Linear and Nonlinear Analysis
Volume 4, Number 3, 2018, 459–472

Yokohama Publishers
ISSN 2188-8167

QUANTIZING MACKEY-GLASS-30 CHAOTIC SERIES AND DECODING SEQUENTIAL STATES BASED ON MANIFOLD NEURAL RECURSIONS

SUO-AN WANG, SIENDONG HUANG, AND JIANN-MING WU*

Abstract. This work explores state quantization of Mackey-Glass-30 (MG30) chaotic series and decoding sequential quantized states based on manifold neural recursions. Quantizing MG30 chaotic series attains sequential states. Each quantized state indicates the best fitting recursion among manifold recursions to a correspondent instance. Conversely, decoding a quantized state involves the process of directly applying the correspondent recursion to generate an instance for approximating the original instance. Embedded manifold recursions within the addressed chaotic series are extracted by learning a set-valued neural mapping and realized by a discrete state-regulated neural network. Numerical simulations verify effectiveness and reliability of learning a set-valued mapping for extracting manifold neural recursions and accuracy of decoding sequential quantized states of long-term MG30 chaotic series.

1. Introduction

Supervised learning of a state-regulated neural network (Wu, et al., 2016) has been successfully extended for constructing a discrete set-valued neural mapping by extracting many approximating functions simultaneously from paired training data. Typically a feed-forward neural network translates an input to an output through the input layer, hidden layers and the output layer for realizing an approximating function. An extension to a state-regulated neural network has been approached by simply concatenating an original input with a unitary vector of binary values, equivalently a regulating state in the input layer. Different settings to the regulating state in the input layer induce distinct network functions. A state-regulated neural network translates an input, x, to a set of values, each corresponding to one possible regulating state.

The famous backpropagation algorithm (Werbos, 1988) and the gradient-based approaches (Hagan & Menhaj, 1994; Nørgaard et al., 2000) are not feasible for learning a state-regulated neural network under assumption that training data is a mixture of many sets of paired predictors and targets, which are respectively oriented from joint single-valued functions. Following the mixture assumption, every paired predictor and target has an exclusive membership to one and only one joint function. All exclusive memberships of paired mixture data are unknown for optimizing adaptive network interconnections. The previous work (Wu, et al.,

2010 *Mathematics Subject Classification.* 26B40.

Key words and phrases. Manifold neural recursions, supervised learning, regulated-state neural neworks, set-valued neural mapping, Levenberg-Marquardt learning, mean field annealing, chaotic time series, state quantization, de-quantization.

*Correspondence.

2016) has shown effectiveness and reliability of a hybrid of mean field annealing and Levenberg-Marquardt methods for learning a set-valued neural mapping, which retrieves unknown exclusive memberships and optimizes network interconnections simultaneously. Learning a set-valued neural mapping is employed for extracting manifold recursions embedded within chaotic series in this work.

Mackey and Glass derived the following Mackey-Glass(MG) differential equation (Mackey & Glass, 1977), trying to describe the blood control system and finding out that it is chaos,

$$\frac{\partial x}{\partial t} = \frac{ax(t-\gamma)}{1+x^c(t-\gamma)} - bx(t), \tag{1.1}$$

where γ denotes time delay, and a, b, c are system parameters. The MG chaotic series oriented from equation 1.1 is unpredictable. Previous works (Moody & Darken,1989; Cechin, Pechmann & Oliveira, 2008; Mirzaee, 2009; Wu, Huang & Wu, 2014; Van Vaerenbergh, 2010) have applied learning a single-valued neural mapping for analyzing the MG17 chaotic series, with $\gamma = 17$, trying to characterize MG chaotic series using the sole extracted recursion. Whenever $\gamma = 30$, the sole extracted recursion (Wu, Huang & Wu, 2014) results in unacceptable performance for long-term characterization of MG30 chaotic series. This motivates the study of extracting manifold neural recursions embedded within MG30 chaotic series.

This work extracts manifold neural recursions by learning a set-valued neural mapping (Wu, et al., 2016), on the basis pioneering state quantization of MG30 chaotic series. Autoregressive sampling translates a segment of MG30 chaotic series to paired training data. Under the mixture assumption, learning a set-valued neural mapping subject to paired training data attains manifold neural recursions. The effectiveness and reliability of derived manifold neural recursions are verified for quantizing MG30 chaotic series and decoding quantized sequential states. In the testing phase, each paired predictor and target is quantized to a state corresponding to the best fitting recursion. Conversely, decoding sequential quantized states is expected to recover original MG30 chaotic series accurately.

This paper is organized as follows. The upcoming section introduces recursive functions, manifold neural recursions and generative models. The architecture of the RBF multilayer neural network and the set-valued RBF multilayer neural network for analysis of MG30 chaotic time series are given in section 3. The proposed quantizing and decoding process is given in section 4, where numerical experiments for performance evaluation of one-step look-ahead prediction, quantization and decoding are also given in section 4. Numerical simulations show learning set-valued mapping well reconstructing manifold recursions for MG30 chaotic series characterization. The conclusion is given in section 5.

2. Manifold recursions of generating MG30 chaotic series

A recursive function typically expresses a recurrent relation embedded with a time series. This work especially emphasizes manifold nonlinear recursive structures for characterizing MG30 chaotic series.

Figure 1 shows MG30 chaotic series generated by the fourth-order Runge-Kutta method. As an initial value problem, tracking the differential equation 1.1, subject

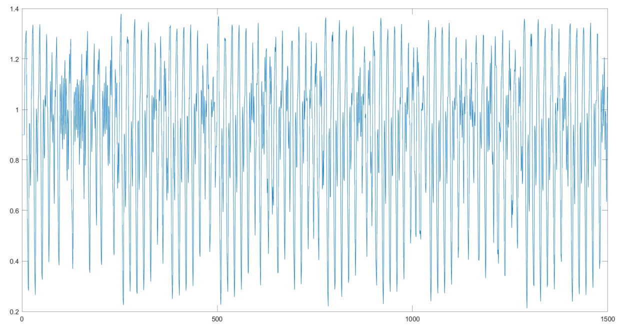

FIGURE 1. A Mackey-Glass chaotic time series with $\gamma = 30$

to an initial condition, is resolved by the fourth-order Runge-Kutta method, where parameters are given as $\gamma = 30, a = 0.2, c = 10, b = 0.1$. The attained chaotic series in discrete form is denoted by $X = \{x[i]\}_{i=1}^{n}$. This works focuses on Mackey-Glass time series with $\gamma = 30$ (MG30) shown in Figure 1. A recursive function expresses $x[i+1]$ as the function output in response to featured previous instances.

$$x[i+1] = f\left(x[i], x[i-L], x[i-2L], ..., x[i-\tau L] | \, \theta\right) \tag{2.1}$$

where $L \geq 1, \tau \geq 1$ and θ denotes the built-in parameters.

MG30 chaotic series is characterized by only one recursive structure 2.1 in previous works (Moody & Darken,1989; Cechin, Pechmann & Oliveira, 2008; Mirzaee, 2009; Wu, Huang & Wu, 2014; Van Vaerenbergh, 2010), where for data driven learning $\{x[t]\}_t$ is transformed into paired training data, denoted by $\{((x[i], x[i-L], ..., x[i-L\tau]), x[i+1])\}_{i=L\tau+1}^{n}$, further by $\{(\tilde{x}[t], y[t]\}_{t=1}^{T}$ where $\tilde{x}[t] = (x[t-1], x[t-L-1], ..., x[t-L\tau-1])$, $y[t] = x[t+1]$, $T = n - L\tau$ and L is time delay. The architecture of a multilayer neural network of radial basis functions (Moody & Darken, 1989) is employed to emulate f and the Levenberg-Marquardt method (Mirzaee, 2009) is applied to optimize adaptable interconnections in θ. However, experiment results of learning a single recursive structure subject to paired training data oriented from MG30 chaotic series still show space for further improvement due to unacceptable errors quickly accumulated for long-term prediction (Wu, Huang & Wu, 2014). This paper releases the constraint of unitary recursion, pioneering learning a set-valued neural mapping (Wu et al., 2016) for extracting manifold neural recursions of characterizing MG30 chaotic series.

Consider a generative model that is composed of manifold neural recursive structures for chaotic series generation in Figure 2. It simply generates a temporal instance, $x[t]$, each time by randomly choosing one of $K (K \geq 2)$ non-linear recursive functions according to a set of prior probabilities. By the stochastic process, each temporal instance $x[t]$ is exactly oriented from a recursion and possesses its exclusive membership, $\delta[t]$, to K joint recursions, where $\delta[t] \in \{e_i\}_i$ and e_i denotes a unitary vector of K bits with the i^{th} bit one and others zero. All exclusive memberships of generated instances, $\delta = \{\delta[t]\}_t$, are missed under the mixture assumption.

The generative model introduces manifold recursions. This work applies learning a set-valued neural mapping (Wu et al., 2016) for reconstructing the model. Learning a set-valued neural mapping subject to paired training data in this work finds out that MG30 chaotic series is actually embedded with manifold neural recursive structures, where K equals to two.

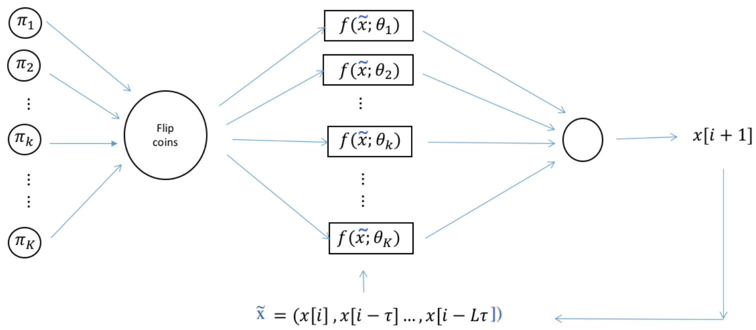

FIGURE 2. A simple generative model

Under the assumption of manifold neural recursive structures in MG30 time series data, paired training data oriented from the i^{th} joint recursion are denoted by $S_k = \{(\tilde{x}[t], y_i[t]) | \delta[t] = e_i, y_i[t] = y[t]\}_t$, where $i = 1, ..., K$. Therefore, based on multiple nonlinear recursive structures of generating the MG30 time series, joint paired training data can be expressed as a union of all S_k, denoted by $\bigcup_i \{(\tilde{x}[t], y_i[t])\}_t$, where y_i denoted the output of the i^{th} non-linear recursive structure.

To resolve the mixture problem, set-valued mapping analysis (Wu et al., 2016) will help reconstructing manifold neural recursive structures and retrieve exclusive memberships of paired training data to joined neural recursive structures. Furthermore, retrieved exclusive memberships allow one to analyze changes between states and determine the state-transient probability.

With transient probabilities a Markov generative model, as shown in figure 3, can be further derived to characterize generation of chaotic series based on manifold neural recursions. Discrete set-valued analysis is capable of retrieving missed exclusive memberships underlying paired training data as well as optimizing manifold neural recursions. The total number of changes between any two consecutive exclusive memberships can be determined and normalized for calculating transition

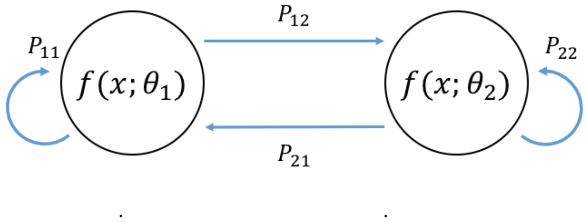

FIGURE 3. A Markov generative model

probabilities among states. Therefore, the transient probability matrix of a Markov generative model is obtained and defined as follows:

$$P_1 = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix}$$

Let 0 and 1 respectively denote two-alternative states of generating chaotic series by two non-linear recursive functions. A second-order Markov generative model, as shown in figure 4, could be also a possible choice for characterizing generation of MG30 time series. The transient probability matrix for a second-order Markov generative model is defined as P_3 and can be obtained by scanning retrieved consecutive exclusive memberships of training data,

$$P_3 = \begin{array}{c} \\ 00 \\ 01 \\ 10 \\ 11 \end{array} \!\!\begin{array}{c} \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{pmatrix} p_{00,00} & p_{00,01} & p_{00,10} & p_{00,11} \\ p_{01,00} & p_{01,01} & p_{01,10} & p_{01,11} \\ p_{10,00} & p_{10,01} & p_{10,10} & p_{10,11} \\ p_{11,00} & p_{11,01} & p_{11,10} & p_{11,11} \end{pmatrix} \end{array}$$

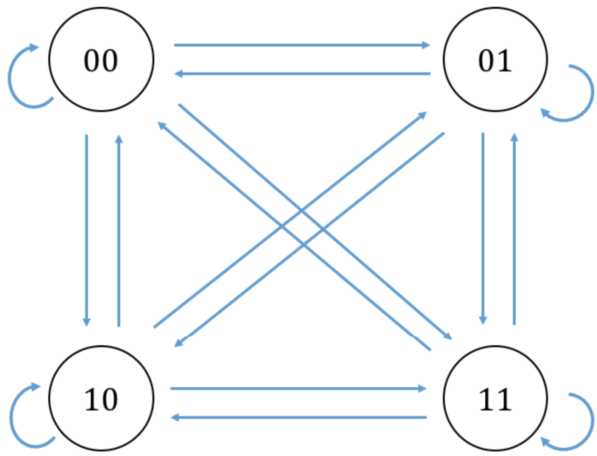

FIGURE 4. A second-ordered Markov generative model

In P_3, 00, 01, 10, and 11 enumerate four second-order states and each entry denotes the probability of a transition among second-ordered states. The transient probability matrix as well as a state-regulated multilayer neural network of radial basis functions constitute a second-order Markov generative model that could be employed for chaotic series identification. Manifold non-linear recursive structures and the transient probability matrix constitute a mixed stochastic and deterministic model for identification of chaotic time series.

3. Architecture and learning of a set-valued neural mapping

A multilayer neural network that is composed of radial basis functions has been extensively employed for translating high dimensional inputs to the network output by non-linear transformation through the hidden layer and posterior linear projection. The learning process involves optimizing radial basis functions in the hidden layer and adaptable posterior interconnections. The network can be mathematically expressed as $y = F(x|\theta)$ where θ collects all adaptive interconnections.

Paired training data, when consisting of multiple input elements and single output element, constrains a multilayer neural network for MISO (multiple inputs and single output) transformation. In this work, paired training data is assumed as a mixture of multiple sets of paired MISO training data, respectively oriented from multiple single-valued functions. Subject to this type of mixture data, supervised learning of a multilayer neural network, which possesses input units identical to the dimension of predictors, is unable to figure out original MISO functions that generate multiple sets of paired MISO data due to unknown exclusive memberships. Set-valued mapping analysis in the previous work (Wu et al., 2016) has been shown effective for retrieving unknown exclusive memberships and deriving the set-valued mapping for reconstructing joined MISO functions.

Set-valued mapping analysis is based on learning a state-regulated neural network, whose input units receive elements of predictors as well as those of a regulating state. A state-regulated neural network is with multiple elements in the input layer and a sole element in the output layer. It is translated to an equivalent set-valued neural network for MIMO transformation in this work. The obtained set-valued neural network possesses input units in number identical to the dimension of predictors, translating a common predictor to a set of distinct values in the output layer. It inherits multilayer neural organization with shared neurons in the hidden layers, avoiding redundancy for robust internal representations. Set-valued mapping analysis finds exclusive memberships as well as single valued mapping functions subject to a mixture of multiple sets of paired MISO training data. Different output units in the constructed set-valued neural network share hidden units. The idea of sharing hidden units in a set-valued neural network fits requirement of extremely high utilization of neurons. The organized set-valued neural network in this work possesses multiple output units, sharing neurons in the hidden layer. In comparison with the architecture of multiple single-valued neural networks, the set-valued neural network utilizes neurons more efficiently in the hidden layer. A hybrid of mean field annealing and Levenberg-Marquardt methods has been proposed to learn adaptive interconnections for set-valued mapping analysis.

3.1. **A multilayer neural network.** The authors in (Moody & Darken, 1989) pioneered learning an RBF multilayer neural network for approximating the nonlinear recursive function embedded within Mackey-Glass chaotic series based on the gradient descent method. An RBF multilayer neural network, as shown in figure 5, performs a single-valued function,

$$G(\tilde{x}|\theta) \quad = \quad r_0 + \sum_{m=1}^{M} r_m exp\left(\frac{\|\tilde{x}-\mu_m\|^2}{-2\sigma_m^2}\right) \tag{3.1}$$

where the vector μ_m denotes a local center, σ_m^2 denotes the variance and r_m denotes a posterior weight. Adaptable built-in parameters are collectively represented by $\theta = \{\mu_1, ..., \mu_M, \sigma_1, ..., \sigma_M, r_0, r_1, ..., r_M\}$

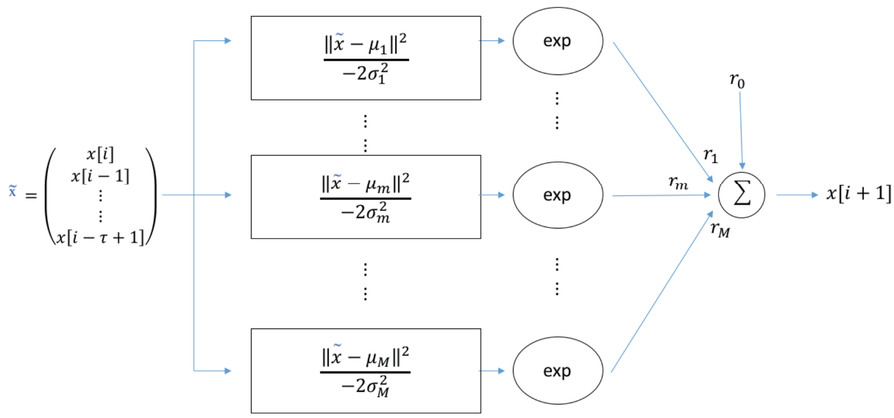

FIGURE 5. An RBF multilayer neural network model with L=1

The architecture of RBF is not enough to reconstruct multiple recursive structures underlying a mixture of multiple paired MISO datasets.

3.2. **A set-valued neural network.** An set-valued neural network is oriented from a state-regulated neural network. For set-valued mapping (Wu et al., 2016), the input of an RBF neural network is extended to recruit a discrete regulating state δ, which is a unitary vector of binary bits for indicating different operating modes. With a fixed regulating state, $\delta = e_k$, a state-regulated neural network performs the following single-valued function,

$$\begin{aligned} y_k(\tilde{x}|\theta) &= F(\tilde{x}, \delta = e_k|\theta) \\ &= r_0 + \sum_{m=1}^{M} r_m exp\left(\frac{\|\tilde{x} - \mu_m\|^2 + \|e_k - a_m\|^2}{-2\sigma_m^2}\right) \\ &= w_{0k} + \sum_{m=1}^{M} w_{mk} exp\left(\frac{\|\tilde{x} - \mu_m\|^2}{-2\sigma_m^2}\right) \end{aligned} \tag{3.2}$$

where $w_{0k} = r_0$, $w_{mk} = r_m exp\left(\frac{a_{mk}}{\sigma_m^2}\right) exp\left(\frac{1+\|a_{mk}\|^2}{-2\sigma_m^2}\right)$

and the vector $\binom{\mu_m}{a_m}$ denotes a center, σ_m^2 denotes the variance and w_{mk} denotes a posterior weight, and e_k denotes a unitary of binary bits with the k^{th} bits one and others zero. The obtained set-valued neural network is composed of multiple output units in response to $\tilde{x}$ in figure 6, performing MIMO (multiple inputs and multiple outputs) transformation.

Supervised learning of a set-valued RBF multilayer neural network involves a mixed integer programming. Adding a state-regulating variable $\delta \subseteq \{e_1, ..., e_K\}$ in the input layer induces a state-regulated neural network. Multiple outputs of the network are expressed by $F_k(\tilde{x}|\theta) = F(\tilde{x}, \delta = e_k|\theta)$, where F_k is the kth non-linear recursive function, as shown figure 6.

As stated previously, paired training data are a mixture of input-output samples from multiple single-valued mappings. A hybrid of Levenberg-Marquardt and

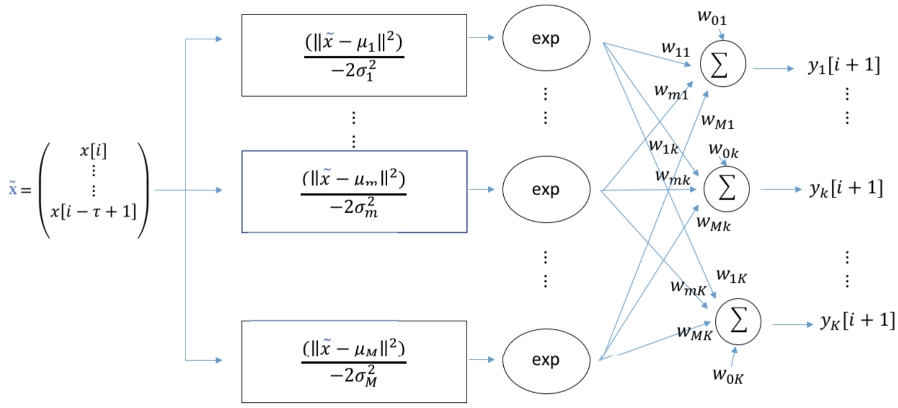

FIGURE 6. A set-valued RBF multilayer neural network

mean field annealing methods has been derived for learning a state-regulated RBF neural network subject to the mixture-type paired MISO training data. Let $z[t] = (\tilde{x}[t] \quad \delta[t])$ denote a result of concatenating $\tilde{x}[t]$ with $\delta[t] = (\delta_1[t], ..., \delta_k[t], ..., \delta_K[t])^T$. Hence, the new paired dataset with unknown state-regulating variables will be $S = \{(z[t], y[t])\}_t$. Let $E(\theta, \Lambda)$ quantify the objective of learning a state-regulated neural network.

$$
\begin{aligned}
E(\theta, \Lambda) &= \frac{1}{N} \sum_t \|y[t] - F(z[t]|\theta)\|^2 \\
&= \frac{1}{N} \sum_t \|y[t] - F(\tilde{x}[t], \delta[t]|\theta)\|^2
\end{aligned} \tag{3.3}
$$

where Λ denotes a collection of all membership vectors. Each $\delta[t]$ is a unitary vector of K binary bits, indicating the source dataset from which paired predictor and target $(\tilde{x}[t], y[t])$ is oriented. For learning a state-regulated neural network, $\delta[t]$ is indeed unknown for all t and considered as a random variable in the previous work (Wu et al., 2016). Learning a state-regulated neural network is expected to transform mixture-type paired training data into K disjoint paired MISO datasets according to the minimizer of $E(\theta, \Lambda)$. A a hybrid of Levenberg-Marquardt (LM) and mean field annealing methods has been proposed for the purpose (Wu et al., 2016). The learning algorithm of a state-regulated RBF multilayer neural network is reviewed as the following step-wise procedure.

Step1: Input $(\tilde{x}[t], y[t])$ for all t and initialize θ, β

Step2: Determine the expectation, $\langle \delta_k[t] \rangle$, by mean field equations for all t, k

$u_{tk} = \frac{\partial E(\theta, \langle \Lambda \rangle)}{\partial \langle \delta_k[t] \rangle} = \|y[t] - F(\tilde{x}[t], e_k|\theta)\|^2$

$\langle \delta_k[t] \rangle = \frac{\exp(\beta u_{tk})}{\sum_h \exp(\beta u_{th})}$

Step3: Minimize

$E(\theta, \langle \Lambda \rangle) = \frac{1}{N} \sum_t \|y[t] - F(\tilde{x}[t], \langle \delta[t] \rangle \, |\theta)\|$

with respect to θ by the LM method

Step4: Increase β by dividing a near one annealing factor and calculate stability ξ
Step5: If the stability ξ does not exceed a pre-defined threshold, repeat Step 2-4 otherwise halt.

$\delta_k[t]$ is regarded as a binary random variable. Its expectation is determined at step 2, where β regulates the randomness. When β is scheduled sufficiently high, the expectation of each $\delta_k[t]$ approaches a binary value and the stability ξ, measuring the mean of $\xi_t = \sum_k \langle \delta_k[t] \rangle^2$ over t reaches its maximal value that exceeds a predetermined threshold eventually. Given all $\langle \delta_k[t] \rangle$ under fixed β, learning a state-regulated neural network is approached by the Levenberg-Marquardt method at step 3. The objective E is a result of substituting $\langle \delta_k[t] \rangle$ to $\delta_k[t]$ and is differential with respect to bulit-in parameters of radial basis functions.

4. Quantizing MG30 chaotic series and de-quantizing sequential states

The source of MG30 chaotic series for numerical simulations here are the same as in (Wu, Huang & Wu, 2014). The MG30 chaotic series is generated by the fourth-order Runge-Kutta method of tracking the differential equation 1.1 with parameters same as for generating series in figure 1. Following non-linear recursive relation 2.1, MG30 chaotic series is transformed into paired MISO data denoted by $\{((x[i], x[i-L], ..., x[i-L\tau]), x[i+1])\}_{i=\tau}^{N}$, which is further represented by $\{(\tilde{x}[t], y[t])\}_{t=1}^{T}$ where $T = N - \tau + 1$, L denotes time delay and $x[i]$ is set to zero for negative i. As described in section 2, the paired MISO training dataset is a union of $\bigcup_k S_k$, subject to which learning a set-valued RBF neural network, denoted by set-valued LM-RBF, by a hybrid of Levenberg-Marquardt and mean field annealing methods retrieves unknown exclusive memberships as well as optimal parameters θ_{opt}.

$F(z|\theta_{opt})$ denotes the obtained set-valued mapping, which is further employed for quantizing each paired data $(\tilde{x}[t], y[t]) = ((x[t-1], x[t-L-1], ..., x[i-L\tau-1]), x[t])$ to a K-state code. The encoder first determines K possible responses by the set-valued mapping,

$$\hat{y}_k = F(\tilde{x}[t], \delta = e_k|\theta_{opt}) \tag{4.1}$$

where k runs from 1 to K, and sets the discrete code $\zeta[t]$ to e_{k^*}, where

$$k^* = arg \min_k \|\hat{y}_k - y[t]\| \tag{4.2}$$

Conversely, a quantized state $\zeta[t] = e_{k^*}$ is decoded to an instance that approximates $x[t]$ for given $\tilde{x}[t]$,

$$\hat{y} = F(\tilde{x}, \zeta = e_k^*|\theta_{opt}) \tag{4.3}$$

where $\tilde{x}[t]$ has been replaced with $\tilde{x}$.

Numerical simulations partition MG30 chaotic series of the first 1500 instances to two non-overlapping segments, denoted by seg_1 and seg_2, respectively containing 800 and 700 instances. By setting $\tau = 4, L = 6$, data preparing translates the first segment to paired training data for reconstructing the embedded set-valued mapping and the second segment to paired testing data for generalization.

The hybrid learning approach in the previous section is employed to analyze paired training data. The obtained set-valued mapping $F(z|\theta_{opt})$ is a model with

$K = 2$. Each pair, $(\tilde{x}[t], y[t])$, is related to an exclusive membership, $\delta[t]$, which is either e_1 or e_2. The approximation of $y[t]$ is a result of substituting the concatenation of $\tilde{x}[t]$ and $\delta[t]$ to the network function. This approximation of $\hat{y}$ to $y[t]$ subject to given $\tilde{x}[t]$ can be recalculated by equations (4.1)-(4.3) without $\delta[t]$. The mean square approximating error is expressed by

$$mse_s = \frac{1}{|s|} \sum_t (y[t] - \hat{y}[t])^2$$

where s collects paired data for training or testing, $y[t]$ is the real output, $\hat{y}[t]$ is a result of substituting $\tilde{x}[t]$ to equations (4.1)-(4.3). The mean square approximating error over paired training data measures 5.3×10^{-5}, which has significantly improved the result of learning an RBF network simply by the LM method for function approximation, as shown in table 1. Due to high reliability of the hybrid learning approach, numerical results in table 1 omit extremely low variances of mean square errors.

	mse_{seg_1}	mse_{seg_2}	quantizing and decoding: mse_{seg_2}
LM-RBF	0.001503	0.001313	0.177402
Set-Valued LM-RBF	0.000053	0.000192	0.005156

Table 1. Mean square error for MG30 chaotic series

The set-valued mapping derived from paired training data is expected to fit paired testing data oriented from seg_2, which is unknown to the hybrid learning approach. Similarly, the target $y[t]$ subject to $\tilde{x}[t]$ in paired testing data is approximated by $\hat{y}$ derived by equations (4.1)-(4.3). The mean square error of approximating the target $y[t]$ subject to $\tilde{x}[t]$ over all paired testing data measures 1.92×10^{-4}, which has significantly improved the result of learning an RBF network simply by the LM method, as shown in table 1. The first column in table 1 shows the mean square training error. In mse_s, $\hat{y}[t]$ is a result of substituting $\tilde{x}[t]$ to equations (4.1)-(4.3). The second column shows the mean square testing error, where paired data is oriented from the segment seg_2. Similarly, $\hat{y}[t]$ is a result of substituting $\tilde{x}[t]$ to equations (4.1)-(4.3). In figure 7-8, a soild line denotes real instances of MG30 chaotic series and points are the results of substituting $\tilde{x}[t]$ to equations (4.1)-(4.3) where * and o respectively denote outputs of two different non-linear recursive functions.

Furthermore the derived set-valued mapping $F(z|\theta_{opt})$ is verified for quantizing MG30 chaotic series and decoding sequential states. It is assumed that both transmitter and receiver have been equipped with $F(z|\theta_{opt})$ derived from paired training data oriented from the segment seg_1. Now the transmitter is expected to quantize instances in seg_2 to binary states. The transmission of sequential quantized states of instances in seg_2 from the transmitter to the receiver is through a link of binary values, where the length of quantized sequential states equals $|seg_2|$. The transmitter substitutes each $\tilde{x}[t]$, which is prepared for quantizing a correspondent instance in seg_2, to equations (4.1)-(4.2) and obtains a quantized state, $\zeta[t] \in \{e_1, e_2\}$. Sequential quantized states are transmitted to the receiver for further decoding. Since

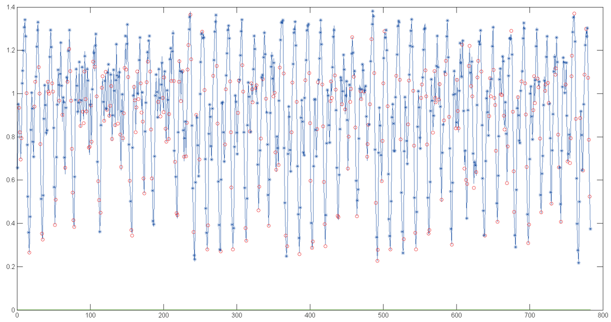

FIGURE 7. Training result

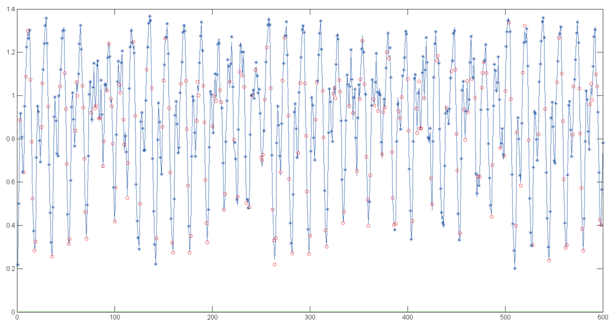

FIGURE 8. One-step look-ahead result

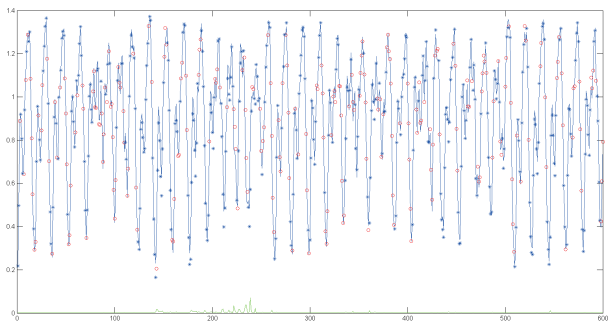

FIGURE 9. A result of decoding sequential quantized states of instances in seg_2

the receiver is not given instances in seg_2, $\tilde{x}[t]$ for decoding $\zeta[t]$ is not actually a collection of real instances in seg_2, but containing decoded results. The decoding process substitutes decoded instances, $\tilde{x} = (\hat{y}[t-1], \hat{y}[t-L-1], ..., \hat{y}[i-L\tau-1])$, as well as a quantized state $\zeta[t]$ to equation (4.3) to generate $\hat{y}[t]$. By the process, sequential quantized states are translated to $\{\hat{y}[t]\}_t$, in figure 9, for approximating seg_2. The mean square error of approximating seg_2 by $\{\hat{y}[t]\}_t$ is listed in the third column of table 1.

5. CONCLUSIONS

This work presents learning a set-valued multilayer neural network for extracting manifold recursions underlying MG30 chaotic series. The derived manifold neural recursions have been successfully applied for quantizing MG30 chaotic series and decoding sequential quantized states.

Numerical simulations verify existence of embedded manifold neural recursions within MG30 chaotic series. The analyzed MG30 chaotic segment is indeed embedded with manifold neural recursions of K equaling 2. Manifold neural recursions have shown acceptable mean square errors for one-step look-ahead prediction in comparison with a single neural recursion. This work also presents a combination of manifold neural recursive structures and the Markov generative model, which respectively characterize deterministic and stochastic parts of generation of MG30 chaotic series.

Learning a state-regulated neural network subject to mixture-type paired training data retrieves exclusive memberships and optimizes adaptive interconnections for set-valued mapping. The optimized state-regulated neural network has been shown equivalent to a set-valued neural network with MIMO neural organization. The equivalent set-valued neural network translates a predictor to multiple outputs, performing essential transformation for quantizing MG30 chaotic series to discrete states. The quantized sequential states are binary series in length identical to original instances. Decoding sequential quantized states recovers original instances accurately. Numerical simulations have shown that decoded instances well approximate original instances, where mean square approximating error has been significantly reduced by the proposed quantizing and decoding process of MG30 chaotic series.

References

[1] A. L. Cechin, D. R. Pechmann, L. P. L. de Oliveira, *Optimizing Markovian modeling of chaotic systems with recurrent neural networks*, Chaos Solitons Fractals **37** (2008), 1317–1327.

[2] M. T. Hagan and M. B. Menhaj, *Training feedforward networks with the marquardt algorithm*, IEEE Transactions on Neural Network **5** (1994), 989–993.

[3] M C Mackey and L Glass. *Oscillation and chaos in physiological control systems*, Science **197** (1977), :287–289.

[4] H. Mirzaee, *Long-term prediction of chaotic time series with multi-step prediction horizons by a neural network with LevenbergMarquardt learning algorithm*, Chaos Solitons Fractals **41** (2009), 1975–1979.

[5] J. Moody and J. Darken, *Fast learning in networks of locally-tuned processing units*, Neural Comput. **1** (1989), 281–294.

[6] M. Nørgaard, O. Ravn, N. K. Poulsen an L. K. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems*, Springer-Verlag, London, UK, 2000.

[7] S. Van Vaerenbergh, *Kernel methods for nonlinear identification, equalization and separation of signals*, PhD thesis, University of Cantabria, February 2010. Software available at https://github.com/steven2358/kmbox.

[8] P. J. Werbos, *Backpropagation: past and the future, in: Neural Networks*, IEEE International Conference, vol. 1, 1988, pp. 343–353.
[9] J.-M. Wu, C.-C. Wu and Y.-L. Lin, J.-C. Chen, *Set-valued functional neural mapping and inverse system approximation*, Neurocomputing **173** (2016), 1276–1278.
[10] J.-M. Wu, C.-W. Huang and C.-C. Wu. *Annealed cooperative competitive learning of mahalanobis-nrbf neural modules for nonlinear and chaotic differential function approximation*, Neurocomputing **136** (2014), 56–70.

Manuscript received 3 October 2018

Suo-An Wang
Department of Applied Mathematics, National Dong Hwa University, Shoufeng, Hualien 974, Taiwan
E-mail address: parkourmass@gmail.com

Siendong Huang
Department of Applied Mathematics, National Dong Hwa University, Shoufeng, Hualien 974, Taiwan
E-mail address: sdhuang@gms.ndhu.edu.tw

Jiann-Ming Wu
Department of Applied Mathematics, National Dong Hwa University, Shoufeng, Hualien 974, Taiwan
E-mail address: jmwu@gms.ndhu.edu.tw; jmwu@livemail.tw