2011

# Discrete-Time Recurrent Neural Network and Its Application to Compression of Infra-Red Spectrum

Leong-Kwan Li
*Hong Kong Polytechnic University*, malblkli@polyu.edu.hk

Sally S. L. Shao
*Cleveland State University*, s.shao@csuohio.edu



## Original Citation

Leong Kwan Li and Sally S. L. Shao. (2011). Discrete-Time Recurrent Neural Network and Its Application to Compression of Infra-Red Spectrum. Neural Network World, 21(5), 393-406.

# DISCRETE-TIME RECURRENT NEURAL NETWORKS AND ITS APPLICATION TO COMPRESSION OF INFRA-RED SPECTRUM

*Leong-Kwan Li, S. Shao*

**Abstract:** We study the discrete-time recurrent neural network that derived from the Leaky-integrator model and its application to compression of infra-red spectrum. Our results show that the discrete-time Leaky-integrator recurrent neural network (RNN) model can be used to approximate the continuous-time model and inherit its dynamical characters if a proper step size is chosen. Moreover, the discrete-time Leaky-integrator RNN model is absolutely stable. By developing the double discrete integral method and employing the state space search algorithm for the discrete-time recurrent neural network model, we demonstrate with quality spectra regenerated from the compressed data how to compress the infra-red spectrum effectively. The information we stored is the parameters of the system and its initial states. The method offers an ideal setting to carry out the recurrent neural network approach to chaotic cases of data compression.

## 1. Introduction

Neural networks are trainable analytic tools that attempt to mimic information processing patterns in the brain. They can be used effectively to automate both routine and ad hoc tasks. Since the human brain is a recurrent neural network (RNN) – a network of neurons with feedback connections, RNNs are biologically more plausible and computationally more powerful than other adaptive models, such as Hidden Markov Models (no continuous internal states), feedforward networks and Support Vector Machines (no internal states at all). One of the reasons is that when the neural signals are exchanged between different cell assemblies, due to the recurrent connections between the neurons, there are typical brain functions involved. A possible way to model this behavior is by describing each cell assembly by a Leaky-integrator unit that integrates input over time while the internal activation is continuously decreased by a dampening leaky term ([4]). With different leakage constants, single neurons can also be described by Leaky-integrator units. Therefore, the Leaky-integrator model – a continue-time RNN model of the neuron can be used to approximate a biological neuron quite well (more details can be found in [4]). Moreover, because of the richer dynamical structures of RNNs it can learn extremely complex temporal patterns to yield good results. Therefore, the Leaky-integrator dynamics are common in computational neuroscience and have been studied by many researchers in the field in the past few decades ([6], [7], [12]).

In this paper, instead of manipulating the RNN's input-output relationship which is represented in the input and output spike trains, we use the internal states of the neurons (that is, the membrane potential, in neuroscience terms) in our derived RNN and not their firing rate or any other form of "output", as they do not fire spikes at all. The reasons of this choice are that as our work attempts to reproduce discretised continuous-time signals, the internal state of the neuron is more appropriate for approximating the signals, rather than some function of the network's spiking.

Although there are more common and handy data compression techniques, such as the Fast Fourier Transforms and the wavelet transforms, available in the recent years, neural networks are considered to be very suitable for nonlinear signal processing problems because of their inherently nonlinear nature (see [9], [18]). Zaknich and Attikiouzel ([18]) pointed out in 1995 that most signal processing problems are related to a time or spatial data series. Li studied in [7] the data compression problems by recurrent neural dynamics. He shows that there exists a discrete-time neural network for any given finite signals ([6]). The difficulty of dealing with the large scale feedforward networks, numerous neurons and the parameters could be skillfully overcome by using the RNN of the dynamical system approach according to Li. Therefore, the RNNs of the dynamical systems can be applied to solve the problems of data compression ([7]). Li, Chau and Leung further applied the RNN of the dynamical system approach to the task of compression of ultraviolet-visible spectrum in ([8]). In this paper, we study the discrete-time recurrent neural network model and its application to compression of infra-red (IR) spectrum. Our results shows that the discrete-time Leaky-integrator RNN model can be used to approximate the continuous-time model and inherit its dynamical characters if a proper step size is chosen, and the discrete-time Leaky-integrator RNN model is absolutely stable. By developing the double discrete integral method which removes the noise around the trend of the data and provides a smoother visual image of the trend, and employing the state space search learning algorithm (SSSA), we demonstrate in the empirical examples, with quality spectra regenerated from the compressed data, that the storage space of the spectral information could be reduced significantly by using the proposed RNN of dynamical system approach. Meanwhile, a double discrete integral method is developed as a digital filtering technique to smoothen the given signal. The method is simple, flexible and very easy to be implemented. Awaiting this, we are inclined to consider the proposed method as the preferred method on account of its simplicity and storage space saving.

The organization of the paper is as follows. In Section 2, we first highlight the Leaky-integrator model and its discrete-time RNN model, then we present some theoretical results of the characters of the solutions of the Leaky-integrator RNN model and the stability properties of the discrete-time system. The compression techniques and methodology are presented in Section 3. Empirical results and the relative merits of the method are discussed in the concluding Section 4.

## 2. Characters of the Leaky-Integrator Recurrent Neural Network

In this section, we introduce some characters of the solutions of the Leaky-integrator RNN model and the stability properties of the discrete-time system.

Consider the continuous-time Leaky-integrator RNN model of the system of nonlinear equations described by

$$\begin{aligned} \frac{dx_i}{dt} &= -a_i x_i + b_i \sigma(\sum_{j=1}^{n} w_{ij} x_i + J_i) \\ i &= 1, 2, 3, \ldots n, \end{aligned} \tag{1}$$

where $x_i$ represents the internal state of the $ith$ neuron. $W = [w_{ij}]_{n\times n}$ is the synaptic connection weight matrix. $A = \mathrm{diag}[a_1, a_2, ..., a_n]$ and $B = \mathrm{diag}[b_1, b_2, ..., b_n]$ are diagonal matrices with *positive diagonal entries*, where $a_i$ of matrix A represents the inverse of the neuron's leakage time constant of $ith$ neuron $\tau_i$ (that is, $1/\tau_i$), and B represents the neuron's resistance. $J = [J_1, J_2, ..., J_n]^T$ is the input bias or threshold vector of the system. $\sigma$ is a neuronal activation function that is bounded, differentiable and monotonic increasing on [-1, 1]. We assume that $\sigma(z) = \tanh(z)$, which is the symmetric sigmoid logistic function, and $\sigma = diag[\sigma_1, \sigma_2, ..., \sigma_n]$ with $\sigma_i$ is defined by

$$\begin{aligned} \sigma_i(Wx + \theta) &= \sigma(\sum_{j=1}^{n} w_{ij} x_i + J_i), \\ i &= 1, 2, ..., n. \end{aligned} \tag{2}$$

It is a fully connected network recurrently. That is the synaptic connection matrix $W = [w_{ij}]_{n\times n}$ is a nonzero matrix. It is possible to have some 0 valued weights in the matrix $W$ during the optimization procedure for some intermediated iterations. However, the resulting optimal $W \neq 0$. A fully connected Leaky-integrator RNN model for $n = 4$ is depicted in Fig. 1.

System (1) has been studied and used in many applied areas of sciences (see [12], [13], [6]) for the past. However, the solution behavior the system (1) and the stability property of its discrete-time model are less understood from the dynamical system point of view. Studying the solution behavior of system (1) will help us to understand the characters of the network behavior in general, which will provide more hints in network design and learning.

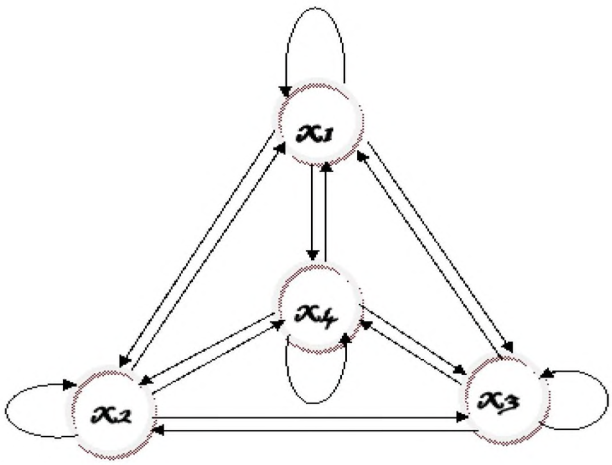


**Fig. 1** *A Leaky-integrator RNN model when* $n = 4$

Before proceeding to discuss the main results, we re-write system (1) as the following matrix form

$$\frac{dx}{dt} = -Ax + B\sigma(Wx + J), \tag{3}$$

we first show that the RNN model (3) has bounded solution trajectory for any initial point in $R^n$.

By Peano's local existence theorem ([1]) for solutions of ordinary differential equations, given any $x_0 \in R^n$, there exists a positive number $t^*(x_0)$ such that the system (3) has a solution $x(t) \in R^n$ for $t \in [0, t^*(x_0))$, which is the maximal right existence interval of the solution $x(t)$ satisfying $x(0)$. Now we show our theorem below.

**Theorem 1.** Given an initial state $x(0) = x_0$, there exists a positive number $t^*(x_0)$ such that the solution $x(t) \in R^n$ of the Leaky integrator model (3)

$$\frac{dx}{dt} = -Ax + B\sigma(Wx + J),$$

is unique and bounded for all $t \in [0, t^*(x_0))$. In addition, this local existence of the solution $x(t)$ exists globally. That is, $t^*(x_0) = +\infty$ for any $x_0 \in R^n$. In other words, the solution of system (3) is globally bounded.

Proof. There are many ways that can be employed to prove the desired results. Here we align with the approach in [11].

Let us consider the linear systems of differential equations

$$\frac{dy}{dt} = -Ay + Bv_p + J \qquad \text{and} \qquad \frac{dz}{dt} = -Az + Bv_q + J, \tag{4}$$

where $A = \text{diag}[a_1, a_2, ..., a_n]$ and $B = \text{diag}[b_1, b_2, ..., b_n]$ are diagonal matrices with positive diagonal entries defined as in (3). $v_p = (p, p, ..., p)^T$ and $v_p =$

$(q, q, ..., q)^T$ are the $n$ dimensional constant vectors to be determined later, and $J = [J_1, J_2, ..., J_n]^T$ is the system input. We set $y(0) = z(0) = x(0) = x_0$.

Claim: $y_i(t) \leq x_i(t) \leq z_i(t)$ for all $i = 1, ..., n$. and for all real $t$.

The proof of the claim is as follow: Let $r = x_j - y_j$, then $r(0) = 0$. Now

$$r' = -a_i r + g,$$

where we choose $p$ such that

$$g = b_i(\sigma_i(u_i) - p) - J_i > 0 \quad \text{with} \quad u_i = \sum_{j=1}^{n} w_{ij} x_i + J_i.$$

Thus, $r(t) = e^{-a_i t} \int_0^t g e^{a_i s} ds > 0$. Therefore, $y_i(t) \leq x_i(t)$ for all $t \in [0, t^*(x_0))$ with some positive number $t^*(x_0)$. Similarly we can show that $z_i(t) \geq x_i(t)$. It follows that there exists a bounded solution $x(t)$ of system (3) for any given initial state $x_0$. Moreover,

$$y_i(t) = [y_i(0) - \frac{b_i p + J_i}{a_i}]e^{-a_i t} + \frac{(b_i p + J_i)}{a_i}$$

and

$$z_i(t) = [z_i(0) - \frac{b_i q + J_i}{a_i}]e^{-a_i t} + \frac{(b_i q + J_i)}{a_i}.$$

Thus,

$$[y_i(0) - \frac{b_i p + J_i}{a_i}]e^{-a_i t} + \frac{(b_i p + J_i)}{a_i} \leq x_i(t) \leq [z_i(0) - \frac{b_i q + J_i}{a_i}]e^{-a_i t} + \frac{(b_i q + J_i)}{a_i}$$

for $i = 1, 2..., n$.

Hence, as time $t$ tends to infinity, each component of the solution $x(t)$ of system (3) is *asymptotically contained* in an interval $[\frac{(b_i p + J_i)}{a_i}, \frac{(b_i q + J_i)}{a_i}]$. It follows that system (3) is globally bounded.

For the uniqueness of the solution $x(t)$ of system (3), let $f \in C^n[R^n]$ defined by

$$f_i = -a_i x_i + b_i \sigma(\sum_{j=1}^{n} w_{ij} x_j) + J_i, \quad i = 1, 2, ..., n. \tag{5}$$

and $\sigma$ is in $C^n$ for $n \geq 2$. Suppose $\sigma'$ is bounded by a positive constant $k$. Then, by the mean-value theorem, $\sigma(a) - \sigma(b) = \sigma'(\phi)(a - b)$ for some $\phi \in (a, b)$, that is, $|\sigma(a) - \sigma(b)| < p|a - b|$. Hence,

$$\begin{aligned}
\| f(x) - f(y) \| &\leq \| A(x - y) \| + \| B(\sigma(Wx) - \sigma(Wy)) \| \\
&\leq \| A \| \| x - y \| + \| B \| \| \sigma' \| \| Wx - Wy \| \\
&\leq \| A \| \| x - y \| + k \| B \| \| W \| \| x - y \| \\
&= (\| A \| + k \| B \| \| W \|) \| x - y \| .
\end{aligned}$$

It implies that $f$ is Lipschitz over $R^n$ with Lipschitz constant $\| A \| + k \| B \| \| W \|$. Thus, the uniqueness of the solution is also guaranteed by the theory of ordinary differential equations. Therefore, for a fixed $W$, there exists a unique solution for system (3) for arbitrary initial state $x_0$. We finish the proof.

Note that if the input $J(t)$ is a bounded function of time $t$, the global boundedness of the system is not affected. Yet, the equilibrium state $x^* = A^{-1}[B\sigma(Wx^* + J)]$ depends on the input $J$.

Mind that for a dynamical system, though all the real parts of the eigenvalues of the Jacobian are negative for every $x \in R^n$, this dynamical system needs not to be globally asymptotically stable. Furthermore, it is important to find a condition for which the system is strictly globally asymptotically stable, so that the output state $x(t)$ will reach the same steady state as $t \to \infty$ for arbitrary given initial state.

Li and Shao ([11]) show that if $\omega \in R^n$ is the solution set of (1), then for any solution trajectory of the RNN model (1) starting from $\Omega$, the solution trajectory *cannot* escape from $\Omega$. Moreover, for any solution trajectory of the RNN model (1) starting from the outside of $\Omega$, it will converge to $\Omega$.

Define set $S^* = \{y^* \in R^n | y^* = A^{-1}B\sigma(Wy^* + J)\}$ of equilibrium points of (3). For simplicity, we set matrix $A = diag[a_1, a_2, ..., a_n]$ of system (1) to be the $n \times n$ identity matrix, then the system (3) becomes

$$\frac{dx}{dt} = -x + B\sigma(Wx + J). \tag{6}$$

We then introduce the stability property of system (6) in Theorem 2 below and its proof using the Lyapunov function can be found in ([11])).

**Theorem 2.** If $W$ is invertible, $(WB)$ is negative semi-definite, and $S^*$ is a singleton, then given initial state $x_0$, system (3) is globally exponentially stable. That is, there exist two positive constants $p \geq 1$ and $q > 0$ such that for any $x_0 \in R^n$ and $t \in [0, \infty)$

$$\|x(t) - x^*\| \leq p\|x_0 - x^*\| \exp(-qt), \tag{7}$$

where $x(t)$ is the solution of (6) and $x^*$ is an equilibrium point of (6) in $S^*$ with $A = I$. That is,

$$x^* = B\sigma(Wx^* + J). \tag{8}$$

The results of Theorem 1-2 provide some important characters of the solutions of system (3). It is known that the continuous-time model (3) and its numerical discretization of system (9) need not share the same dynamical behavior. However, the discrete-time system (9) of system (3) will inherit the same dynamics of system (3) when the step size is "small" (see [17]). Since the dynamical system like the IR spectral signal is digitalized, for practical purpose, we approximate system (3) by Euler's method to obtain the dynamics of a discrete-time RNN

$$x(t+1) = (I - hA)x(t) + hB\sigma(Wx(t) + J), \tag{9}$$

where $h$ is the step size of Euler's discretization, $A$ and $B$ are defined as in (3). The two systems (9) and (3) share the same dynamical behavior when $h \to 0$. In fact, the result in ([11]) shows that if $0 < h_i a_i < 2$ for $i = 1, 2, ..., n$, and if the connection weight matrix $W = [w_{ij}]$ satisfies the inequalities

$$\sum_{j=1}^{n} w_{ij} < \delta_i = \frac{1}{h_i b_i}(1 - |1 - h_i a_i|) \tag{10}$$

or

$$|w_{jj}| + \frac{1}{|h_i b_i|} \sum_{i=1, i \neq j} |h_i b_i w_{ij}| < \delta_i = \frac{1}{h_i b_i}(1 - |1 - h_i a_i|) \tag{11}$$

for each $i = 1, 2, ..., n$, then the discrete-time RNN model (9) is absolutely stable. We notice that inequality (10) implies that the solution space of the connection weight matrix $W$ forms $n$ open convex hyper cones in n-dimensional space. Moreover, $\delta_i = \frac{1}{h_i b_i}(1 - |1 - h_i a_i|) \to \infty$ as $h_i \to 0$. Since the discrete-time model (9) and the continuous-time model (1) share the same dynamical behavior as $h_i \to 0$, we can conclude that as $h_i \to 0$ both (9) and (1) are absolutely stable. It is known that for an absolutely stable neural network model, the system state will converge to one of the asymptotically stable equilibrium points regardless of the initial state ([11]). This result is very important in using system (9) for the data compression.

We will show how system (9) can be used as a suitable dynamical system to compress given signals in the next section.

# 3. Compression Techniques and Methodology

There are many different techniques available to compress a given signal ([2]) including the most well-known methods such as the Fast Fourier Transforms and the wavelet transforms. Both the Fast Fourier Transform and the wavelet transform methods are very popular and handy in signal/data compression. However, the intrinsic properties of the original signal have not been handled effectively. When a signal has some intrinsic properties, it follows some dynamical behavior locally. Hence, the dynamical system approach can aptly be employed for the signal compression.

To show why the RNN of the dynamical system approach can be used for signal compressions, let us consider a finite sequence generated by

$$z(t+1) = \mu z(t)(1 - z(t)) \quad \mu = 3.82. \tag{12}$$

Sequence $\{z(t)\}_{1 \leq t \leq m}$ is chaotic and its dynamics is well-understood. Given a signal $z(t)$, we can restore all the information of $z(t)$ by (12) by storing the parameters $\mu$, the initial state $z(1)$ and $m$. Thus, the finite sequence is compressed into three parameters $\mu, z(1)$ and $m$ only. It is unlikely that an arbitrary dynamical system is capable to approximate a given sequence or function over a finite interval. Therefore, we need a universal approximation theory. It was proved by Li ([6]) that, for an arbitrary finite sequence (signals) $\{z(t) \in R^n 1\}_{1 \leq t \leq m}$, there exists a discrete-time RNN system (9) with network size $N > n1$ for some positive integer $n1$ such that the first $n1$ output of $x(t)$ generates $z(t)$ accurately for $1 \leq t \leq m$. This universal approximation property of RNNs is the foundation to compress arbitrarily sequence or a function.

Before we proceed to discuss the methodology, the following definition will be established.

**Definition.** Dynamical system (9) is said to be *exactly capable* if there exist some neural parameters $W, J$ and $h$ such that the error between the system output $x(t)$ and the approximating sequence $z(t)$ is zero. If the least square error between the system output and the approximating sequence is less than a pre-condition tolerance, then the network is said to be *capable*.

For simplicity, we assume that matrices $A$ and $B$ of (9) are the identity matrices in which

$$x(t+1) = x(t) + h[-x(t) + \sigma(Wx(t) + J)]. \tag{13}$$

Now the task of compression of a given signal with the RNN system (13) approach can be carried out by the following steps (1)-(6):

(1) Segmentation. Given a signal $z(t)$ of length $m$ (or of $m$ data), $z(t)$ is partitioned into $n$ equal segments with the same length $p$, that is, $m = np$. Thus, we obtain a finite sequence $\{z_k(t)\}_{1 \leq k \leq n}$ of segments with length $p$ for each segment.

(2) Mean correction. For each segment $z_k(t), k = 1, 2, ..., n,$ we define

$$\hat{z}_k(t) = z_k(t) - \text{mean}(z_k(t)). \tag{14}$$

Then the new sequence $\{\hat{z}_k(t)\}$ has the property of $\text{mean}(\hat{z}_k(t)) = 0$ for each $k = 1, 2, ..., n$. This mean correction process to $z_k$ is important because we can later set $J = 0$ in the normalization process (see (4)). For simplicity, we replace $\hat{z}_k(t)$'s by $z_k(t)$'s in the remaining discussion.

(3) Discrete integration. Since inherent in the collection of data taken over time is some form of random variation, there exist methods for reducing or canceling the effect due to random variation. An often-used technique in data compression is "smoothing". This technique, when properly applied, reveals more clearly the underlying trend, seasonal and cyclic components. Better trend recognition can lead to more accurate signals. In this paper, instead of opting for a popular moving average smoothing technique, we develop a *double discrete integral method* to smoothen the given noisy signal $\{z_k(t)\}$. The procedures of the discrete double integral method are presented as follows:

For each $k = 1, 2, ..., n$, consider $\{z_k(1), z_k(2), ..., z_k(p)\}$ we perform the following mean zero discrete integration with scaling procedure:

$$
\begin{aligned}
(i)\quad & y_k(1) = 0, \\
& y_k(2) = y_k(1) + z_k(1), \\
& \dots \\
& y_k(t+1) = y_k(t) + z_k(t). \\
(ii)\quad & \phi_k(t) = y_k(t) - \text{mean}\{y_k(t)\}, \quad \text{then} \quad \sum_{1\le s\le p} \phi_k(s) = 0. \\
(iii)\quad & \text{Let} \quad \phi_{kmax} = \max_{1\le s\le p} \phi_k(s), \quad \phi_{kmin} = \min_{1\le s\le p} \phi_k(s), \\
& \phi\phi_{kmax} = \max\{|\phi_{kmax}|, |\phi_{kmin}|\}. \\
(iv)\quad & x_k(t) = \frac{x_k(t)}{\phi\phi_{kmax}}.
\end{aligned} \tag{15}
$$

Now $\{x_k(t)\}$ is a finite sequence of length $p+2$ with mean$\{x_k(t) = 0\}$ and $x_k(1) = x_k(p+2)$. The idea is that we can treat sequence $z_k(t)$ as the discrete derivative of another sequence $y_k(t)$ defined by iteratively in (i), or in other words, $y_k$ is the discrete integral of $z_k$. To assure the new sequence has zero mean, we set $\phi_k(t) = y_k(t) - \text{mean}(y_k(t))$. Then, the processes are repeated in (ii), now $x_k(t)$ is of length $p+2$. Although performing the double discrete integral method causes an increase in the storage space and the cost of computation when the original sequence is to be regenerated, yet it brings much more benefit in reducing the influence of noise upon data compression. We replace $x_k(t)$ by $z_k(t)$ and let $q = p+2$ for simplicity in the remaining discussion. Notice that $z_k(t)$ has leveling property $z_k(0) = z_k(q)$ for all $k$.

As R. W. Hamming observed: Digital filtering includes the process of smoothing, predicting, differentiating, integrating, separation of signals, and removal of noise from a signal ([15]). In effect, the double discrete integrating removes the noise around the trend and provides a smoother visual image of the trend. This method is simple, flexible and very easy to be implemented. Moreover, our claim of a better smoothing method is based on the following theoretical support and analysis.

Consider a continuous-time, $s$-periodic signal $f(t)$, then the N-harmonic Fourier series approximation can be written as

$$
f(t) = \sum_{k=0}^{N} a_k \cos(kt), \tag{16}
$$

where the Fourier cosine coefficients $a_1, a_2 ..., a_N$ are non-negative. In fact, given $\epsilon > 0$, there exists a positive integer $N$ and a Fourier cosine coefficients $a_1, a_2 ..., a_N$ such that

$$
\|f(t) - \sum_{k=0}^{N} a_k \cos(kt)\| < \epsilon. \tag{17}
$$

We call the N-harmonic Fourier series convergent to $f(t)$. To obtain a smooth curve we take integration both sides of (16) twice to obtain

$$g(t) = \int^t \int^y f(z)dzdy = \frac{a_0 t^2}{2} - \sum_{k=1}^{N} \frac{a_k}{k^2}\cos(kt). \tag{18}$$

By chosen $a_0 = 0$ and let $\hat{a}_k = -a_k$, we have

$$g(t) = \int^t \int^y f(z)dzdy = \sum_{k=1}^{N} \frac{\hat{a}_k}{k^2}\cos(kt). \tag{19}$$

By the Second Fundamental Theorem for integration, $g(t)$ is twice differentiable. Equally important, $g(t)$ is square integrable. Thus, the action of double integrating can be used to transform the "higher frequency" from the given signal to make the function smoother or even to emphasize certain informational components contained in the original signal. Setting $a_0 = 0$ in (18), we further guarantee the range of the signal can be re-scaling within [-1,1]. Observing (19) again, we notice that factor $\frac{1}{k^2}$ of $\bar{a}_k \cos(kt)$ converge to zero very fast as $k$ larger, which causes the effect of eliminating the higher frequency terms in (19). It provides another explanation for why the double discrete integral method removes the noise around the trend. Our experiments show that the double discrete integral method is enough for smoothness and handling trends in general, the triple or higher order integral method may not be necessary and is costly.

(4) Normalization. To normalize $z_k(t)$ within the range -1 to 1 (that is, within the range of $\sigma$) instead of 0 to 1, we re-scale the given signal $z_k(t)$ (see $(iii)$ and $(iv)$ in (15)). The parameters $\theta$ can be assigned to be zero vector to save memory space. Again, this setting has very little effect on the efficiency of data compression ([6]).

(5) Approximation. After normalizing, re-scaling and discrete integration, each $z_k(t)$ is a one dimensional sequence of length $q$, where $q = p + 2$ and $m = np$. Let $n$ be the number of neurons $x_1(t), x_2(t), ..., x_n(t)$ to be used in RNN system (13). The first neuron $x_1(t)$ with $t = 1, 2, ..., q$ is utilized to approximate the first subsequence that we assign the initial state equal to the first term of the subsequence, $x_1(1) = z(1)$. Continuing the similar procedures, we have $x_2(1) = z(q+1)$, $x_3(1) = z(2q+1)$, and $x_n(1) = z((n-1)q+1)$, respectively. Hence, $\{x_1(2), ..., x_1(q)\}, \{x_2(q+2), ..., x_2(2q)\}, ...$ and $\{x_n((n-1)q+2), ..., x_n(nq)\}$ can be generated via (13) with given values of the RNN parameters $h$ and $W$. By varying these RNN quantities systematically, we obtain the regenerated spectrum $X(t) = (x_1(t), ..., x_1(q), ..., x_n((n-1)q+1), ..., x_n(nq))$ to have the smallest discrepancies with respect to the original spectrum. In this case, the error may build up as the number of iterations increases. We may reduce the accumulated error by further dividing each subsequence of $z(t)$ into shorter subsequences, that is, *multiple segmentations*. Suppose $k$ multiple segmentations are used and the same parameters $W$ and $h$, the number of variables used is $(n+1)^2 + (k-1)n$.

(6) Optimization. As the network size $n$ is fixed, we need to optimize the parameters $W$ and $h$ so that we can minimize the least square error. That is, the discrepancy between $z(t)$ and $X(t) = [x_1(t), x_2(t), ..., x_n(t)]$ is within the tolerance. We define

the root mean square difference by

$$E = \sum_{i=1}^{n} \sum_{t=1}^{p} [\{x_i(t) - z((i-1)p + t)\}^2 / m]^{\frac{1}{2}}, \tag{20}$$

Equation (20) is used as an indicator of the discrepancy. Our goal is to adjust the RNN parameters so as to minimize $E$ of (20). Unlike the previous work of ([8]), we obtain the parameters by the SSSA (recall: SSSA represents the state space search learning algorithm) for RNNs ([11]). That is, for each $k$,

$$X_{k+1} = \alpha_{k+1}{}^* x + (1 - \alpha_{k+1}{}^*) y(W_{k+1}), \tag{21}$$

where

$$\begin{aligned} &X_k = y(W_{k+1}), \\ &0 < \alpha_k{}^* < 1, \quad \{\alpha_k{}^*\} \subset \{\alpha_i\}, \\ &\alpha_{k+1}{}^* = \alpha_k{}^* \quad \text{if} \quad E(W_{k+1}) < E(W_k), \\ &E(W_{k+1}) \leq E(W_k). \end{aligned} \tag{22}$$

Note that the learning algorithm SSSA is not based on the gradient method, and there is no computation of the partial derivatives along the target trajectory in the procedure. By searching in the neighborhood of the target trajectory in the state space for each iteration, the algorithm performs nonlinear optimization learning process and provides the best feasible solution for the nonlinear optimization problem. The convergence analysis shows that the network convergence to the desired solution is guaranteed, and the stability properties have been studied theoretically ([11]).

As the optimized RNN quantities together with $X(1)$ achieved as the compressed data set, it will be used to regenerate the original signal. For a network of $n$ neurons, the parameters we used include $W$ and $h$ together with the initial state $X(1)$, the total variables to be stored $n^2 + n + 1$. That is, the total number of data retained is $n^2 + n + 1$. Hence, if the network is capable for the task, the compression ratio is $m/[n^2 + n + 1]$. Notice that the vector $J$ is a zero vector as we have normalized the sequence, the stored parameters needed to be reduced to $n^2 + n + 1 + K$, where $K$ is the length of double integration.

The proposed approach is to demonstrate the power using recurrent neural dynamics for compression. As compression techniques use computation time in exchange for storage space, we may use various learning strategies to compute the parameters that lead to an acceptable error level. Some error analysis of this compression method can be found in ([7]).

# 4. Empirical Results

In this section we use 21 infra-red spectra as samples to illustrate our approach. The given signals are assumed to be finite and continuous. We use 21 noise free data sequences of infra-red spectrum which is $1,800$ points each. The size of the recurrent neural networks used is 9. Applying the method that we proposed in

Section 3, we found that all the empirical results are extremely promising. We summarize the least square errors of each sample in Tab. I. To avoid overstuffing the paper we show only one of the reconstructed signals, Benzacid, in Fig. 2 to illustrate our results. The top diagram of Fig. 2 is the original signal, the bottom diagram of Fig. 2 is the reconstructed signal using the RNN of the dynamical system approach. All the samples converge very fast (less than 5 seconds using MATLAB run in Window XP, Pentium 4, CPU 1.60 GHz).

| | *Sample* | *error* |
|---|---|---|
| 1 | benzacid | 0.014392 |
| 2 | brbenzen | 0.024306 |
| 3 | br1butan | 0.037459 |
| 4 | butane | 0.516144 |
| 5 | cl1butan | 0.013203 |
| 6 | cl2butan | 0.002340 |
| 7 | cl2pheno | 0.306229 |
| 8 | cl3pheno | 0.032144 |
| 9 | cl14benz | 0.048176 |
| 10 | cln24pben | 0.639169 |
| 11 | cln24benz | 0.109439 |
| 12 | cyclo5cl | 0.032655 |
| 13 | cyclo6br | 0.072161 |
| 14 | ibenzene | 0.04.9160 |
| 15 | i1butane | 0.514836 |
| 16 | n4benzcl | 0.112637 |
| 17 | no2pheno | 0.088191 |
| 18 | no3pheno | 0.069696 |
| 19 | no4pheno | 0.293058 |
| 20 | no4tulen | 0.029738 |
| 21 | toluene | 0.117859 |

**Tab. I** *Compression Errors of the 21 Samples.*

In fact, if the 7 or 8 neurons are used, the network gives an acceptable approximation, and less than a hundred parameters are stored as we use no hidden unit. The trade off is that the error decreases as the network size increases.

In these experiments, we took the first 1800 points and use no averaging technique. The error between the approximation and the original curve is depending on the different sample. The parameters we stored is $9^2 + 9 + 1 = 91$ plus the 4 neglected points.

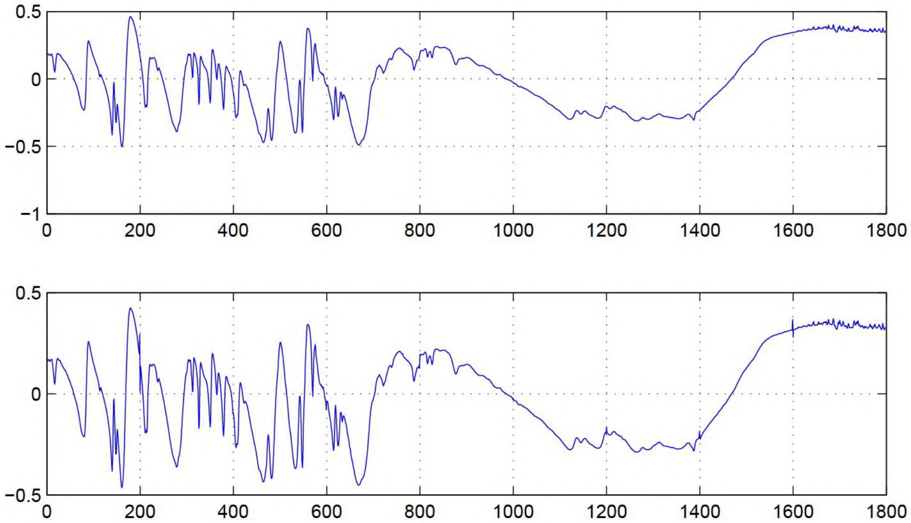


**Fig. 2** *Benzacid (Top: Original signal. Bottom: Reconstructed signal).*

From these experiments, we demonstrates some very promising results by using the RNN of the dynamical system approach for signal compression.

## 5. Concluding Remarks and Future Research Directions

We study the discrete-time RNN model and its application to compression of infra-red spectrum. The characters of the solutions of system (3) have been analyzed. Our results show that the discrete-time leaky integrator RNN model can be used to approximate the continuous-time model and inherit its dynamical characters if a proper step size is chosen. Having developed a novel discrete double integral smoothing technique together with the SSSA, our examples demonstrate some extremely promising results of compressing infra-red spectrum with the RNN of dynamical system approach. We are convinced that the simplicity of discrete double integral method together with the SSSA, especially with the algorithm given in this paper, would then favor our approach over other methods in dealing with the intrinsic properties of the original signals.

Moreover, from the data compression point of view, it is important for us to find a systematic way to determine a smallest network size and proper segmentation such that the error of the approximated sequence is within our tolerance, and the least parameters are used. On the other hand, after segmentation, the one dimensional trajectory $z_k(t)$ becomes a shorter trajectory in $R^n$. It is an interesting question for us to find an optimal network such that the intrinsic behavior is best represented, which may provide us some useful information or physical meaning for a given sequence of signals.

## Acknowledgement

(1) The authors would like to express their sincerely thanks to the referees who gave us valuable comments and suggestions.

(2) This research was supported by the Hong Kong Polytechnic University's Research Award G-YH05. We also would like to thank F. T. Chau and D. Mok of the Department of Applied Biology and Chemical Technology at The Hong Kong Polytechnic University for providing the data of the infra-red spectrum samples to conduct this research.

# References

[1] Hale J. A.: Ordinary differential equations, Wiley, New York, 1969.

[2] Hankerson D., Harris G. A., Johnson P. D.: Introduction to Information Theory and Data Compression, CRC Press, 1998.

[3] Horn R. A., Johnson C. A.: Matrix analysis, Cambridge University Press, Cambridge, UK, 1985.

[4] Graben P., Liebscher T., Kurths J.: Neural, Cognitive Modeling with Networks of Leaky Integrator Units, Understanding Complex Systems, Springer, 2008, pp. 195-223.

[5] Jin L., Nikifork N., Gupta M. M.: Absolute stability conditions for discrete-time neural networks, IEEE Trancs. Neural Networks, **5**, 6, 1994, pp. 954-964.

[6] Li L. K.: Approximation theory and recurrent networks, Proceedings of IJ CNN-Baltimore-92, **2**, 1992, pp. 266-271.

[7] Li L. K.: Data compression by recurrent neural dynamics, Proceedings of 1996 IEEE Region Ten Conference, **1**, 2, 1996, pp. 96-101.

[8] Li L. K., Chau F. T., Leung K. M.: Compression of Ultraviolet-visible Spectrum with recurrent neural network, Chemometrics and Intelligent Laboratory Systems, **52**, 2, 2000, pp. 135-143.

[9] Lapedes A., Farber R.: Nonlinear signal processing using neural networks, Prediction and system modeling, Technical Report LA-UR87-2662, Los Alamos National Laboratory, 1987. Cambridge University Press, Cambridge, UK, 1985.

[10] Li L. K., Shao S.: Dynamic properties of recurrent neural networks and its applications. International Journal of Pure and Applied Mathematics, **39**, 4, 2007, pp. 545-562.

[11] Li L. K., Shao S., Zheleva T.: A state space search algorithm and its application to learning short-term exchange rates. Appl. Math. Sciences, **2**, 35, 2008, pp. 1705-1728.

[12] Liang X. B., Wang J.: A recurrent neural network for nonlinear optimization with a continuously differentiable objective function and bound constraints, IEEE Trancs. Neural Networks, **11**, 6, 2000, pp. 1251-1262.

[13] Li L. K.: Fixed point analysis for discrete-time recurrent neural networks, Advances in Data Mining and Modeling (W. K. Ching, K. P. Ng, Eds), World Science, 2003, pp. 107-115.

[14] Perez-Ilzarbe M. J.: Convergence analysis of a discrete-time recurrent neural network to perform quadratic real optimization with bound constraints, IEEE Trancs. Neural Networks, **9**, 6, 1998, pp. 1344-1351.

[15] Tillson T.: Smoothing Techniques for More Accurate Signals, TASC, January, 1998, 57.

[16] Tuy H., Convex analysis and global optimization, Kluwer Academic Publisher, Dordrecht, 1998.

[17] Wang X., Blum E. K.: Discrete-time versus continuous-time models of neural networks, J. of Comp. Syst. Sci., **45**, 1992, pp. 1-19.

[18] Zaknich A., Attikiouzel Y.: Application of artificial neural networks to nonlinear signal processing, Computational Intelligence: A dynamic system perspective M. Palaniswami, et al., Eds., IEEE Press, 1995.