Sine-Cosine-Taylor-Like Method for Hole-Filler ICNN Simulation

S. Senthilkumar* and Abd Rahni Mt Piah

Universiti Sains Malaysia
School of Mathematical Sciences
11800 USM Pulau Pinang, MALAYSIA

****Corresponding author***

Abstract

Sine-Cosine-Taylor-Like method is employed to improve the performance of image or handwritten character recognition under improved cellular non-linear network environment. The ultimate aim of this paper is focused on developing an efficient design strategy for simulating hole filler under ICNN arrays with a set of inequalities satisfying its output characteristics by considering the parameter range.

Keywords: *Improved Cellular Non-linear Network, Sine-Cosine-Taylor-Like Method, Hole-Filler Template, Edge Detection, Ordinary Differential Equations.*

1. Introduction

Cellular non-linear networks (CNNs), proposed by Chua and Yang [1, 2] are essentially non-linear analog electric circuits, locally interconnected for distributed computation. It is understood that the characteristics of cellular neural networks (CNNs) are analog, time-continuous, non-linear dynamical systems and formally belong to the class of recurrent neural networks. CNNs have been proposed by Chua and Yang [l, 2], and they have found that CNN has many important applications in signal and real-time image processing [12]. Roska et al. [3] have presented the first widely used simulation system which allows simulation of a large class of CNN and is especially suited for image processing applications including signal processing, pattern recognition and solving ordinary and partial differential equations etc.

Ahmad and Yaacob [4-6] introduced sin-cos-Taylor-like method for solving stiff ordinary differential equations and proved the results are better than other methods. Runge-Kutta (RK) methods have become very popular and efficient tools for computational purpose [19-22] and many real-time problems are solved. Particularly Runge-Kutta 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. The RK-Butcher algorithm has been introduced by Bader [7, 8] for finding truncation error estimates, intrinsic accuracies and early detection of stiffness in coupled differential equations that arises in theoretical chemistry problems. Oliveira [9] introduced popular RK-Gill algorithm for evaluation of effectiveness factor of immobilized enzymes. Ponalagusamy and Senthilkumar [10] discussed about the comparison of RK-fourth orders of variety of means and embedded means on multilayer raster CNN simulation.

In this article, the hole filing scheme under ICNN paradigm with sine-cosine-Taylor-like method is carried out and compared with explicit Euler, RK-Gill, RK-classical fourth order. It is significant to note that the explicit sine-cosine-Taylor-like method for solving hole filing CNN simulation is a formulation of the combination of a polynomial and the exponential function. This method requires extra work to evaluate a number of differentiations of the function involved. The result shows smaller errors when compared to the results from the explicit classical fourth-order Runge-Kutta (RK4) is of order-6 [4-6].

Using the existing RK-Butcher fifth order method, the hole filing has been studied via CNN simulation by Murugesh and Badri [23] and the same has been studied by Murugesan and Elango [24] using RK fourth order method. Dalla Betta et al. [25] implemented CMOS implementation of an analogy programmed cellular neural network. Qiang Feng et al. [29] proposed new automatic nucleated cell method with improved cellular networks. Further, they proved that the running speed is comparatively high due to the easy hardware implementation and high speed of CNN.

2. Brief Theoretical Study: Structure and Functions of Improved Cellular Neural Network

The architecture of standard $M \times N$ CNN [1,2] is composed of cells $C(i, j)$'s where $1 \le i \le M$; $1 \le j \le N$. $M \times N$ can be understood as the dimension of the digital image P to be processed. The dynamics of each cell is given via the equations as below

$$c\frac{dx_{ij}(t)}{dt} = \frac{-1}{R}x_{ij} + \sum_{k,l\in S_{i,j}(r)} a_{k,l}y_{i+k,j+l} + \sum_{k,l\in S_{i,j}(r)} b_{k,l}u_{i+k,j+l} + z_{i,j},$$

$$1 \le i \le M;\ 1 \le j \le N.$$

$$= \frac{-1}{R}x_{ij} + \sum_{k=-r}^{r}\sum_{l=-r}^{r} a_{k,l}y_{i+k,j+l} + \sum_{k=-r}^{r}\sum_{l=-r}^{r} b_{k,l}u_{i+k,j+l} + z_{i,j} \quad (1)$$

where $x_{i,j}$, $y_{i,j}$, $u_{i,j}$ and $z_{i,j}$ represent state variable, output variable, input variable and threshold variable respectively; $S_{i,j}(r)$ is the sphere of influence with radius r; The $a_{k,l}$'s and $b_{k,l}$'s are said to be the elements of the A template (feedback template) and B template (feed-forward template), respectively. The output $y_{i,j}$ is the piecewise linear function. C and R_x determine the recall time.

The constraint / conditions are given by

$$|x_{i,j}(0)| \le 1, |u_{i,j}| \le 1 \text{ where } C > 0 \quad (2)$$

and

$R_x > 0$

Take C and R_x at 1. The output function is defined as below:

$$y_{i,j} = f(X_{i,j}) = \begin{cases} 1 & x_{i,j} \ge T^* \\ x_{i,j} & |x_{i,j}| < T^* \\ -1 & x_{i,j} \le -T^* \end{cases} \quad (3)$$

$$T^* = \sum_{i=1}^{M}\sum_{j=1}^{N} P(i,j)/(MN) \quad (4)$$

where $P(i,j)$ denotes the unitary grayscale of the pixel (i,j) in the input image P. $M \times N$ can be understood as the dimension of the digital image P. It is obvious that the range of T^* is (0,1).

3. Hole Filler with ICNN Paradigm

In a bipolar image all the holes are filled and remains unaltered outside the holes in case of hole filing ICNN simulation [26-28]. Allow $R_x = 1, C = 1$ and take +1 represent for the black pixel and –1 for the white pixel. If the bipolar image is input with $U = \{u_{ij}\}$ into CNN and the images having holes enclosed by the black pixels then initial state values are set as $x_{ij}(0) = 1$. The output values are obtained as $y_{ij}(0) = 1, 1 \le i \le M, 1 \le j \le N$ from equation (2).

Consider the templates A, B and the independent current source I are

$$A = \begin{bmatrix} 0 & a & 0 \\ a & b & a \\ 0 & a & 0 \end{bmatrix}, \qquad a > 0, b > 0$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad I = -1 \quad (5)$$

where the template parameters a and b are to be determined. In order to make the outer edge cells become the inner ones, normally auxiliary cells are added along the outer boundary of the image, and their state values are set to be zeros by circuit realization, resulting in the zero output values. The state equation (1) can be rewritten as

$$\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + \sum_{c(k,l)\in N(i,j)} A(i,j;k,l)y_{kl}(t) + 4u_{ij}(t) - I \quad (6)$$

For instance, here the cells $C(i+1,j)$, $C(i-1,j)$, $C(i,j+1)$ and $C(i,j-1)$ are the non-diagonal cells. Designing of hole-filler template and its various sub-problems are discussed using

CNN simulations [26-28]. Consider the same two problems by Yin et al. [26].

Problem 1: The input value $u_{ij} = 1$ for cell C(*i*,*j*), signaling the black pixel. Because the initial state value of the cell C(*i*,*j*), has been set to 1, $x_{ij}(t) = 1$, and from equation (3) its initial output value is also $y_{ij}(t) = 1$. According to the hole-filler demands, its eventual output should be $y_{ij}(\infty) = 1$. To attain this result, set

$$\frac{dx_{ij}(t)}{dt} \geq 0 \qquad (7)$$

Substituting this input $u_{ij} = 1$ and equation (6) into equation (7), we obtain

$$\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + a[y_{(i-1)j}(t) + y_{(i+1)j}(t) + y_{i(j-1)}(t) + y_{i(j+1)}(t)] + by_{ij}(t) + 3 \qquad (8)$$

Combining equations (7) and (8) and considering the minimum value of $x_{ij}(t) = 1$ this case yields

$$a[y_{(i-1)j}(t) + y_{(i+1)j}(t) + y_{i(j-1)}(t) + y_{i(j+1)}(t)] + by_{ij}(t) + 2 \geq 0 \qquad (9)$$

Problem 2: The input value of cell C(*i*,*j*) is $u_{ij} = 1$, signaling the white pixel. Substituting this input value into equation (7) gives

$$\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + a[y_{(i-1)j}(t) + y_{(i+1)j}(t) + y_{i(j-1)}(t) + y_{i(j+1)}(t)] + by_{ij}(t) - 5 \qquad (10)$$

4. Numerical Integration Methods

The ICNN is described by a system of nonlinear differential equations. Therefore, it is necessary to discretize the differential equation for performing behavioral simulation. For computational purposes, a normalized time differential equation describing CNN is used by Nossek et al [11]:

$$f'(x(n\tau)) = \frac{dx_{ij}(n\tau)}{dt} = -x_{ij}(n\tau) + \sum_{k=-r}^{r}\sum_{l=-r}^{r} a_{k,l}y_{i+k,j+1} + \sum_{k=-r}^{r}\sum_{l=-r}^{r} b_{k,l}u_{i+k,j+1} + z_{i,j}, \qquad (11)$$

$1 \leq i \leq M;\ 1 \leq j \leq N$

$$y_{ij}(n\tau) = \frac{1}{2}\left[\left|x_{ij}(n\tau)+1\right| - \left|x_{ij}(n\tau)-1\right|\right],$$

$1 \leq i \leq M;\ 1 \leq j \leq N$ (12)

where τ is the normalized time. For the purpose of solving the initial-value problem, well established single step methods of numerical integration methods are used [13].

These methods can be derived using the definition of the definite integral

$$x_{ij}((n+1)\tau) - x_{ij}(n\tau)) = \int_{\tau_n}^{\tau_{n+1}} f'(x(n\tau))d(n\tau) \qquad (13)$$

Explicit Euler's, the improved Euler predictor-corrector and the fourth-order (quartic) Runge-Kutta are the mostly widely used single step algorithm in the CNN behavioral raster simulation. These methods vary in the way they evaluate the integral presented in [7].

4.1 Explicit Euler's Method

Euler's method is the simplest of all algorithms for solving ordinary differential equations. It is an explicit formula which uses the Taylor-series expansion to calculate the approximation.

$$x_{ij}((n+1)\tau) = x_{ij}(n\tau) + \tau f'(x(n\tau)) \qquad (14)$$

4.2 RK-Gill Method

The RK-Gill method discussed by Oliveira [9] is an explicit method which requires the computation of four derivatives per time step. The increase of the state variable x^{ij} is stored in the constant k_1^{ij}. This result is used in the next iteration for evaluating k_2^{ij} and repeat the same process to obtain the values of k_3^{ij} and k_4^{ij}.

$$k_1^{ij} = f'(x_{ij}(n\tau)),$$
$$k_2^{ij} = f'(x_{ij}(n\tau) + \frac{1}{2}k_1^{ij}),$$
$$k_3^{ij} = f'(x_{ij}(n\tau) + (\frac{1}{\sqrt{2}} - \frac{1}{2})k_1^{ij}) + (1 - \frac{1}{\sqrt{2}})k_2^{ij}), \qquad (15)$$
$$k_4^{ij} = f'(x_{ij}(n\tau) - \frac{1}{\sqrt{2}}k_2^{ij} + (1 + \frac{1}{\sqrt{2}})k_3^{ij}),$$

Therefore, the final integration is a weighted sum of the four calculated derivates per time step given by

$$x_{ij}((n+1)\tau) = x_{ij}(n\tau) + \frac{1}{6}[k_1^{ij} + (2-\sqrt{2})k_2^{ij} + (2+\sqrt{2})k_3^{ij} + k_4^{ij}] \quad (16)$$

4.3 Classical Runge-Kutta Method

The RK-classical fourth order method is an explicit method which requires the computation of four derivatives per time step. The increase of the state variable x^{ij} is stored in the constant k_1^{ij}. This result is used in the next iteration for evaluating k_2^{ij} and repeat the same process to obtain the values of k_3^{ij} and k_4^{ij}.

$$k_1^{ij} = \tau f'(x_{ij}(n\tau)),$$
$$k_2^{ij} = \tau f'(x_{ij}(n\tau) + \frac{1}{2}k_1^{ij}),$$
$$k_3^{ij} = \tau f'(x_{ij}(n\tau)) + \frac{1}{2}k_2^{ij},$$
$$k_4^{ij} = \tau f'(x_{ij}(n\tau)) + k_3^{ij}, \quad (17)$$

Therefore, the final integration is a weighted sum of the four calculated derivatives per time step which is given by

$$x_{ij}((n+1)\tau) = x_{ij}(n\tau) + [(k_1^{ij} + 2k_2^{ij} + 2k_3^{ij} + k_4^{ij})/6] \quad (18)$$

where *f*(.) is computed according to (2).

4.4 Sine-Cosine-Taylor-Like Method

Ahmad and Yaacob [4-6] discussed the explicit one step method by the composition of a polynomial and exponential function.

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

(19)

5. Simulation Results and Comparisons

The hole-filing simulated output presented is performed using a high power workstation, and the simulation time used for comparisons is the actual CPU time used. The settling time *Ts* is the time from start of computation until the last cell leaves the interval [-1.0, 1.0] which is based on a specified limit (e.g., |dx/dt|< 0.01). The computation time *Tc* is the time taken for settling the network and adjusting the cell for proper position once the network is settled. The simulation shows the desired output for every cell. Take +1 and -1 to indicate the black and white pixels, respectively. It is marked that the selected template parameters *a* and *b*, are restricted to the shaded area as shown in figure 2 for the simulation. The speed is one of the major concerns in the simulation. Hence, determination of the maximum step-size (Δt) that still yields convergence for a template can be helpful in speeding up the system. The speed-up can be achieved by selecting an appropriate step-size (Δt) for that particular template. Figure 3 shows the image before and after hole-filling by employing sine-cosine-Taylor-like method.

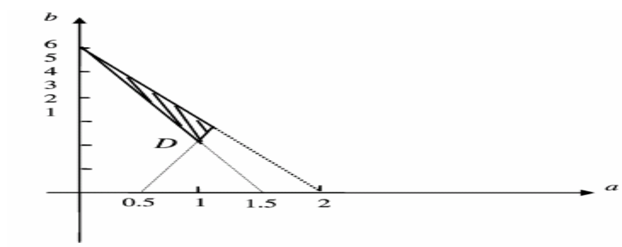

Fig. 2 Range of the template

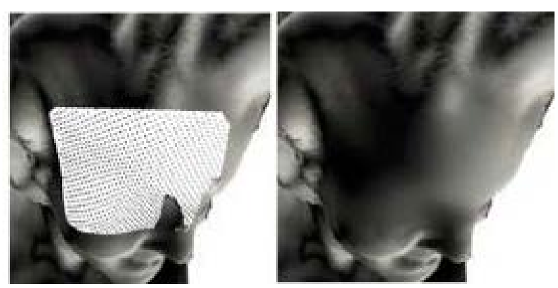
Fig.3 Image before and after hole-filling by employing sine-cosine-Taylor-like method

6. Conclusion

The importance of the hole filing CNN simulator is capable of performing hole-filing simulation for any kind as well as any size of input image or hand written character. It is a powerful tool for researchers to investigate the potential applications of CNN. It is of interest to mention that using the sine-cosine-Taylor-like method in the hole filing simulation shows better feasible and effective output. It is observed that the hole is filled and the outside image remains unaffected that is, the edges of the images are preserved and intact. The templates of the improved cellular neural network are not unique and this is important in its implementation. In many language scripts, numerals and in images etc, there are many holes and the CNN as described above can be used in addition to the connected component detector. It is noteworthy to mention that each pixel receives the information only from its immediate neighbors, while the result $u_{xij}(\infty)$ is completely global in nature.

Acknowledgments

The first author would like to extend his sincere gratitude to Universiti Sains Malaysia for supporting this work under its post-doctoral fellowship scheme. Much of this work was carried out during his stay at Universiti Sains Malaysia in 2011. He wishes to acknowledge Universiti Sains Malaysia's financial support.

References

[1] Chua, L.O., Yang, L., "Cellular neural networks: Theory", IEEE Transactions on Circuits and Systems, Vol.35,1988,pp. 1257 – 1272.

[2] Chua, L.O., Yang, L., "Cellular neural networks: Applications", IEEE Transactions on Circuits and Systems, Vol. 35, 1988,pp. 1273 – 1290.

[3] Roska et al., "CNNM Users Guide", Version 5.3x, Budapest 1994.

[4] R. Ahmad and N. Yaacob, "Sin-Cos-Taylor-like method for solving stiff ordinary differential equations", Journal of Fundamental Sciences, 2005 pp.34-43.

[5] R. Ahmad, N. Yaacob, "Explicit Taylor-like method for solving stiff differential equations", Technical Report. LT/M Bil. 3/2004.

[6] R. Ahmad, N. Yaacob, "Explicit Sine-Taylor-like method for solving stiff differential equations", Technical Report. LT/M Bil. 8/2004.

[7] Bader, M., "A comparative study of new truncation error estimates and intrinsic accuracies of some higher order Runge-Kutta algorithms", Computers & Chemistry, Vol. 11, 1987, pp. 121-124.

[8] Bader, M., "A new technique for the early detection of stiffness in coupled differential equations and application to standard Runge-Kutta algorithms", Theoretical Chemistry Accounts, Vol. 99, 1988, pp. 215-219.

[9] Oliveira, S.C., "Evaluation of effectiveness factor of immobilized enzymes using Runge-Kutta-Gill method: How to solve mathematical undetermination at particle center point?", Bio Process Engineering, Vol.20, 1999, pp. 185-187.

[10] R. Ponalagusamy and S. Senthilkumar, "A comparison of RK-fourth orders of variety of means and embedded means on multilayer raster CNN simulation", Journal of Theoretical and Applied Information Technology, 2007, pp.7-14.

[11] Nossek, J.A., Seiler, G., Roska, T., Chua, L.O., "Cellular neural networks: Theory and circuit design", Int. J. of Circuit Theory and Applications, Vol.20, 1992, pp. 533-553.

[12] L.O. Chua and T. Roska, "The CNN universal machine part 1: The architecture", in Int. Workshop on Cellular Neural Networks and their Applications (CNNA)", 1992, pp. 1-10.

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

[14] Gonzalez, R.C., Woods, R.E. and Eddin, S.L., "Digital Image Processing Using MATLAB", Pearson Education Asia, Upper Saddle River, N.J,2009.

[15] K.K. Lai and P.H.W. Leong,"Implementation of time-multiplexed CNN building block cell", IEEE. Proc. Of Microwave, 1996, pp. 80-85.

[16] K.K. Lai and P.H.W. Leong, "An area efficient implementation of a cellular neural network", IEEE, 1995, pp.51-54.

[17] T. Roska, "CNN software library", Hungarian Academy of Sciences, Analogical and Neural Computing Laboratory, 2000 [Online]. Available:http://lab.analogic.sztaki.hu/Candy/csl.html, 1.1

[18] Roska et al.,"CNNM Users Guide", Version 5.3x, Budapest,1994.

[19] J.C. Butcher, "On Runge processes of higher order", Journal of Australian Mathematical Society, Vol.4, 1964, p.179.

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

[21] J.C. Butcher,(1990), "On order reduction for Runge-Kutta methods applied to differential-algebraic systems and to stiff systems of ODEs", SIAM Journal of Numerical Analysis, Vol.27, pp. 447-456.

[22] Butcher, J.C., "The Numerical Analysis of Ordinary Differential Equations", John Wiley & Sons, U.K, 2003.

[23] V. Murugesh and Krishnan Badri, "An efficient numerical integration algorithm for cellular neural network based hole-filler template design", International Journal of Computers, Communications and Control, Vol.2, 2007, pp. 367-374.

[24] K. Murugesan and P. Elango, "CNN based hole filler template design using numerical integration technique", LNCS 4668, 2007, pp. 490-500.

[25] Dalla Betta, G.F., Graffi, S., Kovacs, M., Masetti, G, "CMOS implementation of an analogy programmed cellular neural network". IEEE Transactions on Circuits and Systems-Part–II Vol. 40(3), 1993, pp. 206–214.

[26] Chun-Li Yin, Jin-Liang Wan, Hai Lin, Wai-Kai Chen, "Brief communication, the cloning template design of a cellular neural network", Journal of the Franklin Institute Vol. 336, No. 199, pp.903-909.

[27] L. O. Chua and P. Thiran, "An analytic method for designing simple cellular neural networks", IEEE Transactions on Circuits and Systems-I, Vol. 38, No. 11, 1991, pp.1332-1341.

[28] Matsumoto, T., Chua, L.O. and Furukawa, R. "CNN cloning template: Hole filler", IEEE Transactions on Circuits and Systems, Vol. 37, Issue 5, 1990, pp. 635-638.

[29] Qiang Feng, Shenglin Yu and Huaiyin Wang, "A new automatic nucleated cell counting method with improved cellular neural networks", (ICNN), IEEE 2006 10th International

Workshop on Cellular Neural Networks and Their Applications, pp.1-4, 2006.

SUKUMAR SENTHILKUMAR was born in Neyveli Township, Cuddalore District, Tamilnadu, India on 18th July 1974. He received his B.Sc in Mathematics from Madras University in 1994, M.Sc in Mathematics from Bharathidasan University in 1996, M.Phil in Mathematics from Bharathidasan University in 1999 and M.Phil in Computer Science & Engineering from Bharathiar University in 2000. Also, he received PGDCA and PGDCH in Computer Science and Applications and Computer Hardware from Bharathidasan University in 1996 and 1997 respectively. He obtained a doctoral degree in the field of Mathematics and Computer Applications from National Institute of Technology [REC], Tiruchirappalli, Tamilnadu, India. Currently he is working as a post doctoral fellow at the School of Mathematical Sciences, Universiti Sains Malaysia in Pulau Pinang Malaysia. He was a lecturer / assistant professor in the Department of Computer Science at Asan Memorial College of Arts and Science, Chennai, Tamilnadu, India. He has published many good research papers in international conference proceedings and peer-reviewed / highly refereed international journals with high impact factor. He has made significant and outstanding contributions to various activities related to research work. He is also an associate editor, editorial board member, reviewer and referee for many scientific international journals. His current research interests include advanced cellular neural networks, advanced digital image processing, advanced numerical analysis and methods, advanced simulation and computing and other related areas.

ABD RAHNI MT PIAH was born in Baling, Kedah Malaysia on 8th May 1956. He received his B.A. (Cum Laude) in Mathematics from Knox College, Illinois, USA in 1979. He received his M.Sc in Mathematics from Universiti Sains Malaysia in 1986. He obtained his Ph.D in Approximation Theory from the University of Dundee, Scotland UK in 1993. He has been an academic staff member of the School of Mathematical Sciences, Universiti Sains Malaysia since 1981 and at present is an Associate Professor. He was a program chairman and deputy dean in the School of Mathematical Sciences, Universiti Sains Malaysia for many years. He has published various research papers in refereed national and international conference proceedings and journals. His current research areas include Computer Aided Geometric Design (CAGD), Medical Imaging, Numerical Analysis and Techniques and other related areas.