Available online at http://scik.org
J. Math. Comput. Sci. 6 (2016), No. 2, 235-246
ISSN: 1927-5307

FUZZY TRANSFORM ALGORITHM FOR PARTIAL FRACTIONAL DIFFERENTIAL EQUATIONS

AWNI M. ABU-SAMAN* AND HUDA S. EL-ZERII
Al-azhar University-Gaza(AUG), Gaza, PALESTINE

Abstract. This paper is devoted for introducing some background theory on the concept of the Fuzzy Transform as new tool for solving fractional order partial differential equations. We then investigate and apply the fuzzy transform on the rate dependent fractional diffusion-wave equation. In the process of this we give a new numerical algorithm, the algorithm will be simulated as a user-subroutine for the mathematical code MATLAB. The time-dependent fractional diffusion equation will be considered to show the efficiency of the algorithm, results will be obtained and compared for different partitions and fractional orders. The approximated results will be compared with the true solution of the differential equation.

keywords: fuzzy set theory; fuzzy transform; fuzzy partition; numerical algorithm and partial fractional differential equations.

2010 AMS Subject Classification: 65N22, 26A33, 41A55, 65B05, 65L05, 65L06, 65D30.

1. Introduction

The concept of differentiation and integration to non-integer order is by no means new, interest in this subject was evident almost as soon as the ideas of the classical calculus were known. In 1859 Leibniz mentioned it in a letter to L'Hospital [19].

Fractional order differential equations are considered to be as a generalization of classical integer order partial differential equations, it is used increasingly to model problems in elasticity, plasticity, fluid mechanics, economics and even finance. Consequently, considerable attention has been given to the solution of fractional order differential equations, since most of the fractional order differential equations do not have analytical solutions, Numerical algorithms,

*Corresponding author

Received October 5, 2015

therefore, are used extensively. Recently numerical solution of differential equations of fractional orders gained increasing interest [1, 2, 7, 8, 9, 24, 25].

Fractional derivatives had been ignored from the scientists for years because there are many nonequivalent definitions of it, but in recent years scientists started to use the fractional derivatives in their researches and it has given more accurate results, so scientists began to study the equations more seriously and finding new ways to solve them [5, 7, 17,19].

Partial fractional differential equations have an important role in mathematical modeling to describe complex processes in different branches of science "physics, biology, chemistry and economics ". In our study we will consider the time-dependent fractional diffusion equations as an example of the partial fractional differential equations.

$$\frac{\partial^\alpha u}{\partial t^\alpha}(x,t) = c\frac{\partial^2 u}{\partial x^2}(x,t) + q(x,t) \qquad (1)$$

where $\frac{\partial^\alpha u}{\partial t^\alpha}(x,t)$ is the derivative of u with respect to time t of order α in the sense of Caputo and $0 < \alpha \leq 2$. We can see that when $\alpha = 1$ the equation is the heat "diffusion" equation, when $0 < \alpha < 1$ the equation is called fractional sub-diffusion equation, when $1 < \alpha < 2$ the equation is called fractional super-diffusion equation, when $\alpha = 2$ the equation will be the Poisson equation [8]. The Fuzzy Transform, which is introduced for the first time by Perfilieva [13 – 18], will be considered to introduce a new technique for approximating the solution of the above rate dependent fractional order differential equation. The fuzzy transform depends basically on Fuzzy Set Theory which was published in 1965 by Lotfi Zadeh [26].

2. Fuzzy Transform

To Avoid repetition the main definition of fuzzy sets, triangular shaped basic functions and Sinusoidal shaped basic functions can be found with details in [12-17, 21, 22, 23, 26]. In this section we will only state main definitions of F-transform that will be used in the next sections of numerical implementations.

Definition 1 Let $A_1, \dots, A_n$ be basic functions which form a fuzzy partition in $[a,b]$, and f be any function in $C[a,b]$. We say that the n-tuple of real numbers $[F_1, \dots, F_n]$ given by

$$F_i = \frac{\int_a^b f(x)A_i(x)dx}{\int_a^b A_i(x)dx}, \qquad i = 1, \dots, n \qquad (2)$$

is the direct (integral) F-transform of f with respect to $A_1, \dots, A_n$.

And to transform the F-transform back we use the next inverse F-transform formula

$$f_{F,n}(x) = \sum_{k=0}^{n} F_k A_k(x) \quad (3)$$

Since we are concerned about partial differential equations we will use the next two definitions of F-transform and inverse F-transform with two variables [20].

Definition 2 Let $u(x,t)$ be an arbitrary continuous function in $\boldsymbol{\mathcal{D}} = [a,b] \times [c,d]$and let basis functions $A_1(x), \dots, A_n(x)$ in $[a,b]$ and $B_1(t), \dots, B_m(t)$ on $[c,d]$ form a uniform fuzzy partitions we say that the real matrix $U_{n\times m}$ given by

$$U_{ij} = \frac{\int_c^d \int_a^b u(x,t) A_i(x) B_j(t) dx dt}{\int_c^d \int_a^b A_i(x) B_j(t) dx dt}, \quad (4)$$

$$i = 1, \dots, n, j = 1, \dots, m$$

is the F-transform of $u(x,t)$ with respect to the given fuzzy partition, and the matrix entries are called the components of the F-transform of $u(x,t)$.

Obviously to transform the F-transform back we use the following inverse F-transform formula [13, 14]

$$u_{n,m}^{F}(x,t) = \sum_{i=1}^{n} \sum_{j=1}^{m} U_{ij} A_i(x) B_j(t) \quad (5)$$

4. Numerical Algorithm

The time dependent fractional diffusion equation can be simply obtained by replacing the time derivative with a fractional derivative of order α.

$$\frac{\partial^\alpha}{\partial t^\alpha} u(x,t) = k \frac{\partial^2}{\partial x^2} u(x,t) + q(x,t), \quad (x,t) \in \mathcal{D}_0^2 \quad (6)$$

Where $\mathcal{D}^2 = [a,b] \times [c,d]$ and $0 < \alpha \leq 2$

With the initial condition

$$u(x,0) = f(x) \qquad a \leq x \leq b$$

And the boundary conditions

$$u(a,t) = u(b,t) = 0 \qquad c \leq t \leq d$$

First for calculations simplicity we will define the uniform partition on $[a,b]$, where $a = x_1 < \cdots < x_n = b$ with steps $h_x = x_{i+1} - x_i$ on $[a,b]$, and define a basic function $\{A_1, \dots, A_n\}$ of $[a,b]$, and define the partition $c = t_1 < \cdots < t_m = d$ with steps $h_t^\alpha = t_{i+1} - t_i$ on $[c,d]$, and define a basic function $\{B_1, \dots, B_m\}$ of $[c,d]$, so we construct an uniform fuzzy partition of $\mathcal{D}^2$.

Now we will apply F-transform on (6), and using the linearity property to get

$$F^2\left[u^{t^{\alpha}}\right] = kF^2[u^{xx}] + F^2[q] \tag{7}$$

Where $F^2\left[u^{t^{\alpha}}\right]$, $F^2[u^{xx}]$ and $F^2[q]$ are nm matrices of the F-Transform components of $\frac{\partial^\alpha}{\partial t^\alpha} u(x,t), \frac{\partial^2}{\partial x^2} u(x,t)$ and $q(x,t)$ respectively, and entries of equation (7) can be written as the next linear combination.

$$\frac{\int_c^d \int_a^b \frac{\partial^\alpha}{\partial t^\alpha} u(x,t) A_i(x) B_j(t)\, dxdt}{\int_c^d \int_a^b A_i(x) B_j(t)\, dxdt} = k \frac{\int_c^d \int_a^b \frac{\partial^2}{\partial x^2} u(x,t) A_i(x) B_j(t)\, dxdt}{\int_c^d \int_a^b A_i(x) B_j(t)\, dxdt} + \frac{\int_c^d \int_a^b \sum_{k=1}^{j} g(x,t)\ A_i(x) B_j(t)\, dxdt}{\int_c^d \int_a^b A_i(x) B_j(t)\, dxdt} \tag{8}$$

To use our algorithm for solving the above fractional problem, the time fractional derivative will be replaced by its finite difference approximation

$$\frac{\partial^\alpha u_{(i,j)}}{\partial t^\alpha} = \frac{1}{h_t^\alpha} \sum_{k=0}^{j} \omega_k^\alpha\, u_{(i,j-k)} \tag{9}$$

where $\omega_k^{(\alpha)}$ is the Grünwald weights and computed with the recurrence relationships.

$$\omega_0^{(\alpha)} = 1\,, \omega_k^{(\alpha)} = \left(1 - \frac{\alpha+1}{k}\right) \omega_{k-1}^{(\alpha)}\,, \quad k = 1,2,\ldots$$

And the space second derivative will be also replaced with its central difference approximation

$$\frac{\partial^2 u_{(i,j)}}{\partial x^2} = \frac{u_{(i+1,j)} - 2u_{(i,j)} + u_{(i-1,j)}}{h_x^2} \tag{10}$$

After substituting equations (9) and (10) into (8) and simplify the equation we will get the following form

$$-rU_{(i-1,j)} + (2r+1)U_{(i,j)} - rU_{(i+1,j)} = h_t^\alpha G_{ij} - \sum_{k=1}^{j} \omega_{k-1}^\alpha\, U_{(i,j-k)} \tag{11}$$

where

$$r = \frac{kh_t^\alpha}{h_x^2}\,, i = 1, \ldots., n-1, \qquad j = 1, \ldots. m$$

and $U_{(i-1,j)}, U_{(i,j)}$, and $U_{(i+1,j)}$ are components in $U_{n\times m}$, and the boundary conditions

$$U_{(1,j)} = u_{(n,j)} = 0 \tag{12}$$

And the initial condition

$$U_{(i,1)} = f(x_i) \tag{13}$$

Equation (11) can be written in a matrix form using the matrix form as

$$A\boldsymbol{U_j} = h_t^{\alpha}\boldsymbol{G_j} - \sum_{k=1}^{j} \omega_k^{\alpha}\, U_{(i,j-k)} \tag{14}$$

where $A_{(n-2)}$ is a square matrix with

$$A_{ij} = \begin{cases} (2r+1) & i = j \\ -r & j = i+1 \\ -r & j = i-1 \\ 0 & elsewhere \end{cases}$$

$\boldsymbol{U_j} = \left(U_{(i,j)}\right)_{i=2}^{n-1}$, and $\boldsymbol{G_j} = \left(G_{(i,j)}\right)_{i=2}^{n-1}$ are vectors for $j = 2, \dots, m$

The components of the F-transform of $u_{(x,t)}$ are computed from equations (12-14)

$$F^2[u] = \begin{bmatrix} U_{11} & U_{12} & \cdots & U_{1m} \\ U_{21} & U_{22} & \cdots & U_{2m} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ U_{n1} & \cdots & \cdots & U_{nm} \end{bmatrix} \tag{15}$$

The continuous approximation of $u_{(x,t)}$ can be obtained using equation (15) and the inverse formula of the F-transform in equation (5).

Example Consider the time dependent fractional diffusion equation

$${}_0^cD_t^{\alpha}u(x,t) - \frac{\partial^2}{\partial x^2}u(x,t) = \sin(x), \qquad 0 \le x \le 1, \qquad 0 \le t \le 1$$

The differential equation is solved in $[0,1] \times [0,1]$ with the initial condition

$$u(x,0) = 0, \frac{\partial u(x,0)}{\partial x} = 0 \;, \qquad 0 < x < 1$$

and boundary conditions

$$u(0,t) = u(1,t) = 0, \qquad 0 < t < 1$$

The exact solution of the above problem is:

$$u(x,y) = \frac{1}{\pi^2}[1 - E_{\alpha}(-\pi^2 t^{\alpha})]\sin(\pi x)$$

Where E_{α} is one-parameter Mittag-Leffler function.

The new algorithm in section (4) is applied to above fractional differential equation, it's simulated as a user-subroutine on the mathematical code MATLAB with double precision calculations. Equations (12) to (14) are applied to find the F-Transform components of u for different partitions and values of $\alpha = 0.7, 1, 1.7$. Fig. (1) and table (1) show the exact solution and approximated computed solutions of the above fractional order differential equation for $\alpha = 0.7$ with different values of n, m. Fig. (2) and table (2) show the exact solution and the approximated computed solutions of the above fractional order differential equation for $\alpha = 1$ with different values of n, m, and Fig. (3) and table (3) show the exact solution and the approximated computed solutions of the above fractional order differential equation for $\alpha = 1.7$ with different values of n, m. Computational results are very close to analytical ones, and get more accurate as the number of partitions m and n increasing.

Table 1-a The Exact solution $u(x, y)$ and the computed solution for $\alpha = 0.7, m = n = 10$

x	t	**Exact Solution**	**Computed solution**			
			Sinusoidal shaped basic functions		**triangular shaped basic functions**	
			n=m=10	**Absolute error**	**n=m=10**	**Absolute error**
0.125	0.125	0.031626418463580	0.024713929935935	6.9125e-003	0.027059073360794	4.5673e-003
0.225	0.325	0.060046860110012	0.058389397119987	1.6575e-003	0.058631302423215	1.4156e-003
0.525	0.425	0.093854416211246	0.092116012051016	1.7384e-003	0.091828779038076	2.0256e-003
0.625	0.825	0.089636270201233	0.087979893983999	1.6564e-003	0.088769252563981	8.6702e-004
0.825	0.925	0.050883961002329	0.051650655718926	7.6669e-004	0.050441238238927	4.4272e-004

Table 1-b The Exact solution $u(x, y)$ and the computed solution for $\alpha = 0.7, m = 20, n = 30$

x	**t**	**Exact solution**	**Computed solution**			
			Sinusoidal shaped basic functions		**triangular shaped basic functions**	
			n=m=20	Absolute error	n=m=20	Absolute error
0.125	0.125	0.031626418463580	0.028422902970668	3.2035e-003	0.029395339869247	2.2311e-003
0.225	0.325	0.060046860110012	0.058041849318320	2.0050e-003	0.059236400359664	8.1046e-004
0.525	0.425	0.093854416211246	0.093266115621467	5.8830e-004	0.093313717256758	5.4070e-004
0.625	0.825	0.089636270201233	0.089073170298655	5.6310e-004	0.089506627066611	1.2964e-004
0.825	0.925	0.050883961002329	0.049590403634651	1.2936e-003	0.050748227928774	1.3573e-004

Table 2-a The Exact solution $u(x,y)$ and the computed solution for $\alpha = 1, m = n = 10$

x	t	Exact solution	Computed Solution			
			Sinusoidal shaped basic functions		triangular shaped basic functions	
			n=m=10	Absolute Error	n=m=10	Absolute Error
0.125	0.125	0.027482466014920	0.019175386311012	8.3071e-003	0.021437048746517	6.0454e-003
0.225	0.325	0.063140941004369	0.058462636621803	4.6783e-003	0.058452754064054	4.6882e-003
0.525	0.425	0.099485929326756	0.095046676665002	4.4393e-003	0.094460624380619	5.0253e-003
0.625	0.825	0.093581334120924	0.091964832520955	1.6165e-003	0.092779455648408	8.0188e-004
0.825	0.925	0.052934432601019	0.053861869431872	9.2744e-004	0.052587173317306	3.4726e-004

Table 2-b The Exact solution $u(x,y)$ and the computed solution for $\alpha = 1, m = 20, n = 30$

x	t	Exact solution	Computed solution			
			Sinusoidal shaped basic functions		Triangular shaped basic functions	
			n=30, m=20	Absolute error	n=30, m=20	Absolute error
0.125	0.125	0.027482466014920	0.024146128895380	3.3363e-003	0.024112511967642	3.3700e-003
0.225	0.325	0.063140941004369	0.060660246820567	2.4807e-003	0.060735980106444	2.4050e-003
0.525	0.425	0.099485929326756	0.097577244599283	1.9087e-003	0.097531234991589	1.9547e-003
0.625	0.825	0.093581334120924	0.093909992464966	3.2866e-004	0.093504450679766	7.6883e-005
0.825	0.925	0.052934432601019	0.052379291000909	5.5514e-004	0.052934442051871	9.4509e-009

Table 3-a The Exact solution $u(x,y)$ and the computed solution for $\alpha = 1.7, m = n = 10$

x	t	Exact solution	Computed solution			
			Sinusoidal shaped basic functions		triangular shaped basic functions	
			n= m=10	Absolute error	n=m=10	Absolute error
0.125	0.125	0.006913105647992	0.007199844503153	2.8674e-004	0.008613020524771	1.6999e-003
0.225	0.325	0.049722851734235	0.047011126290422	2.7117e-003	0.046305279189023	3.4176e-003
0.525	0.425	0.105655786293884	0.094275799837509	1.1380e-002	0.091878673860641	1.3777e-002
0.625	0.825	0.141735239590746	0.120180949616185	2.1554e-002	0.121241683582674	2.0494e-002
0.825	0.925	0.076586729407653	0.069894021004826	6.6927e-003	0.068077511468866	8.5092e-003

Table 3-b The Exact solution $u(x,y)$ and the computed solution for $\alpha = 1.7, m = n = 100$

x	t	Exact solution	Computed solution			
			Sinusoidal shaped basic functions		triangular shaped basic functions	
			n= m=100	Absolute error	n=m=100	Absolute error
0.125	0.125	0.006913105647992	0.007058071887871	1.4497e-004	0.007154639430869	2.4153e-004
0.225	0.325	0.049722851734235	0.049128185844175	5.9467e-004	0.049509917091144	2.1293e-004
0.525	0.425	0.105655786293884	0.104283274387151	1.3725e-003	0.104448645287637	1.2071e-003
0.625	0.825	0.141735239590746	0.139127922419599	2.6073e-003	0.139309323966533	2.4259e-003
0.825	0.925	0.076586729407653	0.075196021343482	1.3907e-003	0.075551096437673	1.0356e-003

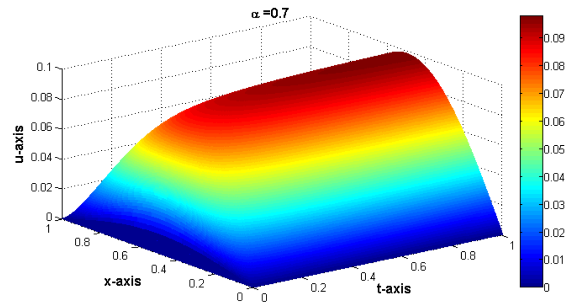

Fig.(3-a) The exact solution $u(x, y)$ for $\alpha = 0.7$

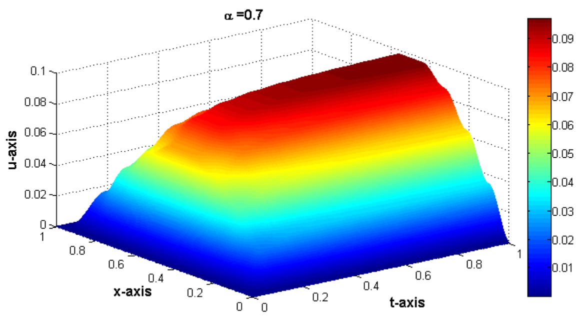

Fig.(1-b) The computed solution $u(x, y)$ for $m = n = 10$ with sinusoidal shaped functions

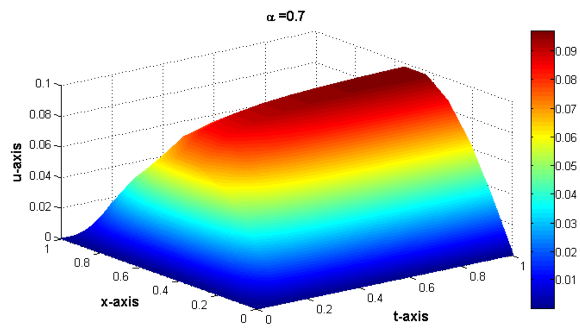

Fig.(1-c) The computed solution $u(x, y)$ for $m = n = 10$ with triangular shaped functions

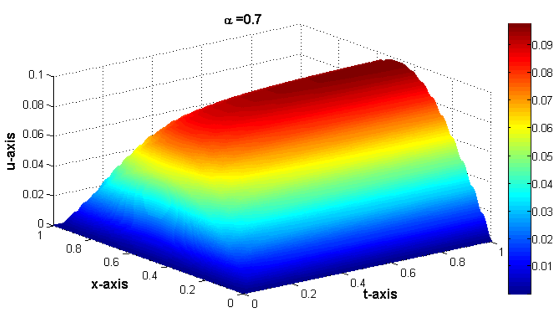

Fig.(1-d) The computed solution $u(x, y)$ for $m = n = 20$ with sinusoidal shaped functions

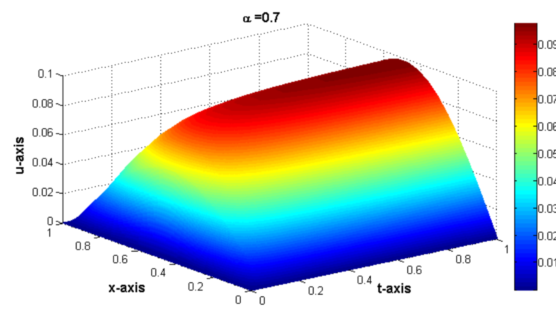

Fig.(1-e) The computed solution $u(x, y)$ for $m = n = 20$ with triangular shaped functions

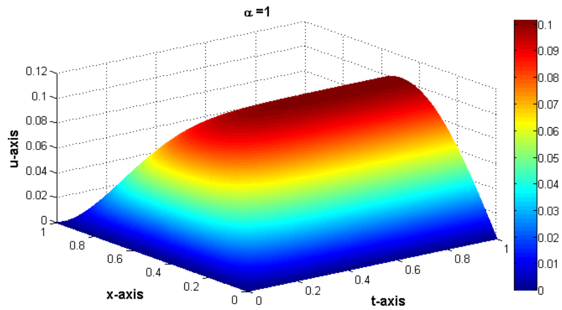

Fig.(2-a) The exact solution $u(x, y)$ for $\alpha = 1$

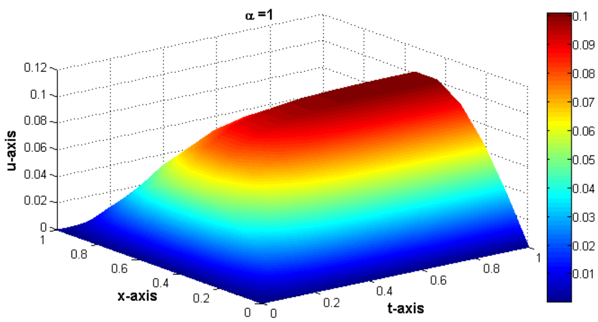

Fig.(2-b) The computed solution $u(x, y)$ for $m = n = 10$ with triangular shaped functions.

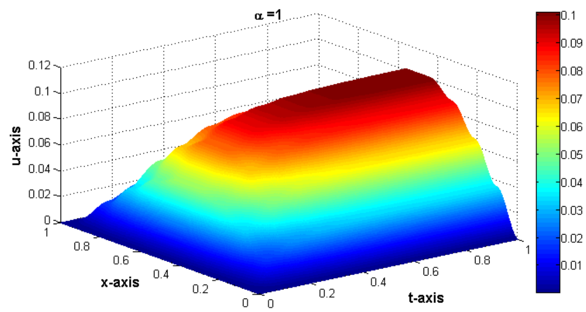

Fig.(2-c) The computed solution $u(x, y)$ for $m = n = 10$ with sinusoidal shaped functions.

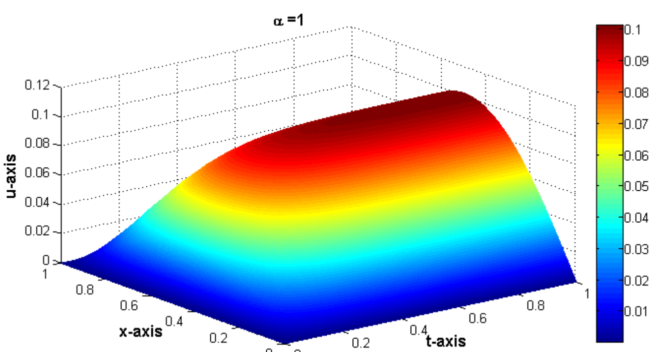

Fig.(2-d) The computed solution $u(x, y)$ for $m = 20, n = 30$ with triangular shaped functions.

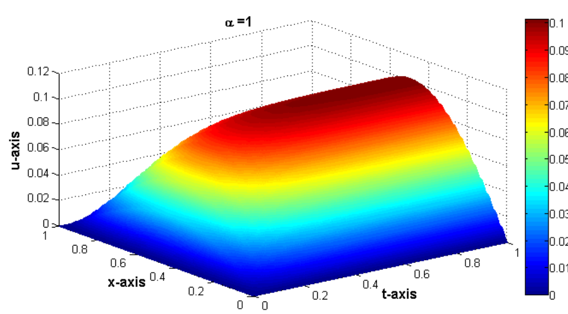

Fig.(2-e) The computed solution $u(x, y)$ for $m = 20$ $n = 30$ with sinusoidal shaped functions.

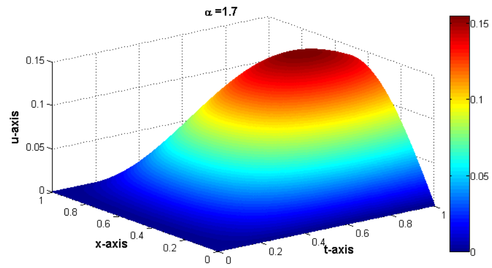

Fig.(3-a) The exact solution $u(x, y)$ for $\alpha = 1.7$

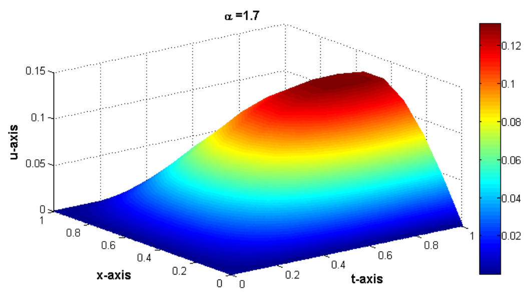

Fig. (3-b) The computed solution $u(x, y)$ for $m = n = 10$ with triangular shaped functions.

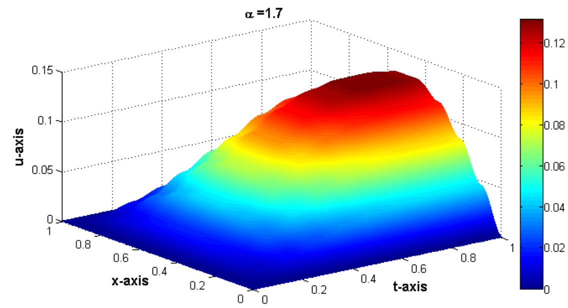

Fig.(3-c) The computed solution $u(x, y)$ for $m = n = 10$ with sinusoidal shaped functions.

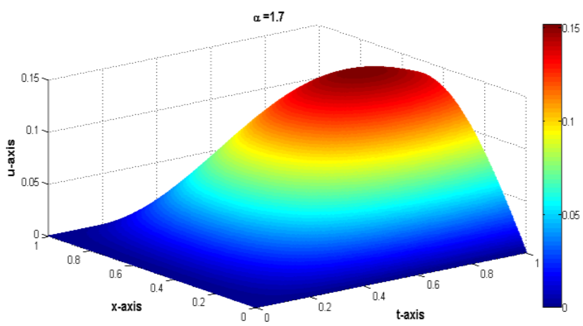

Fig.(3-d) The computed solution $u(x, y)$ for $m = n = 100$ with sinusoidal shaped functions.

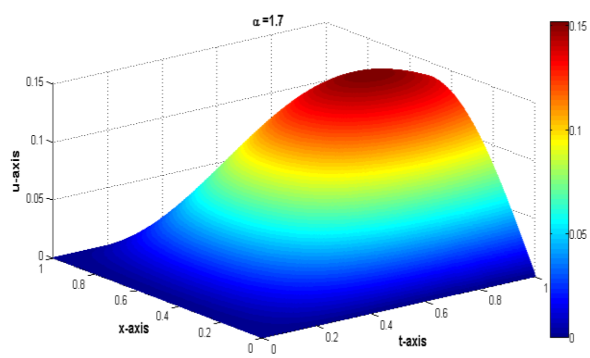

Fig.(3-e) The computed solution $u(x, y)$ for $m = n = 100$ with triangular shaped functions.

5. Conclusions

In this article, The F-Transform is simulated to obtain a new numerical algorithm to find the solution of fractional order partial differential equations, It's shown that the F –transform can be applied efficiently to the fractional order as well as the integer order differential equations [3]. Computational results indicate that, the proposed incremental algorithm provided very promising method in estimating the solution of rate dependent fractional order partial differential equations. It's well known that solving the time dependent fractional diffusion problems is one of the most difficult problems, but partitioning $\boldsymbol{\mathcal{D}}^{\mathbf{2}} = [a, b] \times [c, d]$ and using F-transform to get a continuous computed solution to the problem decreases the number of hard calculations we have to deal with.

In our calculations we have used the numerical fractional forms of Podlubny [19] and we obtained a good solution with an accepted error relative to the analytical solution as the number of partition increased, same calculations can be done using other numerical fractional formulas.

Conflict of Interests

The authors declare that there is no conflict of interests.

REFERENCES

[1] Abu-Saman A. M. and Assaf A., Stability and Convergence of Crank-Nicholson Method for Fractional Advection Dispersion Equation, Advances in Applied Mathematical Analysis, 2(2007), No. 2, 1-9.

[2] Abu-Saman A.M., Numerical Algorithm for Plasma Transport Equations of Fractional Order, Journal of Rajasthan Academy of Physical Sciences, 7(2008), No.2, 163-168.

[3] Abu-Saman A. M. and El-Zerii H., 'Numerical Solution of Initial Value Problems using Fuzzy Transform', International Journal of Applied Engineering Research, 9(2014), No. 23, 13667-13675.

[4] Achar B. N., Hanneken J. W., Enck T., and Clarke T., Dynamics of the fractional oscillator, Physica A 297(2001), 361-367.

[5] Chen G., Tat Pham T., Introduction to fuzzy set, fuzzy logic, and fuzzy control system, CRC, (2000).

[6] Dankova M. and Valasek R., Fuzzy transform and the problem of image fusion, Journal of Electrical Engineering, 57(2006).

[7] Deng Z., Singh V. P., Bengtsson L., Numerical solution of fractional advection-dispersion equation, J. Hydraulic Eng. 130(2004), 422-431.

[8] Diethelm K., Freed A, Modeling and simulating viscoelastic and viscoplastic materials using nonlinear, fractional-order differential equations, submitted for publications.

[9] Dimitrov Y., Numerical Approximations For Fractional Differential, http://www.researchgate.net/publication, 2013.

[10] Liu F., Ahn V., Turner I., Numerical solution of the space fractional Fokker-Planck equation, J. Comput. Appl. Math. 166 (2004), 209-219.

[11] Meerschaert M. M., the fractional calculus Project, MAA Student Lecture, Phoenix January, (2004).

[12] Patane G., fuzzy transform and least-squares approximation, Elsevier, (2010).

[13] Perfilieva I., Fuzzy transforms in image compressions and fusion, Acta Mathematica Universitatis Ostraviensis, 15(2007), 27-37.

[14] Perfilieva I., and Hodakova P., Fuzzy and fourier Transforms, EUSFLAT-LFA, 2011, 452-456.

[15] Perfilieva I., and Valasek R., Data compression on the basis of fuzzy transform, EUSFLAT-LFA conference, 2005, 663-668.

[16] Perfilieva I., Novak V., Dvorak A., Fuzzy Transform in the analysis of data, International Journal of Approximate Reasoning 48 (2008), 36-46.

[17] Perfilieva I., Fuzzy Transforms and Universal Approximation, EUSFLAT conference, 2003, 529-533.

[18] Perfilieva I., Approximating models based on fuzzy transforms, EUSFLAT-LFA conference, 2005, 645-650.

[19] Podlubny I., Fractional Differential Equations, Academic Press, 1999.

[20] Samko S .G., Kilbas A. A. and Marichev O. I., Fractional Integrals and Derivatives Theory and Applications, Gordon and Breach Science Publishers, 1993.

[21] Stefanini L., Fuzzy transform and smooth Functions. IFSA-EUSFLAT, 2009, 579-584.

[22] Stepnicka M. and Valasek R., Fuzzy Transform and Their Application on Wave Equation, Journal of Electrical Engineering, 52(2004), 7-10.

[23] Stepnika M., Fuzzy transform and its Applications to problems in engineering practice, Ph.D. Thesis Ostrava Uni., 2007.

[24] Tadjeran C., Meerschaert M. and Scheffler H., A second-order accurate numerical approximataion for the fractional diffusion equation, Journal of Computational physics,vol. 213 (2006), 205-213.

[25] Tadjeran C., Meerschaert M. and Scheffler H., Finite difference methods for two-dimensional fractional dispersion equation, Journal of Computational physics, 211 (2006), 249-261.

[26] Zadah L., Fuzzy set theory, Information and control 8^{th}, 1965, 338-353.