Merson's Method for Solving nth Order Nonlinear Differential-Difference Equations

Atheer Jawad Kadhim
Department of Applied Sciences / University of Technology
Lahib Ibrahim Zaidan
Agriculture College / Babylon University

Abstract

The paper is devoted to propose a method with algorithms which have been written in Matlab language for constructing the numerical solutions for a system of *nonlinear differential-difference equations* using fourth-order-five-steps *Merson's method*. On the other hand, this method has been presented to find the numerical treatment for nth order nonlinear differential-difference equations where Merson method was efficiency for solving nonlinear problems. Comparison between the numerical and exact results has been given for three numerical examples for solving different types of differential-difference equations for conciliated the accuracy of the results of this method.

Keywords: Nonlinear Differential-Difference Equation, Numerical solution, Merson's method and Algorithms.

1. Introduction

The differential-difference equation "DDE" is an differential equation in which function having delay arguments where it is an equation in an unknown function y(t) and some of its derivatives are evaluated at arguments that differ in any of fixed number of values $\tau_1, \tau_2, ..., \tau_n$. DDE has been developed over twenty years ago. It has been much effort devoted to study *nonlinear differential-difference equations* of the form :

$$F(t, y(t), y(t-\tau_1), ..., y(t-\tau_k), y'(t), y'(t-\tau_1), ..., y'(t-\tau_k), y^{(n)}(t), y^{(n)}(t-\tau_1), ..., y^{(n)}(t-\tau_k)) = 0 \quad ... (1)$$

where F is a given function and $\tau_1, \tau_2, ..., \tau_k$ are given positive numbers called the "time delay" or "difference argument" [1].

In some literature, eq.(1) is called a differential equation with deviating argument [2,3], or an equation with time lag [4].

In this work, fourth order five steps Merson's method has been used to find the numerical solution of nth-order nonlinear differential-difference equation.

To facilitate the presentation of the material that followed, a brief review of some background on the nonlinear differential-difference equations and their types are given in the following section.

2. Differential-Difference Equation :

A differential-difference equation (DDE) is a difference equation in which various derivatives of the function y(t) can be present. DDE arises in many realistic models of problems in science, engineering and medicine, only in the last few years has much effort in behavior of solution of nonlinear differential-difference equation [5,6].

The general form of the nth-order differential-difference equation is given by eq.(1). The main difference between differential-difference equations and ordinary differential equations is the kind of initial condition that should be used in differential-difference equation differs from ordinary differential equation so that one should specify in differential difference equations an initial function on some interval of length τ, say $[t_0-\tau,t_0]$ and then try to find the solution of equation (1) for all $t \geq t_0$ [2,7,8].

Differential-difference equation is classified into three types [1,6,7] :-

- Equation (1) is called a *Neutral* type if the highest-order derivative of unknown function appears both with and without difference argument.
- Equation (1) is called *Retarded* type if the highest-order derivative of unknown function appears without difference argument.
- All other DDE (1) with *advanced* types (sometimes called mixed types), i.e. a combination of the previous two types.

3. Merson's Method:

Merson's method provides efficient mean for the solution of the many problem arising in various fields of science and engineering. It uses only the information from the last step computed; therefore, it is called single-step method [7,9].

Merson's method has been widely used for nonlinear problems. Merson [10] uses this method to find the numerical solution of nonlinear ordinary differential equations.

Consider the following first order nonlinear differential equation :

$$\frac{dy}{dt}=f(t,y) \qquad \text{with initial condition} \qquad y(t_0)=y_0 \qquad \ldots (2)$$

In order to solve the above equation numerically, a fourth order method which is valid for a nonlinear differential equation is derived by Merson [9,10] as:

$$\left.\begin{aligned}
&y_{n+1} = y_n + \frac{h}{6}\left(M_1 + 4M_4 + M_5\right) \quad , \\
&where \\
&M_1 = f(t_n, y_n), \\
&M_2 = f(t_n + \tfrac{1}{3}h, y_n + \tfrac{1}{3}hM_1), \\
&M_3 = f\left(t_n + \tfrac{1}{3}h, y_n + \tfrac{1}{6}h(M_1 + M_2)\right), \\
&M_4 = f\left(t_n + \frac{1}{2}h, y_n + \frac{h}{8}(M_1 + 3M_3)\right), \\
&M_5 = f\left(t_n + h, y_n + \frac{h}{2}M_1 - \frac{3h}{2}M_3 + 2hM_4)\right).
\end{aligned}\right\} \quad \ldots (3)$$

In this work the idea of Merson's method has been used to treat both nonlinear differential-difference equation as well as system of nonlinear differential-difference equations numerically.

3.1 The Solution of a Single First Order Nonlinear Differential-Difference Equation Using Merson Method:

In this subsection Merson method has been used to find the numerical solution for a 1st order nonlinear differential-difference equation.

Consider the first order nonlinear differential-difference equation (DDE) :-

$$y'(t) = f(t, y(t), y(t-\tau), y'(t-\tau)) \quad , \quad t \in [t_0, \infty) \qquad \ldots (4)$$

with initial function

$$y(t) = \phi(t) \quad \text{for} \quad t_0 - \tau \le t \le t_0$$

eq.(4) may be solved if we use the initial function as:

$$y'(t) = f(t, y(t), \phi(t-\tau), \phi'(t-\tau)) \qquad \ldots (5)$$

with initial condition

$$y(t_0) = \phi(t_0)$$

For solving the above nonlinear DDE by Merson method, eq.(3) is written as:

$$\left.\begin{array}{l} y(t_{j+1}) = y(t_j) + \frac{h}{6}\left(M_1 + 4M_4 + M_5\right), \\ where \\ \\ M_1 = f\left(t_j, y(t_j), \phi(t_j - \tau), \phi'(t_j - \tau)\right), \\ M_2 = f\left(t_j + \frac{1}{2}h, y(t_j) + \frac{1}{2}hM_1, \phi((t_j + \frac{1}{2}h) - \tau), \phi'((t_j + \frac{1}{2}h) - \tau)\right), \\ M_3 = f\left(t_j + \frac{1}{3}h, y(t_j) + \frac{1}{6}h(M_1 + M_2), \phi((t_j + \frac{1}{3}h) - \tau), \phi'((t_j + \frac{1}{3}h) - \tau)\right), \\ M_4 = f\left(t_j + \frac{1}{2}h, y(t_j) + \frac{h}{8}(M_1 + 3M_3), \phi((t_j + \frac{1}{2}h) - \tau), \phi'((t_j + \frac{1}{2}h) - \tau)\right), \\ M_5 = f\left(t_j + h, y(t_j) + \frac{h}{2}M_1 - \frac{3h}{2}M_3 + 2hM_4, \phi((t_j + h) - \tau), \phi'((t_j + h) - \tau)\right) \end{array}\right\} \quad \ldots (6)$$

for each $j = 0,1,...., m$, where (m + 1) is the number of points $(t_0, t_1, ..., t_m)$.

The numerical solution using fourth order ***Merson method*** of a **f**irst order ***n****onlinear* ***DDE*** in eq.(4) can be summarized by the following algorithm :

MM-FNDDE Algorithm :

Step 1: Input the step length (h) .

Step 2: Input the initial value t_0 .

Step 3: Set j=0

Step 4: Compute M_1 as in eq.(6) .

Step 5: Compute M_2 as in eq.(6) .

Step 6: Compute M_3 as in eq.(6) .

Step 7: Compute M_4 as in eq.(6) .

Step 8: Compute M_5 as in eq.(6) .

Step 9: Compute :

$$y(t_{j+1}) = y(t_j) + \frac{h}{6}\left(M_1 + 4M_4 + M_5\right)$$

Step 10: Put $j = j+1$

Step 11: If $j = m$ then stop.

Else go to (step 4)

3.2 The Solution of a System of First Order Nonlinear Differential-Difference Equations Using Merson Method:

Consider the following system of the first order nonlinear differential-difference equations :

$$\frac{dy_i(t)}{dt} = f_i(t, y_1(t),..., y_n(t), y_1(t-\tau_1),..., y_n(t-\tau_n), y_1'(t-\tau_1),..., y_n'(t-\tau_n)), \quad ... (7)$$

where $t \in [t_0, \infty)$ and f_i , i=1,2,…,n denotes the ith nonlinear functional relationship.

The initial functions of eq.(7) are:

$$\begin{aligned} y_1(t) &= \phi_1(t) \quad for \quad t_0 - \tau_1 \le t \le t_0 \\ &\vdots \qquad\qquad\qquad \vdots \\ y_n(t) &= \phi_n(t) \quad for \quad t_0 - \tau_n \le t \le t_0 \end{aligned} \quad \ldots (8)$$

Equation (7) can be solved by using Merson method if the initial functions in eq.(8) were used as follows:

$$y_i'(t) = f_i(t, y_1(t),..., y_n(t), \phi_1(t-\tau_1),..., \phi_n(t-\tau_n), \phi_1'(t-\tau_1),..., \phi_n'(t-\tau_n)), \text{ i=1,…,n} \ ... (9)$$

with initial conditions

$$y_1(t_0) = \phi_1(t_0), ..., y_n(t_0) = \phi_n(t_0).$$

For treating a system of a first order of nonlinear DDE's by Merson method eq.(3) is written as:

$$y_i(t_{j+1}) = y_i(t_j) + \frac{h}{6}\left(M_{1i} + 4M_{4i} + M_{5i}\right) \quad \ldots (10)$$

where

$$
\left.\begin{aligned}
M_{1i} &= f_i(t_j, y_1(t_j),\dots, y_n(t_j), \phi_1(t_j-\tau_1),\dots, \phi_n(t_j-\tau_n), \phi_1'(t_j-\tau_1),\dots, \phi_n'(t_j-\tau_n)) \\
M_{2i} &= f_i\begin{pmatrix} t_j+\tfrac{1}{3}h, y_1(t_j)+\tfrac{1}{3}hM_{11},\dots, y_n(t_j)+\tfrac{1}{3}hM_{1n}, \phi_1(t_j+\tfrac{1}{3}h-\tau_1),\dots, \\ \phi_n(t_j+\tfrac{1}{3}h-\tau_n), \phi_1'(t_j+\tfrac{1}{3}h-\tau_1),\dots, \phi_n'(t_j+\tfrac{1}{3}h-\tau_n) \end{pmatrix} \\
M_{3i} &= f_i\begin{pmatrix} t_j+\tfrac{1}{3}h, y_1(t_j)+\dfrac{h}{6}(M_{11}+M_{21}),\dots, y_n(t_j)+\dfrac{h}{6}(M_{1n}+M_{2n}), \\ \phi_1(t_j+\tfrac{1}{3}h-\tau_1),\dots, \phi_n(t_j+\tfrac{1}{3}h-\tau_n), \phi_1'(t_j+\tfrac{1}{3}h-\tau_1),\dots, \phi_n'(t_j+\tfrac{1}{3}h-\tau_n) \end{pmatrix} \\
M_{4i} &= f_i\begin{pmatrix} t_j+\dfrac{1}{2}h, y_1(t_j)+\dfrac{h}{8}(M_{11}+3M_{31}),\dots, y_n(t_j)+\dfrac{h}{8}(M_{1n}+3M_{3n}), \\ \phi_1(t_j+\tfrac{1}{2}h-\tau_1),\dots, \phi_n(t_j+\tfrac{1}{2}h-\tau_n), \phi_1'(t_j+\tfrac{1}{2}h-\tau_1),\dots, \phi_n'(t_j+\tfrac{1}{2}h-\tau_n) \end{pmatrix} \\
M_{5i} &= f_i\begin{pmatrix} t_j+h, y_1(t_j)+\dfrac{h}{2}M_{11}-\dfrac{3h}{2}M_{31}+2hM_{41},\dots, y_n(t_j)+\dfrac{h}{2}M_{1n}-\dfrac{3h}{2}M_{3n}+ \\ 2hM_{4n},\dots, \phi_1(t_j+h-\tau_1),\dots, \phi_n(t_j+h-\tau_n), \phi_1'(t_j+h-\tau_1),\dots, \phi_n'(t_j+h-\tau_n) \end{pmatrix}.
\end{aligned}\right\}
\quad \dots (11)
$$

for each i=1,2,…,n and j=0,1,…, m .

The following (MM-SYNDDE) algorithm summarizes the steps for finding the numerical solution by using ***M**erson **m**ethod* for a ***sy**stem* of ***n**onlinear **d**ifferential-**d**ifference **e**quations* in eq.(7).

MM-SYNDDE Algorithm :

Step 1: Input the step length (h) .

Step 2: Input the initial value t_0.

Step 3: Set j=0

Step 4: For each i=1,2,…,n compute M_{1i} in eq.(11).

Step 5: $\forall\, i=1,2,\dots,n$ compute M_{2i} in eq.(11).

Step 6: $\forall\, i=1,2,\dots,n$ compute M_{3i} in eq.(11).

Step 7: $\forall\, i=1,2,\dots,n$ compute M_{4i} in eq.(11).

Step 8: $\forall\, i=1,2,\dots,n$ compute M_{5i} in eq.(11).

Step 9: $\forall\ i = 1,2,...,n$ compute :

$$y_i(t_{j+1}) = y_i(t_j) + \frac{h}{6}\left(M_{1i} + 4M_{4i} + M_{5i}\right)$$

Step 10: Put $j = j+1$

Step 11: If $j = m$ then stop.

Else go to (step 4)

3.3 The Solution of n^th Order Nonlinear Differential-Difference Equations Using Merson Method:

The general form of nth–order nonlinear differential-difference equation is:

$$y^{(n)}(t) = f(t, y(t), y'(t),...., y^{(n-1)}(t), y(t-\tau), y'(t-\tau),...., y^{(n-1)}(t-\tau)), \quad t \ge t_0 \qquad (12)$$

with initial functions:

$$\left.\begin{array}{l} y(t) = \phi(t) \\ y'(t) = \phi'(t) \\ \quad \vdots \\ y^{(n-1)}(t) = \phi^{(n-1)}(t) \end{array}\right\} \quad for \quad t_0 - \tau \le t \le t_0$$

where $\phi(t)$ and its first (n-1) derivatives $\phi'(t),...., \phi^{n-1}(t)$ are continuous on the interval $[t_0 - \tau, t_0]$

Obviously, the n^th order equation (12) with difference argument may be replaced by a system of nth-equation of first order differential-difference equations as follows:

Let

$$\left.\begin{array}{l} x_1(t) = y(t) \\ x_2(t) = y'(t) \\ \quad \vdots \\ x_{n-1}(t) = y^{(n-2)}(t) \\ x_n(t) = y^{(n-1)}(t) \end{array}\right\} \qquad \ldots (13)$$

Then, one gets the following system of the first order equations :

$$\left.\begin{array}{l} x_1'(t) = x_2(t) \\ x_2'(t) = x_3(t) \\ \quad \vdots \\ x_{n-1}'(t) = x_n(t) \\ x_n'(t) = f(t, x_1(t),...., x_n(t), x_1(t-\tau),...., x_n(t-\tau)) \end{array}\right\} \qquad \ldots (14)$$

The above system eq.(14) of the first order nonlinear DDE's can be treated numerically by using Merson method as it is prescribed in section (3.2).

4. Numerical Examples :

Example (1):

Consider the following nonlinear neutral differential-difference equation of the first order:

$$y'(t) = \cos t\left(1 + \sin(t(1 - \cos^2 t))\right) + y(t)y'(ty(t)^2) - \sin(t + t\sin^2 t) \quad t \geq 0$$

with initial function :

$$y(t) = \sin t \qquad t \leq 0 .$$

The exact solution of the above nonlinear DDE is:

$$y(t) = \sin t \qquad 0 \leq t \leq 1 .$$

When the algorithm (MM-FNDDE) is applied, table (1) presents the comparison between the exact and numerical solution using Merson's method for m=10, h=0.1 , $t_j = jh$, $j = 0,1,..., m$ and m=100, h=0.01 , depending on least square error (L.S.E.).

Table (1) The solution of DDE for Ex.(1).

t	***Exact***	***Merson Method (MM-FNDDE) y(t)***	
		h=0.1	***h=0.01***
0	0	0	0
0.1	0.0998	0.0998	0.0998
0.2	0.1987	0.1987	0.1987
0.3	0.2955	0.2955	0.2955
0.4	0.3894	0.3894	0.3894
0.5	0.4794	0.4794	0.4794
0.6	0.5646	0.5646	0.5646
0.7	0.6442	0.6442	0.6442
0.8	0.7174	0.7174	0.7174
0.9	0.7833	0.7833	0.7833
1	0.8415	0.8415	0.8415
L.S.E.		**0.373e-15**	**0.333e-22**

Example (2):

Consider the following system of the two first order of nonlinear differential-difference equations:

$$y_1'(t) = 1 - y_1(e^{(1-\frac{1}{t})}) \qquad 0 < t \le 2$$

$$y_2'(t) = -y_2(e^{(1-y_2(t))})\, y_2^2(t)\, e^{(1-y_2(t))} \qquad 0 < t \le 2$$

where the initial functions are:

$$\left.\begin{aligned} y_1(t) &= \ln t \\ y_2(t) &= \frac{1}{t} \end{aligned}\right\} \quad t \ge 1$$

The exact solutions are:

$$\begin{pmatrix} exact_1 \\ exact_2 \end{pmatrix} = \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} = \begin{pmatrix} \ln t \\ \frac{1}{t} \end{pmatrix} \quad , \; 0 < t \le 2 \quad .$$

Example (2) is solved by using Merson's method. Table (2) gives a summary of the numerical solution, exact solution and the least square errors (L.E.S.) by applying (MM-SYNDDE) algorithm for m=10, h=0.1, $t_j = t_0 + jh$, $t_0 = 1,\ j = 0,1,..., m$.

Table (2) The solution of nonlinear DDE's for example (2)

t	*Exact₁*	*Merson* $y_1(t)$	*Exact₂*	*Merson* $y_2(t)$
1	**0**	**0**	**1**	**1**
1.1	**0.0953**	**0.0953**	**0.9091**	**0.9091**
1.2	**0.1823**	**0.1823**	**0.8333**	**0.8333**
1.3	**0.2624**	**0.2624**	**0.7692**	**0.7692**
1.4	**0.3365**	**0.3365**	**0.7143**	**0.7143**
1.5	**0.4055**	**0.4055**	**0.6667**	**0.6667**
1.6	**0.4700**	**0.4700**	**0.6250**	**0.6250**
1.7	**0.5306**	**0.5306**	**0.5882**	**0.5882**
1.8	**0.5878**	**0.5878**	**0.5556**	**0.5556**
1.9	**0.6419**	**0.6419**	**0.5263**	**0.5263**
2	**0.6931**	**0.6931**	**0.5000**	**0.5000**
L.S.E		**0.258e-13**	**L.S.E**	**0.461e-12**

Example (3):

Consider the nonlinear differential- difference equation of the third order:-

$$y^{(3)}(t) = y(t-1)y''(t-2) + y(t-2)y''(t-1) + 2y'(t-1)y'(t-2) \qquad t \ge 0$$

with initial functions :

$$y(t) = e^{(t+2)} \qquad t \le 0$$

$$y'(t) = e^{(t+2)} \qquad t \le 0$$

$$y''(t) = e^{(t+2)} \qquad t \le 0$$

and the exact solution:

$$y(t) = \tfrac{1}{2}e^{(2t+1)} + \frac{t^2}{2}(-2e+e^2) + t(-e+e^2) - \tfrac{1}{2}e + e^2 \qquad t \ge 0$$

The above differential-difference equation can be replaced by a system of three first order DDE equations.

$$x_1'(t) = x_2(t), \qquad t \ge 0$$

$$x_2'(t) = x_3(t), \qquad t \ge 0$$

$$x_3'(t) = x_1(t-1)x_3(t-2) + x_1(t-2)x_3(t-1) + 2x_2(t-1)x_2(t-2) \quad t \ge 0$$

with initial conditions:

$$x_1(t) = e^{(t+2)} \qquad t \le 0$$

$$x_2(t) = e^{(t+2)} \qquad t \le 0$$

$$x_3(t) = e^{(t+2)} \qquad t \le 0$$

which has the exact solution:

$$\begin{pmatrix} exact_1 \\ exact_2 \\ exact_3 \end{pmatrix} = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} \tfrac{1}{2}e^{(2t+1)} + \frac{t^2}{2}(-2e+e^2) + t(-e+e^2) - \tfrac{1}{2}e + e^2 & t \ge 0 \\ e^{(2t+1)} + t(-2e+e^2) - e + e^2 & t \ge 0 \\ 2e^{(2t+1)} - 2e + e^2 & t \ge 0 \end{pmatrix}$$

The Merson’s method was calculated for m=10, h=0.1 and m=100, h=0.01 using (MM-SYNDDE) algorithm. Table (3) presents the comparison between the numerical and exact solution depending on the least square error (L.S.E.), where $t_j = jh\,,\quad j = 0,1,\ldots,m$.

Table (3) The solution of nonlinear DDE’s for example (3)

t	***Exact₁***	***Merson Method (MM-SYNDDE) x₁(t)***		***Exact₂***	***Merson Method (MM-SYNDDE) x₂(t)***		***Exact₃***	***Merson Method (MM-SYNDDE) x₃(t)***	
		h=0.1	***h=0.01***		***h=0.1***	***h=0.01***		***h=0.1***	***h=0.01***
0	**7.3891**	**7.3891**	**7.3891**	**7.3891**	**7.3891**	**7.3891**	**7.3891**	**7.3891**	**7.3891**
0.1	**8.1668**	**8.1668**	**8.1668**	**8.1861**	**8.1861**	**8.1861**	**8.5927**	**8.5927**	**8.5927**
0.2	**9.0307**	**9.0307**	**9.0307**	**9.1165**	**9.1165**	**9.1165**	**10.0629**	**10.0629**	**10.0629**
0.3	**9.9955**	**9.9955**	**9.9955**	**10.2096**	**10.2096**	**10.2096**	**11.8586**	**11.8586**	**11.8586**
0.4	**11.0792**	**11.0792**	**11.0792**	**11.5014**	**11.5014**	**11.5014**	**14.0518**	**14.0518**	**14.0518**
0.5	**12.3039**	**12.3039**	**12.3039**	**13.0361**	**13.0361**	**13.0361**	**16.7306**	**16.7306**	**16.7306**
0.6	**13.6963**	**13.6963**	**13.6963**	**14.8673**	**14.8673**	**14.8673**	**20.0025**	**20.0025**	**20.0025**
0.7	**15.2894**	**15.2894**	**15.2894**	**17.0607**	**17.0607**	**17.0607**	**23.9988**	**23.9988**	**23.9988**
0.8	**17.1232**	**17.1232**	**17.1232**	**19.6965**	**19.6965**	**19.6965**	**28.8800**	**28.8800**	**28.8800**
0.9	**19.2467**	**19.2467**	**19.2467**	**22.8727**	**22.8727**	**22.8727**	**34.8418**	**34.8418**	**34.8418**
1	**21.7197**	**21.7197**	**21.7197**	**26.7088**	**26.7088**	**26.7088**	**42.1236**	**42.1236**	**42.1236**
L.S.E.		**0.192e-9**	**0.17e-16**	**L.S.E.**	**0.265e-9**	**0.22e-16**	**L.S.E.**	**0.93e-10**	**0.75e-17**

5. Conclusion:

Merson method has been presented for solving nonlinear differential-difference equations. The results show a marked improvement in the least square errors (L.S.E). From solving some numerical examples the following points are listed :

1- Merson method gives a better accuracy and consistent to the solution of n^{th} order nonlinear differential-difference equation by reducing the equation to a system of first order nonlinear DDE's.

2- Merson method gives qualified way for solving first order nonlinear differential-difference equation as well as system of first order nonlinear differential-difference equations.

3- The good approximation depends on the size of *h*, if *h* is decreased then the number of points (knots) increases and the L.S.E. approaches to zero where this gives the advantage in numerical computation.

References:

1. Bellman, R. and Cooke, K.L., "Differential, Difference Equations " Academic press, Inc, New York 1963.
2. Driver, R.D., "Ordinary and Delay Differential Equations", Springer verlag, New York, Inc. 1977.
3. Elּsgol'ts, L.E.; "Introduction to the theory of Differential Equations with Deviating Arguments", Holden. Day, Inc. 1966.
4. Abood, B.N. "On the Numerical Solution of the Delay Differential Equations", Ph.D. Thesis, College of science, Al- Mustansiriya University, IRAQ, 2004.
5. Byung-Kookim;" Piecewise linear Dynamic systems with time Delay" Ph.D. Thesis, California Institute of Technology, Pasadena, California, 1972.
6. Hussain A.M. ;First order Nonlinear Neutral Delay Differential Equations ,J.of Um-Saluma for science vol 1, No 2, PP.347-349, 2004.
7. Hartung F. and Gyori I., "On Numerical Solution for a Class of Nonlinear Differential-Difference Equation with Time and State-Dependent Delays", http:/www.ma.man.ac.uk/nare parts, 2008.
8. Baker, C.H. , "Retarded Differential Equations", http:/ www.ma.man.ac.uk/nare ports, 2009.
9. Peitgen O. and Glash H. " Numerical Solution of Nonlinear Equations", Springer-Verlag, Berlin Heidel berg, N.Y., 2002.
10. Lambert, J.D., "Computational Methods in Ordinary Differential Equations", John Wiley & Sons Ltd., New York, 1979.

طريقة ميرسن لحل معادلات الفروق- التفاضلية اللاخطية من الرتبة *n*

أثير جواد كاظم
قسم العلوم التطبيقية - الجامعة التكنولوجية
لهيب ابراهيم زيدان
كلية الزراعة – جامعة بابل

الخلاصة :

يقدم البحث طريقة مقترحة مع اشتقاق خوارزميات تمت برمجتها بلغة (Matlab) لإيجاد الحل العددي لمعادلات الفروق التفاضلية اللاخطية باستخدام طريقة ميرسن من الرتبة الرابعة. حيث تمت معالجة معادلة الفروق- التفاضلية اللاخطية من الرتبة الأولى عددياً مثلما لمنظومة من معادلات الفروق- التفاضلية اللاخطية باستخدام طريقة ميرسن من الرتبة الرابعة. ومن ناحية اخرى طورت هذه الطريقة لإيجاد النتائج العددية لمعادلات الفروق- التفاضلية اللاخطية من الرتبة n حيث من الممكن ملاحظة كفاءة الطريقة و دقة حساباتها في حل المعادلات اللاخطية. كما تمت مقارنة النتائج العددية و الحقيقية لأنواع مختلفة من معادلات الفروق- التفاضلية اللاخطية من خلال ثلاثة امثلة وقد تم الحصول على نتائج دقيقة.