Mathematical & Computational Applications, Vol. 7, No. 1, pp. 41-52, 2002
©Association for Scientific Research

NUMERICAL SOLUTION OF FUZZY DIFFERENTIAL EQUATION

S. Abbasbandy
Department of Mathematics, Imam Khomeini Int. University, Qazvin, Iran.
(abbasbandy@yahoo.com)

T. Allah Viranloo
Department of Mathematics, Science and Research Branch,Islamic Azad University,
Tehran, Iran. (alahviranlo@yahoo.com)

Abstract-In this paper numerical algorithms for solving fuzzy ordinary differential equations are considered. A scheme based on the 2nd Taylor method in detail is discussed and this is followed by a complete error analysis. The algorithm is illustrated by solving some linear and nonlinear fuzzy cauchy problems.

Keywords-Fuzzy Differential Equation, 2nd Taylor Method, Fuzzy Cauchy Problem.

1. INTRODUCTION

The topics of numerical methods for solving fuzzy differential equations have been rapidly growing in recent years. The concept of fuzzy derivative was first introduced by S.L. Chang, L.A. Zadeh in [1]. It was followed up by D.Dubois, H. Prade in [2], who defined and used the extension principle. Other methods have been discussed by M.L Puri, D.A. Ralescu in [3] and R. Goetschel, W. Voxman in [4] The fuzzy differential equation and the initial value problem were regularly treated by O. Kaleva in [5] and [6],by S. Seikkala in [7]. The numerical method for solving fuzzy differential equations is introduced by M. Ma, M. Friedman, A. Kandel in [9] by the standard Euler method. The structure of this paper organize as follows:

In section 2 some basic results on fuzzy numbers and definition of a fuzzy derivative, which have been discussed by S. Seikkala in [9] are given. In section 3 we define the problem, this is a fuzzy cauchy problem whose numerical solution is the main interest of this work. The numerically solving fuzzy differential equation by 2nd Taylor method is discussed in section 4. The proposed algorithm is illustrated by solving some examples in section 5 and conclusion is in section 6.

2. PRELIMINARIES

Consider the initial value problem

$$\begin{cases} x'(t) = f(t, x(t)); \quad a \le t \le b, \\ x(a) = \alpha. \end{cases} \tag{1}$$

Let $Y(t)$ be the exact solution of (1) and $Y(t_i)$ approximated by $y_i = y(t_i)$, which in 2nd Taylor method

$$y_{i+1} = y_i + hT(t_i, y_i), \quad i = 0,1,\dots,N-1, \tag{2}$$

and

$$T(t_i, y_i) = f(t_i, y_i) + \frac{h}{2} f'(t_i, y_i), \tag{3}$$

where

$$a = t_0 \le t_1 \le \ldots \le t_N = b \text{ and } h = \frac{b-a}{N} = t_{i+1} - t_i. \tag{4}$$

A triangular fuzzy number v, is defined by three numbers $a_1 < a_2 < a_3$ where the graph of $v(x)$, the membership function of the fuzzy number v, is a triangle with base on the interval $[a_1, a_3]$, and vertex at $x = a_2$. We specify v as $(a_1 / a_2 / a_3)$. We will write: (1)$v > 0$ if $a_1 > 0$; (2)$v \ge 0$ if $a_1 \ge 0$; (3)$v < 0$ if $a_3 < 0$; and (4)$v \le 0$ if $a_3 \le 0$. Let E be the set of all upper semicontinuous normal convex fuzzy numbers with bounded r-level sets It means that if $v \in E$ then the r-evel set

$$[v]_r = \{s \,|\, v(s) \ge r\}, \quad 0 < r \le 1,$$

is a closed bounded interval which is denoted by

$$[v]_r = [v_1(r), v_2(r)].$$

Let I be a real interval. A mapping $x: I \to E$ is called a fuzzy process and its r-level set is denoted by

$$[x(t)]_r = [x_1(t;r), x_2(t;r)], \quad t \in I, \quad r \in (0,1].$$

The derivative $x'(t)$ of a fuzzy process x is defined by

$$[x'(t)]_r = [x'_1(t;r), x'_2(t;r)], \quad t \in I, \quad r \in (0,1],$$

provided that this equation defines a fuzzy number, as in Sikkala [7].

3. A FUZZY CAUCHY PROBLEM

Consider the fuzzy initial value problem

$$\begin{cases} x'(t) = f(t, x(t)), \; t \in I = [0, T], \\ x(0) = x_0, \end{cases} \tag{5}$$

where f is a continuous mapping from $R_+ \times R$ into R and $x_0 \in E$ with r-level set

$$[x_0]_r = [x_1(0;r), x_2(0;r)], \quad r \in (0,1].$$

The extension principle of Zadeh leads to the following definition of $f(t,x)$ when $x = x(t)$ is a fuzzy number

$$f(t,x)(s) = \sup\{x(\tau) \,|\, s = f(t,\tau)\}, \quad s \in R.$$

It follows that

$$[f(t,x)]_r = [f_1(t,x;r), f_2(t,x;r)], \quad r \in (0,1],$$

where

$$f_1(t,x;r)=\min\{f(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\},$$
$$f_2(t,x;r)=\max\{f(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\}. \qquad (6)$$

The mapping $f(t,x)$ is a fuzzy process and the derivative $f'(t,x)$ is defined by

$$[f'(t,x)]_r=[f'_1(t,x;r),f'_2(t,x;r)],\quad r\in(0,1],$$

provided that this equation defines a fuzzy number $f'(t,x)\in E$ where

$$f'_1(t,x;r)=\min\{f'(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\},$$
$$f'_2(t,x;r)=\max\{f'(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\}. \qquad (7)$$

THEOREM 3.1- *Let f satisfy*

$$|f(t,v)-f(t,\bar{v})|\le g(t,|v-\bar{v}|),\quad t\ge 0,\quad v,\bar{v}\in R,$$

where $g:R_+\times R_+\to R_+$ is a continuous mapping such that $r\to g(t,r)$ is nondecreasing, the initial value problem

$$u'(t)=g(t,u(t)),\quad u(0)=u_0, \qquad (8)$$

has a solution on R_+ for $u_0>0$ and that $u(t)=0$ is the only solution of (8) for $u_0=0$. Then the fuzzy initial value problem (6) has a unique fuzzy solution.

Proof [7].

Since

$$x''(t)=f'(t,x)=\frac{\partial f}{\partial t}(t,x(t))+\frac{\partial f}{\partial x}(t,x(t)).f(t,x(t)). \qquad (9)$$

Hence by (7) it follows that

$$f'_1(t,x;r)=\min\{\frac{\partial f}{\partial t}(t,u)+\frac{\partial f}{\partial u}(t,u).f(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\},$$
$$f'_2(t,x;r)=\max\{\frac{\partial f}{\partial t}(t,u)+\frac{\partial f}{\partial u}(t,u).f(t,u)\,|u\in[x_1(t;r),x_2(t;r)]\}. \qquad (10)$$

4. 2ND TAYLOR METHOD

Let the exact solution $[Y(t)]_r=[Y_1(t;r),Y_2(t;r)]$ is approximated by some $[y(t)]_r=[y_1(t;r),y_2(t;r)]$. The 2nd Taylor method is based on the

$$x(t+h;r)=x(t;r)+hx'(t;r)+\frac{h}{2}x''(t;r), \qquad (11)$$

where $x(t;r)$ is Y_1 and Y_2 alternatively. We define

$$F[t,x;r]=f_1(t,x;r)+\frac{h}{2}f'_1(t,x;r),$$
$$G[t,x;r]=f_2(t,x;r)+\frac{h}{2}f'_2(t,x;r). \qquad (12)$$

The exact and approximate solutions at $t_n, 0\le n\le N$ are denoted by $[Y(t_n)]_r=[Y_1(t_n;r),Y_2(t_n;r)]$ and $[y(t_n)]_r=[y_1(t_n;r),y_2(t_n;r)]$ respectively. The solution is calculated by grid points at (4). By 2nd Taylor method and substituting Y_1 and Y_2 in (11) and considering (12) we have

$$Y_1(t_{n+1};r)\approx Y_1(t_n;r)+hF[t_n,Y(t_n);r],$$
$$Y_2(t_{n+1};r)\approx Y_2(t_n;r)+hG[t_n,Y(t_n);r], \qquad (13)$$

hence we have

$$y_1(t_{n+1};r)=y_1(t_n;r)+hF[t_n,y(t_n);r],$$
$$y_2(t_{n+1};r)=y_2(t_n;r)+hG[t_n,y(t_n);r], \qquad (14)$$

where

$$y_1(0;r)=x_1(0;r) \quad , \quad y_2(0;r)=x_2(0;r).$$

The following lemmas will be applied to show convergence of these approximates ,i.e.,

$$\lim_{h\to 0} y_1(t;r)=Y_1(t;r),$$
$$\lim_{h\to 0} y_2(t;r)=Y_2(t;r).$$

LEMMA 4.1- *Let a sequence of numbers* $\{W_n\}_{n=0}^N$ *satisfy*

$$|W_{n+1}|\le A|W_n|+B,\quad 0\le n\le N-1,$$

for some given positive constants A *and* B. *Then*

$$|W_n|\le A^n|W_0|+B\frac{A^n-1}{A-1},\quad 0\le n\le N-1.$$

Proof see [9].

LEMMA 4.2- *Let a sequence of numbers* $\{W_n\}_{n=0}^N$, $\{V_n\}_{n=0}^N$ *satisfy*

$$|W_{n+1}|\le|W_n|+A.\max\{|W_n|,|V_n|\}+B,$$
$$|V_{n+1}|\le|V_n|+A.\max\{|W_n|,|V_n|\}+B,$$

for some given positive constants A *and* B *.; and denote*

$$|U_n|=|W_n|+|V_n|,\quad 0\le n\le N-1.$$

Then

$$|U_n| \le \bar{A}^n |U_0| + \bar{B}\frac{\bar{A}^n - 1}{\bar{A} - 1}, \quad 0 \le n \le N-1, \text{ where } \bar{A} = 1 + 2A \text{ and } \bar{B} = 2B.$$

Proof [9].

Let $F^*(t,u,v)$ and $G^*(t,u,v)$ be the functions F and G in (12), where u and v are constants and $u \le v$. In other words

$$F^*(t,u;v) = \min\{f(t,\tau) \mid \tau \in [u,v]\} + \frac{h}{2}\min\{\frac{\partial f}{\partial t}(t,\tau) + \frac{\partial f}{\partial \tau}(t,\tau) f(t,\tau) \mid \tau \in [u,v]\},$$

$$G^*(t,u;v) = \max\{f(t,\tau) \mid \tau \in [u,v]\} + \frac{h}{2}\max\{\frac{\partial f}{\partial t}(t,\tau) + \frac{\partial f}{\partial \tau}(t,\tau) f(t,\tau) \mid \tau \in [u,v]\},$$

,i.e. $F^*(t,u,v)$ and $G^*(t,u,v)$ are obtained by substituting $[x(t)]_r = [u,v]$ in (12). The domain where F^* and G^* are defined is

$$K = \{(t,u,v) \mid 0 \le t \le T, -\infty < v < \infty, -\infty < u \le v\}.$$

THEOREM 4.1- *Let $F^*(t,u,v)$ and $G^*(t,u,v)$ belong to $C^1(K)$ and let the partial derivatives of F^* and G^* be bounded over K. Then, for arbitrary fixed $r: 0 \le r \le 1$, the approximately solutions (14) converge to the exact solutions $Y_1(t;r)$ and $Y_2(t;r)$ uniformly in t.*

Proof : It is sufficient to show

$$\lim_{h\to 0} y_1(t_N;r) = Y_1(t_N;r),$$
$$\lim_{h\to 0} y_2(t_N;r) = Y_2(t_N;r),$$

where $t_N = T$. For $n = 0,1,\ldots,N-1$, by using Taylor theorem we get

$$\begin{aligned} Y_1(t_{n+1};r) &= Y_1(t_n;r) + hF^*[t_n, Y_1(t_n;r), Y_2(t_n;r)] + \frac{h^3}{6}Y_1^{(3)}(\xi_{n,1}), \\ Y_2(t_{n+1};r) &= Y_2(t_n;r) + hG^*[t_n, Y_1(t_n;r), Y_2(t_n;r)] + \frac{h^3}{6}Y_{12}^{(3)}(\xi_{n,2}), \end{aligned} \tag{15}$$

where $\xi_{n,1}, \xi_{n,2} \in (t_n, t_{n+1})$. Denote

$$W_n = Y_1(t_n;r) - y_1(t_n;r),$$
$$V_n = Y_2(t_n;r) - y_2(t_n;r).$$

Hence from (14) and (15)

$$W_{n+1} = W_n + h\{F^*[t_n, Y_1(t_n;r), Y_2(t_n;r)] - F^*[t_n, y_1(t_n;r), y_2(t_n;r)]\} + \frac{h^3}{6}Y_1^{(3)}(\xi_{n,1}),$$

$$V_{n+1} = V_n + h\{G^*[t_n, Y_1(t_n;r), Y_2(t_n;r)] - G^*[t_n, y_1(t_n;r), y_2(t_n;r)]\} + \frac{h^3}{6} Y_2^{(3)}(\xi_{n,1}).$$

Then

$$|W_{n+1}| \le |W_n| + 2Lh.\max\{|W_n|, |V_n|\} + \frac{h^3}{6} M_1,$$
$$|V_{n+1}| \le |V_n| + 2Lh.\max\{|W_n|, |V_n|\} + \frac{h^3}{6} M_2,$$

where

$$M_1 = \max|Y_1^{(3)}(t;r)|,\ M_2 = \max|Y_2^{(3)}(t;r)|,$$

for $t \in [0,T]$ and $L > 0$ is a bound for the partial derivatives of F^* and G^*. Thus by lemma 2

$$|W_n| \le (1+4Lh)^n |U_0| + \frac{h^3}{3} M_1 \frac{(1+4Lh)^n - 1}{4Lh},$$
$$|V_n| \le (1+4Lh)^n |U_0| + \frac{h^3}{3} M_2 \frac{(1+4Lh)^n - 1}{4Lh},$$

where $|U_0| = |W_0| + |V_0|$. In particular

$$|W_N| \le (1+4Lh)^N |U_0| + \frac{h^3}{3} M_1 \frac{(1+4Lh)^{\frac{T}{h}} - 1}{4Lh},$$
$$|V_N| \le (1+4Lh)^N |U_0| + \frac{h^3}{3} M_2 \frac{(1+4Lh)^{\frac{T}{h}} - 1}{4Lh}.$$

Since $W_0 = V_0 = 0$ we obtain

$$|W_N| \le M_1 \frac{e^{4LT} - 1}{24L} h^2,$$
$$|V_N| \le M_2 \frac{e^{4LT} - 1}{24L} h^2,$$

and if $h \to 0$ we get $W_N \to 0$ and $V_N \to 0$ which completes the proof.

5. EXAMPLES

EXAMPLE 5.1- *Consider the fuzzy initial value problem, [9],*

$$\begin{cases} y'(t) = y(t),\ t \in I = [0,T], \\ y(0) = (0.75 + 0.25r, 1.125 - 0.125r),\ 0 < r \le 1. \end{cases}$$

By using 2nd Taylor method we have

$$y_1(t_{n+1};r) = y_1(t_n;r)[1+h+\frac{h^2}{2}],$$

$$y_2(t_{n+1};r) = y_2(t_n;r)[1+h+\frac{h^2}{2}].$$

The exact solution is given by

$$Y_1(t;r) = y_1(0;r)e^t \quad , \quad Y_2(t;r) = y_2(0;r)e^t$$

which at $t=1$,

$$Y(1;r) = [(0.75+0.25r)e, (1.125-0.125r)e], 0<r\le 1.$$

The exact and approximate solutions are compared and plotted at $t=1$ in Figure 5.1.

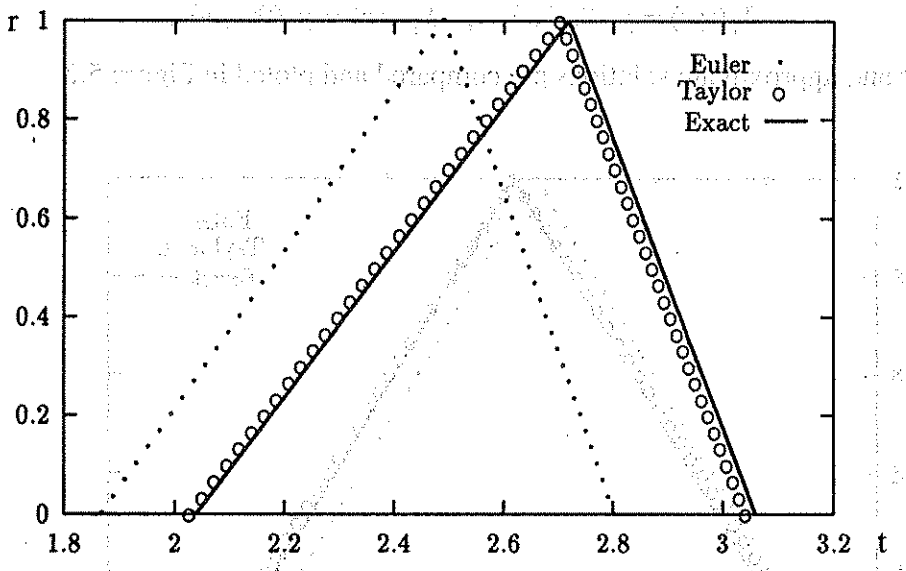

Figure 5.1, ($h=0.2$)

EXAMPLE 5.2.- *Consider the fuzzy initial value problem, [9],*

$$\begin{cases} y'(t)=ty(t), \ [a,b]=[-1,1], \\ y(-1)=(\sqrt{e}-0.5(1-r), \sqrt{e}+0.5(1-r)), 0<r\le 1. \end{cases}$$

We separate between two steps.

(a) $t<0$: The parametric form in this case is

$$y'_1(t;r) = t\,y_2(t;r) \quad , \quad y'_2(t;r) = t\,y_1(t;r),$$

$$y''_1(t;r) = (1+t^2)y_2(t;r) \quad , \quad y''_2(t;r) = (1+t^2)y_1(t;r),$$

with the initial conditions given. The unique exact solution is

$$Y_1(t;r)=\frac{A-B}{2}y_2(0;r)+\frac{A+B}{2}y_1(0;r),$$
$$Y_2(t;r)=\frac{A+B}{2}y_2(0;r)+\frac{A-B}{2}y_1(0;r),$$

where $A=e^{\frac{(t^2-a^2)}{2}}$, $B=\frac{1}{A}$.

$(b)\,t\geq 0$: The parametric equations are

$$y'_1(t;r)=ty_1(t;r)\quad,\quad y'_2(t;r)=ty_2(t;r),$$
$$y''_1(t;r)=(1+t^2)y_1(t;r)\quad,\quad y''_2(t;r)=(1+t^2)y_2(t;r),$$

with the initial conditions given. The unique exact solution at $t>0$ is

$$Y_1(t;r)=y_1(0;r)e^{\frac{t^2}{2}}\quad,\quad Y_2(t;r)=y_2(0;r)e^{\frac{t^2}{2}}\ .$$

The exact and approximate solutions are compared and ploted in Figure 5.2.

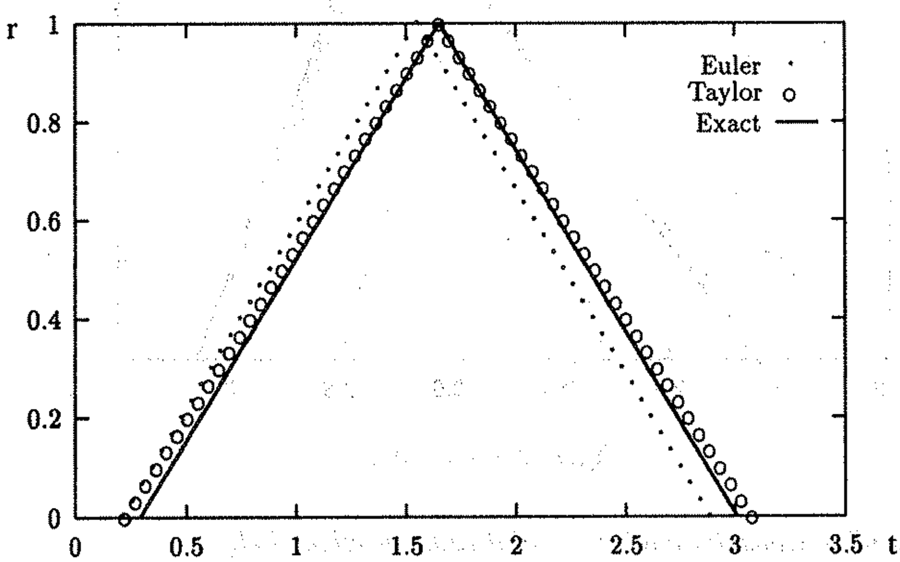

Figure 5.2, $(h=0.5)$

EXAMPLE 5.3- *Consider the fuzzy initial value problem*

$$y'(t)=k_1y^2(t)+k_2\,,\quad y(0)=0,$$

where $k_i>0$, *for* $i=1,2$ *are triangular fuzzy numbers, [10].*

The exact solution is given by

$$Y_1(t;r)=l_1(r)\tan(\omega_1(r)t)\,,$$
$$Y_2(t;r)=l_2(r)\tan(\omega_2(r)t)\,,$$

with

$$l_1(r)=\sqrt{k_{2,1}(r)/k_{1,1}(r)}\quad,\quad l_2(r)=\sqrt{k_{2,2}(r)/k_{1,2}(r)}\,,$$

$$\omega_1(r)=\sqrt{k_{1,1}(r)/k_{2,1}(r)}\quad,\quad \omega_2(r)=\sqrt{k_{1,2}(r)/k_{2,2}(r)}\,,$$

where

$$[k_1]_r=[k_{1,1}(r),k_{1,2}(r)]\quad\text{and}\quad [k_2]_r=[k_{2,1}(r),k_{2,2}(r)]$$

and

$k_{1,1}(r)=0.5+0.5r\,, k_{1,2}(r)=1.5-0.5r$ *and* $k_{2,1}(r)=0.75+0.25r\,, k_{2,2}(r)=1.25-0.25r.$

The r-level sets of $y'(t)$ are

$$Y'_1(t;r)=k_{2,1}(r)\sec^2(\omega_1(r)t)\,,$$

$$Y'_2(t;r)=k_{2,2}(r)\sec^2(\omega_2(r)t),$$

which defines a fuzzy number. We have

$f_1(t,y;r)=\min\{k_1u^2+k_2\mid u\in[y_1(t;r),y_2(t;r)],k_1\in[k_{1,1}(r),k_{1,2}(r)],k_2\in[k_{2,1}(r),k_{2,2}(r)]\},$

$f_2(t,y;r)=\max\{k_1u^2+k_2\mid u\in[y_1(t;r),y_2(t;r)],k_1\in[k_{1,1}(r),k_{1,2}(r)],k_2\in[k_{2,1}(r),k_{2,2}(r)]\},$

$f'_1(t,y;r)=\min\{2k_1^2u^3+2uk_1k_2\mid u\in[y_1(t;r),y_2(t;r)],k_1\in[k_{1,1}(r),k_{1,2}(r)],k_2\in[k_{2,1}(r),k_{2,2}(r)]\},$

$f'_2(t,y;r)=\max\{2k_1^2u^3+2uk_1k_2\mid u\in[y_1(t;r),y_2(t;r)],k_1\in[k_{1,1}(r),k_{1,2}(r)],k_2\in[k_{2,1}(r),k_{2,2}(r)]\}.$

There are two nonlinear programings and can be solving by GAMS software. Thus the suggested Taylor method in this paper can be used. The exact and approximate solutions are shown in Figure 5.3) at $t=1$.

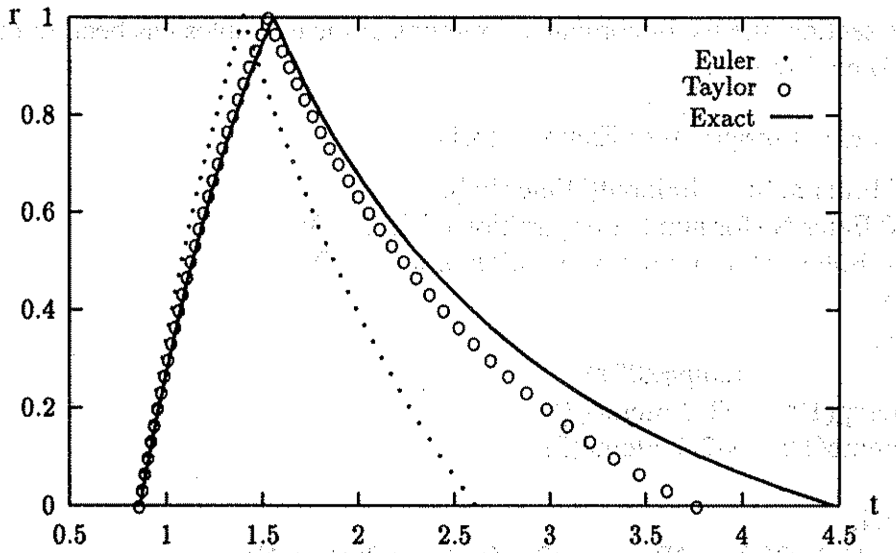

Figure 5.3, $(h=0.1)$

6.CONCLUSION

We note that the convergence order of the Euler method in [9] is $O(h)$. It is shown that in proposed method, the convergence order is $O(h^2)$ and the comparison of solutions of example (1) ,(2) in this paper and [9] shows that this solutions are better for these examples.

REFERENCES

1. S.L. Chang, L.A. Zadeh, On fuzzy mapping and control, *IEEE Trans, Systems Man Cybernet,* **2,** 30-34, 1972.
2. D.Dubois, H. Prade, Towards fuzzy differential calculus: Part 3, differentiation, *Fuzzy Sets and Systems* , **8**, 225-233, 1982.
3. M.L Puri, D.A. Ralescu, Differentials of fuzzy functions, *J Math. Anal. Appl.,* **91**, 321-325, 1983.
4. R. Goetschel, W. Voxman, Elementary fuzzy calculus, *Fuzzy Sets and Systems,* **18,** 31-43, 1986.
5. O. Kaleva, Fuzzy differential equations, *Fuzzy Sets and Systems* ,**24**, 301-317, 1987.
6. O. Kaleva, The cuachy problem for Fuzzy differential equations, *Fuzzy Sets and Systems* , **35,** 389-396, 1990.
7. S. Seikkala, On the fuzzy initial value problem, *Fuzzy Sets and Systems,* **24**, 319-330, 1987.
8. R. L. Burden, J. D. Faires, *Numerical Analysis*, 1997.
9. M. Ma, M. Friedman, A. Kandel, Numerical Solutions of fuzzy differential equations, *Fuzzy Sets and Systems* , **105**, 133-138, 1999.
10. James J. Buckley, Thomas Feuring, Fuzzy differential equations, *Fuzzy Sets and Systems,* **110**, 43-54, 2000.

APPENDIX

In this section, the list of computer programs of the examples has been written in MATLAB environment.

1- The computer programs of Example (5.1).

```
a=input(' Enter a: ');    b=input(' Enter b: ');
N=input(' Enter N (for number of partition of time) : ');
nr=input(' Enter r (for number of partition of level) : ');
h=(b-a)/N;
for ir=0:nr;
 r=ir/nr;            temp=x0f(r);
 y1(1)=temp(1);      y2(1)=temp(2);
 yt1(1)=temp(1);     yt2(1)=temp(2);
 t=a;
 for n=1:N;
  a1=f(t,y1(n),y2(n),r,n,N);        a2=g(t,y1(n),y2(n),r,n,N);
  y1(n+1)=y1(n)+h*a1;               y2(n+1)=y2(n)+h*a2;
  a1=f(t,yt1(n),yt2(n),r,n,N);      a2=g(t,yt1(n),yt2(n),r,n,N);
  yt1(n+1)=yt1(n)+h*(a1+(h/2)*ff(t,yt1(n),yt2(n),r,n,N));
```

```
  yt2(n+1)=yt2(n)+h*(a2+(h/2)*gg(t,yt1(n),yt2(n),t,n,N));
  t=t+h;
 end
 z(ir+1)=y1(N+1);     z(2*nr+2-ir)=y2(N+1);
 zt(ir+1)=yt1(N+1);   zt(2*nr+2-ir)=yt2(N+1);
 alpha(ir+1)=r;       alpha(2*nr+2-ir)=r;
 temp=ytf(r);
 ze(ir+1)=temp(1);    ze(2*nr+2-ir)=temp(2);
end
plot(z,alpha,'y',zt,alpha,'r',ze,alpha,'b');

function x0f=x0f(r)
x0f(1,1)=.75+.25*r;
x0f(1,2)=1.125-.125*r;

function ytf=ytf(r)
ytf(1,1)=(.75+.25*r)*exp(1);
ytf(1,2)=(1.125-.125*r)*exp(1);

function y=F(t,y1,y2,r,n,N)
y=y1;

function y=G(t,y1,y2,r,n,N)
y=y2;

function y=ff(t,y1,y2,r,n,N)
y1=fmin('fd1',y1,y2,[0 1.e-12]);
y=fd1(y1);

function y=gg(t,y1,y2,r,n,N)
y1=fmin('fd2',y1,y2,[0 1.e-12]);
y=fd1(y1);

function d=fd1(x)
d=x;

function d=fd2(x)
d=-fd1(x);
```

2- The computer programs of Example (5.2).

```
function x0f=x0f(r)
x0f(1,1)=sqrt(exp(1))-0.5*(1-r);   x0f(1,2)=sqrt(exp(1))+0.5*(1-r);

function ytf=ytf(r)
temp=x0f(r);
ytf(1,1)=temp(2)*(1-exp(1))/2 + temp(1)*(1+exp(1))/2;
ytf(1,2)=temp(2)*(1+exp(1))/2 - temp(1)*(exp(1)-1)/2;

function y=f(t,y1,y2,r,n,N);
if n<=(N/2)
  y=t*y2;
else
  y=t*y1;
end

function y=g(t,y1,y2,r,n,N)
if n<=(N/2)
  y=t*y1;
else
  y=t*y2;
end

function y=ff(t,y1,y2,r,n,N)
y1=fmin('fd1',y1,y2,[0 1.e-12]);
y=(1+t*t)*fd1(y1);

function y=gg(t,y1,y2,r,n,N)
y1=fmin('fd2',y1,y2,[0 1.e-12]);
y=(1+t*t)*fd1(y1);

function d=fd1(x)

function d=fd2(x)
```

```
d=x;                                  d=-fd1(x);
```

3- The computer programs of Example (5.3).

```
a=input(' Enter a: ');        b=input(' Enter b: ');
N=input(' Enter N (for number of partition of time) : ');
nr=input(' Enter r (for number of partition of level) : ');
h=(b-a)/N;
for ir=0:nr;
  r=ir/nr;                 temp=x0f(r);
  y1(1)=temp(1);     y2(1)=temp(2);
  yt1(1)=temp(1);    yt2(1)=temp(2);
  temp1=k1(r);        temp2=k2(r);
  t=a;
  for n=1:N;
    y1(n+1)=y1(n)+h*(temp1(1)*yp1(y1(n))+temp2(1));
    y2(n+1)=y2(n)+h*(temp1(2)*yp1(y2(n))+temp2(2));
    a1=temp1(1)*yp1(yt1(n))+temp2(1);
    a2=temp1(2)*yp1(yt2(n))+temp2(2);
    yt1(n+1)=yt1(n)+h*(a1+(h/2)*ff(yt1(n),temp1(1),temp2(1)));
    yt2(n+1)=yt2(n)+h*(a2+(h/2)*ff(yt2(n),temp1(2),temp2(2)));
    t=t+h;
  end
  z(ir+1)=y1(N+1);      z(2*nr+2-ir)=y2(N+1);
  zt(ir+1)=yt1(N+1);    zt(2*nr+2-ir)=yt2(N+1);
  alpha(ir+1)=r;          alpha(2*nr+2-ir)=r;
  temp=ytf(r,b);
  ze(ir+1)=temp(1);     ze(2*nr+2-ir)=temp(2);
end
plot(z,alpha,'y',zt,alpha,'r',ze,alpha,'b');

function y=x0f(r)
y(1,1)=0;      y(1,2)=0;

function y=k1(r)                        function y=k2(r)
y(1,1)=.5+.5*r;                         y(1,1)=.75+0.25*r;
y(1,2)=1.5-0.5*r;                       y(1,2)=1.25-.25*r;

function y=yp1(x)                       function z=ff(y,c1,c2);
y=x*x;                                  z=2*c1*c1*y*y*y+2*c1*c2*y;

function y=ytf(r,b)
ko=k1(r);                    kt=k2(r);
lam1=sqrt(kt(1)/ko(1));      om1=sqrt(ko(1)*kt(1));
lam2=sqrt(kt(2)/ko(2));      om2=sqrt(ko(2)*kt(2));
y(1,1)=lam1*tan(om1*b);      y(1,2)=lam2*tan(om2*b);
```