

## On Fuzzy differential equation

Eman Ali Hussain<sup>1</sup>,

Yahya Mourad Abdul – Abbass<sup>2</sup>

<sup>1</sup>Department of Mathematics, Al- Mustansiriya, University, College of Science, Iraq

<sup>2</sup>Ministry of Education, Educational Directorate Babylon, Iraq

<sup>1</sup>[dremansultan@yahoo.com](mailto:dremansultan@yahoo.com)

<sup>2</sup>[yihiamor@yahoo.com](mailto:yihiamor@yahoo.com)

Received : 9\1\2019

Revised : 14 \ 1 \ 2019

Accepted : 24\1\2019

Available online : 22 /4/2019

### Abstract

In this paper, we introduce a hybrid method to use fuzzy differential equation, and Genetic Turing Machine developed for solving  $n^{\text{th}}$  order fuzzy differential equation under Seikkala differentiability concept [14]. The Errors between the exact solutions and the approximate solutions were computed by fitness function and the Genetic Turing Machine results are obtained. After comparing the approximate solution obtained by the GTM method with approximate to the exact solution, the approximate results by Genetic Turing Machine demonstrate the efficiency of hybrid methods for solving fuzzy differential equations (FDE).

**Keywords:** Fuzzy differential equations, Genetic Turing machine, Turing machine,

## 1- INTRODUCTION

The Fuzzy differential equations play an important role in recent years in an multiple models in biology[7], engineering[12], physics and other sciences[11]. The First order fuzzy differential equations appear in many applications. However the form of such an equation is very simple. Chang and Zadeh [16] introduced the concept of fuzzy derivative. Kandel and Byatt [ 1] applied the fuzzy differential equation (FDE) to solved the fuzzy dynamical problems. Kaleva [15], Seikkala [17] introduced the FDE with the initial value problems (Cauhy problem). He and Yi [14], created the numerical methods for solving fuzzy differential equations are introduced. Buckley and Feuring [8] created two analytical methods for solving Nth-order linear differential equations with fuzzy initial value conditions.

In 2009, Nieto et al. [9] showed that any numerical method using to solve ordinary differential equations can be used to solve numerically fuzzy differential equations under generalized differentiability, Allahviranloo et al. [19] solved fuzzy differential equations by used the generalised differentiability and applied differential transformation method . In 2011, Khastan and et al. [2] presented the general form to solve first order linear fuzzy differential equations. Recently, Ghazanfari and et al [4] solving first order fuzzy differential equation, by using the Runge-Kutta-like formulae of order 4

## 2. Fuzzy differential equation by Genetic Turing Machine

### 2.1:Definition: ( Multi-Tape Turing Machines) [3]

Multi-Tape Turing Machines is defined as 7–tuple  $(S, \Sigma, \Gamma, \delta, S_0, B, F)$  where,

1.  $S$  is a set of finite states.
2.  $\Sigma$  is the set of input symbols (alphabet).
3.  $\Gamma$  is the tape alphabet.
4.  $B \in \Gamma$  The blank symbol.
5.  $\delta$  is The transition function where  $\delta: S \times \Gamma^t \rightarrow S \times \Gamma^t \times \{L, R, N\}^t$  where L,R and N , are left, right and null respectively

which indicates the direction to move the read/write head.

6.  $S_0 \in S$  is the start state.
7.  $F$  is the set of final (accepting) states.

### 2.2:Definition: (Multidimensional Turing Machines) [13]

A two–dimensional machine has a transition defined as 7–tuple  $(S, \Sigma, \Gamma, \delta, S_0, B, F)$  where,

- $S$  is a set of finite states.
- $\Sigma$  is the set of input symbols (alphabet).
- $\Gamma$  is the tape alphabet.
- $B \in \Gamma$  the blank symbol.
- $\delta$  is The transition function for an t–tape

Turing machine can be defined

$\delta: S \times \Gamma^t \rightarrow S \times \Gamma^t \times \{L,R,U,D\}^t$  where L ,R ,U, and D are left, right , up, and down respectively which indicates the direction to move the read/write head.

- $S_0 \in S$  is the start state.
- $F$  is the set of final (accepting) states

### 2.3:Definition: Fuzzy Number [18]

A fuzzy number is a fuzzy set  $\tilde{A}$  on  $R$  (real line) with three properties to become as a fuzzy number

- 1-  $\tilde{A}$  must be a normal fuzzy set;
- 2-  $\tilde{A}_\alpha$  must be closed interval for every  $\alpha \in [0, 1]$
- 3- The support of  $\tilde{A}$  must be bounded.

### 2.4: Definition (Fuzzy point) [6]

Let  $\tilde{A}$  be a fuzzy number. If  $\text{supp}(A) = \{x_0\}$  then  $\tilde{A}$  is called a fuzzy point and used the notation  $\tilde{A} = \overline{x_0}$

### 2.5: Definition(Triangular Fuzzy Number) [10]

It is a fuzzy number represented with three points as follows:  $\tilde{A} = (a_1, a_2, a_3)$

This representation is membership functions and holds the following conditions

- (i)  $a_1$  to  $a_2$  is increasing function
- (ii)  $a_2$  to  $a_3$  is decreasing function
- (iii)  $a_1 \leq a_2 \leq a_3$ .

Then the form

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & \text{for } x < a_1 \\ \frac{x - a_1}{a_2 - a_1} & \text{for } a_1 \leq x \leq a_2 \\ \frac{a_3 - x}{a_3 - a_2} & \text{for } a_2 \leq x \leq a_3 \\ 0 & \text{for } x > a_3 \end{cases}$$

**2.9: Definition:  $\alpha$ - cut of a triangular fuzzy number [18]**

By the  $\alpha$ -cut operation can obtain a crisp interval, the interval  $A_\alpha$  is obtained as follows  $\forall \alpha \in [0,1]$ . Thus

$$A_\alpha = [a_1^\alpha, a_3^\alpha] [(a_2 - a_1)\alpha + a_1 - (a_3 - a_2)\alpha + a_3]$$

**2.10: Definition: Positive triangular fuzzy number [19]**

A positive triangular fuzzy number  $\tilde{A}$  is denoted as  $\tilde{A} = (a_1, a_2, a_3)$  where all  $a_i$ 's  $> 0$  for all  $i=1, 2, 3$ .

**2.11: Definition: Negative triangular fuzzy number [19]**

A negative triangular fuzzy number  $\tilde{A}$  is denoted as  $\tilde{A} = (a_1, a_2, a_3)$  where all  $a_i$ 's  $< 0$  for all  $i=1, 2, 3$ .

**2.12: Definition: Equal Triangular fuzzy number [10]**

Let  $\tilde{A} = (a_1, a_2, a_3)$  and  $\tilde{B} = (b_1, b_2, b_3)$  be two triangular fuzzy numbers. Then  $\tilde{A}$  is equal to  $\tilde{B}$  if  $a_1 = b_1, a_2 = b_2$  and  $a_3 = b_3$ .

**2.20: Definition: Operation of Triangular Fuzzy Number [6]**

Let  $\tilde{A} = (a_1, a_2, a_3)$  and  $\tilde{B} = (b_1, b_2, b_3)$  be two triangular fuzzy numbers then,

(i) **Addition:**

$$\tilde{A} + \tilde{B} = (a_1+b_1, a_2+b_2, a_3+b_3).$$

(ii) **Subtraction:**

$$\tilde{A} - \tilde{B} = (a_1-b_1, a_2-b_2, a_3-b_3).$$

(iii) **Multiplication:**

$$\tilde{A} \cdot \tilde{B} = (\min (a_1b_1, a_1b_3, a_3b_1, a_3b_3), a_2b_2, \max (a_1b_1, a_1b_3, a_3b_1, a_3b_3)).$$

(iv) **Division:**

$$\frac{\tilde{A}}{\tilde{B}} = (\min (\frac{a_1}{b_1}, \frac{a_1}{b_3}, \frac{a_3}{b_1}, \frac{a_3}{b_3}), \frac{a_2}{b_2}, \max (\frac{a_1}{b_1}, \frac{a_1}{b_3}, \frac{a_3}{b_1}, \frac{a_3}{b_3})).$$

**2.21 Definition : Fuzzy Turing machine [5]**

A fuzzy Turing machine, FTM, divided into triple-tuple

- 1- NTM (Non-deterministic Turing Machine) of 7-tuple  $(S, \Sigma, \Gamma, \delta, S_0, B, F)$ .
- 2-  $*$  is a t-norm.
- 3-  $\mu$  is a map ( function ) which assign a membership degree to each tuple in the "next move" relation  $\delta$ , where  $\mu: \delta \rightarrow [0,1]$

If the solution in the interval  $[x_0, b]$ , and the optimal solution is the fitness function approach to zero , where the fitness function  $E_r$  is to provide a basis for among available approximation solutions and to obtain the optimal the fitness function for FDE (fuzzy differential equation) is defined in general as

$$E_r = [f(x, y, y', \dots, y^{(n-1)}) - \phi(y(x))]^2 \tag{2-1}$$

On FDE

$$y^{(n)}(x) = f(x, y, y', \dots, y^{(n-1)}) \quad (2-2)$$

$$y(x_0) = k_1, y'(x_0) = k_2, y''(x_0) = k_3, \dots,$$

where the function  $f : [x_0, b] \times F^n \rightarrow F$  is a fuzzy process with fuzzy initial values  $x^{(i-1)}(x_0) = k_i, i = 1, \dots, n$ .

where  $F^n = F \times \dots \times F$  (n-time) be the space of all compact and convex fuzzy sets on  $R^n$ .

### 3 .FDE of first order with TM [1]

Consider the first order fuzzy differential equation is

$$y' = f(x, y(x)), y(x_0) = y_0, \quad (2-3)$$

where  $f : [x_0, b] \times F \rightarrow F$  is a continuous fuzzy mapping and  $y_0$  is a fuzzy interval, since the solution of the fuzzy differential equation (2.3) by [ 1 ] give us an suitable procedure to solve the FDE of (2.3). For this, let

$$[y(x)]^\alpha = [u_\alpha(x), v_\alpha(x)] \quad (2-4)$$

and

$$[f(x, y(x))]^\alpha = [h_\alpha(x, u_\alpha(x), v_\alpha(x)), g_\alpha(x, u_\alpha(x), v_\alpha(x))] \quad (2.5)$$

Where fuzzy sets  $u$ , and  $v$  in  $E$  ( $E$  be a nonempty set) are characterized by its membership functions

$$u : E \rightarrow [0,1], \text{ and } v : E \rightarrow [0,1]$$

Then  $u(x)$ , and  $v(x)$  are interpreted as the degree of membership of an element  $x$  in the fuzzy set  $u$ , and  $v$  for each  $x \in E$ .

### 4 .Genetic FDE by TM algorithm

**Algorithm(4.1):** Genetic Fuzzy differential equation by Turing Machine algorithm

**Input :** FDE eq(2.2)  
 Input of the inputs of the algorithm (4. 2)

**Output:** optimal solution of (2.5)

1. Calculate (2.4) and (2.5).
2. Calculate the initial conditions  $k_i$  is a symmetric triangular fuzzy number with support interval.
3. Using algorithm (4.2).
4. Evaluate fitness value of the chromosome and calculate the eq (2.1).
5. If  $E_r < \epsilon$   
     stop
6. Generate new population using Genetic operations
7. Goto step 2

For each generation, a set of expressions are generated by the chromosomes. If an expression minimizes the fitness function  $E_r$  to zero or very close to zero and satisfies the initial condition, the process may be stopped; otherwise, the GP approach must be continued.

**Algorithm(4.2):** *population Technique for TM*

**Input :** *n size of population,*  
*t the number terms*  
*a minimum number value*  
*b maximum number value*  
 *$\varepsilon$  small number  $0 < \varepsilon \ll 1$*   
*sequence of production.*

**Output:** *sequence of production.*

```

0.  $h=0, x_1 = \frac{a+b}{2}$ 
1. while  $h < n$ 
2.  $z=0, w=0, s=0, f=0,$ 
3. for  $j=1$  to  $t$ 
4.    $P_{ij}=03$ 
5.    $r = \text{random}(0,1)$  // Generate random
      number  $r, 0 \leq r \leq 1$ 
6.    $k = 13[r + 0.75 + 0.25z] - 11$ 
7.   call A1
8.   if  $z=0$  and  $r < 0.25$ 
9.      $z=1$ 
10.     $P_{ij}=P_{ij} || D || F$ 
11.     $\text{term}(i,j)=m$ 
12.    continue
13.  end
14.   $P_{ij}=P_{ij} || D || 2$ 
15.   $e=m$ 
16.   $k = 2[(3 - w)r] - 2$ 
17.   $P_{ij}=P_{ij} || k$ 
18.  if  $w=0$ 
19.     $P_{ij}=P_{ij} || F$ 
20.     $w=1$ 
21.     $\text{term}(i,j)=e*x$ 
22.    continue
23.  end
24.  if  $k=0$ 
25.     $s=s+1$ 
26.    call A1
27.     $P_{ij}=P_{ij} || 443 || D || F$ 
28.     $c = [m]$ 
29.     $u(s)=c$ 
30.    while  $u(L)=u(s) L=1,2,\dots,s-1$ 
31.      call A1
32.    end
33.     $\text{term}(i,j)=e*x^c$ 
34.    continue
35.  end

```

```

36if  $k=2$ 
37.    $P_{ij}=P_{ij} || 2$ 
38.    $f=f+1$ 
39.   if  $f=1$ 
40.      $r=\text{random}(0,1)$ 
41.      $k = [6r]$ 
42.      $v(1)=k$ 
43.   else
44.     A2:  $r=\text{random}(0,1)$ 
45.         $L=1$ 
46.        while  $v(L) \neq [6r]$ 
47.           $L=L+1$ 
48.        end
49.        if  $L < f$ 
50.          Goto A2
51.        end
52.         $k = [6r]$ 
53.         $v(f)=k$ 
54.      end
55.       $P_{ij}=P_{ij} || k$ 
56.      Call A1
57.       $P_{ij}=P_{ij} || 3 || D || 2 || 4 || 1 || F$ 
58.       $\text{Term}(i,j)= e * f_k(mx)$ 
59.    end
60.     $T(x,i)=T(x,i)+ \text{Term}(i,j)$ 
61.    if  $ft(x_1) < \varepsilon$  // see equation (4.4)
62.       $h=h+1$ 
63.       $F\text{terms}(x,i)=T(x,i)$ 
64.    end
65.  end
66. end
67. A1: Call algorithm (4.2),
68.      return random number  $a < m < b,$ 
69.      and sequence
       $D = D_{I_1} D_{I_2} \dots D_{I_{\alpha+\beta+1}}$ 
70. end A1
71.  $f_0(x)=\sin(x)$ 
72.  $f_1(x)=\cos(x)$ 
73.  $f_2(x)=\exp(x)$ 
74.  $f_3(x)=\log(x)$ 
75.  $f_4(x)=\text{sqrt}(x)$ 
76.  $f_5(x)=\ln(x)$ 

```

**Example( 4-1):**

Let us consider the fuzzy differential equation

$$y'(x) - 100 y(x), y(0) = y_0, \quad (2.6)$$

where the initial condition  $y_0$  is a symmetric triangular fuzzy number with support  $[-4,4]$ .

That is,

$$[y_0]^\alpha = [-4(1-\alpha), 4(1-\alpha)] = (1-\alpha)[-4,4]$$

(2.7)

Then from (2.5) the fuzzy differential system will be as given below [2]:

$$\begin{aligned} u'_\alpha(x) &= -100v_\alpha(x), & u_\alpha(0) &= -4(1-\alpha) \\ v'_\alpha(x) &= -100u_\alpha(x), & v_\alpha(0) &= 4(1-\alpha) \end{aligned}$$

By using GTM (Genetic Turing Machine), then the solution of above system is

$$u_\alpha(x) = -4(1-\alpha)e^{100x}$$

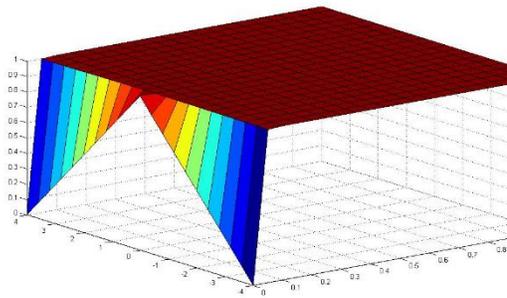
and

$$v_\alpha(x) = 4(1-\alpha)e^{100x}$$

Therefore, the fuzzy function  $y(x)$  solving (5.6) has level sets

$$\begin{aligned} [y(x)]^\alpha &= [-4(1-\alpha)e^{100x}, 4(1-\alpha)e^{100x}] \\ &= (1-\alpha)e^{100x}[-4,4] \quad \forall x \geq 0 \end{aligned}$$

Then approximate  $\alpha$  by using (GTM) algorithm and test it in symmetric triangular fuzzy number.



**Figure (4-1): symmetric triangular fuzzy number for example (4-1)**

**4.2:Example:**

$$y'(x) = y(x) \quad x \in [0, 1] \quad (2.8)$$

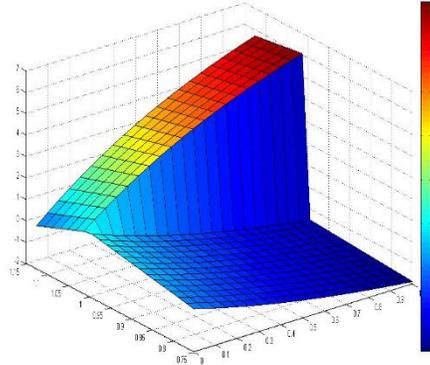
$$y(0) = (0.75 + 0.25\alpha, 1.125 - 0.125\alpha).$$

where  $0 \leq \alpha \leq 1$

The exact solution of (5.8) at  $x = 1$ ,

is given by,

$$y(1, \alpha) = [(0.75 + 0.25\alpha)e^x, (1.125 - 0.125\alpha)e^x],$$



**Figure (4-2): symmetric triangular fuzzy number for example (4.2)**

**4.3: Example**

Let us consider the fuzzy differential second equation

$$\begin{aligned} y''(x) - 100 y(x) &= 0, & y(0) &= y_0, \\ y'(0) &= 2 \end{aligned} \quad (2-9)$$

where  $y_0 \in [-2+3\alpha, 2-\alpha]$ ,  $0 \leq \alpha \leq 1$

The exact solution of (2-9) of ordinary DE is

$$y(x) = c_1 y_1(x) + c_2 y_2(x) = c_1 e^{10x} + c_2 e^{-10x}$$

$$\text{Let } A(x) = \begin{pmatrix} y_1(x) & y_2(x) \\ y'_1(x) & y'_2(x) \end{pmatrix}, \text{ then}$$

$$A = A(0) = \begin{pmatrix} 1 & 1 \\ 10 & -10 \end{pmatrix}$$

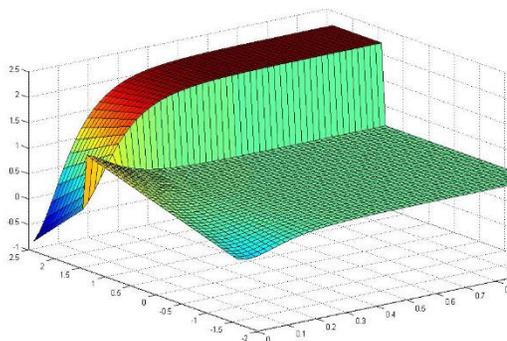
$$A^{-1} = \frac{1}{20} \begin{pmatrix} 10 & 1 \\ 10 & -1 \end{pmatrix}, b = \begin{pmatrix} [-2 + 3\alpha, 2 - \alpha] \\ 2 \end{pmatrix}, c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

$$c = A^{-1}b$$

then

$$\begin{aligned} y(x; \alpha) &= 0.5e^{10x} - 0.1e^{-10x} + 0.5[-2 + 3\alpha, 2 - \alpha](e^{10x} + e^{-10x}) \\ &= 0.6 \sinh(10x) - 0.4 \cosh(10x) + [-2 + 3\alpha, 2 - \alpha] \cosh(10x) \end{aligned}$$

$$\alpha = z(x, y) = \begin{cases} 0.8 + \frac{1}{3}y \operatorname{sech}(10x) - 0.2 \tanh(10x) & y \in [-1.6, 1.4] \\ 1.6 + 0.6 \tanh(10x) - y \operatorname{sech}(10x) & y \in [1.4, 2.4] \end{cases}$$



**Figure (4-3): symmetric triangular fuzzy number for example (4.4)**

## 5. RESULTS AND DISCUSSION

The hybrid method which introduced in this work used for solving  $n^{\text{th}}$  order Fuzzy differential equation, by using Techniques Genetic Algorithms Turing Machine (GTM). The Errors between the exact solutions and the approximate solutions were computed by fitness function and the Genetic Turing Machine results are obtained. After comparing the approximate solution obtained by the (GAT) method with approximate to the exact solution, the approximate results by Genetic Turing Machine demonstrate the efficiency of hybrid methods for solving Fuzzy differential equations (FDE)

## REFERENCES

1. A. Kandel &, W.J. Byatt," Fuzzy differential equations" in Proceedings of the International Conference on Cybernetics and Society, Tokyo, 1213-1216,1972.
2. A. Khastan, & etal ,"Variation of constant formula for first order fuzzy differential

equations" *Fuzzy Sets and Systems* 177(1), 20-33,2011.

3. AMASHINI NAIDOO , “Evolving Automata Using Genetic Programming”, Master of Science thesis in the Faculty of Science and Agriculture, University of KwaZulu-Natal, Durban,2008.
4. B. Ghazanfari,&A. Shakerami, "Numerical solutions of fuzzy differential equations by extended Runge-Kutta-like formulae of order 4" *Fuzzy Sets and Systems* 189(1) , 74-91,2012.
5. barnabas bede, "mathematics of fuzzy sets and fuzzy logic", springer-verlag berlin heidelberg , Series ISSN 1434-9922, edition 1,2013.
6. Dubois, D. & Prade,H. "Towards fuzzy differential calculus.III. Differentiation" *Fuzzy Sets and Systems*, vol. 8, no. 3, pp.225–233, 1982
7. J. Casanovas & F. Rossell,"Averaging fuzzy biopolymers" *Fuzzy Sets and Systems* 152 ,139-158,2005.
8. J. J. Buckley& T. Feuring,"Fuzzy initial value problem for Nth-order fuzzy linear differ-ential equations" *Fuzzy Sets and Systems* 121 , 247-255,2001.
9. J.J. Nieto & et al,"Numerical solution of fuzzy differential equations under generalized differentiability, *Nonlinear Analysis" Hybrid Systems* 3(4), 700-707, 2009.
10. Klir, GJ & Yuan, B., "Fuzzy Sets and Fuzzy Logic, Theory and Applications" Prentice-Hall, Upper Saddle River, NJ, 1995 .

11. M. Guo & et al " Impulsive functional differential inclusions and fuzzy population models" Fuzzy Sets and Systems 138, 601-615,2003.
12. M. Oberguggenberger& S. Pittschmann,"Differential equations with fuzzy parameters" Math. Mod. Syst. 5 ,181-202,1999.
13. Michael C. Loui ,"Simulations among multidimensional Turing machines", Theoretical Computer Science Journal, Volume 21, Issue 2, November 1982, Pages 145-161.
14. O. He &W. Yi "On fuzzy differential equations" Fuzzy Sets Systems, 24, 321-325, 1989.
15. O. Kaleva,"The Cauchy Problem for Fuzzy Differential Equations" Fuzzy Sets and Systems, 35 ,389-396,1990.
16. S. L. Chang & A. Zadeh"On Fuzzy Mapping and Control" IEEE Trans. Systems Man Cybernet., 2 (1972) 30-34,1972.
17. S. Seikkala, "On the Fuzzy Initial Value Problem" Fuzzy Sets Systems, 24,319-330,1987.
18. S. Sindu Devi, and K. Ganesan, "Numerical Solution of First Order Fuzzy Differential Equations by Simpson's Rule ",British Journal of Mathematics & Computer Science 18(1): 1-13, 2016, Article no.BJMCS.26060
19. T. Allahviranloo & et al " Solving fuzzy differential equations by differential transformation method"Information Sciences 179(7),956-966,2009.

## على المعادلات التفاضلية الضبابية

ايمان علي حسين<sup>١</sup> يحيى مراد عبد العباس<sup>٢</sup>

قسم الرياضيات/ كلية العلوم/ الجامعة المستنصرية / العراق<sup>١</sup>

وزارة التربية / المديرية العامة لتربية بابل / العراق<sup>٢</sup>

[1dremansultan@yahoo.com](mailto:1dremansultan@yahoo.com)

[2yihiamor@yahoo.com](mailto:2yihiamor@yahoo.com)

### المستخلص:

في هذه الورقة ، نقدم طريقة هجينة لاستخدام المعادلات التفاضلية الضبابية و آلة تورينج الوراثية التي تم تطويرها لحل معادلة تفاضلية الضبابية حسب مفهوم سيكالا للتفاضل [١٤]. تم حساب الأخطاء بين الحلول الدقيقة والحلول التقريبية من خلال دالة ftj مع TGM وبعد مقارنة الحل التقريبي الذي تم الحصول عليه من خلال MTG الدقيق ، فإن النتائج التقريبية بواسطة آلة MT(التورنج) تُظهر كفاءة الطرق الهجينة لحل المعادلات التفاضلية الضبابية.