.073784286495	−0.073784286494
0.7	−0.048630716853	−0.048630716822	−0.048630717734
0.8	−0.035049432073	−0.035049432116	−0.035049432882
0.9	−0.017086866521	−0.017086866595	−0.01708686623
1	0.000000000000	0.000000000000	0.000000000000

Table 13:The performance of the train with epoch and time

Train Function	Performance of train	Epoch	Time
Trainlm τ_1	7.6643	63	0:00:04
Trainlm τ_2	4.0433	32	0:00:02
Trainlm τ_3	3.7743	43	0:00:03

References

[1] Lange, C.G. and Miura, R.M. 1994,Singular perturbation analysis of boundary-value problems for differential-difference equations. v.small shifts with layer behavior, *SIAM J. Appl. Math.*, 54, 249-272.

[2] Derstine, M.W., Gibbs, F.A.H.H.M.,and Kaplan, D.L. 1982, Bifurcation gap in a hybrid optical system, *Phys. Rev. A*,26, 3720-3722.

[3] Glizer, V.Y. 2003, Asymptotic analysisand solution of a finite-horizon H1control problem for singularly-perturbed linear systems with small state delay, *J. Optim. Theory Appl.*, 117, 295-325.

[4] Lange, C.G. and Miura, R.M. 1994,Singular perturbation analysis of boundary-value problems for differential-difference equations. vi.Small shifts with rapid oscillations, *SIAM J. Appl. Math.*, 54, 273-283.

[5] Kadalbajoo, M.K. and Sharma, K.K. 2004, Numerical analysis of singularly perturbed delay differential equations with layer behavior, *Applied Mathematics and Computation*, 157,11-28.

[6] Kadalbajoo, M.K. and Sharma, K.K. 2008, A numerical method on finite difference for boundary value problems for singularly perturbed delay differential equations, *Applied Mathematics and Computation*, 197, 692-707.

[7] Gabil M. Amiraliyev and Erkan Cimen, 2010, Numerical method for a singularly perturbed convection–diffusion problem with delay, *Applied Mathematics and Computation*, 216, 2351-2359.

[8] Mohapatra, J. and Natesan, S. 2011, Uniformly convergent numerical method for singularly perturbed differential-difference equation using grid equidistribution, *International Journal for Numerical Methods in Biomedical Engineering*, 27, Issue 9, 1427-1445.

[9] Kadalbajoo, M. K. and Kumar, D. 2008, Fitted mesh B-spline collocation method for singularly perturbed differential–difference equations with small delay, *Applied Mathematics and Computation*, 204, 90-98.

[10] Tawfiq Luma N. M., Oraibi Yaseen A., (2013), "Fast Training Algorithms for Feed Forward Neural Networks", Ibn Al-Haitham Journal for Pure and Applied Science , NO. 1, Vol. 26:275-280.

[11] Gemechis File and Y. N. Reddy," Numerical Integration of a Class of Singularly Perturbed Delay Differential Equations with Small Shift ",Department of Mathematics, National Institute of Technology, Warangal 506 004, India, International Journal of Differential Equations Volume 2012, Article ID 572723, 12 pages

[12] M.K.Kadalbajoo, K.K. Sharma, A numerical analysis of singularly perturbed de-lay differential equations with layer behavior, Appl. Math. Comput. 157(2004), 11-28.