Solving Stiff Ordinary Differential Equations and Partial Differential Equations Using Analog Computing Based on Cellular Neural Networks

J. C. Chedjou[1], K. Kyamakya[1], M. A. Latif[1], U. A. Khan[1], I. Moussa[2] and Do Trong Tuan[3]

Abstract— **Setting analog cellular computers based on cellular neural networks systems (CNNs) to change the way analog signals are processed is a revolutionary idea and a proof as well of the high importance devoted to the analog simulation methods. We provide an in-depth description of the concept exploiting analog computing based on the CNN paradigm to solve nonlinear and highly stiff ordinary differential equations (ODEs) and partial differential equations (PDEs). We appply our method to the analysis of the dynamics of two systems modeled by complex and stiff equations. The first system consists of three coupled Rössler oscillators in a Master-Slave-Auxiliary configuration. The capabilities of this coupled system to exhibit regular and chaotic dynamics have been demonstrated so far. The synchronization modes of the coupled system can be exploited in chaotic secure communications. The second system is the Burgers' equation which is a well-known classical model for analyzing macroscopic traffic flow motions/scenarios. As a proof of concept of the proposed approach, the results obtained in this paper are compared with the results available in the relevant literature (benchmarking) and, the proposed concept is validated by a very good agreement obtained. The computation based on CNN paradigm is advantageous as it provides accurate and ultra-fast solutions of very complex ODEs and PDEs and performs real-time computing.**

Keywords— ODE, PDE, Cellular neural network, Field-programmable gate array (FPGA), templates, dicretization, coupling, stiffness.

I. INTRODUCTION

During the last sixty years, the theory of digital computation has been based on the concept of deterministic Turing machine [13]. However, despite the great success of digital computers, the existence of so called hard/complex problems has revealed some weaknesses or limitations on their computing capabilities. Indeed, digital computers are exposed to transient phenomena, stiffness, accumulation of round-off errors, divergences or floating point overflows [7-11] when dealing with the analysis of complex problems/scenarios which in essence are generally modeled by nonlinear ODEs and/or PDEs in the spatio-temporal domain. Further solving these complex and stiff equations with digital computers is very time consuming [14].

1. The authors are with the Transportation Informatics Group, Institute of Smart System Technologies, University of Klagenfurt (Austria), phone:+(43)463 2700 3550 fax:+(43)463 2700 3698, email: jean.chedjou@uni-klu.ac.at, kyandoghere.kyamakya@uni-klu.ac.at
2. The author is with the University of Yaoundé 1(Cameroon), email: moussaildoko@yahoo.fr
3. The author is with the faculty of Electronics and Telecommunications, Hanoi University of Technology(Vietnam), email: dotuan@mail.hut.edu.vn

Analog computers could be considered as an alternative method to analyze the dynamics of complex and stiff systems since these computers are not exposed to problems faced by digital computers when dealing with the simulation of complex models ODEs and/or PDEs. Despite some limitations presented by analog computers e.g. lack of universality, requirement of highly precise electronic components, and limitation of the dynamics of analog computers due to their bias (power supply), the analog method has a great potential to efficiently analyze complex and stiff models and provide ultra-fast and accurate solutions. Precisely, an implementation of analog computers on reprogrammable hardware (e.g. FPGA) could be envisaged to improve and optimize the computing process with analog computers. It should be worth mentioning that a challenging and up-to-date research question concerns the development of efficient, accurate and ultra-fast (high speed) computing platforms to solve complex and stiff ODEs and/or PDEs which are equations modeling the dynamics/motion of real-life engineering systems, scenarios or events [14].

Concerning the methods/approaches for solving stiff PDEs, some interesting works have been carried out. Ref. [15] does present a learning method based on CNN, i.e. a modified online back propagation (BP) algorithm. Ref. [16] deals with the implementation of the CNN paradigm on very large scale integrated circuit (VLSI) to solve PDE. The authors of Ref. [17] consider the discrete CNN paradigm to solve PDEs with applications for image multi-scale analysis. Ref. [18] presented an emulation of the CNN paradigm on digital platforms to solve PDEs. The idea of fixed-point is introduced which is being exploited to decrease the computing precision, leading to the increasing computing speed. Ref. [19] is focused on the CNN-based analog computing paradigm to solve PDEs. The paradigm is shown to be flexible in setting boundary conditions and selecting discretization methodologies as well. Ref. [20] discusses various possibilities of mapping CNN models into PDEs in order to approximate their solutions numerically. It is shown that the mapping process is not a general method as this process is limited (i.e. not valid) for a specific class of PDEs.

The state-of-the-art shows/presents the CNN paradigm as being an attractive alternative solution to conventional numerical computation method [1-2, 15-20]. Indeed, it has been intensively shown that CNN is an analog computing paradigm which performs ultra-fast calculations and provides accurate results [1, 2]. Interestingly, a speed-up of the analog computing process is possible by an implementation on reprogrammable computing (i.e. FPGA).

This paper considers the concept of analog computing based on the CNNs paradigm to solve complex nonlinear and stiff equations (ODEs and/or PDEs). We provide an in-depth presentation of our concept and apply it to two well known and stiff nonlinear differential equations namely the system consisting of three coupled Rössler equations in Master-Slave-Auxiliary configuration (coupled ODE) and the Burger's equations. The first coupled model has been shown appropriate to exhibit regular and chaotic waves which in their synchronization regime can be used in secure communications [10], the second model (Burgers') is the traditional model to describe macroscopic traffic flow scenarios.

We explain and show the possibility of deriving appropriate CNN templates to solve complex ODEs and/or PDEs. Using our approach, these equations are mapped to a CNN array in order to facilitate templates calculation. On the other hand complex/stiff PDEs are transformed into ODEs having array structures. This transformation is achieved by applying the method of finite difference. This method is based on the Taylor's series expansion.

The structure of the paper is as follows. Section 2 provides a brief overview of the CNN paradigm. Section 3 explains/shows the approach to solve complex/stiff ODEs with the CNN paradigm. An application example is provided based on solving coupled Rössler equations with CNN. Section 4 applies the same approach to solve a nonlinear and stiff PDE (Burgers') with CNN. In section 5, the benchmarking of the proposed concept is presented by comparing our results with those provided by the relevant literature. Section 6 deals with conclusions and outlooks.

II. THE CNN PARADIGM/CONCEPT

The concept of CNN was introduced by Leon O. Chua and Yang [1]. The cell, which is the fundamental building block of the CNN processor, is a lumped circuit containing both linear and nonlinear elements (Fig.1a). Fig.1a presents the CNN processor as an array of cells characterized from an input, a state and an output. This CNN processor is built using identical analog processing elements called cells [1-2]. These cells can be arranged in a k-dimensional square grid which is the most commonly used CNN type amongst many others namely the spherical CNN [4] and the star CNN [3], just to name a few. In these types of CNN, cells are locally connected (i.e. each cell is connected to its neighborhood) via programmable weights called templates. These templates are changed to make it programmable the CNN cell array.

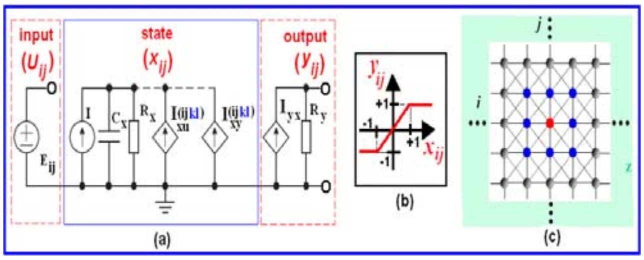

Fig. 1a: (a) The basic CNN cell defined as a nonlinear first order circuit, (b) The PWL sigmoid function, and (c) A CNN architecture composed by a two-dimensional array of MxN cells arranged in a

Hence, the essential/fundamental of the technology based on the CNN paradigm is located in templates. These templates are sets of matrices which are space invariant (i.e. cloning templates) if the values of templates do not depend on the position of the cell [1, 2]. This condition/case is particularly interesting to materialize spatial discretization. A complete (or full) specification of the dynamics of a CNN cell array requires the definition of Dirichlet (or boundary) conditions. The CNN computing platform developed in this paper exploits the structure of the state-control CNN (SC-CNN) [1, 2] which is modeled by (1).

$$\frac{dx_i}{dt} = -x_i + \sum_{j=1}^{M}\left[\hat{A}_{ij}x_j + A_{ij}y_j + B_{ij}u_j\right] + I_i \qquad (1)$$

The coefficients $\hat{A}_{ij}$, A_{ij} and B_{ij} are the self-feedback template, feedback template and control template, respectively. The schematic representation of a state-control CNN cell coupled to (M-1) neighboring cells is shown in Fig. 1b. I_i is the bias value and y_i is the nonlinear output sigmoid function of each cell. u_j denotes the input value and x_i represents the state of each cell.

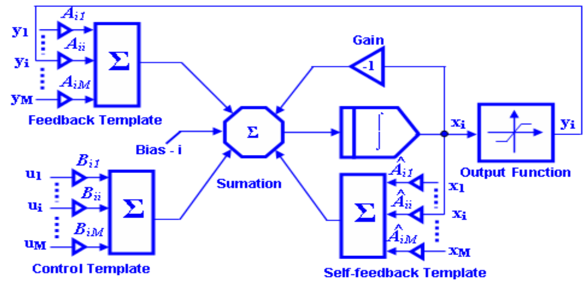

Fig. 1b: SIMULINK graphical representation of the basic model of a state-control CNN cell (SC-CNN) coupled to (M-1) Neighbours.

III. SOLVING COMPLEX AND STIFF ODE WITH CNN

A. *Principle for CNN templates findings*

According to the general theory in nonlinear dynamics based on the linearization of the vector field [12], complex and stiff ODEs can be described by a unique vector field in a bounded region of R^n which is solution in (2).

$$\frac{dx}{dt} = A(x)\left[x - F(x)\right] \qquad (2)$$

Where $A(x)$ is a $n \times n$ matrix function of x, F being the mapping of R^n to itself. It should be worth to mention that (2) can be transformed into the classical model of Chua equation in which components of the matrix A are all linear and F is a piecewise linear function of the variable x. Our approach in this paper transforms complex ODEs into the form described in (2) in order to make them solvable by the CNN paradigm since it is well-known that (2) can easily be mapped into the form of CNN model in (1). Therefore a good identification of the equations leads to determining the appropriate CNN templates to solve the complex and stiff ODEs.

We consider the case of a system consisting of three identical oscillators of the Rössler type coupled in a Master-Slave-Auxiliary configuration. The master (x_1, y_1, z_1) + slave (x_2, y_2, z_2) + auxiliary (x_3, y_3, z_3) system under investigation is modeled by the following set of complex and stiff coupled ODEs:

$$\frac{dx_{1,2,3}}{dt} = -\omega_{1,2,3} y_{1,2,3} - z_{1,2,3} + \varepsilon_{1,2,3}\left(x_{2,1,1} + x_{3,3,2} - x_{1,2,3}\right) \quad (3a)$$

$$\frac{dy_{1,2,3}}{dt} = \omega_{1,2,3} x_{1,2,3} + a_{1,2,3} y_{1,2,3} \quad (3b)$$

$$\frac{dz_{1,2,3}}{dt} = f_{1,2,3} + z_{1,2,3}\left(x_{1,2,3} - U_{1,2,3}\right) \quad (3c)$$

Where ω_i are the natural frequencies of the oscillators, ε_i are the elastic coupling coefficients (couplings through solutions) and a_i, f_i and U_i are the system parameters. This coupled system can be assimilated to a sequential system (i.e. a system which output depends on both inputs and previous state of the output).

We now want to provide an in-depth explanation of our approach for solving complex and stiff ODEs with the CNN paradigm. Therefore we consider (3) which are good prototypes of complex and stiff ODEs.

We first transform (3) into the form

$$\frac{d}{dt}\begin{bmatrix} x_{1,2,3} \\ y_{1,2,3} \\ z_{1,2,3} \end{bmatrix} = \begin{bmatrix} -\varepsilon_{1,2,3} & -\omega_{1,2,3} & -1 \\ +\omega_{1,2,3} & +a_{1,2,3} & 0 \\ 0 & 0 & -U_{1,2,3} \end{bmatrix}\begin{bmatrix} x_{1,2,3} \\ y_{1,2,3} \\ z_{1,2,3} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1,2,3}\left(x_{2,1,1} + \varepsilon_{3,3,2}\right) \\ 0 \\ f_{1,2,3} + x_{1,2,3} . z_{1,2,3} \end{bmatrix} \quad (4)$$

From (4) one can show the existence of fixed points through (5)

$$\frac{d}{dt}\begin{bmatrix} x_{1,2,3} \\ y_{1,2,3} \\ z_{1,2,3} \end{bmatrix} = 0 \quad (5)$$

Under a good specification of the parameters setting (coefficients) of (4) one can evaluate fixed/equilibrium points as follows:

Master system fixed point: $$\hat{X}_1 = \begin{bmatrix} x_{01} \\ y_{01} \\ z_{01} \end{bmatrix} \quad (6a)$$

Slave system fixed point: $$\hat{X}_2 = \begin{bmatrix} x_{02} \\ y_{02} \\ z_{02} \end{bmatrix} \quad (6b)$$

Auxiliary system fixed point: $$\hat{X}_3 = \begin{bmatrix} x_{03} \\ y_{03} \\ z_{03} \end{bmatrix} \quad (6c)$$

It should be worth mentioning that the aim is to linearize the vector field around fixed points. This linearization around a non-zero equilibrium/fixed point provides the possibility of modifying the nonlinear part of the coupled system without changing the qualitative dynamics of the system. This statement can be materialized by: $A\,X_{1,2,3} \rightarrow A\,\hat{X}_{1,2,3}$. Therefore, (2) can be considered to evaluate the linear part of the vector field at fixed points. This linear part is represented or materialized by 3×3 matrices defined as follows:

$$A_{master} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (7a)$$

$$A_{slave} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \quad (7b)$$

$$A_{auxiliary} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad (7c)$$

From which the corresponding CNN templates are derived under precise values of the coefficients of the model in (3).

B. Design of the complete CNN processor

We now want to design a CNN computing platform to investigate the issues of synchronization in the master-slave-auxiliary system modeled by (3). Full insight on synchronization issues can be achieved if it is possible to perform computations in wide ranges of the values of systems parameters in (3), even in cases some of these values can make the system experience stiffness. The efficiency of the calculations using CNN makes it a good candidate to perform computations in the cases of high stiffness and therefore an appropriate tool to tackle the difficulties faced by the classical numerical approach when dealing with computation of the model in (3). Using the structure of the elementary/basic CNN shown in Fig. 1b, we have designed the complete CNN processor to solve (3). We found that a total number of nine basic CNN cells are needed to implement the complete CNN processor on-top of SIMULINK to solve (3). Fig. 2a shows the structure of the three basic CNN cells needed to implement the master system. The same structure is needed to implement both slave and auxiliary systems due to the symmetric nature of the coupled systems under investigation.

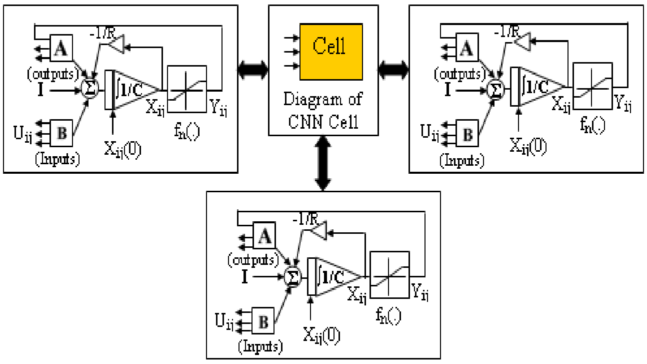

Fig. 2a: Schematic representation of the 3 CNN cells needed to implement each subsystem (i.e. master or slave or auxiliary) on top of SIMULINK.

Fig. 2b shows the implementation on-top of SIMULINK of the complete CNN processor to solve (3). The complete circuit in Fig. 2b is made-up of three layers with a total number of nine coupled CNN cells (Yellow blocs) whereby each layer uses three coupled CNN cells and represents the Rössler oscillator (i.e. Master or Slave, or Auxiliary subsystem).

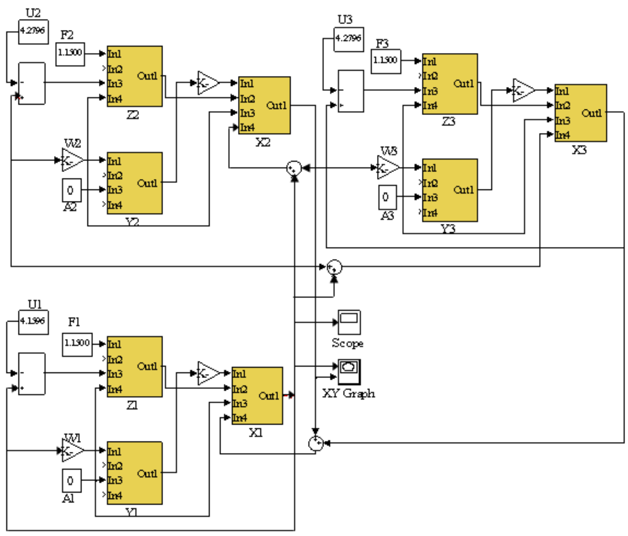

Fig. 2b: Schematic representation of the complete CNN- processor (master-slave-auxiliary) implemented on-top of SIMULINK.

C. *Results and proof of concept*

To illustrate the concepts, we have chosen the following values of the system parameters: $\omega_1 = 0.9700$, $a_1 = 0.2650$, $f_1 = 1.1500$, $U_1 = 4.1596$, $\varepsilon_1 = 0.0176$, $\omega_2 = 0.9750$, $a_2 = 0.2650$, $f_2 = 1.1500$, $U_2 = 4.2796$, $\varepsilon_2 = 0.0460$, $\omega_3 = 0.9650$, $a_3 = 0.2650$, $f_3 = 1.1500$, $U_3 = 4.2796$.

To verify the state of the complete CNN processor in Fig. 2b, we investigate the states of chaotic and regular synchronization in the coupled system when monitoring the control parameter ε_3. The results obtained from the complete CNN processor are shown in figures 3 revealing the possibility for the coupled system to behave in synchrony in the regular state (Fig. 3a) and in the chaotic state (Fig. 3b) as well.

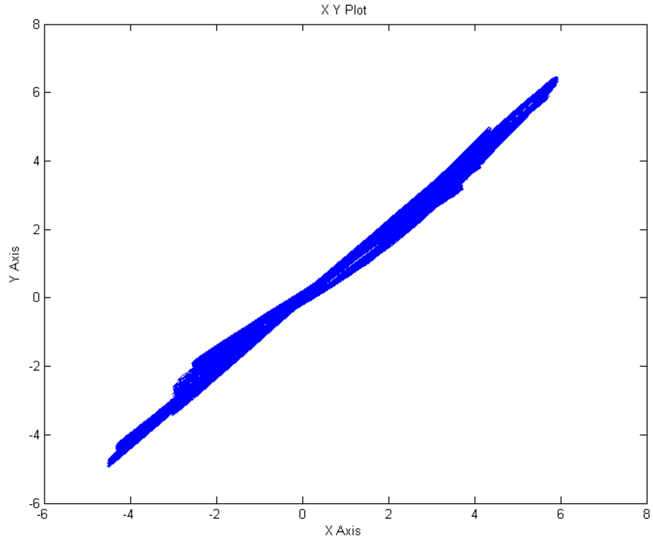

Fig. 3a. Simulation results using CNN on top of SIMULINK showing the projection of the regular attractors of the master and slave systems in the plane (x_1, x_2) for $\varepsilon_3 = 0.0050$. This projection shows the regime of regular synchronization in the master-slave-auxiliary coupled system. The other parameters are defined in the text.

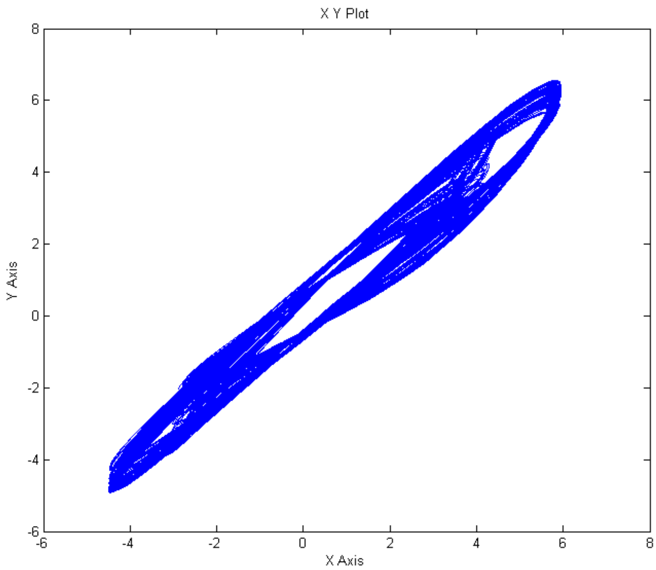

Fig. 3b. Simulation results using CNN on top of SIMULINK showing the projection of the chaotic attractors of the master and slave systems in the plane (x_1, x_2) for $\varepsilon_3 = 0.0155$. This projection shows the regime of chaotic synchronization in the master-slave-auxiliary coupled system. The other parameters are defined in the text.

In order to verify the results obtained from the CNN processor, we use the same sets of system parameters to perform a direct numerical simulation of (3). The direct numerical simulation performed using Turbo-C has led to results in Figs. 4. A very good agreement obtained when comparing results in Figs. 3 with the results in Figs. 4 is a proof to validate the concept we have developed in this paper showing the possibility of using the CNN to solve complex and stiff ODEs.

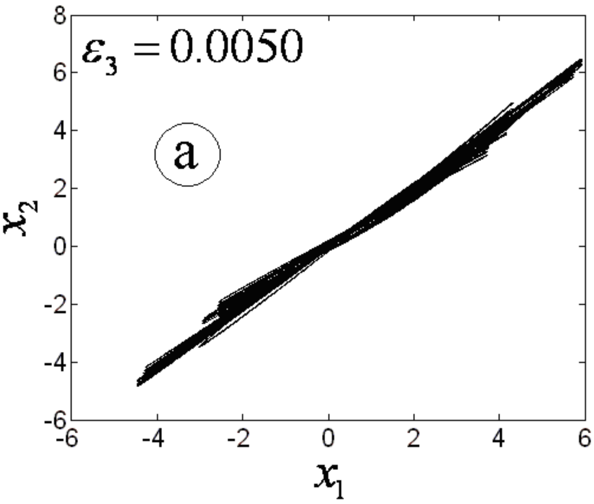

Fig. 4a. Direct numerical simulation results (Turbo-c) showing the projection of the regular attractors of the master and slave systems in the plans (x_1, x_2) for $\varepsilon_3 = 0.0050$. This projection shows the regime of regular synchronization in the Master-slave-auxiliary coupled system. Same values of parameters as in Fig. 3a.

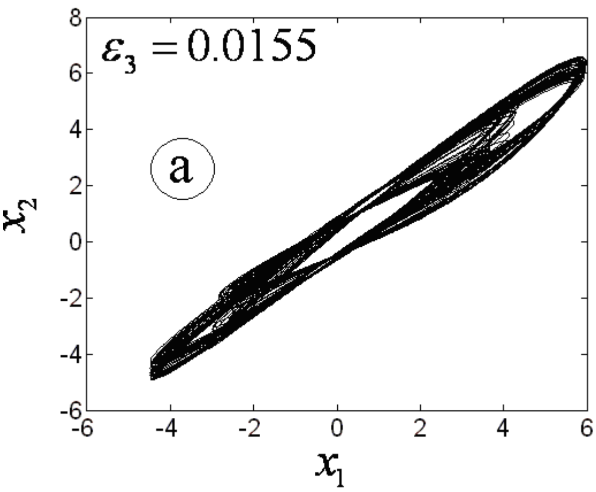

Fig. 4b. Direct numerical simulation results (Turbo-c) showing the projection of the chaotic attractors of the master and slave systems in the plans (x_1, x_2) for $\varepsilon_3 = 0.0155$. This projection shows the regime of chaotic synchronization in the Master-slave-auxiliary coupled system. Same values of parameters as in Fig. 3b.

IV. SOLVING NONLINEAR PDE WITH CNN

Four main variables (discrete or continuous) are considered when solving PDEs: time, value of the state variable, interaction of parameters, and space. The overall approach is based on transforming PDEs into ODEs and, arranged these ODEs into a form which can be identified with the CNN models for templates calculations.

PDEs are good prototypes for modeling traffic flow which is an interesting phenomenon of modern world. The modeling process though very important to get insights on/of the traffic dynamics appears challenging as we all experience it daily traffic problems are still not well understood. The reasons of this misunderstanding are justified by: the dynamic nature of the traffic dynamics which is time varying; the unpredictable or stochastic nature of the traffic dynamics which could be caused/due by/to accidents, maintenances of roads, or unpredictable events/situations influencing the traffic. These are some examples showing the nonlinear dynamical character of the traffic dynamics. Thus, nonlinear equations are good candidates to describe the dynamics of traffic. At the macroscopic level, nonlinear PDEs are used for the modeling process. Specifically, the Burgers' equation has been intensively used for this purpose. However, it is well known that deriving exact analytical solutions of complex/stiff nonlinear PDEs is challenging/impossible. The numerical simulation of complex nonlinear PDEs is exposed to transient phenomena, accumulation of round-off errors during computation and stiffness caused by the high degree of nonlinearity. This simulation is very time consuming as well when dealing with complex nonlinear PDEs. The key issue when simulating complex PDEs is the Dirichlet conditions (i.e. boundary conditions) which are generally very difficult to define during the numerical simulation of equations modeling engineering systems. This justifies the need of appropriate and efficient simulation tools which are robust to the problems faced by the classical simulation tools for solving complex PDEs. It should be worth noticing that this issue is still unsolved by the state-of-the-art as the available traffic simulation algorithms/tools are slow and provide results with low accuracy.

This paper develops a conceptual computing framework for complex/stiff nonlinear ODEs and PDEs. The concept developed is based on the paradigm of CNNs which could be implemented on FPGA.

Amongst equations of practical interest in engineering, the Burgers' equation (8) has been intensively used for many applications (e.g. modeling Traffic flow, modeling Shock Waves propagation, modeling Fluids dynamics, and modeling Heat propagation) [5]. This is a nonlinear wave equation which can be used to describe the corresponding density waves from several models of traffic flow. Another interesting application of the Burgers equation is the modeling and optimal control of traffic flow.

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} - \beta u(x,t) \frac{\partial u(x,t)}{\partial x} \qquad (8)$$

α and β are constants and $u(x,t)$ is the density of the wave. Equation (8) is a nonlinear equation with traveling wave solutions. In the conservation law form (i.e. $\alpha = 0$) the Burgers' equation reduces to a first order hyperbolic PDE. This equation is illustrated with the well-known model of highway traffic theory, namely the Lighthit-Whitham-Richards (LWR) [6].

It can be shown that the solution of (8) is envisaged in the form

$$u(x,t) = g[x - vt] \qquad (9a)$$

$$v(x,t) = \alpha + \beta u(x,t) \qquad (9b)$$

g is an unknown function which depends on the initial-boundary value problem (to be specified/defined). The solution in (9) describes a right-moving wave with a velocity $v(x,t)$ depending on the density of the wave. This dependence can lead to many striking effects namely breaking and formation of shock fronts. This last corresponds to bunching of cars on a highway.

In order to solve the PDE in (8) with the paradigm of CNNs, the spatial discretization method (i.e. the finite difference method) is performed in order to transform the proposed PDEs into sets of ODEs and arrange them into a suitable form the CNN paradigm can solve. Specifically, we first discretize the function spatially by using the Taylor series expansion and next we use the CNN paradigm to account for temporal change (i.e. solving ODE in time domain).

The second order Taylor's series expansion (of the solution in (8)) around a fixed point x_0 can be envisaged in the form

$$u(x,t) = u(x_0,t) + (x - x_0)u'(x_0,t) + \frac{(x - x_0)^2}{2} u''(x_0,t) \quad (10)$$

Considering two neighboring points situated at a distance Δx_0 around the fixed point x_0 (i.e. left and right to x_0) can lead to the following mathematical formulation:

$$x_i = x_0 \pm \Delta x_0 \quad (11)$$

(11) Can be used to derive new forms of (10) as follows:

$$u(x_0 + \Delta x_0,t) = u(x_0,t) + \Delta x_0 u'(x_0,t) + \frac{(\Delta x_0)^2}{2} u''(x_0,t) \quad (12a)$$

$$u(x_0 - \Delta x_0,t) = u(x_0,t) - \Delta x_0 u'(x_0,t) + \frac{(\Delta x_0)^2}{2} u''(x_0,t) \quad (12b)$$

Thus, one can use (12) to deduce the Taylor's series expansion of the second and first derivatives around the fixed point x_0 as follows:

$$u''(x_0,t) = \frac{u(x_0 + \Delta x_0,t) + u(x_0 - \Delta x_0,t) - 2u(x_0,t)}{(\Delta x_0)^2} \quad (13a)$$

$$u'(x_0,t) = \frac{u(x_0 + \Delta x_0,t) - u(x_0 - \Delta x_0,t)}{2\Delta x_0} \quad (13b)$$

Equation (11) can be used to construct a spatial domain which is made-up of a number of grid-points x_i arranged in a regular form, Δx_0 being the distance between them (i.e. grid-points). Therefore the time evolution of the solution $u(x,t)$ in (8) is obtained at each grid-point x_i. This leads to a general solution $u(x_i,t) = u_i$ which can be obtained from the following first order ODE derived by substituting (13) into (8).

$$\frac{du_i}{dt} = \left(\frac{\alpha}{(\Delta x_0)^2}\right)[u_{i+1} - 2u_i + u_{i-1}] - \left(\frac{\beta}{2\Delta x_0}\right)u_i[u_{i+1} - u_{i-1}] \quad (14)$$

The spatial domain is built from a number of grid-points localized by the position x_i, the index i being an integer. Therefore (14) clearly shows that the analog computing of PDEs is possible by transforming them into ODEs which are expressed in the form of (14). This form is a set of coupled first order ODEs, the number of equations being fixed by the index i.

To solve (14) by the CNN paradigm, we apply some algebraic manipulation on (14) to transform it into the form of i first order ODEs which are further identified with the SC-CNN model in (1). Here, the total number of cells used to build the complete CNN processor is fixed by the index i.

Our various numerical computations using the CNN paradigm were very slow as for the sake of obtaining accurate results we were obliged to use a huge amount of CNN cells. It was possible to obtain some sample results up to the maximal number of 10 CNN cells ($i_{\max} = 10$). Fig. 5a shows the temporal evolution of the solution in (8) obtained by our CNN processor built with 10 cells for the parameters $\alpha = \frac{1}{2}$ and $\beta = 1$. Using the same set of parameters, we have also performed the direct numerical simulation of (8) on MATLAB. Fig. 5b shows the spatial evolution of the solution in (8) while Fig. 5c shows the spatio-temporal evolution of the solution in (8). The results in Fig. 5a obtained using CNN simulator were generally very close to the same plots obtained using simulation on MATLAB. However, a divergence was observed for $i \le 8$. This divergence can be explained by the fact that the accuracy of the CNN simulator decreases with decreasing number of cells constituting the CNN simulator.

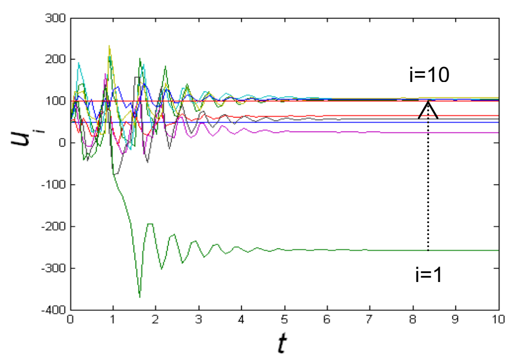

Fig. 5a. Results of the CNN simulator showing the temporal evolution of the solution in (8) for $\alpha = 1/2$ $\beta = 1$ and $i = 1$ *to* 10.

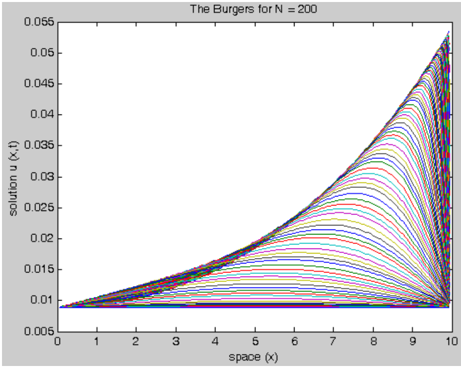

Fig. 5b. Results of a direct simulation on MATLAB showing the spatial evolution of the solution in (8) for $\alpha = 1/2$, $\beta = 1$ and $i = 200$

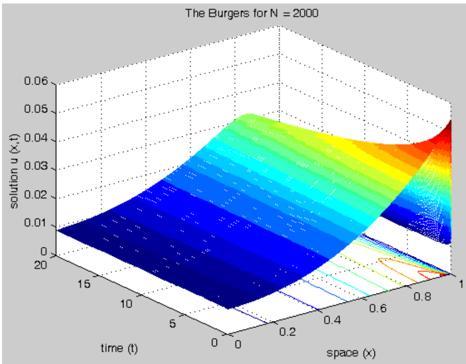

Fig. 5c. Results of a direct simulation on MATLAB showing the spatio-temporal evolution of the solution in (8) for $\alpha = 1/2$, $\beta = 1$ and $i = 2000$

V. BENCHMARKING

The proof of concepts of the approach developed in this paper (i.e. an approach based on the linearization of the vector field around fixed points associated with the CNN paradigm) is now performed by comparing the results obtained with those available in the relevant literature.

We have considered two types of well-known nonlinear and stiff models/equations (ODE & PDE) which have already been considered in Refs. [10, 21, 5] in other to make it possible a comparison of the results obtained by the approach developed in this paper.

Considering ODE, the approach developed in this paper has been used to solve a well-know model describing the dynamics of a system of coupled Rössler type non-identical self-sustained chaotic oscillators which has been intensively used in Ultra-Wide-Band communication and positioning systems [10, 21]. Our approach has revealed that the model of coupled Rössler equations can exhibit synchronized dynamics both in its regular and chaotic states for specific values/settings of system parameters. Using different values/settings of system parameters, similar results were reported in Ref. [10] by both numerical and experimental methods. Therefore, performing a fine tuning of the parameter- settings, the approach developed in this paper could lead to similar results (e.g. plots) as in Ref. [10]. This remark can be drawn from the fact that our approach has been used under the same values/settings of system parameters in Ref. [21] and, similar results were obtained.

Considering PDE, we have considered the nonlinear Burger's equation which is a well-known prototype commonly used to describe (in time domain) the mean density of moving particles along a well- specified axis [5]. This equation has also been presented in this paper as a good prototype for both modeling and simulation of macroscopic traffic scenarios. Although the approach developed in this paper and that developed in Ref. [5] are based on the application of a concept based on the CNN paradigm to solve the Burger's, the main difference is the challenging idea developed in our paper which consists of linearizing the vector field around fixed points in order to modify the nonlinear part of the model/equation without changing the qualitative dynamics of the system. Using the same values/settings of system parameters in Ref. [5], a very good agreement has been obtained when comparing the results (e.g. plots) in this paper with those in Ref. [5].

The main and challenging contribution of this paper is the proposal of a systematic method to derive for both nonlinear and stiff ODE and PDE the corresponding templates to make them solvable by the CNN paradigm. It should be worth noticing that although the approach based on the CNN paradigm to solve nonlinear and stiff ODE and PDE has been intensively developed in the relevant literature [1-6], it is still not clearly explained or shown the process leading to the derivation of the CNN templates. The main objective of this paper has been focused on the development of an analytical framework to clarify this issue.

VI. CONCLUSION

This work has presented a general concept for solving complex/stiff equations with the CNN paradigm. A systematic analytical study has been carried out showing the mapping of ODEs and PDEs into the CNN model in order to deduce the values of corresponding templates. The concept presented in this work has been applied for solving two specific types of well-known complex and stiff equations namely the system consisting of three coupled Rössler oscillators (ODE) and the Burgers' equation (PDE). It has been shown that solving nonlinear ODEs with the CNN paradigm is possible through the linearization process. Further, the space discretization process is essential to transform nonlinear PDEs into ODEs in order to make them solvable by the CNN paradigm. In this paper we use the second order Taylor series expansion for the discretization process. The results obtained using the approach developed in this paper were very close to those in the relevant literature. The parameter settings used in this paper were obtained from the relevant literature. The motivation of using the settings of parameters in the relevant literature was the possibility of comparing our results with the existing ones.

The limitations we found with the approach developed in this paper are twofold. The first is related to the fact that solving complex/stiff equations with CNN is possible only by performing a systematic analytical study in order to derive conditions in which the qualitative dynamics of the system might not be strongly sensitive to the degree of nonlinearity of the system. This analytical development appeared very challenging as the difficulty of performing this task increases with the degree of nonlinearity of the equation under investigation. Therefore an interesting research question would have been developing a general theoretical framework to linearize nonlinear ODEs. The second limitation is related to the discretization of PDEs in space. In fact it was observed that the accuracy of the method increases with increasing quantity of grid points. However, increasing the number of grid points make it very difficult the numerical solution of PDEs with the approach developed in this work. This can be justified by the fact that the numerical simulation of a huge amount of CNN cells on the MATLAB platform is very time consuming (very slow). To optimize the results of the approach proposed in this work it will be of great interest considering the analog computing based on reprogrammable hardware emulation on FPGA or on GPU to attempt the ultra fast computing process.

REFERENCES

[1] L.O Chua and L. Yang, "Cellular Neural Networks: Theory", *IEEE Trans. On Circuits and Systems*, Vol.35, 1988, 1257-1272

[2] G. Manganaro, P. Arena, L. Fortuna, "Cellular Neural Networks: Chaos, Complexity and VLSI Processing," *Springer-Verlag Berlin Heidelberg*, 1999, 44-45.

[3] M. Ito, and L.O Chua, "Star cellular neural networks for associative and dynamic memories", International Journal of Bifurcation and Chaos, vol. 14, 2004, 1725-1772.

[4] T. Yang, K. R. Crounse and L.O Chua, "Spherical cellular nonlinear networks", International Journal of Bifurcation and Chaos, vol. 11, 2001, 241-257.

[5] T. Roska, L.O Chua, D. Wolf, T. Kozek, R. Tetzlaff, and F. Puffer, "Simulating nonlinear waves and partial differential equations via CNN- Part I: Basic techniques", *IEEE Trans. On Circuits and Systems*, Vol. 42, 1995, 807-815.

[6] J.P. Aubin, A.M. Bayen, P. T. Saint-Pierre, "Computation and control of solutions to the Burgers equation using viability theory", proceeding of the American Control Conference, vol. 6, 2005, 3906-3911.

[7] J. C. Chedjou, K. Kyamakya, I. Moussa, H. –P. Kuchenbecker, W. Matis, "Behavior of a self-sustained Electromehanichal Transducer and Routes to Chaos", Journal of vibrations and Acoustics, ASME Transactions, Vol. 128, pp 282-293, 2006.

[8] J. C. Chedjou, K. Kyamakya, Van Duc Nguyen, I. Moussa and J. Kengne," Performance evaluation of analog systems simulation methods for the analysis of nonlinear and chaotic modules in communications", ISAST Transactions on Electronics and Signal processing, vol.2, 2008, pp71-82.

[9] J.C. Chedjou, H. B. Fotsin, P. Woafo, and S. Domngang, "Analog-Simulation of the Dynamics of a van der Pol oscillator coupled to a Duffing oscillator," IEEE Transactions on Circuits and Systems-I, vol. 48, pp. 748-757, 2001.

[10] J. C. Chedjou, K. Kyamakya, W. Mathis, I. Moussa, A. Fome, A. V. Fono, "Chaotic Synchronization in Ultra Wide Band Communication and Positioning Systems", Journal of Vibration and Acoustics, Transactions on the ASME, Vol. 130, 2008, 011012/1-011012/12.

[11] J. C. Chedjou, "On the Analysis of Nonlinear Electromechanical Systems With Applications," Doctorate Dissertation, University of Hanover, Germany, Published under the code:ISBN 3-8322-3750-X, Shaker Verlag (Germany), march 2005.

[12] R. Brown, "Generalizations of the Chua Equations," IEEE Transactions on Circuits and Systems-I, vol. 40, pp. 878-884, 1993.

[13] Rolf Herken, "The Universal Turing Machine. A Half-Century Survey", ISBN-13:978-3211826379, Springer, Wien, Feb 1995.

[14] J. Mailen Kootsey, "Future Directions in Computer Simulations", ISSN:0092-8240(print) 1522-9602(online), Journal:Bulletin of Mathematical Biology, Springer New York 1986.

[15] M. J. Aein, H. A. Talebi, "Introducing a training methodology for Cellular Neural Networks solving partial differential equations", International joint conference on neural networks 2009, ISBN:978-1-4244-3548-7.

[16] K. Hadad, A. Piroozmand, "Application of Cellular Neural Network(CNN) method to the Nuclear Reactor Dynamic Equations", Elsevier, Annals of nuclear energy 34(2007) 406-416.

[17] F. Corinto, M. Biey, M. Gilli, "Non-linear coupled CNN models for Multiscale Image Analysis", International journal of circuit theory and applications(special issue on CNN technology), vol. 34 issue 1, pp. 77-88 2006.

[18] Nagy Zoltan, Vöröshazi Zsolt, Szolgay Peter, "Emulated digital CNN-UM solution of partial differential equations", International journal of circuit theory and applications, ISSN: 0098-9886, vol. 34, pp. 445-470, 2006.

[19] Fausto Sargeni, Vincenzo Bonaiuto, "Programmable CNN Analogue chip for RD-PDE multi method simulations", Analog integrated circuits and signal processing, ISSN: 0925-1030, vol. 44, pp 283-292, issue 3 (September 2005).

[20] M. Gilli, T. Roska, L. O. Chua, and P.P. Civalleri, "On the relationship between CNNs and PDEs", Proceedings of the 7th IEEE workshop on Cellular Neural Networks and their applications, IEEE circuits and systems society 2002.

[21] I. Moussa, J. C. Chedjou, K. Kyamakya and Van Duc Nguyen, "Dynamics of a secure communication module based on chaotic synchronizations", ISAST Transactions on communication and networking, No.1, vol.2, 2008, pp.14-23.

Jean Chamberlain Chedjou received in 2004 his doctorate in Electrical Engineering at the Leibniz University of Hanover, Germany. He has been a DAAD (Germany) scholar and also an AUF research Fellow (Postdoc.). From 2000 to date he has been a Junior Associate researcher in the Condensed Matter section of the ICTP (Abdus Salam International Centre for Theoretical Physics) Trieste, Italy. Currently, he is a senior researcher at the Institute for Smart Systems Technologies of the Alpen-Adria University of Klagenfurt in Austria. His research interests include Electronics Circuits Engineering, Chaos Theory, Analog Systems Simulation, Cellular Neural Networks, Nonlinear Dynamics, Synchronization and related Applications in Engineering. He has authored and co-authored 3 books and more than 40 journals and conference papers.

Kyandoghere Kyamakya obtained the M.S. in Electrical Engineering in 1990 at the University of Kinshasa. In 1999 he received his Doctorate in Electrical Engineering at the University of Hagen in Germany. He then worked three years as post-doctorate researcher at the Leibniz University of Hannover in the field of Mobility Management in Wireless Networks. From 2002 to 2005 he was junior professor for Positioning Location Based Services at Leibniz University of Hannover. Since 2005 he is full Professor for Transportation Informatics and Director of the Institute for Smart Systems Technologies at the University of Klagenfurt in Austria.

M. Ahsan Latif obtained his MSc in computer sciences in 2001 at the University of Agriculture, Faisalababad, Pakistan. He has served Pakistan defence ministry as programmer for five years. Currently he is doing his Doctorate under prof. Kyandoghere Kyamakya and Dr. Jean Chamberlain Chedjou. His research domain comprises of nonlinear dynamics in image processing, cellular neural networks and solving stiff ODEs and PDEs.

Umair Ali Khan completed his bachlor degree in Computer Engineering from Quaid-e-Awam University, Nawabshah, Pakistan. He joined the same university as a lecturer in 2005. In 2008, he started his Masters degree in Information Technology in Alpen Adria University, Klagenfurt, Austria where he is also currently working as a research assistant in Transportation Informatics Group. His research interests include nonlinear dynamics, image processing, cellular neural networks and robotics. Before this, he has 3 international publications on obstacle detection using cellular neural networks, benchmarking of traditional genetic algorithm with a novel approach for image processing using cellular neural networks and traffic light controller based on reinforcement learning.

Moussa Ildoko holds a MSc. in control and signal processing and a Doctorate degree in Electronics from the University of "Valen- ciennes et du Hainaut-Cambrésis" in France, respectively in 1982 and 1985. He obtained his final Doctorate, the so-called "Doctorat d'Etat" in French educational system in 2008 at the University of Yaoundé I (Cameroon). He is currently an Associate researcher at UDETIME (Doctorate School of Electronics, Information Technology, and Experimental Mechanics) at the University of Dschang, Cameroon. Besides, he is a Senior Lecturer at the University of Yaoundé 1, Cameroon. His research interests are related to Nonlinear Dynamics, Analog Circuits Design, Chaos-based Secure Communications.

Do Trong Tuan received B.E., M.E., and Dr.Eng. degrees in Electronics and Telecommunications from Hanoi University of Technology, Vietnam in 1997, 2000 and 2006, respectively. He has been a lecturer at Hanoi University of Technology since 1999. During 2008-2009, he studied at Institute for Smart-Systems Technologies, University of Klagenfurt, Austria as a post-doctoral fellow. His fields of research are Mobile Communications, Electronic Navigations, Random/Stochastic Graph Optimization and Cellular Neural Networks.