Solution of Fuzzy Differential Equation under Generalized Differentiability by Genetic Programming

N. Kumaresan, J. Kavikumar, M. Kumudthaa, and Kuru Ratnavelu

Abstract—In this paper, solution of fuzzy differential equation under general differentiability is obtained by genetic programming (GP). The obtained solution in this method is equivalent or very close to the exact solution of the problem. Accuracy of the solution to this problem is qualitatively better. An illustrative numerical example is presented for the proposed method.

Keywords—Fuzzy differential equation, Generalized differentiability, Genetic programming and H-difference.

I. INTRODUCTION

FUZZY set theory is a powerful tool for modelling uncertainty and for processing vague or subjective information in mathematical models. The main directions of development of this subject have been diverse with applications to variety of real problems like the golden mean [8], quantum optics, gravity [10], synchronize hyperchaotic systems [34], chaotic system, medicine [1], [3], and engineering problems [15]. Particularly, fuzzy differential equation is an important topic from the theoretical point of view (see [2], [11], [17], [22]) as well as its applications like in population models [13], [14], civil engineering and hydraulics.

Differentiable fuzzy valued mappings were initially studied by Puri and Ralescu [25]. They generalized and extended the concept of Hukuhara differentiability (H-derivative) for set valued mappings to the class of fuzzy mappings. Subsequently, using H-derivative, Kaleva [16] started to develop a theory for fuzzy differential equations.

In the last few years, many works have been done by several authors in theoretical and applied fields for fuzzy differential equations with H-derivative (see [28], [29], [32], [33]). Now, in some cases this approach suffers certain disadvantages since the diameter diam(x(t)) of the solution is unbounded as time t increases [9]. This problem demonstrates that in some case this interpretation is not a good generalization of the associated crisp case.

The generalized differentiability was introduced and studied in [4], [5], [6], [7]. This concept allows us to resolve the above mentioned shortcoming. Indeed, the generalized derivative is defined for a larger class of fuzzy number valued functions than Hukuhara derivative. Hence, this differentiability concept

N. Kumaresan and Kuru Ratnavelu are with the Institute of Mathematical Science, Universiti Malaya, Kuala Lumpur, Malaysia.(drnk2008@um.edu.my)

J. Kavikumar is with the Department of Science and Mathematics, Faculty of Science, Technology and Human Development, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, Malaysia.(kavi@uthm.edu.my)

M. Kumudthaa is with the SMK SERI TASIK, 6/106 Bandar Sri Permaisuri, 56000, Kuala Lumpur, Malaysia.(kumudthaa@yahoo.com)

is used in the present paper. Under appropriate conditions, the fuzzy initial value problem considered under this interpretation has locally two solutions. In this paper, the genetic algorithm is used in GP to compute the solution of fuzzy differential equation.

Genetic Programming(GP) is an evolutionary algorithm that attempts to evolve solution to the given problem by using concepts taken from naturally occurring evolving process. The technique is based on the evolution of a large number of candidate solutions through genetic operations such as reproduction, crossover and mutation. It is based upon the Genetic algorithm(GA) [12], which exploits the process of natural selection based on a fitness measure to breed a population of trial solution that improves over time. While GA usually operates on (coded) strings of numbers but GP uses the principles and ideas from biological evolution to guide the computer to acquire desired solution. The search space is too large to attempt a brute force search, the method must be utilized to reduce the number of examined solutions. In this search, initially the population looks a bit like a cloud of randomly selected points, but that generation after generation it moves in the search space following a well defined trajectory. The generation is achieved with the help of grammatical evolution, because grammatical evolution can produce programmes in an arbitrary language, the genetic operations are faster and also because it is more convenient to symbolically differentiate mathematical expression. The code production is performed using a mapping process governed by grammar expressed in Backus Naur Form(BNF)[23]. In analogy to nature, the potential solution is an individual in some collection or population of potential solutions. The individuals who are stronger, meaning higher ranked according to fitness function, will be used to determine the next collection of potential solution. A new generation will be arisen by employing analogs of reproduction and mutation.

This means that GP has advantages over other algorithms as it can perform optimization at a structural level. This enabled Koza [19] to demonstrate the application of GP algorithm to a number of problem domains, including regression, control and classification. Research in this area has grown rapidly and encompassed a wide range of problems. GP techniques have been successfully applied in various engineering fields like signal processing [26], electrical circuit design [18], scheduling [21], process controller evolution [27] and modelling of both steady-state and dynamic processes [20].

Although parallel algorithms can compute the solutions

World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:8, 2011

faster than sequential algorithms, there have been no report on genetic programming solutions [31] for matrix Riccati differential equation. This paper focuses upon the implementation of genetic programming approach for solving fuzzy differential equation in order to get the optimal solution.

This paper is organized as follows. In section 2, the basic concepts and fuzzy differential equation are described. In section 3, genetic programming method is presented. In section 4, numerical example is discussed. The final conclusion section demonstrates the efficiency of the method.

II. BASIC CONCEPTS AND FUZZY DIFFERENTIAL EQUATION

Let X be a nonempty set. A fuzzy set u in X is characterized by its membership function $u : X \rightarrow [0,1]$. Then $u(x)$ is interpreted as the degree of membership of a element x in the fuzzy set u for each $x \in X$.

Definition 1: Let $\mathcal{F}^n$ be the space of all compact and convex fuzzy sets on $\mathbf{R}^n$. Let $u, v \in \mathcal{F}^n$. If there exists $w \in \mathcal{F}^n$ such that $u = v \oplus w$, then w is called the H-difference of u and v and it is denoted by $u \ominus v$.

Definition 2: Let $F : T \rightarrow \mathcal{F}^n$ and $t_0 \in T$. The function F is said to be differentiable at t_0 if

(I) an element $F'(t_0) \in \mathcal{F}^n$ exist such that, for all $h > 0$ sufficiently near 0, there are $F(t_0+h) \ominus F(t_0), F(t_0) \ominus F(t_0 - h)$ and the limits

$$\lim_{h \to 0^+} \frac{F(t_0 + h) \ominus F(t_0)}{h} = \lim_{h \to 0^+} \frac{F(t_0) \ominus F(t_0 - h)}{h}$$

are equal to $F'(t_0)$.

(or)

(II) there is an element $F'(t_0) \in \mathcal{F}^n$ exist such that, for all $h < 0$ sufficiently near 0, there are $F(t_0+h) \ominus F(t_0), F(t_0) \ominus F(t_0 - h)$ and the limits

$$\lim_{h \to 0^-} \frac{F(t_0 + h) \ominus F(t_0)}{h} = \lim_{h \to 0^-} \frac{F(t_0) \ominus F(t_0 - h)}{h}$$

are equal to $F'(t_0)$.

Note that if F is differentiable in the first form (I), then it is not differentiable in the second form (II) and viceversa.

Given that $F : T \rightarrow \mathcal{F}$ is a function and $[F(t)]^\alpha = [f_\alpha(t), g_\alpha(t)]$, for each $\alpha \in [0, 1]$. The following result is the fundamental for solving a fuzzy differential equation.

Theorem 1: Let $F : T \rightarrow \mathcal{F}$ be a function. Then
(i) If F is differentiable in the first form (I), then f_α and g_α are differentiable functions and

$$[F'(t)]^\alpha = [f'_\alpha(t), g'_\alpha(t)]. \tag{1}$$

(ii) If F is differentiable in the second form (II), then f_α and g_α are differentiable functions and

$$[F'(t)]^\alpha = [g'_\alpha(t), f'_\alpha(t)]. \tag{2}$$

Theorem 2: Let $F : T \rightarrow \mathcal{F}$ be a continuous function. Then
(i) If F is differentiable in the first form (I), then F' is integrable if and only if $F(a) \prec F(t)$ for all $t \in T$.

(ii) If F is differentiable in the second form (II), then F' is integrable if and only if $F(t) \prec F(a)$ for all $t \in T$.

A. Fuzzy differential equation

Consider the fuzzy differential equation

$$x' = F(t, x(t)), \quad x(a) = x_0, \tag{3}$$

where $F : [a, b] \times \mathcal{F} \rightarrow \mathcal{F}$ is a continuous fuzzy mapping and x_0 is a fuzzy interval.

The solution of the fuzzy differential equation (3) is dependent of the choice of the derivative: in the first form or in the second form. The equations (1) and (2) in Theorem 1 give us an useful procedure to solve the fuzzy differential equation (3). For this, let

$$[x(t)]^\alpha = [u_\alpha(t), v_\alpha(t)]$$

and

$$[F(t, x(t))]^\alpha = [f_\alpha(t, u_\alpha(t), v_\alpha(t)), g_\alpha(t, u_\alpha(t), v_\alpha(t))].$$

Example: Let us consider the fuzzy differential equation

$$x'(t) = -\lambda x(t), \quad x(0) = x_0, \tag{4}$$

where $\lambda > 0$ and the initial condition x_0 is a symmetric triangular fuzzy number with support $[-a, a]$. That is,

$$[x_0]^\alpha = [-a(1 - \alpha), a(1 - \alpha)] = (1 - \alpha)[-a, a].$$

If $x'(t)$ is considered in the first form(I), the fuzzy differential system will be as given below:

$$u'_\alpha(t) = -\lambda v_\alpha(t), \quad u_\alpha(0) = -a(1 - \alpha)$$
$$v'_\alpha(t) = -\lambda u_\alpha(t), \quad v_\alpha(0) = a(1 - \alpha).$$

The solution of this system is $u_\alpha(t) = -a(1 - \alpha)e^{\lambda t}$ and $v_\alpha(t) = a(1-\alpha)e^{\lambda t}$. Therefore, the fuzzy function x(t) solving (4) has level sets

$$[x(t)]^\alpha = [-a(1 - \alpha)e^{\lambda t}, a(1 - \alpha)e^{\lambda t}]$$

for all $t \geq 0$.

If $x'(t)$ is considered in the second form(II), the fuzzy differential system will be as given below:

$$u'_\alpha(t) = -\lambda u_\alpha(t), \quad u_\alpha(0) = -a(1 - \alpha)$$
$$v'_\alpha(t) = -\lambda v_\alpha(t), \quad v_\alpha(0) = a(1 - \alpha).$$

The solution of this system is $u_\alpha(t) = -a(1 - \alpha)e^{-\lambda t}$ and $v_\alpha(t) = a(1 - \alpha)e^{-\lambda t}$. Therefore, the fuzzy function x(t) solving (4) has level sets

$$[x(t)]^\alpha = [-a(1 - \alpha)e^{-\lambda t}, a(1 - \alpha)e^{-\lambda t}]$$

for all $t \geq 0$.

III. GENETIC PROGRAMMING METHOD

In this approach, GP is used to obtain a set of expressions. If the required number expressions satisfy the fitness function, it will be the optimal solution of fuzzy differential equation. The scheme of computing optimal solution is given in Figure 1.

World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:8, 2011

Fig. 1.   Flow Chart

TABLE I
THE GRAMMAR OF THE PROPOSED METHOD

S : := <exp>		(0)
<exp> ::= <exp> <op> <exp>		(0)
	|(<exp>)	(1)
	| <fun>(<exp>)	(2)
	| <digit>	(3)
	| x	(4)
	| y	(5)
	| z	(6)
<op> : : = +		(0)
	| -	(1)
	| *	(2)
	| /	(3)
<fun> : : = sin		(0)
	| cos	(1)
	| exp	(2)
	| log	(3)
<digit> : : = 0		(0)
	| 1	(1)
	| 2	(2)
	| 3	(3)
	| 4	(4)
	| 5	(5)
	| 6	(6)
	| 7	(7)
	| 8	(8)
	| 9	(9)

TABLE II
EXAMPLE OF CONSTRUCTION OF EXPRESSION USING BNF GRAMMAR

String	Chromosome	Operation
<exp>	7, 2, 10, 4, 4, 2,11,16 ,30,5	7 mod 7 = 0
<exp><op><exp>	2, 10, 4, 4, 2, 11, 16, 30, 5	2 mod 7 = 2
fun(<exp><op><exp>)	10, 4, 4, 2, 11, 16, 30, 5	10 mod 4 = 2
Exp <exp><op><exp>	4, 4, 2, 11, 16, 30, 5	4 mod 7 = 4
Exp <x><op><exp>	4, 2,11, 16, 30, 5	4 mod 4 = 0
Exp<x>+<exp>	2, 11, 16, 30, 5	2 mod 7 = 2
Exp<x>+ fun <exp>	11, 16, 30, 5	11 mod 4 = 3
Exp<x>+ (log)(<exp>)	16, 30, 5	16 mod 7 = 2
Exp<x>+ log(fun<exp>)	30,5	30 mod 4 = 2
Exp<x>+ log(Exp<exp>)	5	5 mod 7 = 5
Exp<x>+ log(Exp<y>)		

A. Initialization of the Population

The first step is to initialize the population. An initial population of the desired size is generated randomly. The length of each chromosome is to be set according to the nature of the problem. Each program or individual in the population is generally represented as a Parse tree composed of function and data/terminals appropriate to the problem domain.

B. Grammatical Evolution

Grammatical evolution is an evolutionary algorithm that can produce code in any programming language. The algorithm starts from the start symbol of the grammar and gradually creates the program string, by replacing non terminal symbol with right hand of the selected production rule, first read an element from the chromosome (with value V) and compute Rule = V mod NR, where NR is the number of rules for the specific non terminal symbol which is shown in Table 1.

The symbol S in the grammar denotes the start symbol of the grammar. For example, suppose we have the chromosome x = {1, 2, 10, 4, 4, 2, 11, 16, 30, 5}. The method of producing a valid expression is shown in Table 2. The rule = V mod NR is applied in each row of Table 2. In first line V=7, NR=7 (Number of elements in <exp> group in Table 1). The next line <exp><op><exp> is obtained from Table 1 for the result zero of the first line. Similarly the remaining rows of the Table 2 can be found. The function which corresponds to the chromosome is Exp(x) + log(Exp(y)), (for details see [24], [30]).

C. Fitness function

The aim of the fitness function is to provide a basis for competition among available solutions and to obtain the opti-

mal solution. Hence, the fitness function for fuzzy differential equation is defined as

$$E_r = \left(\dot{x}(t) - \phi(x(t)) \right)^2. \tag{5}$$

D. Genetic operators:

The genetic operators such as reproduction, crossover and mutation explained below:

1) Reproduction :: In reproduction process, best chromosome in a population is probabilistically assigned a large number of copies according to their fitness value. It is important to note that no new strings are formed in the reproduction phase. Koza [19] allowed 10 percentage of the population to reproduce.

2) Crossover :: The crossover is applied every generation in order to create new chromosome from the old ones, that will replace the worst individuals in the population. There are many types of crossover operator in which we use single point crossover technique. In that operation, for each couple of new chromosomes two parents are selected, we cut these parent -

World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:8, 2011

chromosomes at a randomly chosen point and exchange the right-hand-side sub-chromosomes, as shown in Figure 2. A random number is generated for each chromosome. If the random number is less than the crossover probability, then the chromosome is chosen for crossover with parent previously chosen.
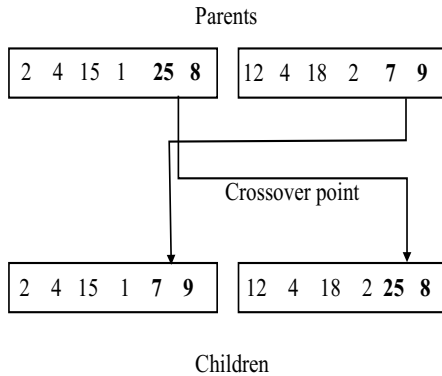
Parents

| 2 | 4 | 15 | 1 | **25** | **8** |

| 12 | 4 | 18 | 2 | **7** | **9** |

Crossover point

| 2 | 4 | 15 | 1 | **7** | **9** |

| 12 | 4 | 18 | 2 | **25** | **8** |

Children

Fig. 2.   one point crossover

3) Mutation :: In the mutation process, for every element in a chromosome a random number in the range [0, 1] is chosen. If the number is less than or equal to the mutation rate the corresponding element is changed randomly, otherwise it remains intact.

E. Termination control

In each generation, a set of expressions are generated by the chromosomes. If an expression minimizes the fitness function E_r (5) to zero or very close to zero and satisfies the initial condition, the process may be stopped; otherwise, the GP approach must be continued.

F. Genetic Programming Algorithm

The algorithm has the following steps

ep 1.   Initialize random population.
ep 2.   Create valid function using grammar.
ep 3.   Evaluate fitness value of the chromosome.
ep 4.   If Fitness tends to zero, stop the procedure. Otherwise proceed to next step.
ep 5.   Generate new population using Genetic operations, Go to step 2.

IV. NUMERICAL EXAMPLE

Consider the fuzzy differential equation

$$x'(t) = -x(t) + 1, \quad x(0) = x_0,$$

where $\lambda > 0$ and the initial condition x_0 is a symmetric triangular fuzzy number with support $[-1, 1]$. That is,

$$[x_0]^\alpha = [-(1-\alpha), (1-\alpha)] = (1-\alpha)[-1, 1].$$

If $x'(t)$ is considered in the first form(I), the fuzzy differential system will be as given below:

$$u'_\alpha(t) = -v_\alpha(t) + 1, \quad u_\alpha(0) = -(1-\alpha)$$
$$v'_\alpha(t) = -u_\alpha(t) + 1, \quad v_\alpha(0) = (1-\alpha).$$

If $x'(t)$ is considered in the second form(II), the fuzzy differential system will be as given below:

$$u'_\alpha(t) = -u_\alpha(t) + 1, \quad u_\alpha(0) = -(1-\alpha)$$
$$v'_\alpha(t) = -v_\alpha(t) + 1, \quad v_\alpha(0) = (1-\alpha).$$

A. Solution obtained using genetic programming

To solve the above fuzzy differential equation, each chromosome is split uniformly in M parts, where M is the number of equations in the system. In the computation process, the parameters are taken as 0.01 for replication rate, 0.9 for the crossover probability and 0.05 for the mutation rate.

After 150 generations, the following expressions satisfy the fitness function and the value of the fitness function tends to 0. The expressions also satisfy the initial conditions. Hence the solution of the first system is

$$u_\alpha(t) = 1 - (exp(-t)) - (exp(t)) + (\alpha * (exp(t)))$$
$$= -(1-\alpha)e^t - e^{-t} + 1,$$

$$v_\alpha(t) = 1 + (exp(t)) - (\alpha * (exp(t)))$$
$$= (1-\alpha)e^t - e^{-t} + 1.$$

Therefore, the fuzzy function x(t) has level sets

$$[x(t)]^\alpha = [-(1-\alpha)e^t - e^{-t} + 1, (1-\alpha)e^t - e^{-t} + 1]$$

for all $t \geq 0$. The solution of the second system is

$$u_\alpha(t) = 1 - (2 * (exp(-t))) + (\alpha * (exp(-t)))$$
$$= -(2-\alpha)e^{-t} + 1,$$

$$v_\alpha(t) = exp(-t) - (\alpha * (exp(-t)))$$
$$= (1-\alpha)e^{-t}.$$

Therefore, the fuzzy function x(t) has level sets

$$[x(t)]^\alpha = [-(2-\alpha)e^{-t} + 1, (1-\alpha)e^{-t}]$$

for all $t \geq 0$.

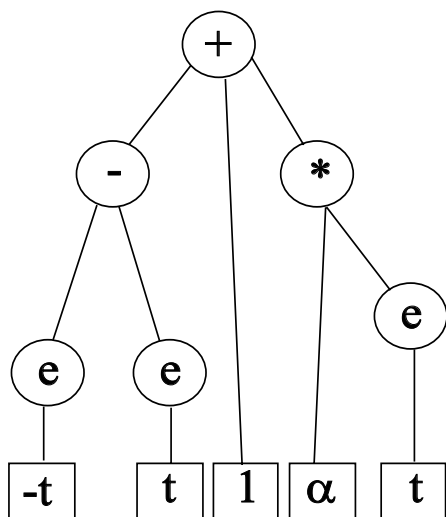The parse trees for the solutions are given in Figures 3, 4, 5 and 6.

World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:8, 2011

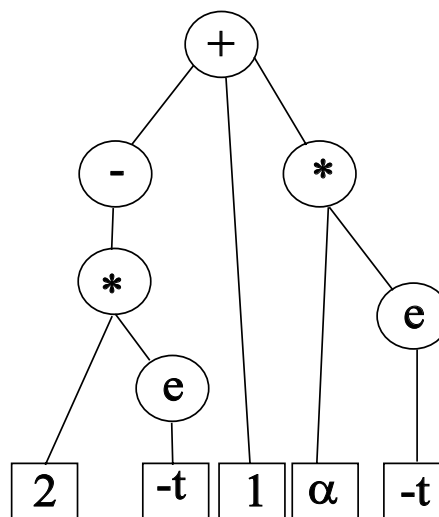Fig. 3.   Tree for $u_\alpha(t)$ for the first system

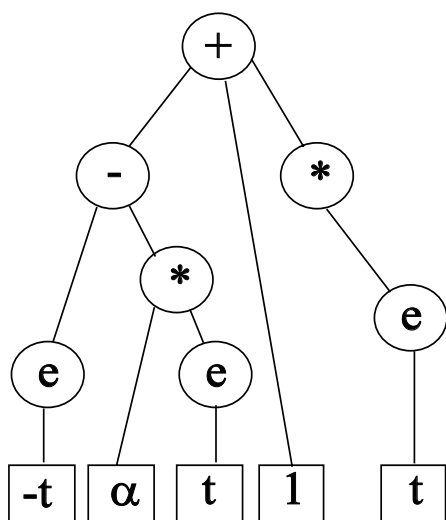Fig. 5.   Tree for $u_\alpha(t)$ for the second system

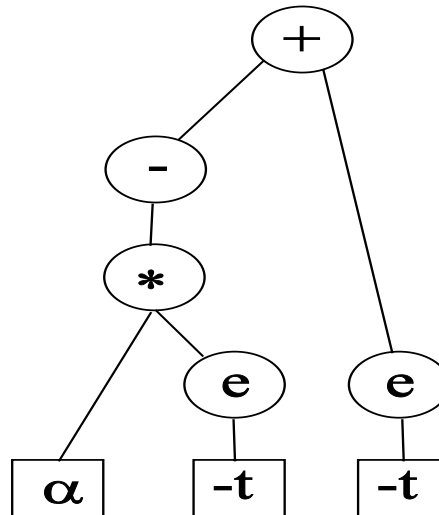Fig. 4.   Tree for $v_\alpha(t)$ for the first system

Fig. 6.   Tree for $v_\alpha(t)$ for the second system

V. Conclusion

The solution of fuzzy differential equation under generalized differentiability can be obtained by using GP approach. In GP approach, the grammar creates some expressions from chromosomes. Then the optimal solution is obtained from the expression with the help of fitness function and terminal criteria. The obtained solution in this method is equivalent or close to the exact solution of the problem. A numerical example is given to illustrate the derived results. In future, GP can be used to solve linear and nonlinear stochastic differential equation in the fuzzy environment.

Acknowledgment

References

[1] M. F. Abbod, D. G. Von Keyserlingk, D. A. Linkens, M. Mahfouf, Survey of utilisation of fuzzy technology in medicine and healthcare, *Fuzzy Sets and Systems*, **120** (2001), 331–349.
[2] S. Abbasbandy, J. J. Nieto, M. Alavi, Tuning of reachable set in one dimensional fuzzy differential inclusions, *Chaos, Solitons & Fractals*, **26** (2005), 1337–1341.
[3] S. Barro, R. Marín, *Fuzzy logic in medicine*, Heidelberg: Physica-Verlag, 2002.
[4] B. Bede, S. G. Gal, Almost periodic fuzzy number valued functions, *Fuzzy Sets and Systems*, **147** (2004), 385–403.
[5] B. Bede, S. G. Gal, Generalizations of the differentiability of fuzzy number valued functions with applications to fuzzy differential equations, *Fuzzy Sets and Systems*, **151** (2005), 581–599.
[6] B. Bede, I. J. Rudas, A. L. Bencsik, First order linear fuzzy differential equations under generalized differentiability, *Inform. Sciences*, **177** (2007), 1648–1662.
[7] Y. Chalco-Cano, H. Román-Flores, On new solutions of fuzzy differential equations, *Chaos, Solitons & Fractals*, **38** (2008), 112–119.
[8] D. P. Datta, The Golden mean, scale free extension of real number system, fuzzy sets and 1/f spectrum in physics and biology, *Chaos, Solitons & Fractals*, **17** (2003), 781–788.
[9] P. Diamond, Brief note on the variation of constants formula for fuzzy differential equations, *Fuzzy Sets and Systems*, **129** (2002), 65–71.

World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:8, 2011

[10] M. S. El Naschie, From experimental quantum optics to quantum gravity via a fuzzy Kähler manifold, *Chaos, Solitons & Fractals*, **25** (2005), 969–977.

[11] T. Gnana Bhaskar, V. Lakshmikantham, V. Devi, Revisiting fuzzy differential equations, *Nonlinear Analysis*, **58** (2004), 351–358.

[12] D. E. Goldberg, Genetic algorithms in search, Optimization and Machine Learning, Addision Wesley, 1989.

[13] M. Guo, X. Xue, R. Li, The oscillation of delay differential inclusions and fuzzy biodynamics models, *Math. Comput. Model.*, **37** (2003), 651–658.

[14] M. Guo, R. Li, Impulsive functional differential inclusions and fuzzy population models, *Fuzzy Sets and Systems*, **138** (2003), 601–615.

[15] M. Hanss, *Applied fuzzy arithmetic: An introduction with engineering applications*, Springer Verlag, Berlin, 2005.

[16] O. Kaleva, Fuzzy differential equations, *Fuzzy Sets and Systems*, **24** (1987), 301–317.

[17] O. Kaleva, A note on fuzzy differential equations, *Nonlinear Analysis*, **64** (2006), 895–900.

[18] J. R. Koza, F.H. Bennett D. Andre, M.A. Keane, Genetic Programming III:*Darwinian Invention and Problem Solving*. Morgan Kaufmann, 1999.

[19] J. R. Koza, Genetic Programming: *On the programming of Computer by means of Natural Selection*. MIT Press: Cambridge, MA, 1992.

[20] B. Mckay, M. Wills G. Barton, Steady-state modelling of chemical process systems using genetic programming, *Comput. Chem. Eng.*, **1** (1997), 981–996.

[21] D.J. Monana, S. Czerwinski, Evolving Control Laws for a Network of Traffic Signals'. *genetic programming, Proceedings of the First International Conference*, Stanford University, California (1996), 333–338.

[22] J. J. Nieto, R. Rodríguez-López, Bounded solutions for fuzzy differential and integral equations, *Chaos, Solitons & Fractals*, **27** (2006), 1376–1386.

[23] M. O'Neill, C. Ryan, Grammatical Evoulation, *IEEE Trans. Evol. Comput.*, **5** (2001), 349–358 .

[24] M. O'Neill, C. Ryan Grammatical Evolution: Evolutionary Automatic programming in an Arbitaty Language, volume 4 of Genetic Programming, Kluwer Academic Publishers, 2003.

[25] M. Puri, D. Ralescu, Differential and fuzzy functions, *J. Math. Anal. Appl.*, **91** (1983), 552–558.

[26] K. C Sharman, A. I. Esparcia Alcazar, Y. Li, Evolving Signal Processing Algorithms by Genetic Programming, *Proceeding of the Third Annual Conference*, (1998), 359–364.

[27] D. Searson, M. Wills, G. Montague, Chemical Process Controller Design Using Genetic Programming, *IEEE Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, (1995), 473–480.

[28] S. Song, Lei Guo, Chumbo Feng, Global existence of solutions to fuzzy differential equations, *Fuzzy Sets and Systems*, **115** (2000), 371–376.

[29] S. Song, C. Wu, Existence and uniqueness of solutions to Cauchy problem of fuzzy differential equations, *Fuzzy Sets and Systems*, **110** (2000), 55–67.

[30] I. G.Tsoulos, I. E.Lagaris, Solving differential equations with genetic programming, *Genet. Program Evolvable Mach*, **7** (2006), 33–54.

[31] A. Vincent Antony Kumar, P. Balasubramaniam, Optimal control for linear singular system using genetic programming, *Appl. Math. Comput.*, **192** (2007), 78–89.

[32] C. X. Wu, S. Song, Stanley Lee, Approximate solutions, existence and uniqueness of the Cauchy problem of fuzzy differential equations, *J. Math. Anal. Appl.*, **202** (1996), 629–644.

[33] C. X. Wu, S. Song, Existence theorem to the Cauchy problem of fuzzy differential equations under compactness-type conditions, *Inform. Sciences*, **108** (1998), 123–134.

[34] H. Zhang, X. Liao, J. Yu, Fuzzy modeling and synchronization of hyperchaotic systems, *Chaos, Solitons & Fractals*, **26** (2005), 835–843.

J. Kavikumar received his MSc and Ph.D from Annamalai University, India. Since 2006 he has been at Universiti Tun Hussein Onn Malaysia. He has published many papers in national and international journals. His research interests center on the Fuzzy algebra, Fuzzy functional analysis and Fuzzy numerical analysis.

M. Kumudthaa is working as a Mathematics Teacher at SMK Seri Tasik, Kuala Lumpur. Her research interest is focus on Differential Equation and Fuzzy Mathematics.

Kuru Ratnavelu is working as a Professor of Mathematics at Institute of Mathematical Science, Universiti Malaya. He has published many research papers in international and national journals.

N. Kumaresan recieved his Ph.D from Anna University, India. He is working as a Senior Lecturer in Institute of Mathematical Science at Universiti Malaysia. His research interest is focus on Control theory, Genetic Algorithm and Fuzzy Mathematics. He has published many research papers in international and national journals.