EMPLOYING DIFFERENT TYPES OF TAYLOR-LIKE METHODS FOR IMAGE PROCESSING AND INITIAL VALUE PROBLEMS

SUKUMAR SENTHILKUMAR

ABSTRACT

Many articles have addressed different methods to solve numerical problems but in this paper, Sine-Cosine-Taylor like, Cosine Taylor like and Explicit Taylor like methods are used which is utmost importance to solve image processing and initial value problems. It is pertinent to point out that, the employed methods are an elegant coupling of Taylor's series and the results particularly disclose the effectiveness of adapted approaches to solve the considered problems. Simulation outputs and numerical results explicitly reveal the complete reliability of adapted algorithms.

Keywords: Sine-Cosine-Taylor like Method; Cosine Taylor like Method; Explicit Taylor like Method; Fuzzy Cellular Neural Networks; Image Processing; Initial Value Problems; Simulation.

INTRODUCTION

Recently, there has been much interest in designing novel numerical methods and algorithms for solving ordinary differential equations because they are employed for the description of many real-life phenomena in various fields, including biology and physics, population dynamics, economics and finance, engineering and physical disciplines etc. Moreover, because of the difficulty in obtaining true solutions to such equations, numerical methods are required and essential. It is noticed that many problems in science and engineering including system of equations can be formulated into ordinary differential equations by satisfying certain given conditions or constraints. Also, it is known that, analytical methods are applicable only to a particular class of differential equations. Moreover, differential equations (DEs) appearing in the fields of science and engineering are owing to physical and natural phenomena. DEs do not belong to any of the particular class of differential equations and therefore, cannot have closed or finite form of solution. Computations of approximate values are known as numerical solution to the differential equations. The difference between computed value y_i and true value $f(x_i)$ say, ε_i, is termed as truncation error at $x = x_i$. Also, note that there are many numerical algorithms for obtaining approximate solution to ODEs with initial conditions. Moreover, the fundamental mathematical notion to successfully execute ordinary differential equation is to replace differentiation by differencing to obtain better numerical solutions. It is well known that computer cannot differentiate but it can easily perform a difference operation. Hence, it is clear that differentiation is a continuous process and differencing is a discrete process (Press et al., 1986; Butcher, 2003). Still researchers employ Runge-Kutta (RK) techniques because of its accuracy and efficiency for better computational purpose (Butcher, 1987) therefore; many real-time problems can be solved easily to obtain required solution. Particularly RK algorithms are used to solve differential equations efficiently that are equivalent to approximate the exact solutions by matching *'n'* terms of the Taylor series expansion. Different higher order RK methods have also been introduced by various researchers (Butcher, 2003) to yield approximate solutions equivalent to true solutions. In addition, it is well known that, many authors have further developed and implemented Taylor algorithm for solving various numerical problems in real time, see in detail (Gibbons, 1960; Moore, 1979; Jorba & Zou, 2004; Miletics, Moln'arka

& Taylor, 2002, Ponalagusamy & Senthilkumar, 2009, 2011). The fundamental notion of these developments is due to recursive calculation of the coefficients of Taylor series. Sine-Cosine-Taylor-Like, Cosine Taylor Like and Explicit Taylor like technique for solving stiff ordinary differential equations are developed by Rokiah & Nazeeruddin (2004, 2005, 2010) and proved their results are better than other techniques by illustrating some examples. It is pertinent to note that explicit Sine-Cosine-Taylor-Like technique is a formulation of the combination of polynomial and exponential function. This technique requires extra work to evaluate a number of differentiations of the function involved. The result shows smaller errors when compared to the results from explicit classical fourth-order Runge-Kutta (RK4) explicit technique is of order-6. The rest of the paper is organized as follows. A short note on different types of Taylor-like methods is given in section 2. For solving different image processing problems, various types of Taylor like methods are employed under FCNN environment and the simulation results are depicted in section 3. Similarly, some numerical problems are solved and the outputs are reported in section 4. Conclusion is given in section 5.

A BRIEF NOTE ON DIFFERENT TYPES OF TAYLOR-LIKE METHODS

SINE-COSINE-TAYLOR-LIKE TECHNIQUE

Rokiah & Nazeeruddin (2005) discussed about explicit one step technique by the composition of a polynomial and exponential function which is given by,

$$y_{n+1} = y_n + h(f_n + h(\frac{f_n^1}{2} + h(\frac{f_n^2}{6} + h(\frac{f_n^3}{24} + h\frac{f_n^4}{120})))) + \frac{f_n^5(\sin(z_n h) + \cos z_n h))}{z_n^6}$$
$$(\exp z_n h) - 1 - hz_n(1 + hz_n(\frac{1}{2} + hz_n(\frac{1}{6} + hz_n(\frac{1}{24} + \frac{z_n h}{120}))))) \tag{1}$$

where h is a step-length.

COSINE-TAYLOR-LIKE METHOD

Rokiah & Nazeeruddin (2010) demonstrated the explicit one step Cosine technique by the composition of a polynomial and exponential function is given below.

$$y_{n+1} = y_n + h(f_n + h(\frac{f_n^1}{2} + h(\frac{f_n^2}{6} + h(\frac{f_n^3}{24} + h\frac{f_n^4}{120})))) + \frac{f_n^5(\cos(z_n h)}{z_n^6}$$
$$(\exp z_n h) - 1 - hz_n(1 + hz_n(\frac{1}{2} + hz_n(\frac{1}{6} + hz_n(\frac{1}{24} + \frac{z_n h}{120}))))) \tag{2}$$

where h is a step-size.

EXPLICIT TAYLOR-LIKE FORMULA

Rokiah & Nazeeruddin (2004) illustrated about the explicit Taylor like one step Taylor-L7 technique for solving stiff differential equations and illustrated about the advantage of using explicit method which is much cheaper than the implicit techniques. Furthermore, they pointed out that explicit method of order-7 is of A-stable and L-stable.

$$y_{n+1} = y_n + h(f_n + h(\frac{f_n^1}{2} + h(\frac{f_n^2}{6} + h(\frac{f_n^3}{24} + h\frac{f_n^4}{120})))) + \frac{f_n^5}{z_n^6}$$

$$(\exp z_n h) - 1 - hz_n(1 + hz_n(\frac{1}{2} + hz_n(\frac{1}{6} + hz_n(\frac{1}{24} + \frac{z_n h}{120})))))$$

(3)

where h = Δt which represents the temporal time step.

Some of the advantages of traditional Taylor's series method are as follows.

  1. It is one step method and explicit in nature.
  2. It can be high order.
  3. Easy to illustrate that global error is the same order as local truncation error.

One of the disadvantage of Taylor's method is explicit form of derivatives of f is needed.

ADAPTING VARIOUS TYPES OF TAYLOR LIKE-METHODS FOR IMAGE PROCESSING PROBLEM

The dynamics of a standard cellular neural network with a neighborhood of radius r are governed by a system of n = MN differential equations (Angela Slavova & Valeri Mladenov, 2004; Chi-Chien & De Gyvez, 1994; Chua & Yang, 1988a; Chua & Yang, 1988b; Chua & Roska, 1992; Roska, 2000; Roska & K'ek, 1994, Roska et al. 1994; Nossek et al. 1992). CNN is a dynamic nonlinear system defined by coupling identical simple dynamical systems called cells located within a prescribed sphere of influence, such as nearest neighbors. Fuzzy cellular neural network (FCNN) is a generalization of cellular neural networks (CNNs) by using fuzzy operations in the synaptic law computation, which allows us to combine the low level information processing capability of CNNs with the high level information processing capability, such as image understanding of fuzzy systems (Yang & Yang, 1996, Yang & Yang, 1997a; Yang & Yang, 1997b; Yang & Yang, 1997c; Itoh & Chua, 2003).

The recasted state equation of cell C_{ij} is defined by the following,

$$c\frac{dx_{ij}(t)}{dt} = \frac{-1}{R_x}x_{ij}(t) + \sum_{c(k,l)\in N_r(i,j)} A(i,j;k,l)y_{kl}(t) +$$

$$\sum_{c(k,l)\in N_r(i,j)} B(i,j;k,l)u_{kl}(t)$$

$$+I_{ij} + \underset{c(k,l)\in N_r(i,j)}{\tilde{\wedge}}(A_{f\min}(i,j;k,l)y_{kl}(t)) +$$

$$\underset{c(k,l)\in N_r(i,j)}{\tilde{\vee}}(A_{f\max}(i,j;k,l)y_{kl}(t)) +$$

$$+\underset{c(k,l)\in N_r(i,j)}{\tilde{\wedge}}(B_{f\min}(i,j;k,l)u_{kl}(t)) +$$

$$\underset{c(k,l)\in N_r(i,j)}{\tilde{\vee}}(B_{f\max}(i,j;k,l)u_{kl}(t)) +$$

(4)

where $A_{f\min}, A_{f\max}, B_{f\min}, B_{f\max}$ is feedback MIN, feedback MAX, feedforward MIN, feedforward MAX templates, respectively. The input equation of C_{ij} is given by

$$u_{ij} = E_{ij} \geq 0, 1 \leq i \leq M;\ 1 \leq j \leq N.$$

(5)

The output equation of C_{ij} is given by

$$y_{ij} = f(x_{ij}) = \frac{1}{2}(|x_{ij} + 1| - |x_{ij} - 1|),\ \text{ for } 1 \leq i \leq M;\ 1 \leq j \leq N.$$

(6)

SIMULATION RESULTS FOR IMAGE PROCESSING PROBLEMS

Digital image processing algorithms are used to perform different image processing operations on digital images Gonzalez, Woods & Eddin, (2009). A detailed discussion on single layer or raster scheme and time-multiplexing approach for edge detection using cellular neural network paradigm by new fourth order four stage algorithms is given by Senthilkumar (2009). A systematic design methodology for finding CNN parameters with prescribed functions are proposed by Itoh and Chua (2003), but it is interest to note that, in this paper a different attempt has been carried out and employed templates under fuzzy neural network paradigm. A detailed discussion about theory and implementation of hole filling using cellular neural network and connected component detector by improved cellular neural network can be seen in (Senthilkumar & Abdul Rahni, 2011; Senthilkumar & Abdul Rahni, 2012; Betta et al., 1993; Chum-Li et al. 1999; Chua & Thiran, 1991; Matsumoto, Chua & Furukawa, 1990).

HOLE FILLING TASK PRESCRIPTION

The template fills all holes (see fig.1) in a given image.

$$\text{Template Set: } A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0 \end{bmatrix}; I = -1 \tag{7}$$

Global Task Prescription: Fill the interior of all closed contours in a binary image P1. The input image is given by P1, and the initial state is given by $x_{ij}(0) = 1$. The boundaries are constrained to -1.

Local Task Prescription:

1. Let us assume that the input u_{ij} of cell C_{ij} is coded in black. Then, the output y_{ij} is printed in black.
2. Let us consider that the input u_{ij} is coded in white and all neighbors y_{kl} are colored in black. Then, the output y_{ij} is printed in black.
3. Furthermore, assume that the input u_{ij} is coded in white and at least one neighbor y_{kl} is colored in white. Then, the output y_{ij} is printed in white.
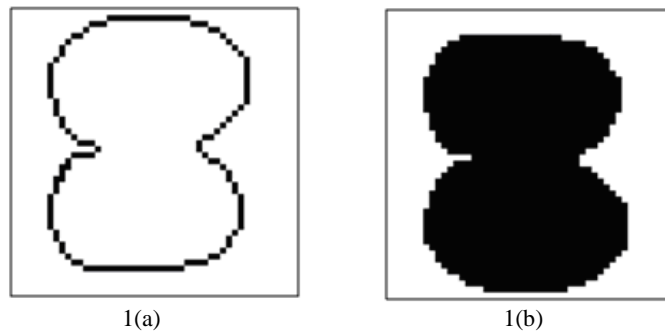

| 1(a) | 1(b) |

FIGURE. 1(a). Input Image (Before Hole Filling); 1(b). Output Image (After Hole Filling).

HALF TONING TASK PRESCRIPTION

Conversion of a grayscale image into a binary image is called half-toning. The following template performs half-toning (see fig.2) of a given image. Half-toning is one of the main image compression techniques employed for transmitting graphics over low-bandwidth channels. Particularly, it converts a gray-scale image into a binary image in such a way that the average gray level for corresponding regions of both images is same. Owing to the averaging property

of human visual system, these images will appear to be the same when observed from an appropriate distance.

$$\text{Template Set: } A = \begin{matrix} -0.07 & -0.1 & -0.07 \\ -0.1 & 3.68 & -0.1 \\ -0.07 & -0.1 & -0.07 \end{matrix} ; \ B = \begin{matrix} 0.07 & 0.1 & 0.07 \\ 0.1 & 0.32 & 0.1 \\ 0.07 & 0.1 & 0.07 \end{matrix} ; \ I = 0 \qquad (8)$$

Global Task Prescrption: Transforming a given gray-scale image P1 into a half-tone binary image. The CNN input image and the initial state is given by P1, and the boundaries are constrained to be -1.

Local Task Prescription: (a) Suppose, if the averaged value of outputs are greater (respectively smaller) than the average value of the inputs, then the output y_{ij} of cell C_{ij} is printed in white (black).

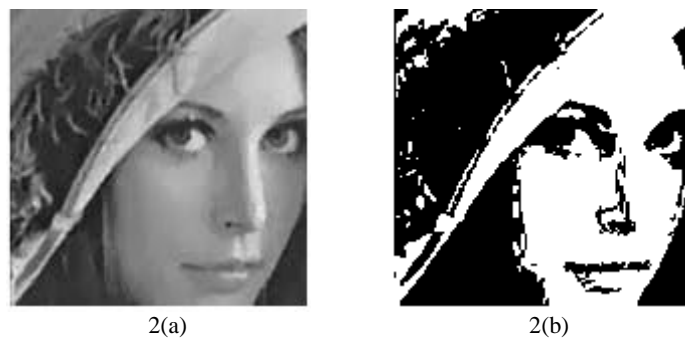2(a)   2(b)

FIGURE. 2(a). Input Image (Before half-toning); 2(b). Output Image (After half-toning).

SOLVING NUMERICAL PROBLEMS BY VARIOUS TYPES OF TAYLOR-LIKE METHODS

In case of solving systems of ordinary differential equations, through Taylor series method is one of the conventional way but analytic-numeric algorithms for calculating higher order derivatives formally is an over-elaborate task, and therefore, current Taylor algorithm is not applied frequently. Moreover, other reason is that only the explicit version of this algorithm is known. In other words, from the application point of view, the traditional Taylor series method has a major disadvantage. i.e. The method requires evaluation of partial derivatives of higher orders manually. This is not possible in any practical applications. Hence, end users need to develop methods which do not require evaluations and computation of higher order derivatives. The most important class of methods in this direction is Runge-Kutta methods. However, all these methods should compare with the Taylor series method when they are expanded about a point $t = t_j$.

It is observed from literature study that, modern numerical methods and its corresponding algorithms to obtain solution of ordinary differential equations are also based on Taylor series. Note that, each algorithm, such as the Runge-Kutta or the multistep methods are constructed so that they give an expression depending on a parameter called step-size (h) as an approximate solution and the first terms of Taylor series of this expression must be identical with the terms of the Taylor series of the exact solution. Also, these can be called as consistency and order conditions for the algorithms. These expressions potentially can be evaluated at any value of the parameter (h), but in real time the evaluation can be realized only at grid points. Such algorithms give value of the approximate solution at grid points and main cost is the number of function evaluation. (Some algorithm may contain more than one parameter.)

Moreover, these algorithms differ from others in its order, stability properties and its cost of realization can be seen from (Harier, Norsett & Wanner, 1987; Harier, Norsett & Wanner, 1991; Miletics & Moln'arka, 2004).

ORDINARY DIFFERENTIAL EQUATION PROBLEM AND RESULTS

Let us consider a simple problem $y' = y - t^2 + 1; y(0) = 0.2$ to compute approximate solution using Sine-Cosine Taylor Like method. The corresponding exact solution is given by $y = t^2 + +2t + 1 - \frac{1}{2}e^t; 0 \le t \le 2$.

TABLE 1. Comparison between Exact and Sine-Cosine Taylor-Like Method

Time (t)	Exact Solution	Sine-Cosine Taylor-Like
0.00	0.5	0.5
0.20	0.8292	0.82929333333333
0.40	1.2140	1.21407621066666
0.60	1.6489	1.64892201704160
0.80	2.1272	2.12720268494794
1.00	2.6408	2.64082269272875

NUMERICAL PROBLEM AND OUTPUTS

Let us consider a simple problem $y' = (x - y)/2, y(0) = 1$ to compute approximate solution using explicit Cosine Taylor Like method.

TABLE 2. Comparison between Exact and Cosine Taylor-Like method

Time (t)	Exact Solution	Cosine Taylor-Like method
0.00	1.0	1.0
0.50	0.8364	0.83642578125
1.00	0.8196	0.8196284770965576
1.50	0.9171	0.9171422953950241
2.00	1.1036	1.1036825982202458
2.50	1.3595	1.3595574922662559
3.00	1.6694	1.6694307617991593

INITIAL VALUE PROBLEM AND RESULTS

Let us consider a simple problem $y' = Sin(xy), y(0) = \pi$ to compute approximate solution using explicit Taylor Like method.

TABLE 3. Comparison between Exact and Explicit Taylor-Like Method

Time (t)	Exact Solution	Explicit Taylor- Like Method
0.00	3.1416	3.1416
0.10	3.1572	3.1417
0.20	3.2029	3.1725
0.30	3.2750	3.2318
0.40	3.3663	3.3142
0.50	3.4656	3.4112
0.60	3.5585	3.5103
0.70	3.6299	3.5963
0.80	3.6688	3.6548
0.90	3.6708	3.6764
1.00	3.6383	3.6598

IVP PROBLEM AND SOLUTIONS

Let us consider a simple problem $y' = 5y - 3$, $y(0) = 0.8$ to compute approximate solution using Classical Runge-Kutta Technique and the true solution is given by $y = \frac{1}{5}\exp(5t) + \frac{3}{5}$.

TABLE 4. Comparison between Exact and Standard Runge-Kutta Technique

Time (t)	Exact Solution	Standard Runge-Kutta Technique
0.0000	0.8000	0.8000
0.1000	0.9297	0.9297
0.2000	1.1437	1.1435
0.3000	1.4936	1.4959
0.4000	2.0778	2.0768
0.5000	3.0365	3.0344
0.6000	4.6171	4.6130
0.7000	7.2231	7.2151
0.8000	11.5196	11.5046
0.9000	18.6034	18.5756
1.0000	30.2826	30.2315

It is to be observed that, up to some extend the approximated solution matches with the true solution (see table 1, 2, 3 & 4). If the step-size (*h*), is small one can obtain almost equivalent to exact solution but takes more time to complete its task.

CONCLUSION

In this paper, Sine-Cosine-Taylor like, Cosine Taylor like and Explicit Taylor like methods are successfully employed to examine image processing and numerical problems.

In particular, three different Taylor like techniques are typically a composition of a polynomial and exponential function which yields stable and almost accurate approximate result which is closer to true solution. Under fuzzy cellular non-linear network simulation, the image processing problems such as hole filing and half toning have been solved initially by adapting various types of explicit Taylor's like methods but to make the numerical computation successful, on the other hand specifically initial value problems are solved directly with the help of different types of explicit Taylor's like techniques. To illustrate the efficiency of the employed methods, simple numerical problems are solved and results shown which are computationally competent.

ACKNOWLEDGEMENT

REFERENCES

Angela Slavova, Valeri Mladenov, 2004. *Cellular Neural Networks: Theory and Applications.* New York: Nova Publishers.

Betta, D.G.F., Graffi, S., Kovacs, M., Masetti, G.1993.CMOS Implementation of an Analogy Programmed Cellular Neural Network. *IEEE Transactions on Circuits and Systems-Part–II,* Vol.40 (3): 206–214.

Butcher, J.C. 1987. *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. U.K: John Wiley & Sons.

Butcher, J.C. 2003. *The Numerical Analysis of Ordinary Differential Equations*. U.K, : John Wiley & Sons.

Chi-Chien, L., De Gyvez, J.P. 1994. Single-Layer CNN Simulator. *IEEE International Symposium on Circuits and Systems*, Vol. 6: 217–220.

Chua, L.O., Yang, L. 1988a. Cellular Neural Networks: Theory. *IEEE Transactions on Circuits and Systems*, Vol.35: 1257–1272.

Chua, L.O., Yang, L. 1988b. Cellular Neural Networks: Applications. *IEEE Transactions on Circuits and Systems,* Vol.35: 1273–1290.

Chua, L.O. and Roska, T. 1992. The CNN Universal Machine Part 1: The Architecture. In *International Workshop on Cellular Neural Networks and their Applications (CNNA)*, pp.1–10.

Chua, L.O. and Thiran, P. 1991. An Analytic Method for Designing Simple Cellular Neural Networks, *IEEE Transactions on Circuits and Systems-I*, Vol. 38(11): 1332–1341.

Chum-Li, Y., Jin-Liang, W., Hai, L, and Wai-Kai, C. 1999. Brief Communication, The Cloning Template Design of a Cellular Neural Network. *Journal of the Franklin Institute,* Vol.336: 903–909.

Harier, E., Norsett, S.P. and Wanner,G. 1987. *Solving Ordinary Differential Equations, I.Nonstiff Problems*. Berlin:Springer Verlag.

Harier, E., Norsett, S.P. and Wanner, G. 1991. *Solving Ordinary Differential Equations, II. Stiff and Differential-Algebraic Problems.* Berlin:Springer Verlag..

Jorba, A. and Zou, M. 2004. A Software Package for the Numerical Integration of ODE by Means of High-Order Taylor Methods, *Experimental Mathematics,* Vol.14 (1): 99–117. http://www.maia.ub.es/˜angel/ [27 January 2004].

Gibbons, A. 1960. A Program for the Automatic Integration of Differential Equations using the Method of Taylor Series. *Computer Journal*, Vol. 3: 108–111.

Gonzalez, R.C., Woods, R.E. and Eddin, S.L. 2009. *Digital Image Processing using MATLAB*. Upper Saddle River, New Jersey: Pearson Education Asia.

Itoh, M. and Chua, L.O. 2003. Designing CNN Genes, *International Journal of Bifurcation and Chaos*, Vol. 13(10): 2739–2824.

Matsumoto, T., Chua, L.O. and Furukawa, R. 1990. CNN Cloning Template: Holefiller. *IEEE Transactions on Circuits and Systems,* Vol. 37(5): 635–638.

Moore, R.E. 1979. Methods and Applications of Interval Analysis. *SIAM studies in Applied Mathematics*.

Miletics, P.E. and Moln´arka, G. Taylor. 2002. Series Methods with Numerical Derivatives for Initial Value Problems. In *Computational and Mathematical Methods on Science and Engineering*, Vol. I. eds.: J. Vigo Aguiar and B. A. Wade. Proc. of CMMSE-, Alicante, Spain, pp.258-270.

Miletics, E. and Moln'arka, G. 2004. Taylor Series Method with Numerical Derivatives for Numerical Solution of ODE Initial Value Problems, *Journal of Computational Methods in Sciences and Engineering - Computational and Mathematical Methods for Science and Engineering Conference 2002 - CMMSE-2002*, Volume 4(1-2):105-114.

Nossek, J.A., Seiler, G., Roska, T., Chua, L.O. 1992. Cellular Neural Networks: Theory and Circuit Design. *International Journal of Circuit Theory and Applications*, Vol.20: 533–553.

Ponalagusamy, R. and Senthilkumar, S. 2009. A New RK-Embedded Fourth Order with Four Stages Technique to Study Raster CNN Simulation, *International Journal of Automation and Computing*, Vol.6 (3), pp.285-294.

Ponalagusamy, R. and Senthilkumar, S. 2011. Investigation on Time-Multiplexing Cellular Neural Network Simulation by RKAHeM(4,4) Technique, *International Journal of Advanced Intelligence Paradigms*, Vol.3, pp.43-66.

Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. 1986. *Numerical Recipes. The Art of Scientific Computing*. New York: Cambridge University Press.

Rokiah Ahmed and Nazeeruddin Yaacob. 2004. Explicit Taylor-Like Method for Solving Stiff Differential Equations. *Technical Report*. LT/M Bil.3.

Rokiah Ahmed and Nazeeruddin Yaacob. 2005. Sin-Cos-Taylor-Like Method for Solving Stiff Ordinary Differential Equations. *Journal of Fundamental Sciences*, Vol.1, pp.34–43.

Rokiah Ahmed and Nazeeruddin, Yaacob. 2010. One Step Cosine Taylor Like Method for Solving Stiff Equations. *Jurnal Teknologi*, Vol.53: 41–47.

Roska, T. 2000. CNN Software Library, Hungarian Academy of Sciences, Analogical and Neural Computing Laboratory. Available: http:// lab.analogic.sztaki.hu/Candy /csl.html, 1.1.

Roska, T. and K´ek, L.1994. *Analogic CNN program library* Version 6.1 Budapest, Hungary.

Roska et al.: C 1994. *CNNM Users Guide*, Version 5.3x, Budapest, Hungary.

Senthilkumar, S. 2009. New Embedded Runge-Kutta Fourth order Four Stage Algorithms for Raster and Time-Multiplexing Cellular Neural Networks Simulation, Ph.D. Thesis, Department of Mathematics, National Institute of Technology [REC], Tiruchirappalli, Tamilnadu, India.

Senthilkumar, S. and Abd Rahni Mt. Piah. 2011. Sine-Cosine-Taylor-Like Method for Hole-Filler CNN Simulation. *International Journal of Computer Science Issues*, Vol.8 (4): 117–122.

Senthilkumar, S. and Abd Rahni Mt. Piah. 2012. Sine-Cosine-Taylor-Like Technique for Connected Component Detector by ICNN Simulation. *International Journal of Image, Graphics and Signal Processing*, Vol.3: 28–34.

Yang, T. and. Yang, L.B. 1996. The Global Stability of Fuzzy Cellular Neural Networks. *IEEE, Transactions on Circuit and Systems-I*, Vol.43 (10): 880–883.

Yang, T. and Yang, L.B. 1997a. Fuzzy Cellular Neural Network: A New Paradigm for Image Processing. *International Journal of Circuit Theory and Applications*, Vol.25: 469–481.

Yang, T. and Yang, L.B. 1997b. Application of Fuzzy Cellular Neural Networks to Euclidean Distance Transformation. *IEEE Transactions on circuits and Systems I, CAS-44*, pp.242-246.

Yang, T. and Yang, L.B. 1997c. Application of Fuzzy Cellular Neural Network To Morphological Grey-Scale Reconstruction. *International Journal of Circuit Theory and Applications*, Vol.25: 153–165.

Yang, T., Yang, L.B., Wu, C.W. and Chua, L.O. 1996. Fuzzy Cellular Neural Networks: Theory. In *Proceedings of IEEE International Workshop on Cellular Neural Networks and Applications*, pp.181–186.

AUTHOR'S BIOGRAPHY

Sukumar Senthilkumar received his B.Sc in Mathematics from Madras University in 1994, M.Sc and M.Phil in Mathematics from Bharathidasan University in 1996 and 1999, PGDCA and PGDCH from Bharathidasan University in 1996 and 1997 respectively and M.Phil in Computer Science & Engineering from Bharathiar University in 2000. He has a research doctoral degree in the fields of Mathematics and Computer Applications from National Institute of Technology [An Institute of National Importance-formerly Regional Engineering College], Tiruchirappalli, Tamilnadu, India. He worked as a Post-Doctoral Fellow at the School of Mathematical Sciences, Universiti Sains Malaysia [An Apex University] in Pulau Pinang, Malaysia. Also, he worked as a Post-Doctoral Researcher at the Center for Advanced Image and Information Technology, School of Electronics and Information Engineering, Division of Computer Science and Engineering, Chon Buk National University, Jeonju, Chon Buk, South Korea. He was a Lecturer / Assistant professor in the Department of Computer Science at Asan Memorial College of Arts and Science, Chennai, Tamilnadu, India. At present, he is working as an Associate Professor in the School of Computing Science and Engineering at Vellore Institute of Technology-University, Vellore, Tamilnadu, India. He has published many good scientific research papers in peer reviewed international journals with high impact factor. He is also an editor, associate editor, editorial board member, advisory board member, reviewer and referee for many scientific international journals. His current research interests include cellular neural networks, digital image processing, numerical analysis & methods, simulation & computing and other related areas including astrophysics and oceanography.

Sukumar Senthilkumar
School of Computing Science and Engineering,
Vellore Institute of Technology-University,
Vellore-632014, Tamilnadu, India.
senthilkumar@vit.ac.in; ssenthilkumar1974@yahoo.co.in