

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Multi Layer Neural Networks for Solving Initial Values Problems of Ordinary Differential Equations

Dr. Abdelaziz Hamad Elawad Elfadul

Associate Professor of Mathematics, Department of Mathematics, Faculty of Science, Al Baha University, Al Baha,
Saudi Arabia

ABSTRACT: This paper was focused on the development of a new method base on multi Layer neural network to solve initial values problems of ordinary differential Equation of first order. The concept of initial values problems of ordinary differential equations of first order and a new method base on multi Layer neural network has been proposed. We wrote a program on Mat lab software for initial values problems of ordinary differential equation and we call multi layer neural network for training process. After training neural network give us the solution of initial values problems of ordinary differential equations of first order. The performance of the proposed the new method has been measured on numerical examples. This is the first time a multi Layer neural network was use to resolve the problem of initial values problems of ordinary differential Equation of first order and the results are very encouraging for further investigation.

KEYWORDS: Initial values problems of ordinary differential equation, Multi Layer neural network

I. INTRODUCTION

In this Section we present the concept of initial values problem of ordinary differential equations for first order. Differential equations are an essential tool in a wide range of applications. Many phenomena can be modeled by a relationship between a function and its derivatives. Ordinary differential equations frequently occur as mathematical models in many branches of science, engineering and economy (Myron, B. and Eli.I,1998). Unfortunately it is seldom that these equations have solutions expressed in closed form, so it is common to obtain approximate solutions by means of numerical methods. Nowadays this can usually be achieved very inexpensively to high accuracy and with a reliable bound on the error between the analytical solutions and its numerical approximation. Normally a numerical method use to solve an initial values problem of ordinary differential equations which produces approximate solution at particular points using only the addition, subtraction, multiplication, and division and functional evaluation. In Section II we present related work. In Section III we talk about some classical method from literature reviews for solving initial values problem of ordinary differential equation first order so as to compare between our results and their classical method results. In Section IV, we present our Multi layer neural networks to solve our problem with numerical examples. In the final Section we give our conclusion

II. RELATED WORK

Our idea to used neural network back propagation to solve initial values problems of ordinary differential equation, comes from our works on scheduling and neural network areas and our knowledge about multi layer neural network, that multi layer neural can be solve any problem deals with inputs and outputs units (Abdelaziz and Baharom, 2011). Also Neural networks back propagation has ability to learn from the connection between the inputs and the outputs vectors. For this reason we use it to solve initial values problem of ordinary differential system equation first order. This may be first time in literature review using the concept of neural network back propagation system to solve this problem.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

III. SOME CALSSICAL METHODS

In this Section we present some classical and famous numerical methods to solve initial value problem of the first order from formula like,

$$y' = \frac{dy}{dx} = f(x, y) \quad (3.1)$$

In the interval $[a, b]$ if we know the initial value $y(a) = y_0$

3.1 Euler's Method

This method is simple one belong to the single step method use to solve the initial value problem of first order differential equation. In this Method normally divided the interval $[a, b]$ into sets of equal interval

$$x_n = a + nh, \quad \text{where } n = 0, 1, 2, 3, \dots \quad \text{and } h = x_n - x_{n-1} \quad (3.2)$$

We define the Euler's Method as below,

$$y_{n+1} = y_n + hy'_n \quad (3.3)$$

From equation (3.1), equation (3.2) become as follows,

$$y_{n+1} = y_n + hf(x_n, y_n) \quad (3.4)$$

3.2 Heun's Method:

This Method is an improvement of Euler's methods and is given by

$$y_{n+1} = y_n + \frac{h}{2}(y'_n + f(x_n + h, y_n + hy'_n)) \quad (3.5)$$

From, Equation (3.1) and (3.5) we get

$$y_{n+1} = y_n + \frac{h}{2}(f(x_n, y_n) + f(x_{n+1}, y_{n+1})) \quad (3.6)$$

IV. NEURAL NETWORKS

In This Section we propose a neural network method to solve our initial value problem of ordinary differential equation first order. Our ideas is to make a relation between initial values interval of the ordinary differential equation as inputs variables units and the result already given by exact solution as output units for training the neural network. After Training, we call up the neural network function from Mat Lab which was well trained to find our solutions results.

NEUMERICAL EXAMPLES

EXAMPLE 1.

Find $y(1)$ if, $y' = y - x$, $y(0) = 2$, $h = 0.1$ and $x_0 = 0$

In this case the exact solution for above equation is obtain from the equation

$$y = e^x + x + 1 \quad (4.1)$$

To solve this example we write a program depending on the exact solution after that we call the neural network back propagation function using Mat Lab language, to learn between the initial values interval as inputs vectors and values of exact solutions as outputs vectors. After that the neural network gives us the solutions in the form of output units

International Journal of Innovative Research in Science, Engineering and Technology

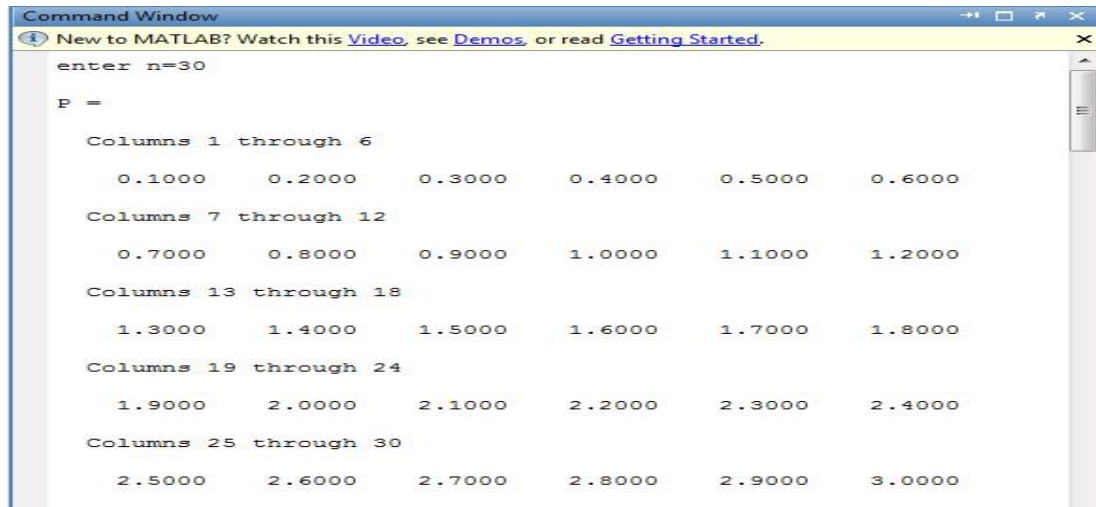
(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Tables 1. and **Table 2.** Illustrate the input units and output units which is a solution of neural networks respectively.

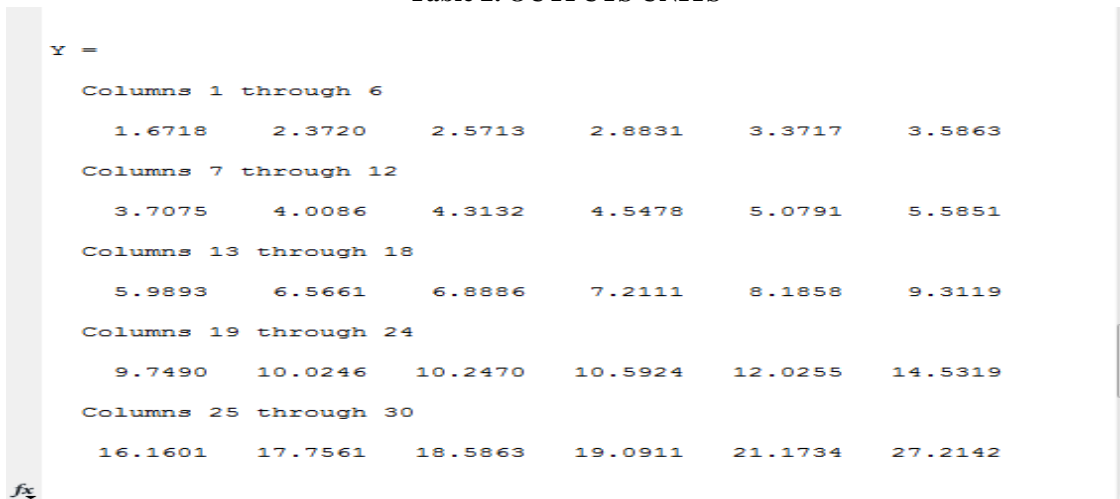
Table 1. INPUT UNITS



```

Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
enter n=30
P =
Columns 1 through 6
    0.1000    0.2000    0.3000    0.4000    0.5000    0.6000
Columns 7 through 12
    0.7000    0.8000    0.9000    1.0000    1.1000    1.2000
Columns 13 through 18
    1.3000    1.4000    1.5000    1.6000    1.7000    1.8000
Columns 19 through 24
    1.9000    2.0000    2.1000    2.2000    2.3000    2.4000
Columns 25 through 30
    2.5000    2.6000    2.7000    2.8000    2.9000    3.0000
    
```

Table 2. OUTPUTS UNITS



```

Y =
Columns 1 through 6
    1.6718    2.3720    2.5713    2.8831    3.3717    3.5863
Columns 7 through 12
    3.7075    4.0086    4.3132    4.5478    5.0791    5.5851
Columns 13 through 18
    5.9893    6.5661    6.8886    7.2111    8.1858    9.3119
Columns 19 through 24
    9.7490    10.0246    10.2470    10.5924    12.0255    14.5319
Columns 25 through 30
    16.1601    17.7561    18.5863    19.0911    21.1734    27.2142
    
```

Table 3. Illustrate the approximation solutions of classical methods and neural networks solution for example 1. When we compare between the results of classical methods with neural networks one. The neural networks method gets better results and near exact solution with very fast and even for large system until **n** equals more than **50**.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

TABLE 3. SOLUTIONS OF CALSSICAL METHODS AND NEURAL NETWORKS

x_n	Euler's Method	Heun's Method	Neural Method	Exact Solution
0.1	2.200	2.2050	1.6718	2.2052
0.2	2.4100	2.4210	2.3720	2.4214
0.3	2.6310	2.6492	2.5713	2.6499
0.4	2.8641	2.8909	2.8831	2.8918
0.5	3.1105	3.1474	3.3717	3.1487
0.6	3.3716	3.4204	3.5863	3.4221
0.7	3.6487	3.7115	3.7075	3.7138
0.8	3.9436	4.0227	4.0086	4.0255
0.9	4.3579	4.356	4.3132	4.3596
1.0	4.6937	4.7140	4.5478	4.7182
1.1	-----	-----	5.0791	5.0795
1.2	-----	-----	5.5551	5.5556
1.3	-----	-----	5.9893	5.9896
1.4	-----	-----	6.5661	6.5665
1.5	-----	-----	6.8886	6.8899

EXAMLE 2.

Find $y(1)$ for $y' = y$ and $y(0) = 1$ and $h = 0.1$

In this case the exact solution for above equation is obtain from the equation

$$y = e^x \quad (4.2)$$

To solve this example we write program depending on the exact solution as outputs units and the initial values as inputs units after that we call the neural network back propagation function using Mat Lab language, to set the result depending on initial value problem as the inputs vectors units. The neural network gives us the solution in the form of output units with very fast and even for large system. The neural method gets better result when we compare with classical one.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

TABLE 4. SOLUTION OF NEURAL NETWORKS.

```

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

initParam: (none)
performParam: (none)
trainParam: .show, .showWindow, .showCommandLine, .epochs,
            .time, .goal, .max_fail, .mem_reduc,
            .min_grad, .mu, .mu_dec, .mu_inc,
            .mu_max, .lr

weight and bias values:

            IW: {2x1 cell} containing 1 input weight matrix
            LW: {2x2 cell} containing 1 layer weight matrix
            b: {2x1 cell} containing 2 bias vectors

other:

            name: ''
            userdata: (user information)

Y =

Columns 1 through 9

    1.0973    1.3765    1.5819    1.4963    1.4091    1.7304    2.0171    2.0124    2.4645

Column 10

    2.8098
    
```

TABLE 5. SOLUTIONS OF CALSSICAL METHOD AND NEURAL NETWORKS

X_n	Euler's	Heun's	Neural Networks	Exact Solution
0.1	1.100	1.1050	1.0937	1.1052
0.2	1.2100	1.2210	1.3765	1.3214
0.3	1.3310	1.3492	1.5819	1.3499
0.4	1.4640	1.4909	1.4963	1.4918
0.5	1.6105	1.6474	1.4091	1.6487
0.6	1.7716	1.8204	1.7304	1.8221
0.7	1.9487	2.0115	2.0171	2.0138
0.8	2.1436	2.2227	2.0124	2.2255
0.9	2.3579	2.4561	2.4546	2.4596
1	2.5937	2.7140	2.8098	2.7183
1.1	-----		----- 2.9849	3.0042
1.2			----- 3.3644	3.3201
1.3			----- 3.6574	3.6693
1.4			----- 4.1250	4.0552
1.5			----- 4.4182	4.4817

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

V. CONCLUSION

The concept of initial values problem of ordinary differential equation of first order has been studied. New method base on multi Layer neural network has been proposed. We found that a new method of neural networks multi layer for solving initial values problem of ordinary differential equation of first order working very well when we compare it between the exact and the classical one. The performance of the new method has been measured on numerical examples. This is the first time a multi Layer neural network was use to resolve the problem of initial values problems of ordinary differential equation of first order and the results are very encouraging for second and higher orders investigation.

REFERENCES

- [1] Myron.B and Eli.I (1998) " Numerical analysis for applied science" John willey and sons.
- [2] Abdelaziz .H.A and Baharom.S (2011)."A neural network and jobs scheduling problems" Lap Lambert Academic
- [3] Abdelaziz Hamed (2002) "Application of neural network for solving Job scheduling problems" PhD thesis. UTM, Malaysia.
- [4] Bahrom Sanugi (1986) "New numerical strategies for initial value type ordinary differential I equations"
PhD Thesis, Loughborough University, United Kingdom.
- [5] Baker, K. and Scudder, G.). "On the Assignment of Optimal Due Date.", Journal of Operational Research Society. 40;pp. 93-95, 1989.
- [6] Baker, K.R. (1974). "Introduction to Sequencing and Scheduling". New York: John Wiley.
- [7] Baker, K. and Scudder, G. "Sequencing with Earliness and Tardiness Penalties: A review." Operations Research. 38; pp .22 – 36, 1990.
- [8] Lambert, J.D. (19731) Computational Methods in Ordinary Differential Equations, John Wiley and Sons, London.
- [9] Lambert, J.D., and S.T. Sigurdsson "Multistep methods with variable matrix coefficients," SIAM J. Numer. Anal., 9, pp.715-733, 1972.
- [10] Nordin Haji and Fatimah Said "Solving Single machine Scheduling problem with common due date " , Business Management Dynamics. 4; pp 63-72, 2011.
- [11] El-Bouri, A., Balakrishnan, S. and Popplewell, N. "Sequencing Jobs on a Single Machine: A Neural Network Approach", European Journal of Operational Research, 126; pp. 474-490, 2000..
- [12] Zahedi, F. "An Introduction to Neural Networks and Comparison with Artificial Intelligence and Expert Systems." Interfaces. 21; pp. 25-38, 1991.