5–10 July 2004, Antalya, Turkey — Dynamical Systems and Applications, Proceedings, pp. 32–47

Neural Networks: Modelling with Impulsive Differential Equations

Haydar Akça, Rajai Alassar and Yaqoub M. Shebadeh
King Fahd University of Petroleum and Minerals
Department of Mathematical Science
Dhahran 31261, Saudi Arabia
E-mail: akca@kfupm.edu.sa, alassar@kfupm.edu.sa, shehadeh@kfupm.edu.sa

Valéry Covachev*
Department of Mathematics & Statistics
College of Science, Sultan Qaboos University
Muscat 123, Sultanate of Oman
E-mail: vcovachev@hotmail.com, vcovachev@yahoo.com, valery@squ.edu.om

Abstract

Neural networks are being used to solve all kinds of problems from a wide range of disciplines. The topic is highly interdisciplinary in nature, and so it is extremely difficult to develop an introductory and comprehensive treatise on the subject in a short manuscript. A brief historical introduction is given and recent research works are summarized. In addition, we provide an example of the study of the stability characteristics of a time-dependent system of impulsive logistic equations by using discrete modelling.

1 Introduction

Mathematical modelling in neural networks has been based on "neurons" that are different both from real biological neurons and from the realistic functioning of simple electronic circuits. Unfortunately, many real-world problems do not come prepackaged with mathematical equations, and often the equations derived might not be accurate or suitable. The neuronal model is made up of four basic components: an input vector, a set of synaptic weights, summing function with an activation, or transfer function, and an output [4]–[9], [11]–[13], [15]–[17]. The bias increases or decreases the net input of the activation function. Throughout history, scientists have attempted to model physical systems using mathematical equations.

*Permanent address: Institute of Mathematics, Bulgarian Academy of Sciences, Sofia 1113, Bulgaria

This approach has been quite successful in some scientific fields, but not in all. For example, what equations would a doctor use to diagnose an illness and then prescribe a treatment? How can we tell whether somebody is telling the truth? These questions have been successfully dealt with by the adoption of "neural networks", or "artificial neural networks", as they are sometimes referred to, using machine learning or data mining. See for more details [4]–[6] and reference therein. McCulloch and Pitts (1943) assumed that a neuron simply counted the number of active inputs and produced an output after a short delay if the number exceeded a threshold value in any given time interval. This formulation permitted algebraic solutions for many network configurations, but omitted several properties considered important by later researchers. Two major divergences between the model and biological or physical systems stand out. Real neurons (and real physical devices) have continuous input-output relations. The original model used two-state threshold "neurons" that followed a stochastic algorithm. Each model neuron i had two states characterized by the output V_i^0 or V_i^1 which may often be taken as 0 and 1, respectively. The input of each neuron came from two sources, external inputs I_i and inputs from other neurons. The total input to neuron i is then

$$\text{Input to } i = H_i = \sum_{j \neq i} T_{ij} V_j + I_i,$$

where I_i are external inputs and T_{ij} can be biologically viewed as a description of the synaptic interconnection strength from neuron j to neuron i. Real neurons and real physical circuits have integrative time delays due to capacitance, and the time evolution of the state of such systems might be represented in a more meaningful way by a differential equation. Among the most popular models in the literature are artificial neural networks, which include additive Hopfield neural networks. The model is described by a set of differential equations with delays, namely, functional differential equations. For example, recently Zhou Jin *et al.* [15] studied the following form of the functional differential equations

$$\dot{x}_i = -c_i x_i(t) + \sum_{j=1}^{n} a_{ij}^0 f_j\big(x_j(t)\big) + \sum_{j=1}^{n} a_{ij}^\tau f_j\big(x_j(t-\tau_{ij})\big) + u_i, \quad i = 1, 2, \ldots, n,$$

or simply in the form

$$\dot{x} = -Cx(t) + Af(x(t)) + A^\tau f(x(t-\tau)) + u.$$

The main objective of the work is to give some sufficient conditions for the existence (or encoding) and globally exponential stability (associative) of periodic solutions for periodic delay neural networks without assuming the smoothness, monotonicity and unboundedness of the activation function. The model introduced is a generalization of some additive delayed neural networks such as delayed Hopfield neural networks and delayed cellular neural networks. Such type of models can well simulate biological neural networks and artificial intelligence systems, from the view point of reality, it should also naturally take into account evolutionary processes of the biological systems as well as disturbances of external influence, particularly under a periodically varying environment.

Delayed neural networks have attracted increasing interest in both theoretical studies and engineering applications. One of the most investigated and attractive problems in the dynamic of behaviors of Hopfield neural networks and design of the delay neural networks, is that of the existence, uniqueness, and global asymptotic stability of the equilibrium point.

The following system of integro-differential equations as a model for Hopfield neural networks with continuously distributed delays have been investigated by many researchers [1]–[3], [11]–[13], [15]–[17] and references therein:

$$\begin{aligned} \dot{x}_i &= -a_i x_i(t) + \sum_{j=1}^{n} a_{ij} g_j\big(x_j(t)\big) + \sum_{j=1}^{n} b_{ij} \int_{-\infty}^{t} k_{ij}(t-s) g_j\big(x_j(s)\big)\, ds + I_i, \quad t \ge 0, \\ & x_i(t) = \phi_i(t), \qquad -\infty < t \le 0, \qquad i = 1, 2, \ldots, n. \end{aligned}$$

Existence and uniqueness, and global asymptotic stability of the equilibrium point of Hopfield neural networks with distributed delays and under different assumptions have been discussed in detail by the researchers. In particular, a novel method of obtaining a discrete time dynamical system whose dynamics is inherited from the continuous time dynamical system has been suggested. Numerical algorithms of Hopfield type differential equations lead to discrete time dynamical systems and such discrete time systems should not give rise to any spurious behavior. The nonlinear neural activation functions $f_i(\cdot)$, $i \in \mathbb{Z}^+$, are usually chosen to be continuous and differentiable nonlinear sigmoid functions satisfying the following conditions:

(a) $f_i(x) \to \mp 1$ as $x \to \mp\infty$;

(b) $f_i(x)$ is bounded above by 1 and below by -1;

(c) $f_i(x) = 0$ at a unique point $x = 0$;

(d) $f_i'(x) > 0$ and $f_i'(x) \to 0$ as $x \to \mp\infty$;

(e) $f_i'(x)$ has a global maximum value of 1 at the unique point $x = 0$.

Some examples of activation functions $f_i(\cdot)$ are

$$f_i(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \qquad f_i(x) = \frac{1 - e^{-x}}{1 + e^{-x}} = \tanh(x/2),$$
$$f_i(x) = \frac{2}{\pi} \arctan\left(\frac{\pi}{2}x\right), \qquad f_i(x) = \frac{x^2}{1+x^2}\operatorname{sgn}(x),$$

where $\operatorname{sgn}(\cdot)$ is a signum function and all the above nonlinear functions are bounded, monotonic and nondecreasing functions. It has been shown that the absolute capacity of an associative memory network can be improved by replacing the usual sigmoid activation functions. There, it seems appropriate that nonmonotonic functions might be better candidates for neuron activation in designing and implementing an artificial neural network. Akça, Covachev *et al.* consider the problem under the presence of impulses in a series of papers [1]–[3]. The global stability characteristics of the systems supplemented with impulse conditions in the continuous-time case have been investigated [1]. The presence of impulses requires some modifications and the imposing of additional conditions on the systems:

$$\begin{aligned} \frac{dx_i}{dt} &= -a_i x_i(t) + \sum_{j=1}^{m} b_{ij} f_j(x_j(t)) + c_i, \quad t > 0,\ t \neq t_k, \\ \Delta x_i(t_k) &= I_i(x_i(t_k)), \quad i \in \{1, \dots, m\}, \ \ k = 1, 2, \dots, \end{aligned} \tag{1}$$

where $\Delta x(t_k) = x(t_k+0) - x(t_k-0)$ are the impulses at moments t_k and $t_1 < t_2 < \dots$ is a strictly increasing sequence such that $\lim\limits_{k\to\infty} t_k = +\infty$; $x_i(t)$ corresponds to the membrane potential of the unit i at time t; $f_j(\cdot)$ denotes a measure of response or activation to its incoming potentials; b_{ij} denotes the synaptic connection weight of the unit j on the unit i; the constants c_i correspond to the external bias or input from outside the network to the unit i; the coefficient a_i is the rate with which the unit self-regulates or resets its potential when isolated from other units and inputs.

The system (1) can be generalized by inserting time delays in neural networks. Then, we will consider the following system

$$\frac{dx_i}{dt} = -a_i x_i(t) + \sum_{j=1}^{m} b_{ij} f_j(x_j(t - \tau_{ij})) + c_i, \quad t > 0,\ t \neq t_k, \tag{2}$$

in which $i \in \{1, 2, \dots, m\}$ and $\tau_{ij} \geq 0$ corresponds to the transmission delay for $i, j \in \{1, 2, \dots, m\}$. The impulsive conditions are

$$\Delta x_i(t) = I_i(x_i(t)), \qquad t = t_k, \quad k = 1, 2, \dots$$

This system is supplemented with initial functions of the form

$$x_i(s) = \psi_i(s), \quad s \in [-\tau, 0],\ i \in \{1, \dots, m\},\ \tau = \max_{i,j \in \{1,\dots,m\}} \{\tau_{ij}\},$$

where $\psi_i(s)$ is continuous for $s \in [-\tau, 0]$.

For an integro-differential equation an impulsive condition including both the functional value and its integral also seems natural. Therefore the impulse conditions can be introduced in the form

$$\Delta x_i(t_k) = I_i(x_i(t_k)) = B_{ik} x_i(t_k) + \int_{t_{k-1}}^{t_k} c_{ik}(s) x_i(s)\, ds + \alpha_{ik}, \quad k \in \mathbb{Z}^+,$$

where $t_k > t_0 = 0$ and $c_{ik} : [t_{k-1}, t_k] \to \mathbb{R}$ are measurable functions, essentially bounded on the respective interval, B_{ik} and α_{ik} are some real constants.

A more satisfactory hypothesis is that the time delays are continuously distributed over a certain duration of time. System (2) is modified to a system of integro-differential equations

$$\begin{aligned} \frac{dx_i(t)}{dt} &= -a_i x_i(t) + \sum_{j=1}^{m} b_{ij} f_j \left(\int_{-\infty}^{t} K_{ij}(t-s) x_j(s)\, ds \right) + c_i, \\ & \qquad i = 1, 2, \dots, m, \quad t > 0, \end{aligned}$$

or

$$\begin{aligned} \frac{dx_i(t)}{dt} &= -a_i x_i(t) + \sum_{j=1}^{m} b_{ij} f_j \left(\int_{0}^{\infty} K_{ij}(s) x_j(t-s)\, ds \right) + c_i, \\ & \qquad i = 1, 2, \dots, m, \qquad t > 0, \end{aligned} \tag{3}$$

where for $i, j \in \{1, \dots, m\}$ the delay kernels $K_{ij}(s)$ are assumed to satisfy the conditions:

- $K_{ij} : [0, \infty) \to [0, \infty)$ are bounded and continuous.
- $\displaystyle\int_0^\infty K_{ij}(s) ds = 1.$
- There exists a positive number μ such that $\displaystyle\int_0^\infty K_{ij}(s) e^{\mu s} ds < \infty.$

The investigation of the stability of the respective discrete systems with impulse effect has been studied in detail in [2].

We rewrite the equation (1) in the form

$$\frac{d}{ds}\left(x_i(s)e^{a_i s}\right) = e^{a_i s}\left(\sum_{j=1}^{m} b_{ij} f_j(x_j(n)) + c_i\right),$$
$$i = \overline{1,m},\ s \in [nh,(n+1)h),$$

and integrate it over the interval $[nh, t]$ for $t < (n+1)h$ to obtain

$$x_i(t)e^{a_i t} - x_i(n)e^{a_i nh} = \frac{e^{a_i t} - e^{a_i nh}}{a_i}\left(\sum_{j=1}^{m} b_{ij} f_j(x_j(n)) + c_j\right), \quad i = \overline{1,m}.$$

In the last equality letting $t \to (n+1)h$ we obtain

$$x_i(n+1) = e^{-a_i h}x_i(n) + \frac{1 - e^{-a_i h}}{a_i}\left(\sum_{j=1}^{m} b_{ij} f_j(x_j(n)) + c_i\right), \quad i = \overline{1,m}.$$

We take this equation for $n \neq n_k = [t_k/h]$, and approximate the impulsive conditions by

$$x_i(n_k + 1) - x_i(n_k) = I_i(x_i(n_k)), \qquad i = \overline{1,m}, \quad k \in \mathbb{Z}^+.$$

Thus we obtain the discrete-time analogue of the system (1).

The discrete-time analogue of (2) is obtained in a similar way:

$$\begin{aligned} x_i(n+1) &= e^{-a_i h}x_i(n) + \phi_i(h)\sum_{j=1}^{m} b_{ij} f_j(x_j(n - \kappa_{ij})) \\ &\quad +\phi_i(h)c_i,\ \ i = \overline{1,m},\ n \in \mathbb{Z}_0^+ \setminus \{n_1, n_2, \ldots\}, \end{aligned}$$

where $\kappa_{ij} = [\tau_{ij}/h]$, $\phi_i(h) = (1 - e^{-a_i h})/a_i$, with initial condition

$$x_i(\ell) = \psi_i(\ell),\ \ i = \overline{1,m},\ \ \ell = \overline{-\kappa, 0},\ \ \kappa = \max_{i,j\in\{1,\ldots,m\}}\{\kappa_{ij}\}.$$

Finally, the discrete-time analogue of (3) is obtained in the same manner:

$$\begin{aligned} x_i(n+1) &= e^{-a_i h}x_i(n) + \phi_i(h)\sum_{j=1}^{n} b_{ij} f_i\left(\sum_{p=1}^{\infty} K_{ij}(p)x_j(n-p)\right) \\ &+ \ \ \phi_i(h)c_i, \quad i = \overline{1,m},\ n \in \mathbb{Z}_0^+ \setminus \{n_1, n_2, \ldots\}. \end{aligned}$$

and it is supplemented with initial values of the form

$$x_i(r) = \psi_i(r), \qquad r \in \mathbb{Z}_0^-,$$

and the sequence $\{\psi_i(r)\}_{r=-\infty}^0$ is bounded for all $i = \overline{1, m}$. The impulsive conditions are approximated by

$$x_i(n_k + 1) - x_i(n_k) = \sum_{\ell=n_{k-1}+1}^{n_k} B_{ik\ell} x_i(\ell) + \alpha_{ik}, \quad i = \overline{1, m},\ k \in \mathbb{Z}^+,$$

where, for convenience, $n_0 = -1$ and the constants $B_{ik\ell}$ satisfy additional conditions. The study [2] is devoted to the investigation of the stability of the discrete time analogue of impulsive systems.

2 Artificial neural networks

An artificial neural network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.

Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras. Many important advances have been boosted by the use of inexpensive computer emulations. Following an initial period of enthusiasm, the field survived a period of frustration and disrepute.

The first artificial neuron was produced in 1943 by the neurophysiologist Warren McCulloch and the logician Walter Pits. But the technology available at that time did not allow them to do too much. Neural networks process information in a similar way the human brain does. The network is composed of a large number of highly interconnected processing elements (neurons) working in parallel to solve a specific problem. Neural networks learn by example. Much is still unknown about how the brain trains itself to process information, so theories abound.

In the human brain, a typical neuron collects signals from others through a host of fine structures called *dendrites*. The neuron sends out spikes of electrical activity through a long, thin stand known as an *axon*, which splits into thousands of branches. At the end of each branch, a structure called a *synapse* converts the activity from the axon into electrical effects that inhibit or excite activity from the axon into electrical effects that inhibit or excite activity in the connected neurons (Figures 1, 2). When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes.

An artificial neuron is a device with many inputs and one output (Figure 3). The neuron has two modes of operation; the training mode and the using mode. In the training mode, the neuron can be trained to fire (or not), for particular input patterns. In the using mode, when a taught input pattern is detected at the input, its associated output becomes the current output. If the input pattern does not belong in the taught list of input patterns, the firing rule is used to determine whether to fire or not.

An important application of neural networks is pattern recognition. Pattern recognition can be implemented by using a feed-forward (Figure 4) neural network that has been trained accordingly. During training, the network is trained to associate outputs with input patterns. When the network is used, it identifies the input pattern and tries to output the associated output pattern. The power of neural networks comes to life when a pattern that has no output associated with it, is given as an input. In this case, the network gives the output that corresponds to a taught input pattern that is least different from the given pattern [4]–[6].

The above neuron does not do anything that conventional computers do not already do. A more sophisticated neuron (Figure 5) is the McCulloch and Pitts model (MCP). The difference from the previous model is that the inputs are 'weighted', the effect that each input has at decision making is dependent on the weight of the particular input. The weight of an input is a number which when multiplied with the input gives the weighted input. These weighted inputs are then added together and if they exceed a pre-set threshold value, the neuron fires. In any other case the neuron does not fire. In mathematical terms, the neuron fires if and only if

$$X_1W_1 + X_2W_2 + X_3W_3 + \cdots > T,$$

where $w_i,\ i = 1, 2, \ldots,$ are weights, $x_i,\ i = 1, 2, \ldots,$ inputs, and T a threshold. The addition of input weights and of the threshold makes this neuron a very flexible andpowerful one. The MCP neuron has the ability to adapt to a particular situation by changing its weights and/or threshold. Various algorithms exist that cause the neuron to 'adapt'; the most used ones are the Delta rule and the back error propagation. The former is used in feed-forward networks and the latter in feedback networks.

The most influential work on neural nets in the 60's went under the heading of 'perceptrons', a term coined by Frank Rosenblatt. The perceptron (Figure 6) turns out to be an MCP model (neuron with weighted inputs) with some additional, fixed, pre-processing.

Every neural network possesses knowledge which is contained in the values of the connections weights. Modifying the knowledge stored in the network as a function of experience implies a learning rule for changing the values of the weights (Figure 7).

Information is stored in the weight matrix W of a neural network. Learning is the determination of the weights. We can distinguish two major categories of neural networks:

- Fixed networks in which the weights cannot be changed, *i.e.*, $\frac{dw}{dt} = 0$. In such networks, the weights are fixed *a priori* according to the problem to solve.
- Adaptive networks which are able to change their weights, *i.e.*, $\frac{dw}{dt} \neq 0$.

The behavior of an ANN depends on both the weights and the input-output function (transfer function) that is specified for the units. We denote by w_{ij} the weight of the connection from unit u_i to unit u_j. It is then convenient to represent the pattern of connectivity in the network by a weight matrix w whose elements are the weights w_{ij}. Two types of connection are usually distinguished: excitatory and inhibitory. A positive weight represents an excitatory connection whereas a negative weight represents an inhibitory connection. The pattern of connectivity characterizes the architecture of the network.

This function typically falls into one of three categories:

- Linear (or ramp), the output activity is proportional to the total weighted output
  $$x_j = \sum_i y_i w_{ij},$$
  where y_i is the activity level of the i-th unit in the previous layer and w_{ij} is the weight of the connection between the i-th and the j-th unit.
- Threshold, the output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.
- In the form of sigmoid, the output varies continuously but not linearly as the input changes, *i.e.*,
  $$y_j = \frac{1}{1 + e^{-x_j}}.$$

Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations.

To make a neural network that performs some specific task, we must choose how the units are connected to one another (Figure 8) and we must set the weights on the connections appropriately. The connections determine whether it is possible for one unit to influence another. The weights specify the strength of the influence.

Once the activities of all output units have been determined, the network computes the error E which is defined by the expression

$$E = \frac{1}{2} \sum_i (y_i - d_i)^2 ,$$

where y_i is the activity level of the i-th unit in the top layer and d_i is the desired output of the i-th unit. Neural networks have wide applicability to real world business problems. In fact, they have already been successfully applied in many industries. Since neural networks are best at identifying patterns or trends in data, they are well suited for prediction or forecasting needs including: sales forecasting, industrial process control, customer research, data validation, risk management, target marketing.

ANN are also used in the following specific paradigms: recognition of speakers in communications; diagnosis of hepatitis; recovery of telecommunications from faulty software; interpretation of multi-meaning Chinese words; undersea mine detection; texture analysis; three-dimensional object recognition; hand-written word recognition; and facial recognition.

3 Example: an impulsive logistic equation

Consider the nonautonomous logistic equation

$$\begin{aligned} \frac{dx}{dt} &= r(t)x(t)\left(1 - \frac{x(t)}{K(t)}\right), \quad t > 0,\ t \neq \tau_k, & (4)\\ \Delta x(t) &= I_k(x(t)), \qquad t = \tau_k, \quad k = 1, 2, \ldots, & (5) \end{aligned}$$

in which $r(t)$ is nonnegative and $K(t)$ is a strictly positive continuous function, and I_k are bounded operators. In the real evolutionary processes of the population, the perturbation or the influence from outside occurs "instantly" as impulses, and not continuously. The duration of these perturbations is negligible compared to the duration of the whole process, more details about the theory of impulsive differential equations and applications see [10, 14] and references therein. Also impulsive perturbations (harvest, taking out, hunting, fishing, etc.) are more practical and realistic compared to the any kind of continuous harvest. For instance, a fisherman can not fish 24 hours a day and furthermore, the seasons also determine the fishing period. Similar considerations are applicable for hunting and taking away a huge part of any biomass. The logistic equation (4) has been intensively studied by various researchers [1, 2, 4] and [5, 6], considering existence of the solutions, asymptotic properties of the solutions, sufficient conditions for the oscillation of the solutions and so on.

As in §1 we obtain the discrete counterpart of (5), (6):

$$\begin{aligned} x(n+1) &= \frac{e^{r(n)h}x(n)}{1+\left(\frac{e^{r(n)h}-1}{K(n)}\right)x(n)}, \quad n \neq m_k, \\ x(m_k+1) &= x(m_k)+I_k(x(m_k)), \qquad k=1,2,\ldots \end{aligned} \tag{6}$$

Theorem 1 *Let the following conditions hold:*

$$0 \le \inf_{n\in\mathbb{Z}^+} r(n),\ R=\sup_{n\in\mathbb{Z}^+} r(n)<\infty, 0<K_* \le \inf_{n\in\mathbb{Z}^+} K(n),$$

$$\sup_{n\in\mathbb{Z}^+} K(n)<\infty \text{ and } I_k(x(m_k))=cx(m_k),$$

where $c>0$. Then for $h>0$ satisfying the inequality $h \le \ln(1+c)/R$ a solution $x(n)$ of (6) corresponding to $x(0)>0$ satisfies the inequality

$$\frac{1}{x(n)} \le \frac{1}{x(0)} e^{-\sum\limits_{i=0}^{n-1} r(i)h} + \sum_{j=1}^{n} \left(\frac{1-e^{-r(n-j)h}}{K(n-j)}\right) e^{-\sum\limits_{\ell=1}^{j-1} r(n-\ell)h}.$$

Theorem 2 *Let all assumptions of Theorem 1 hold and suppose further that there exists a number $\hat{r}>0$ such that*

$$\lim_{m\to\infty} \frac{1}{m}\left\{\sum_{j=1}^{m} r(n-j)\right\} = \hat{r},\ m\in\mathbb{Z}^+,\ \text{uniformly on } n\in\mathbb{Z}.$$

Then the solution $x(n)$ of the system (6) tends to $x^*(n)$ as $n\to\infty$, where $x^*(n)$ is given by

$$x^*(n) = \left[\sum_{j=1}^{\infty}\left(\frac{1-e^{-r(n-j)h}}{K(n-j)}\right) e^{-\sum\limits_{\ell=1}^{j-1} r(n-\ell)h}\right]^{-1}$$

in the sense that $x(n)-x^*(n)\to 0$ as $n\to\infty$.

The proofs of Theorem 1 and Theorem 2 will be given elsewhere.

Acknowledgement

The first three authors would like to thank King Fahd University of Petroleum and Minerals, Department of Mathematical Sciences for providing excellent research facilities.

References

[1] Akça H., Alassar R., Covachev V., Covacheva Z. and Al-Zahrani E. A., *Continuous-time additive Hopfield-type neural networks with impulses*, Journal of Mathematical Analysis and Applications, **290** (2004), No. 2, 436–451.

[2] Akça H., Alassar R., Covachev V. and Covacheva Z., *Discrete counterparts of continuous-time additive Hopfield-type neural networks with impulses*, Dynamic Systems and Applications, **13** (2004), 75–90.

[3] Covachev V., Akça H., Covacheva Z. and Al-Zahrani E. A., *A discrete counterpart of a continuous-time additive hopfiled-type neural network with impulses in an integral form*, Studies of the University of Žilina Mathematical Series, **17** (2003), 11–18.

[4] Freeman J. and Sakura D., *Neural Networks: Algorithms, Applications, and Programming Techniques*, Addision-Wesley, Reading, Mass., 2001.

[5] Haykin S., *Neural Networks, A Comprehensive Foundation*, MacMillan, New York, 2000.

[6] Hecht-Nielsen R., *Neuro Computing*, Addision Wesley, Reading, Mass., 2000.

[7] Hertz J., Krogh A. and Palmer G. R., *Introduction to the Theory of Neural Computation*, Addision-Wesley Publishing Company, Redwood City, 1991.

[8] Hopfield J. J., *Neural networks and physical systems with collective computational abilities*, Proc. Natl. Acad. Sci. USA, **79** (1982), 2554–2558.

[9] Iyengar S. S., Cho E. C. and Phoha V. V., *Foundations of Wavelet Network and Applications*, Chapman and Hall / CRC Press, New York, 2002.

[10] Lakshmikantham V., Bainov D. D. and Simeonov P. S., *Theory of Impulsive Differential Equations*, Series in Modern Applied Mathematics, **6**, World Scientific, Singapore, 1989.

[11] Mohamad S., *Continuous and Discrete Dynamical Systems with Applications*, Ph. D. Thesis, The Flinders University of South Australia, April 17, 2000.

[12] Mohamad S. and Gopalsamy K., *Neuronal dynamics in time varying environments: Continuous and discrete time models*, Discrete and Continuous Dynamical Systems, **6** (2000), No. 4, 841–860.

[13] Mohamad S. and Gopalsamy K., *Dynamics of a class of discrete-time neural networks and their continuous-time counterparts*, Mathematics and Computers in Simulation, **53** (2000), 1–39.

[14] Samoilenko A. M. and Perestyuk N. A., *Impulsive Differential Equations*, World Scientific Series on Nonlinear Science. Ser. A: Monographs and Treatises, **14**, World Scientific, Singapore, 1995.

[15] Zhou Jin, Zengrong Liu and Chen G., *Dynamics of periodic delayed neural networks*, Neural Networks, **17** (2004), 87–101.

[16] Zhang Q. and Benveniste A., *Wavelet networks*, IEEE Transaction on Neural Networks, **3** (1992), 889–898.

[17] Zhao Hongyong, *Global asymptotic stability of Hopfield neural network involving distributed delays*, Neural Networks, **17** (2004), 47–53.