

Dynamic Nonlinear System Identification Using a Wiener-Type Recurrent Network with OKID Algorithm

JEEN-SHING WANG AND YU-LIANG HSU

Department of Electrical Engineering

National Cheng Kung University

Tainan, 701 Taiwan

This paper presents a novel Wiener-type recurrent neural network with the observer/Kalman filter identification (OKID) algorithm for unknown dynamic nonlinear system identification. The proposed Wiener-type recurrent network resembles the conventional Wiener model that consists of a dynamic linear subsystem cascaded with a static nonlinear subsystem. The novelties of our approach include: (1) the realization of a conventional Wiener model into a simple connectionist recurrent network whose output can be expressed by a nonlinear transformation of a linear state-space equation; (2) the overall network structure can be determined by the OKID algorithm effectively using only the input-output measurements; and (3) the proposed network is capable of accurately identifying nonlinear dynamic systems using fewer parameters. Computer simulations and comparisons with some existing recurrent networks and learning algorithms have successfully confirmed the effectiveness and superiority of the proposed Wiener-type network with the OKID algorithm.

Keywords: Wiener models, recurrent neural networks, observer/Kalman filter identification, minimal state-space model realization, dynamic system identification

1. INTRODUCTION

Dynamic system identification is the model estimation process of capturing system dynamics using measured input-output data, which is a very important prerequisite for analysis and controller design in most control applications [15]. A good model representation offers a good capability of representing different systems in terms of modeling accuracy and structure compactness. Therefore, to obtain a best fit for data with few parameters has become a top priority in the selection of model representation. Much of the literature has widely described model representations for nonlinear system identification problems [16]. Among the diverse model representations, the block-oriented (BO) models that are composed of dynamic linear blocks and static nonlinear blocks possess the flexibility of selecting blocks to represent the features of a given unknown system. The choices of different linear and nonlinear blocks result in various structures. One of the notable nonlinear models is the Wiener model, consisting of a dynamic linear part cascaded with a static nonlinear component [1]. Wiener models have been widely used in industry such as in polymerization reactor control, fluid flow control, pH neutralization control, and identification of nonlinear biological systems [11]. The advantages of Wiener models include: (1) the complexity of system dynamics is contained in the linear subsystem whereas the complexity of nonlinearity only in the static subsystem and (2) the overall output of Wiener models can be written analytically as a kernel function ex-

Received April 3, 2007; revised August 10 & October 12, 2007; accepted October 25, 2007.

Communicated by Chin-Teng Lin.

pansion [2]. Another effective model representation is neural networks that have been treated as a powerful model for nonlinear system identification problems. To name a few, Kalinli and Sagioglu [12] presented a new recurrent neural network named ENEM (Elman network with embedded memory) composed of Elman network and NARX neural network for dynamic nonlinear system identification. Lin [13] proposed a wavelet neural network with an online partition method and the gradient descent method to identify the nonlinear dynamic system. Lin and Xu [14] designed neuro-fuzzy systems with a modified modified variable-length genetic algorithm to solve identification and control problems. Wang and Chen [4, 18] presented a Hammerstein-type recurrent neural network with a self-construction algorithm to identify nonlinear dynamic systems. Recently, many researchers have integrated neural networks with some linear systems to form Wiener models. To name a few, AI-Duwaish *et al.* [1] used a linear autoregressive moving average (ARMA) model to represent the dynamic linear block and a multilayer feedforward neural network to model the static nonlinear element. Chen *et al.* [2] utilized the Wiener model to identify chaotic systems. The dynamic element was represented by a simple linear plant, and the static nonlinear element was represented as a feedforward neural network.

With a priori knowledge of the given nonlinear system, conventional system identification approaches that use either frequency domain or time domain methods can explicitly approximate and simplify the nonlinear system dynamics in terms of a linear model. Among these approaches, fast Fourier transforms, maximum likelihood estimation, and least squares are three representative methods. In the early 1960s, the problem of realization of state-space representations using input-output descriptions has received considerable attention, which resulted in a wide variety of algorithms to solve the problem. To name a few, Ho and Kalman [5] introduced an important principle of minimum realization theory based on the sequence of Markov parameters of the system. The realization of a state-space model is based on the Hankel matrix constructed by Markov parameters. The eigensystem realization algorithm (ERA), proposed by Juang and Pappa [6], is a powerful identification method to deal with the realization problem with noisy input-output measurements. The ERA can provide accurate parameter estimation and determine the system order for multivariable linear state-space models. Later on, Juang *et al.* [7] extended the ERA algorithm to the observer/Kalman filter identification (OKID) algorithm that is a non-recursive least-squares approach to compute the observer Markov parameters. With the concepts of stochastic estimation and deterministic Markov parameter identification techniques, the OKID is capable of generating a discrete state-space locally linear model representation of the nonlinear system. The aforementioned approaches are very effective in determining the system order and provide a powerful tool to transfer the relations of input-output measurements into a state-space representation. Much of the literature has used the OKID algorithm to realize the state-space models from the input-output data of nonlinear systems. To name a few, Juang and Phan [8] presented the backward OKID to identify an autoregressive with exogenous input (ARX) model and an autoregressive moving average with exogenous input (ARMAX) model. Clarke and Sun [3] realized the minimal state-space model of a nonlinear helicopter via the OKID algorithm. They also showed that the OKID approach can produce more effective models than the conventional linear models generated by the small perturbation approach.

In this paper, we identify dynamic nonlinear systems by two procedures: model selection and parameter estimation. First, we propose a Wiener-type recurrent neural network that expresses the nonlinear dynamics of a given system via a nonlinear transformation of a linear state-space equation. The linear state-space equation is realized by a recurrent neural network that embeds the recurrence of internal state variables in the feedback connections and delay (memory) elements. The advantages of this network include: (1) the two subsystems of a conventional Wiener model are integrated into a single network whose output is expressed by a nonlinear transformation of a linear state-space equation; (2) the characteristics of the network can be analyzed by its associated state-space equation using the well-developed theory of linear systems; and (3) determining the network structure is equivalent to finding the number of state variables (system order) of the unknown system. Our idea for the structure identification and parameter estimation is to transfer a complex nonlinear problem to a simple linear one by eliminating the effect of the static nonlinear subsystem and then utilize the OKID algorithm to estimate the system order and parameters of the linear subsystem of the proposed recurrent network. Computer simulations of nonlinear dynamic applications have successfully validated the effectiveness of the proposed network and algorithm in constructing a quality network with satisfactory performance.

The rest of this paper is organized as follows. In section 2, we introduce the proposed Wiener-type recurrent network. The OKID-based identification algorithm for establishing a parsimonious network with satisfactory parameter estimation is presented in section 3. Section 4 provides computer simulations of dynamic system identification applications to validate the effectiveness of the proposed approach. Finally, the conclusions are presented in section 5.

2. STRUCTURE OF WIENER-TYPE RECURRENT NEURAL NETWORK

In this paper, we realize a conventional Wiener model by a novel recurrent neural network structure. The Wiener model shown in Fig. 1 consists of a dynamic linear subsystem cascaded with a nonlinear static subsystem. One of the advantages of the Wiener model is that the complexity of system dynamics is contained in the linear subsystem whereas the complexity of nonlinearity is only in the static subsystem. This advantage greatly reduces the difficulty in the design of controllers for unknown plants identified by the Wiener model because a divide-and-conquer strategy using well-known linear/nonlinear system theories can be applied to deal with the two subsystems separately. In addition, the overall output of the Wiener model can be written analytically as a kernel function expansion. Various nonlinear kernel functions can be employed to fit different non-linearities for different applications. A large number of research studies have indicated the superior capability and effectiveness of Wiener models in nonlinear dynamic system identification and control [11].

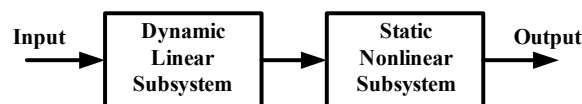


Fig. 1. The block diagram of a Wiener model.

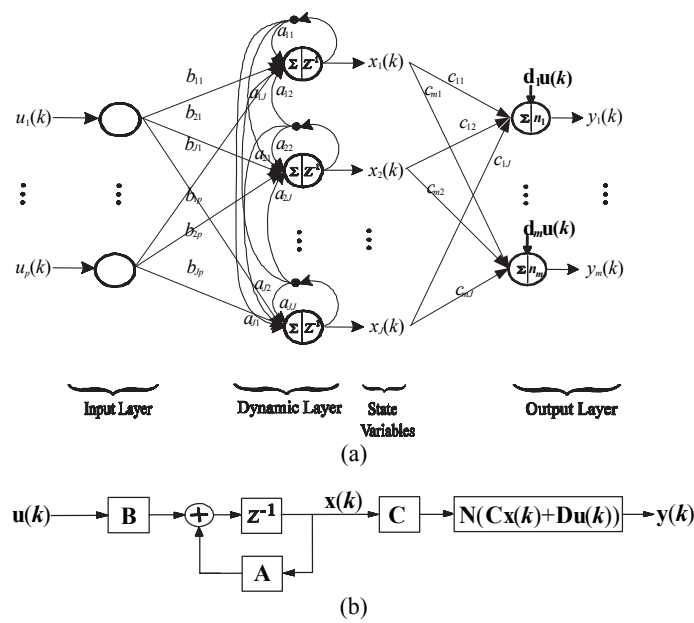


Fig. 2. (a) The topology of the proposed Wiener-type recurrent neural network; (b) The block diagram of the proposed network.

Based on the basic principle of Wiener models, we propose a simple recurrent neural network that integrates the two cascaded subsystems into a three-layered connectionist structure. Fig. 2 (a) shows the proposed recurrent structure, which can be expressed by a block diagram illustrated in Fig. 2 (b). The network structure consists of one input layer, one hidden layer, and one output layer. The input layer transmits the input values to the network. The hidden layer is the dynamic linear layer which maps the input space into a state-space via fully self-feedback connections embedded with time-delay elements (memories). That is, the dynamic layer integrates the current input information from the input layer and the state history stored in the memories of the neurons in the dynamic layer to infer the current states of the network. Finally, the linear combination of the input variables and the state variables are transformed into the output space through a nonlinear mapping in the output layer. The link weights between input layer and output layer are $[\mathbf{d}_1, \dots, \mathbf{d}_m]^T$. The input layer coupled with the dynamic layer forms the linear dynamic subsystem. The output layer, whose neurons combine the input variables and the state variables with different weights and then perform a nonlinear transformation via nonlinear activation functions, acts as the static nonlinear subsystem.

The proposed network can be expressed by the following state-space equations:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \\ \mathbf{y}(k) &= \mathbf{N}(\mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k)), \end{aligned} \tag{1}$$

where $\mathbf{x} = [x_1, \dots, x_J]^T$ is the state vector, $\mathbf{u} = [u_1, \dots, u_p]^T$ is the input vector, $\mathbf{y} = [y_1, \dots,$

$y_m]^T$ is the output vector, $\mathbf{N} = [n_1, \dots, n_m]^T$ is the nonlinear function vector, and J , p and m are the total number of state variables, the numbers of the inputs and outputs, respectively. The components of matrix $\mathbf{A} \in \mathbb{R}^{J \times J}$ are the weights of the self-feedback connections and stand for the degrees of the inter-correlations among the state variables. The elements of matrix $\mathbf{B} \in \mathbb{R}^{J \times p}$ are the link weights between the input layer and the dynamic layer. The elements of matrix $\mathbf{C} \in \mathbb{R}^{m \times J}$ are the link weights between the dynamic layer and the output layer. The elements of matrix $\mathbf{D} \in \mathbb{R}^{m \times p}$ are the coefficients of the linear combination of the input variables. The current output $y_i(k)$ and state variables $x_j(k)$ are obtained by calculating the activities of all nodes in each layer. The corresponding functions are summarized as follows.

$$x_j(k) = \sum_{i=1}^J a_{ji} x_i(k-1) + \sum_{h=1}^p b_{jh} u_h(k-1), \quad (2)$$

$$z_i = \mathbf{C}_i \mathbf{x}(k) + \mathbf{D}_i \mathbf{u}(k) = \sum_{j=1}^J c_{ij} x_j(k) + \sum_{h=1}^p d_{ih} u_h(k), \quad (3)$$

$$y_i(k) = f(z_i) = \frac{\exp(z_i) - \exp(-z_i)}{\exp(z_i) + \exp(-z_i)}. \quad (4)$$

The proposed network possesses some notable advantages. First, since the linear dynamic subsystem of our Wiener-type recurrent network can be translated into state-space equations, the characteristics and theoretical analysis, such as the stability, controllability, and observability of the network become straightforward. Second, the network size is determined by the number of state variables (or the system order). The total number of parameters equals $J \times (J + p + m) + (p \times m)$. That is, if the system order of the plant to be identified is available, the identification task comes down to parameter estimations for matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} . Based on these advantages, we have developed a system identification method based on the OKID algorithm for our Wiener-type recurrent network.

3. OKID-BASED IDENTIFICATION ALGORITHM

Given an unknown nonlinear dynamic plant, we have developed a system identification algorithm for the proposed Wiener-type recurrent neural network to automatically perform the identification task using the input-output measurements of the nonlinear system. Note that the system input-output measurements are normalized to $(-1, 1)$ beforehand and denormalized the values to the actual ones in the end of the identification process. First, we obtain the inverse values of the output measurements through the inverse function of the hyperbolic tangent sigmoid function in Eq. (4). Then, we utilize the input measurements and the inverse values to obtain an optimal state-space model representation by the OKID algorithm. We now introduce the proposed OKID-based identification algorithm in detail.

3.1 OKID Fundamentals

The idea of our system identification approach is to transfer a complex nonlinear problem to a simple linear one. Due to the invertible nonlinear activation function in Eq. (4) is employed in the output layer, we can acquire the output training data for the linear dynamic subsystem by inverting the original output data through the inverse function of the nonlinear activation function. With the original input data and the inverse output data, the realization of the proposed network can be cast as a linear state-space realization problem and can be solved by the OKID algorithm. In our system identification algorithm, we first use the given normalized input-output patterns $[\mathbf{u}(k), \mathbf{y}(k)]$ to generate another set of patterns $[\mathbf{u}(k), \mathbf{s}(k)]$, where $\mathbf{s}(k)$ is obtained by the inverse function of Eq. (4):

$$\mathbf{s}(k) = f^{-1}(\mathbf{y}(k)) = 0.5 \ln \frac{-\mathbf{y}(k) - 1}{\mathbf{y}(k) - 1}. \quad (5)$$

The acquisition of $\mathbf{s}(k)$ is regarded as the data transformation from the nonlinear output space to a linear one through the inverse mapping. The identification of the linear subsystem of our Wiener-type recurrent network becomes the search for the best representative linear state-space model from the input-output patterns $[\mathbf{u}(k), \mathbf{s}(k)]$.

The target representation of the dynamic linear subsystem is written as:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \\ \mathbf{s}(k) &= \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k). \end{aligned} \quad (6)$$

In general, the state variables of Eq. (6) can not be measured directly. Hence, an observer is required to estimate the state variables of the system from the input-output measurements. Therefore, we add and subtract an observer term $\mathbf{K}\mathbf{s}(k)$ to the right-hand side of the state equation in (6) to yield the observer equations as follows.

$$\begin{aligned} \bar{\mathbf{x}}(k+1) &= \bar{\mathbf{A}}\bar{\mathbf{x}}(k) + \bar{\mathbf{B}}\mathbf{v}(k), \\ \mathbf{s}(k) &= \mathbf{C}\bar{\mathbf{x}}(k) + \mathbf{D}\mathbf{u}(k), \end{aligned} \quad (7)$$

where

$$\begin{aligned} \bar{\mathbf{A}} &= \mathbf{A} + \mathbf{K}\mathbf{C}, \\ \bar{\mathbf{B}} &= [\mathbf{B} + \mathbf{K}\mathbf{D}, \quad -\mathbf{K}], \\ \mathbf{v}(k) &= \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{s}(k) \end{bmatrix}, \end{aligned} \quad (8)$$

and $\mathbf{K} \in \mathbb{R}^{J \times m}$ is an arbitrary matrix chosen to make the matrix $\bar{\mathbf{A}}$ as stable as desired. Note that $\mathbf{v}(k)$ consisting of the inputs and outputs in Eq. (6) is the input vector to the observer equations. The Markov parameters of the observer system are called as the observer Markov parameters. The input-output description can be expressed as the following matrix form:

$$\mathbf{s}_{m \times l} = \bar{\mathbf{Y}}_{m \times [(m+p)(l-1)+p]} \bar{\mathbf{V}}_{[(m+p)(l-1)+p] \times l}, \quad (9)$$

where

$$\mathbf{s} = [\mathbf{s}(0) \ \mathbf{s}(1) \ \mathbf{s}(2) \ \cdots \ \mathbf{s}(g) \ \cdots \ \mathbf{s}(l-1)]; \quad (10)$$

$$\bar{\mathbf{Y}} = [\mathbf{D} \ \mathbf{C}\bar{\mathbf{B}} \ \mathbf{C}\bar{\mathbf{A}}\bar{\mathbf{B}} \ \cdots \ \mathbf{C}\bar{\mathbf{A}}^{g-1}\bar{\mathbf{B}} \ \cdots \ \mathbf{C}\bar{\mathbf{A}}^{l-2}\bar{\mathbf{B}}]; \quad (11)$$

$$\bar{\mathbf{V}} = \begin{bmatrix} \mathbf{u}(0) & \mathbf{u}(1) & \mathbf{u}(2) & \cdots & \mathbf{u}(g) & \cdots & \mathbf{u}(l-1) \\ 0 & \mathbf{v}(0) & \mathbf{v}(1) & \cdots & \mathbf{v}(g-1) & \cdots & \mathbf{v}(l-2) \\ 0 & 0 & \mathbf{v}(0) & \cdots & \mathbf{v}(g-2) & \cdots & \mathbf{v}(l-3) \\ \vdots & \ddots & \ddots & \ddots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \ddots & \mathbf{v}(0) & \cdots & \mathbf{v}(l-g-1) \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \mathbf{v}(0) \end{bmatrix}. \quad (12)$$

l is the total number of training samples. If $\bar{\mathbf{A}}$ is asymptotically stable, then we have $\bar{\mathbf{A}}^k \approx \mathbf{0}$ for all $k \geq g$, where g is a sufficiently large integer. The input and output relationship can be written as the following matrix representation:

$$\mathbf{s}_{m \times l} = \hat{\mathbf{Y}}_{m \times [(m+p)g+p]} \mathbf{V}_{[(m+p)g+p] \times l}, \quad (13)$$

where

$$\mathbf{s} = [\mathbf{s}(0) \ \mathbf{s}(1) \ \mathbf{s}(2) \ \cdots \ \mathbf{s}(g) \ \cdots \ \mathbf{s}(l-1)]; \quad (14)$$

$$\hat{\mathbf{Y}} = [\mathbf{D} \ \mathbf{C}\bar{\mathbf{B}} \ \mathbf{C}\bar{\mathbf{A}}\bar{\mathbf{B}} \ \cdots \ \mathbf{C}\bar{\mathbf{A}}^{g-1}\bar{\mathbf{B}}]; \quad (15)$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{u}(0) & \mathbf{u}(1) & \mathbf{u}(2) & \cdots & \mathbf{u}(g) & \cdots & \mathbf{u}(l-1) \\ 0 & \mathbf{v}(0) & \mathbf{v}(1) & \cdots & \mathbf{v}(g-1) & \cdots & \mathbf{v}(l-2) \\ 0 & 0 & \mathbf{v}(0) & \cdots & \mathbf{v}(g-2) & \cdots & \mathbf{v}(l-3) \\ \vdots & \ddots & \ddots & \ddots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & 0 & \mathbf{v}(0) & \cdots & \mathbf{v}(l-g-1) \end{bmatrix}. \quad (16)$$

Note that $\hat{\mathbf{Y}}$ and \mathbf{V} refer to truncated versions of the $\bar{\mathbf{Y}}$ and $\bar{\mathbf{V}}$. Subsequently, the first g observer Markov parameters can be obtained by the following least-squares method.

$$\hat{\mathbf{Y}} = \mathbf{s} \mathbf{V}^T [\mathbf{V} \mathbf{V}^T]^{-1}. \quad (17)$$

If $[\mathbf{V} \mathbf{V}^T]^{-1}$ does not exist, then $\mathbf{V}^T [\mathbf{V} \mathbf{V}^T]^{-1}$ should be replaced by \mathbf{V}^+ , where \mathbf{V}^+ is the pseudo-inverse of the matrix \mathbf{V} . The observer Markov parameters $\hat{\mathbf{Y}}$ can be represented as:

$$\hat{\mathbf{Y}} = [\hat{\mathbf{Y}}_0 \ \hat{\mathbf{Y}}_1 \ \hat{\mathbf{Y}}_2 \ \cdots \ \hat{\mathbf{Y}}_g], \quad (18)$$

where

$$\begin{aligned}
 \hat{Y}_0 &= \mathbf{D}, \\
 \hat{Y}_k &= \mathbf{C}\bar{\mathbf{A}}^{k-1}\bar{\mathbf{B}} \\
 &= [\mathbf{C}(\mathbf{A} + \mathbf{K}\mathbf{C})^{k-1}(\mathbf{B} + \mathbf{K}\mathbf{D}) - \mathbf{C}(\mathbf{A} + \mathbf{K}\mathbf{C})^{k-1}\mathbf{K}] \\
 &= [\hat{Y}_k^{(1)} - \hat{Y}_k^{(2)}], k = 1, 2, \dots, g.
 \end{aligned} \tag{19}$$

3.2 Minimal State-Space Realization

The observer Markov parameters can be decomposed into two parts: the system Markov parameters and the observer gain Markov parameters. The system matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathbf{D} can be computed from the system Markov parameters that are obtained by the following equations.

$$\begin{aligned}
 Y_0^s &= \mathbf{D} = \hat{Y}_0, \\
 Y_k^s &= \mathbf{C}\mathbf{A}^{k-1}\mathbf{B} = \hat{Y}_k^{(1)} - \sum_{i=1}^k \hat{Y}_i^{(2)} Y_{k-i}^s; \text{ for } k = 1, 2, \dots, g, \\
 Y_k^s &= \mathbf{C}\mathbf{A}^{k-1}\mathbf{B} = -\sum_{i=1}^g \hat{Y}_i^{(2)} Y_{k-i}^s; \text{ for } k > g.
 \end{aligned} \tag{20}$$

The observer gain matrix \mathbf{K} can be estimated by the observer gain Markov parameters using the following equations:

$$\begin{aligned}
 Y_1^o &= \mathbf{C}\mathbf{K} = \hat{Y}_1^{(2)}, \\
 Y_k^o &= \mathbf{C}\mathbf{A}^{k-1}\mathbf{K} = \hat{Y}_k^{(2)} - \sum_{i=1}^{k-1} \hat{Y}_i^{(2)} Y_{k-i}^o; \text{ for } k = 2, 3, \dots, g, \\
 Y_k^o &= \mathbf{C}\mathbf{A}^{k-1}\mathbf{K} = -\sum_{i=1}^g \hat{Y}_i^{(2)} Y_{k-i}^o; \text{ for } k > g.
 \end{aligned} \tag{21}$$

Using the combined Markov parameters $\Lambda_k = [\mathbf{C}\mathbf{A}^{k-1}\mathbf{B} \quad \mathbf{C}\mathbf{A}^{k-1}\mathbf{K}]$, we can obtain matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{K} by the eigensystem realization algorithm (ERA). In the ERA, the order of a linear system can be determined by examining the singular values of the Hankel matrix $\bar{\mathbf{H}}_p(0)$. To apply the ERA, we form a generalized Hankel matrix $\bar{\mathbf{H}}_p(0)$ and a shifted Hankel matrix $\bar{\mathbf{H}}_p(1)$ using the Markov parameters and describe these matrices as follows.

$$\bar{\mathbf{H}}_p(0) = \begin{bmatrix} \Lambda_1 & \Lambda_2 & \cdots & \Lambda_\beta \\ \Lambda_2 & \Lambda_3 & \cdots & \Lambda_{\beta+1} \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda_\alpha & \Lambda_{\alpha+1} & \cdots & \Lambda_{\alpha+\beta-1} \end{bmatrix}, \tag{22}$$

$$\bar{H}_p(1) = \begin{bmatrix} \Lambda_2 & \Lambda_3 & \cdots & \Lambda_{\beta+1} \\ \Lambda_3 & \Lambda_4 & \cdots & \Lambda_{\beta+2} \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda_{\alpha+1} & \Lambda_{\alpha+2} & \cdots & \Lambda_{\alpha+\beta} \end{bmatrix}, \quad (23)$$

where α and β are arbitrary integers. To determine the system order, the singular value decomposition (SVD) is applied to decompose the Hankel matrix $\bar{H}_p(0)$:

$$\bar{H}_p(0) = W \Sigma Q^T, \quad (24)$$

where W and Q are orthogonal matrices and Σ is a rectangular matrix. For a J th-order linear system with noise-free input-output measurements, $\Sigma = \begin{bmatrix} \Sigma_J & 0 \\ 0 & 0 \end{bmatrix}$, where $\Sigma_J = \text{diag}[\sigma_1, \sigma_2, \dots, \sigma_J, \sigma_{J+1}, \dots, \sigma_j]$ with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_J \geq \sigma_{J+1} \geq \dots \geq \sigma_j \geq 0$. If some singular values, $\sigma_{J+1}, \dots, \sigma_j$, are relatively small ($\sigma_J \gg \sigma_{J+1}$), the matrix $\bar{H}_p(0)$ can be expressed as follows.

$$\bar{H}_p(0) = [W_J \quad W_0] \begin{bmatrix} \Sigma_J & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Q_J^T \\ Q_0^T \end{bmatrix} = W_J \Sigma_J Q_J^T = [W_J \Sigma_J^{1/2}] [\Sigma_J^{1/2} Q_J^T], \quad (25)$$

where W_J and Q_J are the matrices formed by the first J columns of W and Q , and $\Sigma_J = \text{diag}[\sigma_1, \sigma_2, \dots, \sigma_J]$. Then, we can also obtain the shifted Hankel matrix by

$$\bar{H}_p(1) = [W_J \Sigma_J^{1/2}] A [\Sigma_J^{1/2} Q_J^T]. \quad (26)$$

Based on the result of the SVD, we can obtain A , B , C and K by the following equations.

$$A = \Sigma_J^{-1/2} W_J^T \bar{H}_p(1) Q_J \Sigma_J^{-1/2}, \quad (27)$$

$$[B \quad K] = \text{First } (p+m) \text{ columns of } \Sigma_J^{1/2} Q_J^T, \quad (28)$$

$$C = \text{First } m \text{ rows of } W_J \Sigma_J^{1/2}. \quad (29)$$

Since matrices A , B , C and K contain very important dynamic property about the controllability and observability of the system, selecting a suitable system order is an important issue. In general, the SVD is frequently used to estimate the system order of linear systems. For a J th-order system with m outputs, the minimum number of observer Markov parameters required to be identified is $g_{\min} = J/m$. If the minimum number of observer Markov parameters, g , is identified, then the maximum order of the system is gm . The more detailed information about the OKID can be found in [9].

3.3 Steps of OKID-Based Identification Algorithm

We now summarize our OKID-based identification algorithm in the following steps.

- Step 1:** Obtain the input-output training patterns $[\mathbf{u}(k), \mathbf{s}(k)]$ for the linear dynamic subsystem through the inverse of hyperbolic tangent sigmoid functions.
- Step 2:** Estimate the first g observer Markov parameters from the input-output patterns obtained in step 1.
- Step 3:** Compute the system Markov parameters and observer gain Markov parameters using the observer Markov parameters.
- Step 4:** Realize matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{D} and the observer gain \mathbf{K} by the ERA algorithm using Eqs. (20)-(29).
- Step 5:** Assign the components of \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} as the corresponding parameters of the linear subsystem of the proposed Wiener-type network.

Upon the completion of the OKID-based identification algorithm, we can establish the recurrent network to emulate the unknown system with good performance.

4. SIMULATION RESULTS

To validate the effectiveness of our Wiener-type recurrent network and the proposed identification algorithm, we have conducted extensive computer simulations on dynamic system identification problems. Here, we present two examples of dynamic nonlinear system identification: a multi-input-multi-output (MIMO) plant and a nonlinear Wiener system. The simulation results are compared with those of some existing methods found in the literature.

Example 1: MIMO dynamic plant identification. The following plant is adopted from [15] and is described as

$$\begin{aligned} y_{p1}(k+1) &= 0.5 \left[\frac{y_{p1}(k)}{(1 + y_{p2}^2(k))} + u_1(k) \right], \\ y_{p2}(k+1) &= 0.5 \left[\frac{y_{p1}(k)y_{p2}(k)}{(1 + y_{p2}^2(k))} + u_2(k) \right]. \end{aligned} \quad (30)$$

This example is utilized to demonstrate the capability of the proposed network and algorithm in identifying MIMO plants. A total of 1100 time steps of two *i.i.d* uniform sequences within the limits $[-2, 2]$ are generated to identify the proposed network. Theoretically, $g_{\min} = J/m$ for a J th-order system with m outputs, and g and J are integers. The selection of g is related to the selection of J . Since the value of J is unavailable for unknown system identification problems, in this study we started the value of g from 1 and increased the value incrementally until we found satisfactory system performance. By satisfactory system performance, we consider that the size of the state-space representation as well as the values of MSE should be minimal. Fig. 3 shows the curve of the MSEs for different values of g . We set $g = 1$ and $J = gm = 2$. The parameters identified by the OKID algorithm are as follows. $\Sigma_J = \text{diag}[0.80702, 0.6492]$ and the state-space equation is written as

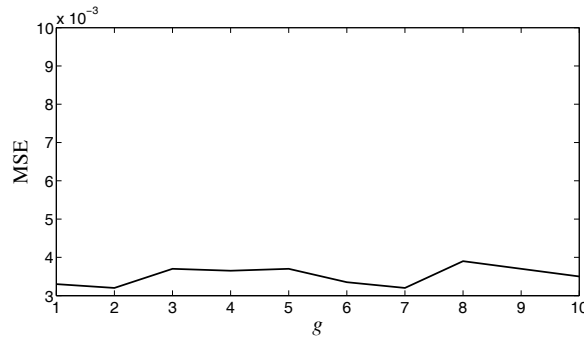


Fig. 3. The MSEs of the proposed recurrent network with different values of g from 1 to 10 for Example 1.

$$\begin{aligned} \mathbf{x}(k+1) &= \begin{bmatrix} 0.4086 & 0.0098436 \\ 0.0019655 & 0.0273 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} -0.47582 & 0.023821 \\ -0.026463 & -0.6498 \end{bmatrix} \mathbf{u}(k), \\ s(k) &= \begin{bmatrix} -0.73257 & -0.026368 \\ 0.03369 & -0.65112 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} -0.0015273 & -0.00042854 \\ -0.0029158 & -0.0047174 \end{bmatrix} \mathbf{u}(k). \end{aligned} \quad (31)$$

The eigenvalues of matrix \mathbf{A} are (0.4087, 0.0272). The total number of the network parameters is 16. After the model is identified, we use the following testing input signal ($u(k) = u_1(k) = u_2(k)$) to verify their identification performance.

$$u(k) = \begin{cases} \sin(\pi k/25), & 0 \leq k < 250 \\ 1.0, & 250 \leq k < 500 \\ -1.0, & 500 \leq k < 750 \\ 0.3\sin(\pi k/25) + 0.1\sin(\pi k/32) + 0.6\sin(\pi k/10), & 750 \leq k < 1000 \end{cases} \quad (32)$$

In the testing phase, the network output is obtained by transferring $s(k)$ to $\mathbf{y}(k)$ via Eq. (4). We compare the Wiener-type network with two existing recurrent networks. From Table 1, we can see that the performance of the proposed network with fewer parameters performs better than those of the two existing networks. From Table 2, we can see that the OKID algorithm performs better than the other two learning algorithms. Fig. 4 depicts the outputs of the MIMO plant and the proposed network for the testing signal.

Table 1. Performance comparisons of the proposed recurrent network with two existing recurrent networks.

| Network Type | No. of Parameters | Training Time (time steps) | MSE |
|---------------------|-------------------|----------------------------|--|
| Wiener-type network | 16 | 1,100 | $y_1 = 1.30 \times 10^{-3}$ $y_2 = 4.90 \times 10^{-3}$ |
| RSONFIN [10] | 77 | 11,000 | $y_1 = 1.24 \times 10^{-2}$ $y_2 = 1.97 \times 10^{-2}$ |
| MNN [17] | 131 | 77,000 | $y_1 = 1.86 \times 10^{-2}$ $y_2 = 3.27 \times 10^{-2}$ |

Table 2. Performance comparisons of the proposed OKID algorithm with two existing learning algorithms.

| Learning Algorithm | No. of Parameters | Training Time (time steps) | MSE |
|--|-------------------|----------------------------|--|
| OKID | 16 | 1,100 | $y_1 = 1.30 \times 10^{-3}$ $y_2 = 4.90 \times 10^{-3}$ |
| Levenberg-Marquardt | 16 | 1,100 | $y_1 = 5.10 \times 10^{-3}$ $y_2 = 2.46 \times 10^{-1}$ |
| Variable learning-rate backpropagation | 16 | 1,100 | $y_1 = 9.10 \times 10^{-3}$ $y_2 = 2.30 \times 10^{-2}$ |

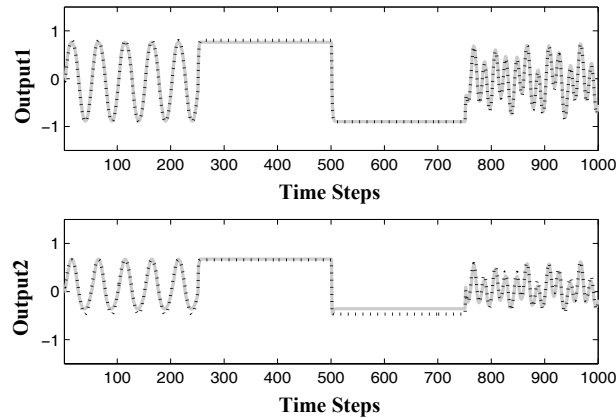


Fig. 4. The outputs of the MIMO plant (solid curve) and the proposed recurrent network (dotted curve) for Example 1.

Example 2: Nonlinear Wiener system identification. The following fluid-flow control problem is adopted from [1].

$$\begin{aligned}
 x(k) &= 1.4138x(k-1) - 0.6065x(k-2) + 0.1044u(k-1) + 0.0883u(k-2), \\
 y(k) &= f[x(k)] = \frac{0.3163x(k)}{\sqrt{0.10 + 0.90[x(k)]^2}},
 \end{aligned} \tag{33}$$

where $u(k)$ is the pneumatic control signal, $x(k)$ is the stem position, and the resulting flow through the valve is given by the nonlinear function $f[x(k)]$ of the stem position. This example is utilized to demonstrate the capability of the proposed network with the OKID algorithm in identifying nonlinear Wiener systems. A total of 1000 time steps of one *i.i.d* uniform sequence within the limits $[-0.05, 0.05]$ were generated to identify the proposed network. To verify the identification performance, the testing signal generated by $u(k) = 0.5\sin(k)$ was applied to the identified model. We used the same procedure as Example 1 to determine a suitable g for the testing signal. Therefore, we set $g = 2$ and $J = gm = 2$. The total number of the network parameters is 9. The state-space equation of the Wiener-type recurrent network identified by the OKID algorithm is written as

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.71785 & 0.32751 \\ -0.32657 & 0.69668 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} -0.2372 \\ -0.15061 \end{bmatrix} \mathbf{u}(k), \quad (34)$$

$$s(k) = [-1.2828 \quad 0.46128] \mathbf{x}(k) + 0.10388 \mathbf{u}(k).$$

The eigenvalues of matrix A of this model are $(0.7073 + 0.3269i, 0.7073 - 0.3269i)$ that indicate the identified network is stable. In the testing phase, the network output is obtained by transferring $s(k)$ to $y(k)$ via Eq. (4). The MSE of the proposed Wiener-type network for the testing signal is 3.85×10^{-5} . Fig. 5 depicts the outputs of the nonlinear Wiener system and the proposed network for the testing signal.

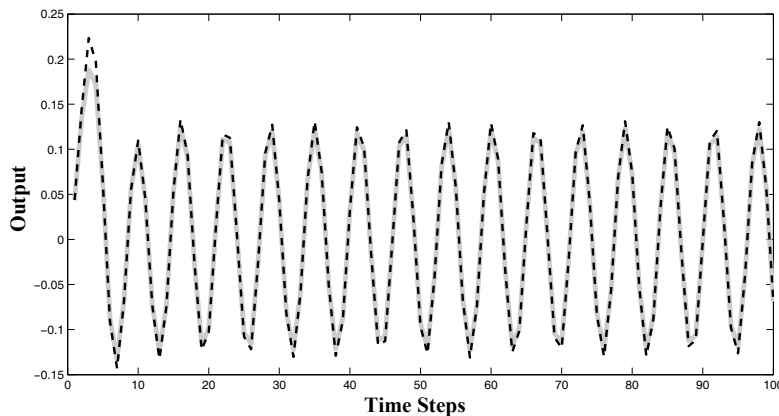


Fig. 5. The outputs of the nonlinear Wiener system (solid curve) and the proposed recurrent network (dotted curve) for Example 2.

5. CONCLUSION

A novel Wiener-type recurrent neural network with an OKID-based identification algorithm has been proposed for nonlinear unknown dynamic system identification problems. The advantages of our approach include: (1) the realization of a conventional Wiener model into a simple connectionist recurrent network whose output can be expressed by a nonlinear transformation of a linear state-space equation; (2) the overall network structure can be determined by the OKID algorithm effectively using only the input-output patterns; and (3) the proposed network is capable of accurately identifying nonlinear dynamic systems using fewer parameters. However, from a theoretical point of view, the OKID algorithm cannot be performed online. This is one of the limitations for using the OKID algorithm in the model construction phase of system identification problems. One possible remedy for such a limitation is to use error correction methods, *e.g.* back-propagation, for tuning the network parameters online to achieve better performance. The effectiveness and superiority of our proposed approach have been successfully validated by computer simulations on nonlinear unknown dynamic examples and comparisons with some existing recurrent networks and learning algorithms.

REFERENCES

1. H. Al-Duwaish, M. N. Karim, and V. Chandrasekar, "Use of multilayer feedforward neural networks in identification and control of Wiener model," *IEE Proceedings of Control Theory and Applications*, Vol. 143, 1996, pp. 255-258.
2. G. Chen, Y. Chen, and H. Ogmen, "Identifying chaotic systems via a Wiener-type cascade model," *IEEE Control System Magazine*, Vol. 17, 1997, pp. 29-36.
3. T. Clarke and X. D. Sun, "Minimal state-space model realization of a nonlinear helicopter," *IEE Proceedings of Control Theory and Applications*, Vol. 145, 1998, pp. 415-422.
4. Y. P. Chen and J. S. Wang, "An automated Hammerstein recurrent neural network for dynamic applications," in *Proceeding of IASTED International Conference on Computational Intelligence*, 2005, pp. 193-198.
5. B. L. Ho and R. E. Kalman, "Effective construction of linear state-variable models from input/output data," in *Proceedings of the 3rd Annual Allerton Conference on Circuit and System Theory*, 1965, pp. 449-459.
6. J. N. Juang and R. S. Pappa, "An eigensystem realization algorithm for modal parameter identification and model reduction," *Journal of Guidance*, Vol. 8, 1985, pp. 620-627.
7. J. N. Juang, M. Phan, L. G. Horta, and R. W. Longman, "Identification of observer/Kalman filter Markov parameters: theory and experiments," *Journal of Guidance, Control, and Dynamics*, Vol. 16, 1993, pp. 320-329.
8. J. N. Juang and M. Phan, "Linear system identification via backward-time observer models," *Journal of Guidance, Control, and Dynamics*, Vol. 17, 1994, pp. 505-512.
9. J. N. Juang, *Applied System Identification*, Prentice-Hall, New Jersey, 1994.
10. C. F. Juang and C. T. Lin, "A recurrent self-organizing neural fuzzy inference network," *IEEE Transactions on Neural Networks*, Vol. 10, 1999, pp. 828-845.
11. A. Janczak, *Identification of Nonlinear Systems Using Neural Networks and Polynomial Models*, Springer-Verlag, New York, 2005.
12. A. Kalinli and S. Sagioglu, "Elman network with embedded memory for system identification," *Journal of Information Science and Engineering*, Vol. 22, 2006, pp. 1555-1568.
13. C. J. Lin, "Wavelet neural networks with a hybrid learning approach," *Journal of Information Science and Engineering*, Vol. 22, 2006, pp. 1367-1387.
14. C. J. Lin and Y. J. Xu, "Design of neuro-fuzzy systems using a hybrid evolutionary learning algorithm," *Journal of Information Science and Engineering*, Vol. 23, 2007, pp. 463-477.
15. K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural network," *IEEE Transactions on Neural Networks*, Vol. 1, 1990, pp. 4-27.
16. O. Nelles, *Nonlinear System Identification*, Springer-Verlag, New York, 2001.
17. P. S. Sastry, G. Santharam, and K. P. Unnikrishnan, "Memory neuron networks for identification and control of dynamical systems," *IEEE Transactions on Neural Networks*, Vol. 5, 1994, pp. 306-319.
18. J. S. Wang and Y. P. Chen, "A fully automated recurrent neural network for unknown dynamic system identification and control," *IEEE Transactions on Circuits and Systems-I*, Vol. 53, 2006, pp. 1363-1372.



Jeen-Shing Wang (王振興) received the B.S. and M.S. degrees in Electrical Engineering from the University of Missouri, Columbia, MO, in 1996 and 1997, respectively, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 2001. He is currently an Associate Professor with the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan. His research interests include computational intelligence, intelligent control, clustering analysis, and optimization.



Yu-Liang Hsu (許煜亮) received the B.S. degree in Automatic Control Engineering from the Feng Chia University in 2004, and the M.S. degree in Electrical Engineering from National Cheng Kung University, Taiwan, R.O.C., in 2007. He is currently working toward the Ph.D. degree in Electrical Engineering in the Department of Electrical Engineering, National Cheng Kung University. His research interests include intelligent control and nonlinear system identification.