EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Stochastic analysis of neural network modeling and identification of nonlinear memoryless MIMO systems

Mohamed Ibnkahla*

Abstract

Neural network (NN) approaches have been widely applied for modeling and identification of nonlinear multiple-input multiple-output (MIMO) systems. This paper proposes a stochastic analysis of a class of these NN algorithms. The class of MIMO systems considered in this paper is composed of a set of single-input nonlinearities followed by a linear combiner. The NN model consists of a set of single-input memoryless NN blocks followed by a linear combiner. A gradient descent algorithm is used for the learning process. Here we give analytical expressions for the mean squared error (MSE), explore the stationary points of the algorithm, evaluate the misadjustment error due to weight fluctuations, and derive recursions for the mean weight transient behavior during the learning process. The paper shows that in the case of independent inputs, the adaptive linear combiner identifies the linear combining matrix of the MIMO system (to within a scaling diagonal matrix) and that each NN block identifies the corresponding unknown nonlinearity to within a scale factor. The paper also investigates the particular case of linear identification of the nonlinear MIMO system. It is shown in this case that, for independent inputs, the adaptive linear combiner identifies a scaled version of the unknown linear combining matrix. The paper is supported with computer simulations which confirm the theoretical results.

Keywords: Nonlinear system identification, Neural networks, Gradient descent, Statistical analysis

Introduction

Neural network [1] approaches have been extensively used in the past few years for nonlinear MIMO system modeling, identification and control where they have shown very good performances compared to classical techniques [2-6].

If these NN approaches are to be used in real systems, it is important for the algorithm designer and the user to understand their learning behavior and performance capabilities. Several authors have analyzed NN algorithms during the last two decades which considerably helped the neural network community to better understand the mechanisms of neural networks [1,7-15]. For example, the authors in [13] have studied a simple structure consisting of two inputs and a single neuron. The authors in [8] studied a memoryless single-input single-output (SISO) system identification model for the single neuron case. In [9] the authors proposed a stochastic analysis of gradient adaptive identification of nonlinear Wiener systems composed of a linear filter followed with a Zero-memory nonlinearity. The model was composed of a linear adaptive filter followed by an adaptive parameterized version of the nonlinearity. This study has been later generalized [16] for the analysis of stochastic gradient tracking of time-varying polynomial Wiener systems. In [12] the author analyzed NN identification of nonlinear SISO Wiener systems with memory for the case where the adaptive nonlinearity is a memoryless NN with an arbitrary number of neurons. The case of a nonlinear SISO Wiener-Hammerstein system (i.e., an adaptive filter followed by an adaptive Zero-memory NN followed by an adaptive filter) has been analyzed in [11].

This paper deals with a typical class of nonlinear MIMO systems (Figure 1) which is composed of M inputs, M memoryless nonlinearities, a linear combiner, and L outputs. This corresponds, for example, to MIMO channels used in wireless terrestrial communications

Correspondence: ibnkahla@post.queensu.ca
Electrical and Computer Engineering Department, Queen's University, Kingston, Ontario K7L 3N6, Canada

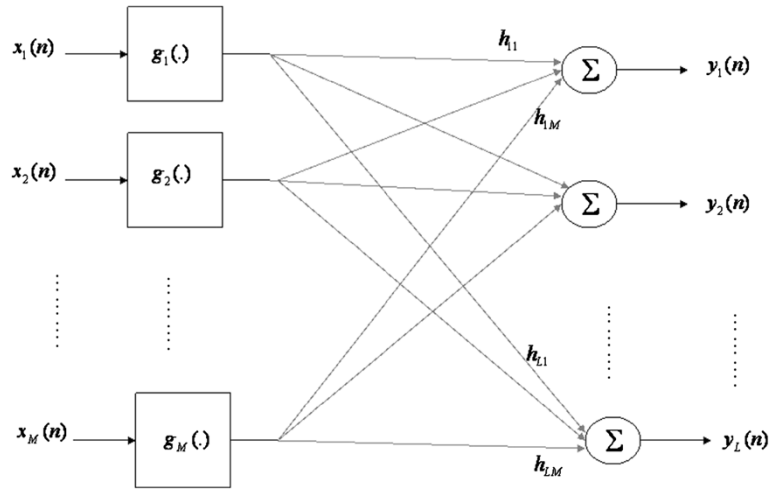

Figure 1 Nonlinear MIMO system.

[17-22], satellite communications [23,24], amplifier modeling [25], control of nonlinear MIMO systems [6], etc. Recently, a neural network approach has been proposed to adaptively identify the overall input–output transfer function of this class of MIMO systems and to characterize each component of the system (i.e., the memoryless nonlinearities and the linear combiner) [4]. The proposed NN model is composed of a set of memoryless NN blocks followed by an adaptive linear combiner. Each part of the adaptive system aims at identifying the corresponding part in the unknown MIMO system. The algorithm has been successfully applied to system modeling, channel tracking, and fault detection.

The purpose of this paper is to provide a stochastic analysis of NN modeling of this class of MIMO systems. The paper provides a general methodology that may be used to solve other problems in stochastic NN learning analysis. The methodology consists of splitting the study into simple structures, before studying the complete structure. Here, as a first step we start by analyzing a simple linear adaptive MIMO scheme (consisting of an adaptive matrix) that identifies the nonlinear MIMO system (i.e., problem of linear identification of a nonlinear MIMO system). Then we analyze a nonlinear adaptive system in which the nonlinearities are assumed to be known and frozen during the learning process, only the linear combiner is made adaptive. Finally, the complete adaptive scheme is analyzed taking into account the insights given by the analysis of the simpler structures. In our analytical approach, we derive the general formulas and recursions, which we apply to a case study that we believe is illustrative to the reader.

The paper is organized as follows. The problem statement is given in Section 2. The study of the simple structures is detailed in Section 3. Section 4 presents the analysis for the complete structure. Simulation results and illustrations are given in Section 5. Finally, conclusions and future work are given in Section 6.

Problem statement

Nonlinear MIMO system

The class of nonlinear MIMO systems discussed in this paper is presented in Figure 1. Each input x_i (n) $(i = 1,\ldots, M)$ is nonlinearly transformed by a memoryless nonlinearity g_i (.). The outputs of these nonlinearities are then linearly combined by an $L \times M$ matrix $H = [h_{ji}]$ (assumed in this paper to be constant). Matrix H is defined by the unknown system to be identified. For example, in wireless MIMO communication systems, M is the propagation matrix representing the channel between M transmitting antennas and L receiving antennas.

The j^{th} output of the MIMO system is expressed as:

$$y_j(n) = \sum_{i=1}^{M} h_{ji}(n) g_i(x_i(n)) + N_j(n) \tag{1}$$

where N_j is a white Gaussian noise with variance σ_0^2.

Let $X(n) = [x_1(n) x_2(n) \ldots x_M(n)]^t, g(X(n)) = [g_1(x_1(n)) g_2(x_2(n)) \ldots g_M(x_M(n))]^t$, $Y(n) = [y_1(n) y_2(n) \ldots y_L(n)]^t$, *and* $N(n) = [N_1(n) N_2(n) \ldots N_L(n)]^t$.

The system input–output relationship can be expressed in a matrix form as:

$$Y(n) = H \times g(X(n)) + N(n). \tag{2}$$

Neural Network identification structure and algorithm

The neural network (Figure 2) is composed of M blocks. Each block k has a scalar input x_k (n) $(k = 1,\ldots,M)$, N

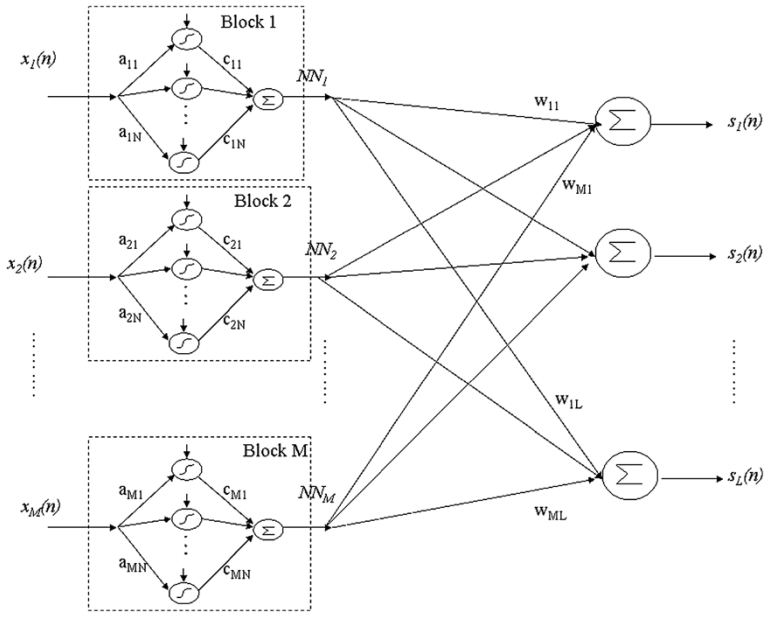

Figure 2 NN identification structure.

neurons and a scalar output. The output of the k[th] block is expressed as:

$$NN_k(n) = \sum_{i=1}^{N} c_{ki} f(a_{ki} x_k(n) + b_{ki}), k = 1, \ldots, M \quad (3)$$

Where f is the NN activation function. a_{ki}, c_{ki}, b_{ki} are, respectively, the input weight, bias term, and output weight of the i^{th} neuron in the k^{th} block. The output NN_k of the k^{th} block is connected to the j^{th} output of the system through weight w_{jk}. The system j^{th} output is then expressed as:

$$s_j(n) = \sum_{k=1}^{M} w_{jk} NN_k(n), j = 1, , L \quad (4)$$

Weights w_{jk} will be represented by an LxM matrix: $W = [w_{jk}]$.

Let $S(n) = [s_1(n) s_2(n) \ldots s_L(n)]^t$ *and* $NN(n) = [NN_1(n)\ NN_2(n) \ldots NN_M(n)]^t$.

Equations (4) can then be expressed in a matrix form as:

$$S(n) = W \times NN(n). \quad (5)$$

For the learning process, the NN parameters are updated so that to minimize the sum of the squared errors between the unknown system outputs and the corresponding outputs of the model (Figure 3):

$$\|e(n)\|^2 = \sum_{j=1}^{L} e_j^2(n). \quad (6)$$

Here $e_j(n) = y_j(n) - s_j(n)$ *and* $e(n) = [e_1(n) e_2(n) \ldots e_L(n)]^t$.

The gradient descent recursions for weight adaptation are:

$$W(n+1) = W(n) + 2\mu e(n) NN^t(n) \quad (7)$$

$$c_{ki}(n+1) = c_{ki}(n) + 2\mu f(a_{ki} x_k(n) + b_{ki}) \sum_{l=1}^{L} w_{lk} e_l(n) \quad (8)$$

$$a_{ki}(n+1) = a_{ki}(n) + 2\mu c_{ki} x_k(n) f'(a_{ki} x_k(n) + b_{ki}) \sum_{l=1}^{L} w_{lk} e_l(n) \quad (9)$$

$$b_{ki}(n+1) = b_{ki}(n) + 2\mu c_{ki} f'(a_{ki} x_k(n) + b_{ki}) \times \sum_{l=1}^{L} w_{lk} e_l(n) \quad (10)$$

where μ is a small positive constant and $f'()$ represents the derivative: $f'(x) = \frac{\partial f(x)}{\partial x}$.

Case study

After the derivation of the general formulas, it is important that we apply them to special cases in order to get closed-form expressions of the different recursions that can be illustrated to the reader. We have chosen here a case study that we think is good to illustrate our results.

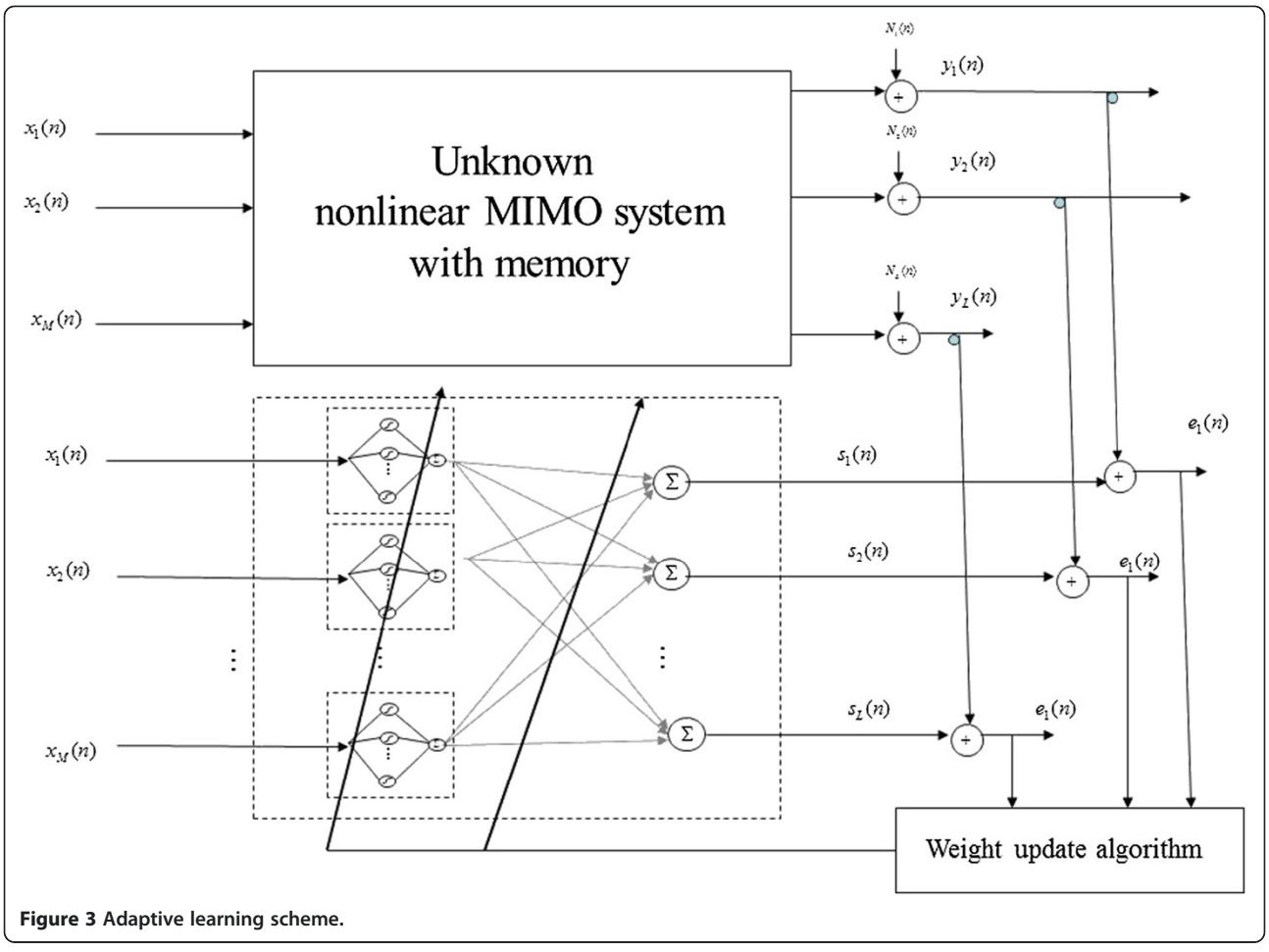

Figure 3 Adaptive learning scheme.

In this case study, the inputs x_i (n) will be assumed uncorrelated Zero-mean Gaussian variables with variance $\sigma^2_{x_i}$. The NN activation function will be taken as the *erf* function. The unknown nonlinear transfer functions are taken from a family of nonlinear functions of the form $g_i(x) = \alpha_i x \exp\left(\frac{-\beta_i x^2}{2}\right)$, where α_i and β_i are positive constants. These nonlinear functions are reasonable models for amplitude conversions of nonlinear high power amplifiers (HPA) used in digital communications [12,25,26]. Note that other nonlinear functions may be considered, however, explicit closed-form solutions of the different derivations may not be possible.

Study of simplified structures: Linear adaptation

Before analyzing the full structure, we will analyze the following simplified schemes which will help us understand the complete structure:

1. The adaptive system is composed of an adaptive linear combiner W (Section 3.1).
2. The adaptive system is composed of W and scaled versions of the unknown nonlinearities (Section 3.2).

Linear adaptive system

This section studies the linear adaptive system that tries to model the nonlinear MIMO system (Figure 4):

Mean weight behavior and Wiener solution

Since matrix W is linear, it will not be able to identify the nonlinear blocks. However, we will see that it is able to identify matrix H to within a diagonal scaling matrix if the inputs are Zero-mean and independent.

The gradient descent update of matrix W is expressed as:

$$\begin{aligned} W(n+1) &= W(n) + 2\mu e(n)X^t(n) \\ &= W(n) + 2\mu((HgX(n) + N(n) \\ &\quad - W(n)X(n))X^t(n) \end{aligned} \tag{11}$$

Averaging both sides of (11) and using the standard LMS assumption of small μ [10], we obtain:

$$\begin{aligned} E(W(n+1)) &\approx E(W(n)) + 2\mu\big(HR_{g(X)X} \\ &\quad - E(W(n))R_{XX}\big) \\ &= E(W(n))(I - 2\mu R_{XX}) + 2\mu H R_{g(X)X} \end{aligned} \tag{12}$$

Where $R_{XX} = E(XX^t), R_{g(X)X} = E(g(X)X^t)$.

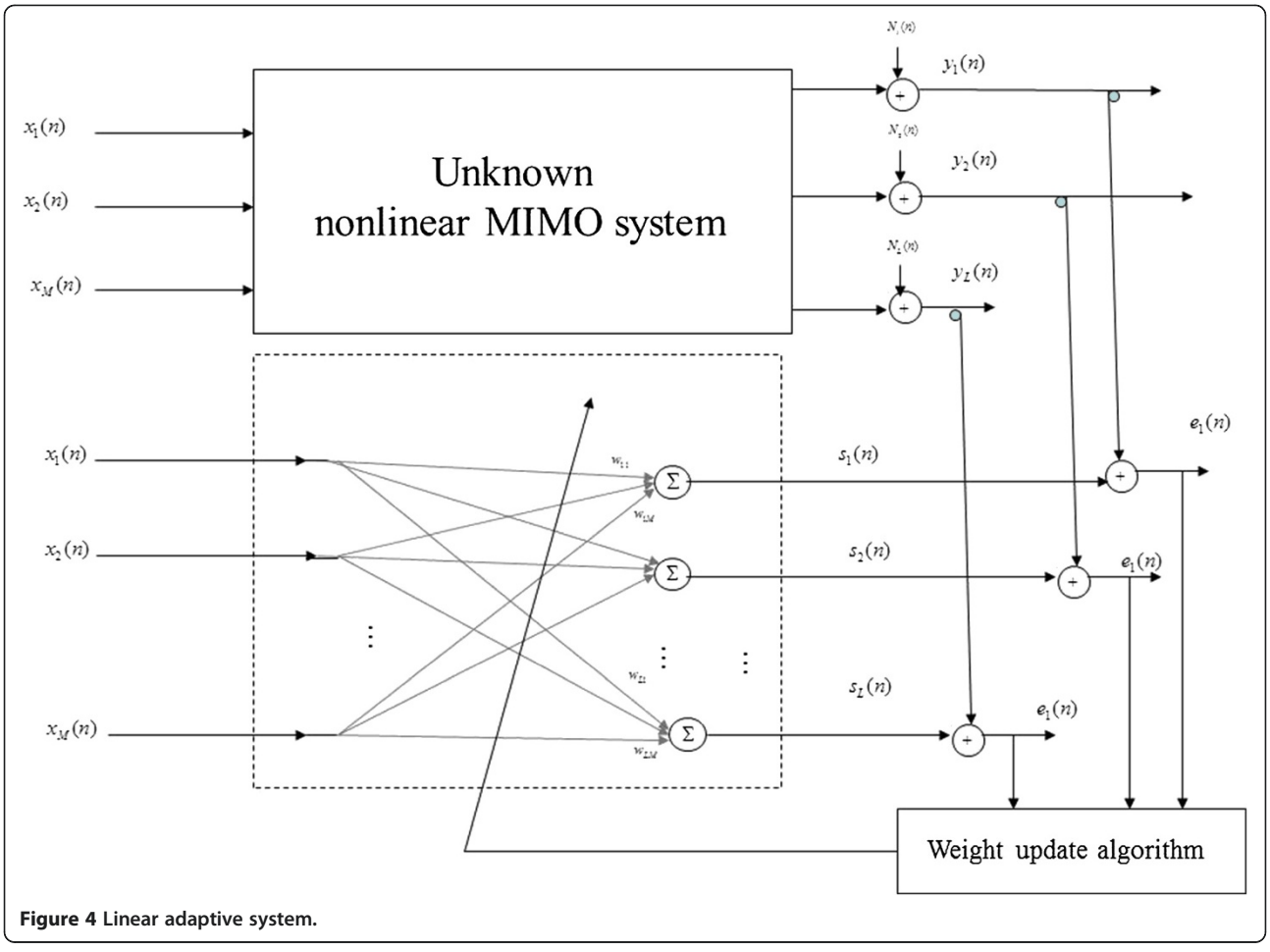

Figure 4 Linear adaptive system.

By setting the updating gradient term to Zero, it can be shown that this equation has a single stationary point (Wiener solution [10]) which is expressed by:

$$W = W_0 = HU \quad where \quad U = R_{g(X)X}R_{XX}^{-1} \tag{13}$$

Following Equation (12), the mean weights can be expressed as a function of the initial condition as:

$$E(W(n)) = W(0)(I - 2\mu R_{XX})^n + 2\mu H R_{g(X)X} \sum_{p=0}^{n-1} (I - 2\mu R_{XX})^p \tag{14}$$

If μ is sufficiently small, the first term converges to 0 and the second term converges to $HR_{g(X)X}R_{XX}^{-1}$.

Hence, the mean weights converge to the Wiener solution:

$$W(\infty) = W_0 = HR_{g(X)X}R_{XX}^{-1} \tag{15}$$

It can be easily shown that the stability condition on μ is [10]:

$$0 < \mu < \frac{1}{\lambda_{\max}} \tag{16}$$

where $\lambda_{\max}$ is the largest eigenvalue of the covariance matrix R_{XX}.

Note that for Zero-mean independent inputs, U is a diagonal matrix:

$$U = R_{g(X)X}R_{XX}^{-1} = \begin{bmatrix} \frac{E[g_1(x_1)x_1]}{\sigma_{x_1}^2} & 0\ldots & 0 \\ 0 & \frac{E[g_2(x_2)x_2]}{\sigma_{x_2}^2} & \ldots \\ 0 & 0\ldots & \frac{E[g_M(x_M)x_M]}{\sigma_{x_M}^2} \end{bmatrix} \tag{17}$$

In this case, the linear adaptation allows the identification of matrix W to within a scaling matrix, which depends on the nonlinearities and the input signals. As

expected, the scaling matrix reduces to the identity matrix if $g_k(x_k) = x_k$.

Application to the case study:

For the particular nonlinear functions given in the case study (see Section 2.3), it is easy to show:

$$E(x_i(n)g_i(x_i(n))) = \frac{\alpha_i {\sigma_{x_i}}^2}{\left(1+\beta_i {\sigma_{x_i}}^2\right)^{\frac{3}{2}}} \quad and$$
$$E\left(g_i^2(x_i(n))\right) = \frac{\alpha_i^2 {\sigma_{x_i}}^2}{\left(1+2\beta_i {\sigma_{x_i}}^2\right)^{\frac{3}{2}}} \tag{18}$$

The mean weight transient recursions are expressed as:

$$E\left(w_{jk}(n+1)\right) = E\left(w_{jk}(n)\right)\left(1-2\mu\sigma_{x_k}^2\right) + 2\mu\, h_{jk}\frac{\alpha_k \sigma_{x_k}^2}{\left(1+\beta_k\sigma_{x_k}^2\right)^{\frac{3}{2}}} \tag{19}$$

Matrix U reduces to the following diagonal matrix:

$$U = \begin{bmatrix} \frac{\alpha_1}{\left(1+\beta_1\sigma_{x_1}^2\right)^{\frac{3}{2}}} & 0\ldots & 0 \\ 0 & \frac{\alpha_2}{\left(1+\beta_2\sigma_{x_2}^2\right)^{\frac{3}{2}}} & \cdots \\ 0 & 0\ldots & \frac{\alpha_M}{\left(1+\beta_M\sigma_{x_M}^2\right)^{\frac{3}{2}}} \end{bmatrix} \tag{20}$$

Transient MSE and Wiener MSE

The transient MSE is determined by:

$$E\left(\|e(n)\|^2\right) = E\left(\|Hg(X(n)) + N(n) - W(n)X(n)\|^2\right) = \sum_{j=1}^{L} E\left[e_j^2(n)\right] \tag{21}$$

where:

$$E\left(e_j^2(n)\right) = E\left(\left\|H_j g(X(n)) + N_j(n) - W_j(n)X(n)\right\|^2\right) \tag{22}$$

where $W_j(n) = \left[w_{j1}(n)\, w_{j2}(n)\ \ldots\ w_{jM}(n)\right]^t$ *and* $H_j = \left[h_{j1}\, h_{j2} \ldots h_{jM}\right]^t$.

Using the independence of noise and weights at time n, we get:

$$\begin{aligned} E\left(e_j^2(n)\right) &= \sigma_0^2 + E\left(\left\|(H_j g(X(n)) - W_j(n)X(n)\right\|^2\right) \\ &= \sigma_0^2 + H_j^t R_{g(X)g(X)} H_j - 2H_j^t R_{g(X)X} E\left(W_j(n)\right) \\ &\quad + E\left(W_j(n) R_{XX} W_j^t(n)\right) \end{aligned} \tag{23}$$

The total MSE is therefore expressed as:

$$\begin{aligned} E\left(\|e(n)\|^2\right) &= L\sigma_0^2 + \sum_{j=1}^{L} H_j^t R_{g(X)g(X)} H_j \\ &\quad - 2H_j^t R_{g(X)X} E\left(W_j(n)\right) \\ &\quad + E\left(W_j^t(n) R_{XX} W_j(n)\right). \end{aligned} \tag{24}$$

Wiener MSE:

The Wiener MSE, $\zeta_0 = E\left(\|e_{W_0}(n)\|^2\right)$, is the minimum MSE that can be reached by the system if W is equal to the Wiener solution $W_0 = HU$. It can be easily shown that:

$$\begin{aligned} \zeta_0 &= E\left(\|e_{W_0}(n)\|^2\right) \\ &= L\sigma_0^2 + E\left(\|Hg(X(n)) - W_0X(n)\|^2\right) \\ &= L\sigma_0^2 + E\left(\|H(g(X(n)) - UX(n))\|^2\right) \end{aligned} \tag{25}$$

It is clear from this equation that if the unknown functions are linear, then the Wiener MSE reduces to the noise power. The MSE is always larger than ζ_0 because of the misadjustment error introduced by the weight fluctuations.

Now we can write the MSE as a function of the Wiener MSE:

$$\begin{aligned} E\left(\|e(n)\|^2\right) &= E\left(\|H(g(X(n)) + N(n) - W(n)X(n))\|^2\right) \\ &= E\left(\|e_{W_0}(n) - (W(n) - W_0)X(n)]\|^2\right) \end{aligned} \tag{26}$$

Let the instantaneous deviation of the matrix weights with respect to the Wiener solution be denoted by:

$$V(n) = \left[v_{jk}(n)\right] = W(n) - W_0. \tag{27}$$

We have:

$$E\left(\|e(n)\|^2\right) = E\left(\|e_{W_0}(n) - V(n)X(n)\|^2\right). \tag{28}$$

This expression is similar to that of the well-known LMS algorithm [10], and can be evaluated as the sum of the minimum error and excess error (or misadjustment) as:

$$E\left(\|e(n)\|^2\right) = \zeta_0 + \sum_{j=1}^{L} tr\left(R_{XX} K_{V_jV_j}(n)\right) \tag{29}$$

where $V_j(n) = \left[v_{j1}\, v_{j2} \ldots v_{jM}\right]^t$ *and* $K_{V_jV_j}(n) = E\left(V_j(n)\, V_j^t(n)\right)$.

The misadjustment is expressed as:

$$\Delta(n) = tr\left(R_{XX}\sum_{j=1}^{L} R_{V_jV_j}(n)\right). \tag{30}$$

At the convergence, we have:

$$E\left(\|e(\infty)\|^2\right) = \zeta_0 + \Delta(\infty). \tag{31}$$

Derivation of the misadjustment:

From Equation (11) it is easy to show that the weight fluctuations follow the recursion:

$$V(n+1) = V(n) + 2\mu(e_{W_0}(n) - V(n)X(n))X^t(n) \tag{32}$$

Taking the mean of this equation and applying the orthogonality principle between the input vector and the Wiener error, we get:

$$E(V(n+1)) = E(V(n))(1 - 2\mu R_{XX}) \tag{33}$$

Thus, as expected, if μ is sufficiently small $E(\underline{V}(n))$ converges to 0.

Similarly, for each vector V_j we can obtain the following recursion:

$$V_j(n+1) = V_j(n) + 2\mu\left(e_{W_{0j}}(n) - X^t(n)V_j(n)\right)X(n)\,) \tag{34}$$

The evaluation of the covariance matrix of the weight fluctuations is obtained by multiplying both sides of Equation (34) by $V_j^t(n+1)$ and averaging:

$$\begin{aligned} K_{V_jV_j^t}(n+1) = {} & K_{V_j^tV_j^t}(n) - 2\mu R_{XX}K_{V_jV_j}(n) \\ & - 2\mu K_{V_jV_j}(n)R_{XX} \\ & + 2\mu E\left[e_{W_{0j}}(n)XV_j^t(I - 2\mu XX^t)\right] \\ & + 2\mu E\left[e_{W_{0j}}(n)XV_j^t(I - 2\mu XX^t)\right]^t \\ & + 4\mu^2 E\left[XX^tK_{V_jV_j}XX^t\right] \\ & + 4\mu^2 E\left[e_{W_{0j}}^2(n)XX^t\right] \end{aligned} \tag{35}$$

These expectations are derived in Appendix III, which yields:

$$\begin{aligned} K_{V_jV_j^t}(n+1) = {} & K_{V_j^tV_j^t}(n) - 2\mu R_{XX}K_{V_jV_j}(n) \\ & - 2\mu K_{V_jV_j}(n)R_{XX} \\ & + 4\mu^2\Big(-E\left[H_j^tg(X)XE\left(V_j^t(n)\right)XX^t\right] \\ & + tr\left(R_{XX}W_0E\left(V_j^t(n)\right)\right)R_{XX} \\ & + R_{XX}W_0E\left(V_j^t(n)\right)R_{XX}\Big) \\ & + 4\mu^2\Big(-E\left[H_j^tg(X)XE\left(V_j^t(n)\right)XX^t\right] \\ & + tr\left(R_{XX}W_0E\left(V_j^t(n)\right)\right)R_{XX} \\ & + R_{XX}W_0E\left(V_j^t(n)\right)R_{XX}\Big)^t \\ & + 4\mu^2\big(tr\left(R_{XX}K_{V_jV_j}(n)\right)R_{XX} \\ & + 2R_{XX}K_{V_jV_j}(n)R_{XX}\big) \\ & + 4\mu^2\Big(E\left[g(X)g^t(X)H_jH_j^tXX^t\right] \\ & + \sigma_0^2R_{XX} - tr\left(R_{XX}W_{0j}^tW_{0j}^t\right)R_{XX} \\ & - 2R_{XX}W_{0j}^tW_{0j}^tR_{XX}\Big) \end{aligned} \tag{36}$$

Taking into account that $E(V_j(\infty)) = 0$, K_{VjVj} can be obtained by solving the following equation:

$$\begin{aligned} & R_{XX}K_{V_jV_j}(\infty) + K_{V_jV_j}(\infty)R_{XX} - 2\mu\, tr\left(R_{XX}K_{V_jV_j}(\infty)\right)R_{XX} \\ & \quad - 4\mu R_{XX}K_{V_jV_j}(\infty)R_{XX} \\ & \quad = 2\mu\Big[E\left[g(X)g^t(X)H_jH_j^tXX^t\right] + 2\mu\sigma_0^2R_{XX} \\ & \quad - 2\mu tr\left(R_{XX}W_{0j}^tW_{0j}^t\right)R_{XX} - 4\mu R_{XX}W_{0j}^tW_{0j}^tR_{XX}\Big] \end{aligned} \tag{37}$$

This expression holds for any input signal. It can be simplified if $R_{XX} = \sigma_x^2 I$. In this case we have:

$$tr\left(R_{XX}K_{V_jV_j}(\infty)\right) = \mu\frac{\sigma_0^2\sigma_x^2M + tr\left(E\left(g(X)g^t(X)H_jH_j^tXX^t\right)\right) - \sigma_x^4(M+2)tr\left(W_{0j}W_{0j}^t\right)}{1 - \mu\sigma_x^2(M+2)} \tag{38}$$

It is now easy to determine the total misadjustment:

$$\Delta(\infty) = \sum_{j=1}^{L} tr\left(R_{XX}K_{V_jV_j}(\infty)\right) = \mu\frac{\sigma_0^2\sigma_x^2ML + tr\left[E\left(g(X)g^t(X)\sum_{j=1}^{L}H_jH_j^tXX^t\right)\right] - \sigma_x^4(M+2)tr\left(W_0W_0^t\right)}{1 - \mu\sigma_x^2(M+2)} \tag{39}$$

Note that, as expected, in the case of linear functions $\Delta(\infty)$ reduces to:

$$\Delta(\infty)\big|_{g(X)=X} = \frac{\mu\sigma_0^2\sigma_x^2 ML}{1-\mu\sigma_x^2(M+2)}. \tag{40}$$

The additional terms are due to the nonlinearities and they should be calculated specifically for each nonlinearity.

Application to the case study:

The MSE is expressed as:

$$E\left(\|e(n)\|^2\right) = L{\sigma_0}^2 + \sum_{j=1}^{L}\sum_{k=1}^{M}\alpha_k\sigma_{x_k}^2\left[\frac{\alpha_k}{\left(1+2\beta_i{\sigma_{x_k}}^2\right)^{\frac{3}{2}}}h_{jk}^2 - \frac{2}{\left(1+\beta_k{\sigma_{x_k}}^2\right)^{\frac{3}{2}}}h_{jk}w_{jk}(n) + w_{jk}^2(n)\right] \tag{41}$$

The Wiener MSE is expressed in this case as:

$$\zeta_0 = L{\sigma_0}^2 + \sum_{i=1}^{M}\alpha_i^2\sigma_{x_i}^2\frac{\left(1+2\beta_i{\sigma_{x_i}}^2\right)^{\frac{3}{2}}-1}{\left(1+\beta_i{\sigma_{x_i}}^2\right)^3}\sum_{j=1}^{L}h_{ji}^2 \tag{42}$$

Adaptive W, the nonlinearities are frozen and known with scale factors

In this section, matrix W is adaptive, the nonlinearities are frozen and known with scale factors (Figure 5).

Mean weight behavior and stationary points

The gradient descent update of matrix W is expressed as:

$$\begin{aligned} W(n+1) &= W(n) + 2\mu e(n)\Omega g(X(n))^t \\ &= W(n) + 2\mu[(Hg(X(n)) + N(n) \\ &\quad - W(n)\Omega g(X(n))]\Omega g(X(n))^t \end{aligned} \tag{43}$$

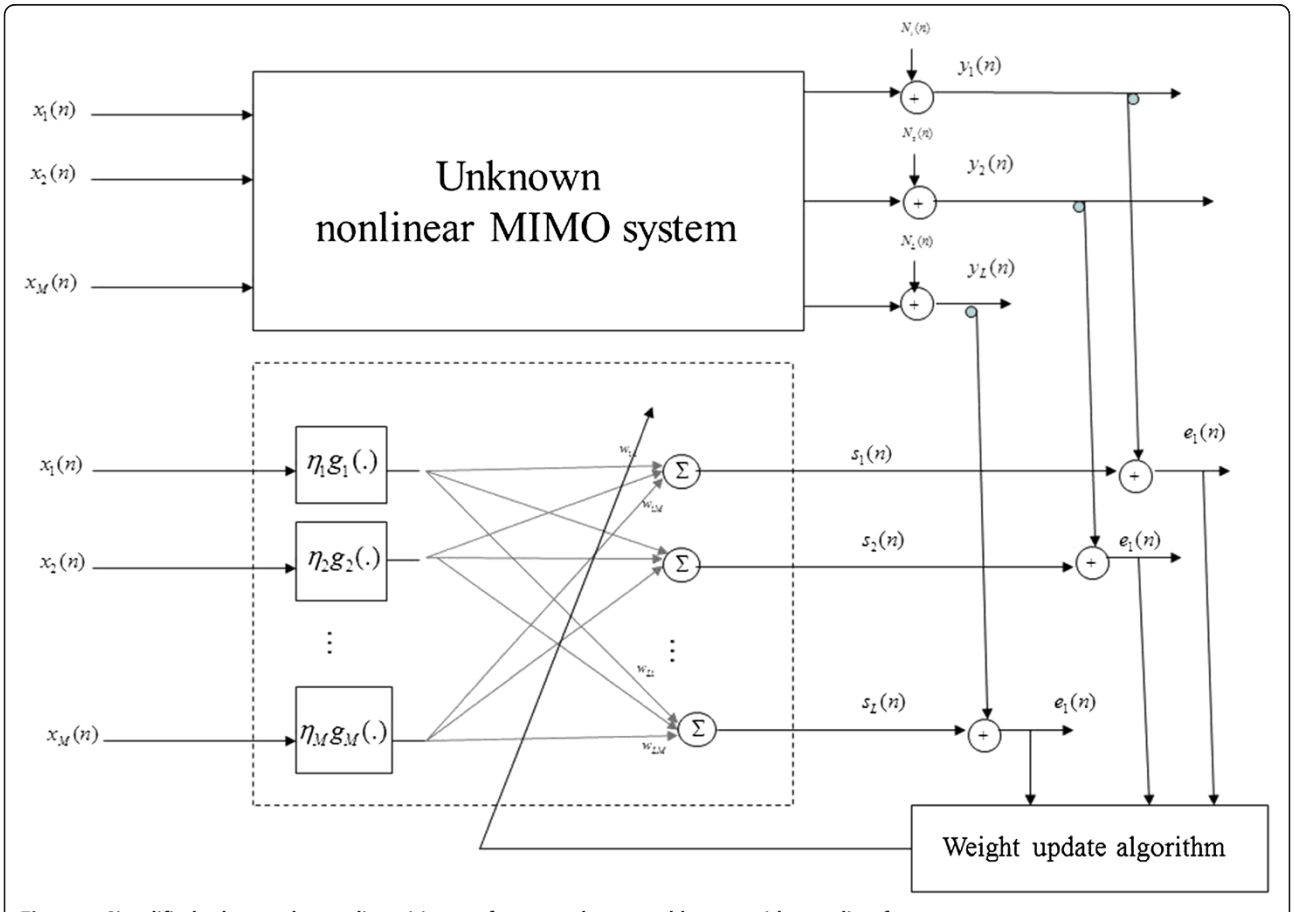

Figure 5 Simplified scheme, the nonlinearities are frozen and assumed known with a scaling factor.

where $\Omega = \begin{bmatrix} \eta_1 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & & \eta_M \end{bmatrix}$.

Averaging both sides of (43) and using the standard LMS assumption of small μ, we obtain:

$$\begin{aligned} E(W(n+1)) &\approx E(W(n)) + 2\mu\big(H\,\Omega\,R_{g(X)g(X)} \\ &\quad - E(W(n))\Omega^2 R_{g(X)g(X)}\big) \\ &= E(W(n))\big(I - 2\mu\,\Omega^2 R_{g(X)g(X)}\big) \\ &\quad + 2\mu H\,\Omega R_{g(X)g(X)} \end{aligned} \tag{44}$$

These recursions have a single stationary point (Wiener solution) which is:

$$W = W_0 = H \times \Omega^{-1} \tag{45}$$

Following Equation (44), the mean weight behavior can be expressed as function of the initial condition as:

$$\begin{aligned} E(W(n)) = W(0)\big(I - 2\mu\Omega^2 R_{g(X)g(X)}\big)^n \\ +2\mu H\,\Omega R_{g(X)g(X)} \sum_{p=0}^{n-1}\big(I-2\mu\,\Omega^2 R_{g(X)g(X)}\big)^p \end{aligned} \tag{46}$$

Hence, if μ is sufficiently small, it can be shown that the mean weights converge to the Wiener solution:

$$W(\infty) = W_0 = H\,\Omega^{-1}. \tag{47}$$

The stability condition on μ is: $0 < \mu < \frac{1}{\lambda_{\max}}$

Where $\lambda_{\max}$ is the largest eigenvalue of the covariance matrix $\Omega^2 R_{g\,(X)g(X)}$.

Thus, if each nonlinear function g_k (.) is known with a scaling factor η_k, then weights h_{jk} will be identified by w_{jk} (to the inverse of the scaling factor).

MSE

We have:

$$\begin{aligned} E\big(\|e(n)\|^2\big) &= E(\|Hg(X(n)) + N(n) \\ &\quad - W(n)\Omega g(X(n))\|^2\big) \\ &= \sum_{j=1}^{L} E\Big[e_j^2(n)\Big] \end{aligned} \tag{48}$$

where:

$$\begin{aligned} E\Big(e_j^2(n)\Big) &= E\Big(\big\|H_j g(X(n)) + N_j(n) \\ &\quad - W_j(n)\Omega g(X(n))\big\|^2\Big) \end{aligned} \tag{49}$$

Using the independence of noise and weights at time n, we obtain:

$$\begin{aligned} E\Big(e_j^2(n)\Big) &= \sigma_0^2 + E\Big(\big\|\big(H_j - \Omega\,W_j(n)\big)g(X(n))\big\|^2\Big) \\ &= \sigma_0^2 + H_j^t R_{g(X)g(X)} H_j \\ &\quad - 2H_j^t\,\Omega R_{g(X)g(X)} E\big(W_j(n)\big) \\ &\quad + E\Big(W_j^t(n)\Omega^2 R_{g(X)g(X)}\,W_j(n)\Big) \end{aligned}$$

The MSE is therefore expressed as:

$$\begin{aligned} E\big(\|e(n)\|^2\big) = L\sigma_0^2 + \sum_{j=1}^{L} H_j^t\,R_{g(X)g(X)}H_j \\ - 2H_j^t\,\Omega R_{g(X)g(X)}E\big(W_j(n)\big) \\ + E\Big(W_j^t(n)\Omega^2 R_{g(X)g(X)}\,W_j(n)\Big) \end{aligned} \tag{50}$$

Wiener MSE:

The Wiener MSE can be easily expressed as:

$$\begin{aligned} \zeta_0 &= E\Big(e_{W_0}^2(n)\Big) \\ &= L\sigma_0^2 + E\big(\|(H - W_0\Omega)g(X(n))\|^2\big) = L\sigma_0^2 \end{aligned} \tag{51}$$

Therefore the Wiener MSE is equal to the noise floor: There are no terms due to the nonlinearities. This is expected since the nonlinearities are known with a scaling matrix Ω (we have seen that the scaling matrix is canceled by W_0 since $W_0 = H\Omega^{-1}$).

Let $Z(n) = \Omega\, g(X(n))$, we can then express the MSE as a function of ζ_0, the weight fluctuation vector $V(n) = W(n) - W_0$, and $Z(n)$:

$$\begin{aligned} E\big(\|e(n)\|^2\big) &= E\big(\|e_{W_0}(n) - (W(n) - W_0)Z(n)\|^2\big) \\ &= \zeta_0 + E\big(\|V(n)Z(n)\|^2\big) \\ &= \zeta_0 + \sum_{j=1}^{L} tr\big(R_{ZZ}K_{V_jV_j}(n)\big) \end{aligned} \tag{52}$$

Similarly to Equation (29), the misadjustment is expressed as:

$$\Delta(n) = tr\left(R_{ZZ}\sum_{j=1}^{L} K_{V_jV_j}(n)\right). \tag{53}$$

The steady state MSE is then expressed as:

$$E\big(\|e(\infty)\|^2\big) = \zeta_0 + \Delta(\infty). \tag{54}$$

Derivation of the misadjustment:

It is easy to show that the weight fluctuations follow the recursion:

$$V(n+1) = V(n) + 2\mu(e_{W_0}(n) - V(n)Z(n))Z^t(n)\,). \tag{55}$$

Taking the mean of this equation and applying the orthogonality principle between the input vector and the Wiener error, we obtain:

$$E(V(n+1)) = E(V(n)) \times (1 - 2\mu R_{ZZ}). \tag{56}$$

Thus, as expected, if μ is sufficiently small, E(V(n)) converges to 0.

For each vector V_j we have similar recursions:

$$V_j(n+1) = V_j(n) + 2\mu\big(e_{W_{0j}}(n) - Z(n)^t V_j(n)\big)Z(n)\,). \tag{57}$$

The evaluation of the covariance matrix of the weight fluctuations is obtained by multiplying both sides of Equation (57) by $V_j^t(n+1)$ and averaging:

$$\begin{aligned} &K_{V_jV_j^t}(n+1) \\ &\quad = K_{V_j^tV_j^t}(n) - 2\mu R_{ZZ}K_{V_jV_j}(n) - 2\mu K_{V_jV_j}(n)R_{ZZ} \\ &\qquad + 2\mu E\Big[e_{W_{0j}}(n)Z(n)V_j^t\big(I - 2\mu Z(n)Z(n)^t\big)\Big] \\ &\qquad + 2\mu E\Big[e_{W_{0j}}(n)Z(n)V_j^t\big(I - 2\mu Z(n)Z(n)^t\big)\Big]^t \\ &\qquad + 4\mu^2 E\big[Z(n)Z(n)^t K_{V_jV_j}Z(n)Z(n)^t\big] \\ &\qquad + 4\mu^2 E\Big[e_{W_{0j}}^2(n)Z(n)Z(n)^t\Big] \end{aligned} \tag{58}$$

In a similar way as in Appendix III, K_{VjVj} (∞) can be obtained by solving the following equation:

$$\begin{aligned} &R_{ZZ}K_{V_jV_j}(\infty) + K_{V_jV_j}(\infty)R_{ZZ} - 2\mu\, tr\big(R_{ZZ}K_{V_jV_j}(\infty)\big)R_{ZZ} \\ &\quad - 4\mu R_{ZZ}K_{V_jV_j}(\infty)R_{ZZ} \\ &\qquad = 2\mu\Big[E\Big[g(X)g^t(X)H_jH_j^tZZ^t\Big] + 2\mu\sigma_0^2R_{ZZ} \\ &\qquad -2\mu tr\Big(R_{ZZ}W_{0j}^tW_{0j}^t\Big)R_{ZZ} - 4\mu R_{ZZ}W_{0j}^tW_{0j}^tR_{ZZ}\Big] \end{aligned} \tag{59}$$

This expression can not be further simplified because R_{ZZ} is not necessarily of the form $\sigma_z^2 I$.

Therefore, tr(R_{ZZ} K_{VjVj} (∞)) should be calculated for each nonlinearity and for each Ω.

It is interesting to study the case where nonlinearities are known with the same scaling factor, i.e., Ω = ηI. In this case, and if the input vectors are independent and the outputs of the nonlinearities are Zero-mean and of equal variance σ_g^2, we have:

$$tr\big(R_{ZZ}K_{V_jV_j}(\infty)\big) = \mu\frac{\sigma_0^2\eta^2\sigma_g^2M}{1 - \mu\eta^2\sigma_g^2(M+2)} \tag{60}$$

As expected, the total misadjustment reduces to:

$$\Delta(\infty) = \sum_{j=1}^{L} tr\big(R_{ZZ}K_{V_jV_j}(\infty)\big) = \mu\frac{\sigma_0^2\eta^2\sigma_g^2ML}{1 - \mu\eta^2\sigma_g^2(M+2)}. \tag{61}$$

Here the value of the misadjustment is similar to that of linear identification of a linear system (LMS algorithm). This is expected since in this case there are no errors due to the approximation of the nonlinearities.

Case study

For the case study, it is easy to show that $G_{ii} = E\big(g\big(x_i^2(n)\big)\big) = \frac{{\alpha_i}^2{\sigma_x}^2}{\big(1+2\beta_i{\sigma_{x_i}}^2\big)^{\frac{3}{2}}}$. This yields:

$$E\Big(e_j^2(n)\Big) = {\sigma_0}^2 + \sum_i^M \frac{{\alpha_i}^2{\sigma_{x_i}}^2}{\big(1+2\beta_i{\sigma_{x_i}}^2\big)^{\frac{3}{2}}}\big(h_{ji} - w_{ji}(n)\eta_i\big)^2 \tag{62}$$

Study of the full structure

This section deals with the full structure (Figures 2, 3). All the NN and matrix W weights are updated.

Mean weight transient behavior

We take the following notations for the weights: $E\big(w_{jk}(n)\big) = \bar{w}_{jk}(n), E(c_{ki}(n)) = \bar{c}_{ki}(n), E(a_{ki}(n)) = \bar{a}_{ki}(n)$, $E(b_{ki}(n)) = \bar{b}_{ki}(n)$.

The update of matrix W is expressed as:

$$\begin{aligned} W(n+1) &= W(n) + 2\mu\, e(n)NN(X(n))^t \\ &= W(n) + 2\mu[Hg(X(n)) + N(n) \\ &\quad -W(n)NN(X(n))]NN(X(n))^t \end{aligned} \tag{63}$$

Averaging both sides of (63) and using the standard LMS assumption of small μ, we obtain:

$$\begin{aligned} E(W(n+1)) &\approx E(W(n)) + 2\mu\big(HR_{g(X)NN(X)}(n) \\ &\quad -E(W(n))R_{NN(X)NN(X)}(n)\big) \\ &= E(W(n))\big(I - 2\mu R_{NN(X)NN(X)}(n)\big) \\ &\quad + 2\mu HR_{g(X)NN(X)}(n) \end{aligned} \tag{64}$$

Where $R_{NN(X)NN(X)}(n) = E\big[NN(X(n))NN(X(n))^t\big]$, $R_{g(X)NN(X)}(n) = E\big[g(X(n))NN(X(n))^t\big]$.

These matrices are time-dependent since they depend on the NN block weights which are updated through time.

Using the scalar notation we have:

$$\begin{aligned}
&\bar{w}_{jk}(n+1) = \bar{w}_{jk}(n) \\
&+ 2\mu E\left[\left(\sum_{i=1}^{M} h_{ji} g_i(x_i(n)) - \sum_{l=1}^{M} w_{jl} NN_l(n)\right)\right) \\
&\times \sum_{m=1}^{N} c_{km} f(a_{km} x_k(n) + b_{km})\Bigg] \\
&\approx \bar{w}_{jk}(n) + 2\mu\left[\sum_{i,m}^{M,N} h_{ji}\bar{c}_{km} E\left(g_i(x_i(n)) f\left(\bar{a}_{km} x_k(n) + \bar{b}_{km}\right)\right)\right. \\
&\left. - \sum_{l,m,i}^{M,N,N} \bar{w}_{jl}\bar{c}_{li}\bar{c}_{km} E\left(f\left(\bar{a}_{li} x_l(n) + \bar{b}_{li}\right) f\left(\bar{a}_{km} x_k(n) + \bar{b}_{km}\right)\right)\right]
\end{aligned} \tag{65}$$

Let: $K_i(a_{km}, b_{km}) = E(g_i(x_i(n)) f(a_{km} x_k(n) + b_{km}))$, *and* $F_{lk}(a_{li}, b_{li}, a_{km}, b_{km}) = E(f(a_{li} x_l(n) + b_{li}) f(a_{km} x_k(n) + b_{km}))$

With these notations we have:

$$\begin{aligned}
\bar{w}_{jk}(n+1) = \bar{w}_{jk}(n) + 2\mu\Bigg[&\sum_{i,m}^{M,N} h_{ji} c_{km} K_i\left(\bar{a}_{km}, \bar{b}_{km}\right) \\
&- \sum_{l,m,i}^{M,N,N} \bar{w}_{jl}\bar{c}_{li}\bar{c}_{km} F_{lk}\left(\bar{a}_{li}, \bar{b}_{li}, \bar{a}_{km}, \bar{b}_{km}\right)\Bigg]
\end{aligned} \tag{66}$$

For the NN block weights we have:

$$\begin{aligned}
&\bar{c}_{ki}(n+1) = \bar{c}_{ki}(n) + 2\mu E\left(\sum_{l=1}^{L} w_{lk} e_l(n) f(a_{ki} x_k(n) + b_{ki})\right) \\
&\approx \bar{c}_{ki}(n) + 2\mu \sum_{l=1}^{L} \bar{w}_{lk} E\left(\sum_{p=1}^{M} h_{lp} g_p(x(n)) f\left(\bar{a}_{ki} x(n) + \bar{b}_{ki}\right)\right. \\
&\left. - \sum_{m=1}^{M} w_{lm}\left(\sum_{q=1}^{N} \bar{c}_{mq} f\left(\bar{a}_{mq} x(n) + \bar{b}_{mq}\right)\right) f\left(\bar{a}_{ki} x(n) + \bar{b}_{ki}\right)\right) \\
&= \bar{c}_{ki}(n) + 2\mu \sum_{l=1}^{L} \bar{w}_{lk}\left(\sum_{p=1}^{M} h_{lp} K_p\left(\bar{a}_{ki}, \bar{b}_{ki}\right)\right. \\
&\left. - \sum_{m=1}^{M} \bar{w}_{lm}\left(\sum_{q=1}^{N} \bar{c}_{mq} F_{mk}\left(\bar{a}_{mq}, \bar{b}_{mq}, \bar{a}_{ki}, \bar{b}_{ki}\right)\right)\right)
\end{aligned} \tag{67}$$

$$\begin{aligned}
&\bar{a}_{ki}(n+1) = \bar{a}_{ki}(n) \\
&+ 2\mu E\left[c_{ki} \sum_{l=1}^{L} w_{lk} e_l(n) x(n) f'(a_{ki} x(n) + b_{ki})\right] \\
&\approx \bar{a}_{ki}(n) + 2\mu\bar{c}_{ki} \sum_{l=1}^{L} \bar{w}_{lk}\left(\sum_{p=1}^{M} h_{lp} E\left(g_p(x(n)) x(n)\right.\right. \\
&\left. \times f'\left(\bar{a}_{ki} x(n) + \bar{b}_{ki}\right)\right) - \sum_{m=1}^{M} \bar{w}_{lm}\left(\sum_{q=1}^{N} \bar{c}_{mq} E\left(f\left(\bar{a}_{mq} x(n)\right.\right.\right. \\
&\left.\left.\left. + \bar{b}_{mq}\right) x(n) f'\left(\bar{a}_{ki} x(n) + \bar{b}_{ki}\right)\right)\right)\Bigg) \\
&= \bar{a}_{ki}(n) + 2\mu\bar{c}_{ki} \sum_{l=1}^{L} \bar{w}_{lk}\left(\sum_{p=1}^{M} h_{lp} \frac{\partial K_p\left(\bar{a}_{ki}, \bar{b}_{ki}\right)}{\partial \bar{a}_{ki}}\right. \\
&- \sum_{m,q,(m,q)\neq(k,i)}^{M,N} \bar{w}_{lm}\bar{c}_{mq} \frac{\partial F_{mk}\left(\bar{a}_{mq}, \bar{b}_{mq}, \bar{a}_{ki}, \bar{b}_{ki}\right)}{\partial \bar{a}_{ki}} \\
&\left. - \frac{1}{2}\bar{w}_{lk}\bar{c}_{ki} \frac{\partial F_{ki}\left(\bar{a}_{ki}, \bar{b}_{ki}, \bar{a}_{ki}, \bar{b}_{ki}\right)}{\partial \bar{a}_{ki}}\right)
\end{aligned} \tag{68}$$

$$\begin{aligned}
&\bar{b}_{ki}(n+1) = \bar{b}_{ki}(n) \\
&+ 2\mu E\left[c_{ki} \sum_{l=1}^{L} w_{lk} e_l(n) f'(a_{ki} x(n) + b_{ki})\right] \\
&\approx \bar{b}_{ki}(n) + 2\mu\bar{c}_{ki} \sum_{l=1}^{L} \bar{w}_{lk}\left(\sum_{p=1}^{M} h_{lp} E\left(g_p(x(n)) f'\left(\bar{a}_{ki} x(n)\right.\right.\right. \\
&\left.\left. + \bar{b}_{ki}\right)\right) - \sum_{m=1}^{M} \bar{w}_{lm}\left(\sum_{q=1}^{N} \bar{c}_{mq} E\left(f\left(\bar{a}_{mq} x(n)\right.\right.\right. \\
&\left.\left.\left. + \bar{b}_{mq}\right) f'\left(\bar{a}_{ki} x(n) + \bar{b}_{ki}\right)\right)\right)\Bigg) \\
&= \bar{b}_{ki}(n) + 2\mu\bar{c}_{ki} \sum_{l=1}^{L} \bar{w}_{lk}\left(\sum_{p=1}^{M} h_{lp} \frac{\partial K_p\left(\bar{a}_{ki}, \bar{b}_{ki}\right)}{\partial \bar{b}_{ki}}\right. \\
&\left. - \sum_{m,q}^{M,N} \bar{w}_{lm}\bar{c}_{mq} \frac{\partial F_{mk}\left(\bar{a}_{mq}, \bar{b}_{mq}, \bar{a}_{ki}, \bar{b}_{ki}\right)}{\partial \bar{b}_{ki}}\right)
\end{aligned} \tag{69}$$

These equations hold for any nonlinearity. In the following, we will calculate them explicitly for the case study described in Section 2.3

Application to the case study:

Since the inputs are independent and Zero-mean, we have $K_i(a_{km}, b_{km}) = 0$, $k \neq i$, and (see Appendix I)

$$\begin{aligned}
K_i(a_{im}, b_{im}) &= E(g_i(x_i(n)) f(a_{im} x(n) + b_{im})) \\
&= \sqrt{\frac{2}{\pi}} \frac{\alpha_i \sigma_x^2}{\left(1 + \beta_i \sigma_x^2\right)} \frac{a_{im}}{\sqrt{\sigma_x^2\left(a_{im}^2 + \beta_i\right) + 1}} \\
&\quad - \frac{1}{2}\sqrt{\frac{2}{\pi}} \alpha_i \sigma_x^2 b_{im}^2 \frac{a_{im}}{\left(1 + \sigma_x^2\left(\beta_i + a_{im}^2\right)\right)^{\frac{3}{2}}}
\end{aligned} \tag{70}$$

In the other hand we have: $F_{lk}(a_{li}, b_{li}, a_{km}, b_{km}) = 0$, $l \neq k$, and (see Appendix I)

$$\begin{aligned}
&F_{kk}(a_{ki}, b_{ki}, a_{km}, b_{km}) \\
&= E(f(a_{ki}x(n) + b_{ki})f(a_{km}x(n) + b_{km})) \\
&= \frac{2}{\pi}\sin^{-1}\left(\frac{a_{ki}a_{km}\sigma_x^2}{\sqrt{1+\sigma_x^2a_{ki}^2+\sigma_x^2a_{km}^2+\sigma_x^4a_{ki}^2a_{km}^2}}\right) \\
&\quad -\frac{1}{\pi}b_{ki}^2\frac{\sigma_x^2\, a_{ki}a_{km}}{(1+\sigma_x^2a_{ki}^2)\sqrt{1+\sigma_x^2(a_{ki}^2+a_{km}^2)}} \\
&\quad -\frac{1}{\pi}b_{km}^2\frac{\sigma_x^2\, a_{ki}a_{km}}{(1+\sigma_x^2a_{km}^2)\sqrt{1+\sigma_x^2(a_{ki}^2+a_{km}^2)}} \\
&\quad + b_{ki}b_{km}\frac{2}{\pi}\frac{1}{\sqrt{1+\sigma_x^2(a_{ki}^2+a_{km}^2)}}
\end{aligned} \tag{71}$$

Inserting these expressions in equations (66)-(69), we obtain:

$$\begin{aligned}
\bar{w}_{jk}(n+1) = \bar{w}_{jk}(n) + 2\mu\Bigg[&h_{jk}\sum_{m=1}^{N}c_{km}K_k(\bar{a}_{km},\bar{b}_{km}) \\
&-\bar{w}_{jk}\sum_{m,i}^{N}\bar{c}_{ki}\bar{c}_{km}F_{kk}(\bar{a}_{ki},\bar{b}_{ki},\bar{a}_{km},\bar{b}_{km})\Bigg]
\end{aligned} \tag{72}$$

$$\begin{aligned}
\bar{c}_{ki}(n+1) = \bar{c}_{ki}(n) + 2\mu\sum_{l=1}^{L}\bar{w}_{lk}\Big(&h_{lk}K_k(\bar{a}_{ki},\bar{b}_{ki}) \\
&-\bar{w}_{lk}\left(\sum_{q=1}^{N}\bar{c}_{kq}F_{kk}(\bar{a}_{kq},\bar{b}_{kq},\bar{a}_{ki},\bar{b}_{ki})\right)\Big)
\end{aligned} \tag{73}$$

$$\begin{aligned}
\bar{a}_{ki}(n+1) = \bar{a}_{ki}(n) + 2\mu\bar{c}_{ki}\sum_{l=1}^{L}\bar{w}_{lk}\Bigg(&h_{lk}\frac{\partial K_k(\bar{a}_{ki},\bar{b}_{ki})}{\partial\bar{a}_{ki}} \\
&-\bar{w}_{lk}\sum_{q\neq k}^{N}\bar{c}_{kq}\frac{\partial F_{kk}(\bar{a}_{kq},\bar{b}_{kq},\bar{a}_{ki},\bar{b}_{ki})}{\partial\bar{a}_{ki}} \\
&-\frac{1}{2}\bar{w}_{lk}\bar{c}_{kk}\frac{\partial F_{kk}(\bar{a}_{kk},\bar{b}_{kk},\bar{a}_{ki},\bar{b}_{ki})}{\partial\bar{a}_{kk}}\Bigg)
\end{aligned} \tag{74}$$

$$\begin{aligned}
\bar{b}_{ki}(n+1) = \bar{b}_{ki}(n) + 2\mu\bar{c}_{ki}\sum_{l=1}^{L}\bar{w}_{lk}\Bigg(&h_{lk}\frac{\partial K_k(\bar{a}_{ki},\bar{b}_{ki})}{\partial\bar{b}_{ki}} \\
&-\sum_{q}^{N}\bar{w}_{lk}\bar{c}_{kq}\frac{\partial F_{kk}(\bar{a}_{kq},\bar{b}_{kq},\bar{a}_{ki},\bar{b}_{ki})}{\partial\bar{b}_{ki}}\Bigg)
\end{aligned} \tag{75}$$

The explicit expressions of the different derivatives are detailed in Appendix II.

Stationary points

We obtain the stationary points by setting to 0 the expectations of the updating gradient terms in (64) and (4.5-7).

For W, we obtain:

$$\begin{aligned}
&W_0 = H \times R_{g(X)NN(X)}R_{NN(X)NN(X)}^{-1} = H \times U, \textit{where} \\
&U = R_{g(X)NN(X)}R_{NN(X)NN(X)}^{-1}.
\end{aligned} \tag{76}$$

For c_{ki} we obtain the equations:

$$\begin{aligned}
\sum_{l=1}^{L}w_{lk}\Big(&h_{lk}K_k(a_{ki},b_{ki}) \\
&-\bar{w}_{lk}\left(\sum_{q=1}^{N}c_{kq}F_{kk}(a_{kq},b_{kq},a_{ki},b_{ki})\right)\Big) = 0
\end{aligned} \tag{77}$$

For a_{ki} we obtain the equations:

$$\begin{aligned}
\sum_{l=1}^{L}w_{lk}\Bigg(&h_{lk}\frac{\partial K_k(a_{ki},b_{ki})}{\partial a_{ki}} \\
&-\bar{w}_{lk}\sum_{q\neq k}^{N}\bar{c}_{kq}\frac{\partial F_{kk}(a_{kq},b_{kq},a_{ki},b_{ki})}{\partial a_{ki}} \\
&-\frac{1}{2}w_{lk}c_{kk}\frac{\partial F_{kk}(a_{kk},b_{kk},a_{ki},b_{ki})}{\partial a_{kk}}\Bigg) = 0
\end{aligned} \tag{78}$$

For b_{ki} we obtain:

$$\begin{aligned}
\sum_{l=1}^{L}w_{lk}\Bigg(&h_{lk}\frac{\partial K_k(a_{ki},b_{ki})}{\partial b_{ki}} \\
&-\sum_{q}^{N}w_{lk}c_{kq}\frac{\partial F_{kk}(a_{kq},b_{kq},a_{ki},b_{ki})}{\partial b_{ki}}\Bigg) = 0
\end{aligned} \tag{79}$$

The above equations are nonlinear in the NN variables. They can be solved numerically, but they are very difficult to solve analytically.

Convergence of the algorithm to the stationary points:

It is always interesting to show whether an algorithm is capable of converging to its stationary points or not. In our case it is difficult to establish this, since the updating equations of the weights are nonlinear, except for W.

In the case where the NN weights are frozen we can establish the convergence condition for W.

In this case we have:

$$\begin{aligned}
E(W(n+1)) = E(W(n))\big(&I - 2\mu R_{NN(X)NN(X)}\big) \\
&+ 2\mu H R_{g(X)NN(X)}
\end{aligned} \tag{80}$$

The covariance matrices are fixed, since in this case the NN weights are frozen.

$E(W(n))$ can be expressed as a function of the initial condition as:

$$E(W(n)) = W(0)\left(I - 2\mu R_{NN(X)NN(X)}\right)^n + 2\mu H R_{g(X)NN(X)} \times \sum_{p=0}^{n-1}\left(I - 2\mu R_{NN(X)NN(X)}\right)^p \quad (81)$$

If μ is sufficiently small, the steady state solution to (81) is:

$$W(\infty) = W_0 = H R_{g(X)NN(X)} R_{NN(X)NN(X)}^{-1}. \quad (82)$$

Hence, the mean weights converge to the stationary point W_0, and the stability condition on μ is:

$$0 < \mu < \frac{1}{\lambda_{\max}} \quad (83)$$

Where $\lambda_{\max}$ is the largest eigenvalue of the correlation matrix $R_{NN(X)NN(X)}$.

Application to the case study:

For the case study it can be shown that U reduces to a diagonal matrix:

$$U = R_{g(X)NN(X)} R_{NN(X)NN(X)}^{-1} = \begin{bmatrix} \gamma_1 & 0 \dots & 0 \\ 0 & \gamma_2 \dots & 0 \\ 0 & 0 \dots & \gamma_M \end{bmatrix}, \quad (84)$$

where:

$$\gamma_k = \frac{\sum_m^N K_k(a_{km}, b_{km})}{\sum_{m,i}^{N,M} c_{ki} c_{km} F_{kk}(a_{ki}, b_{ki}, a_{km}, b_{km})} \quad (85)$$

This indicates that weights w_{jk} are scaled versions of the unknown weights h_{jk}, the scale factor γ_k is the same for all the weights connecting the k^{th} NN block to the outputs and it depends only on block k weights. If the error is sufficiently small, the k^{th} block NN will approximate the k^{th} nonlinearity to the inverse of the scale factor.

MSE expression

The transient MSE is determined by:

$$E\left(\|e(n)\|^2\right) = \sum_{j=1}^{L} E\left[e_j^2(n)\right] = E\left(\|Hg(X(n)) + N(n) - W(n)NN(X(n))\|^2\right) \quad (86)$$

where:

$$E\left(e_j^2(n)\right) = E\left(\left\|H_j g(X(n)) + N_j(n) - W_j(n)NN(X(n))\right\|^2\right) = E\left(\left(\sum_{i=1}^{M} h_{ji} g_i(x_i(n)) + N_j(n) - \sum_{k-1}^{M} w_{jk} \times \sum_{i=1}^{N} c_{ki} f(a_{ki} x_k(n) + b_{ki})\right)^2\right) \quad (87)$$

Which can be expressed as:

$$E\left(e_j^2(n)\right) = {\sigma_0}^2 + \sum_{i,l}^{M} h_{ji} h_{jl} E(g_i(x_i(n)) g_l(x_l(n))) - 2\sum_{i,k}^{M} h_{ji}(n) w_{jk}\left(\sum_{m=1}^{N} c_{km} E(g_i(x_i(n)) f(a_{km} x_k(n) + b_{km}))\right) + \sum_{k,l}^{M}\sum_{i,m}^{N} w_{jl} w_{jk} c_{li} c_{ki} E(f(a_{li} x_l(n) + b_{li}) f(a_{km} x_k(n) + b_{km})) \quad (88)$$

Let $G_{il} = E(g_i(x_i(n)) gl(x_i(n)))$. Using the notations of Section 4.1, we have:

$$E\left(e_j^2(n)\right) = {\sigma_0}^2 + \sum_{i,l}^{M} h_{ji} h_{jl} G_{il} - 2\sum_{i,k}^{M} h_{ji} w_{jk}\left(\sum_{m=1}^{N} c_{km} K_i(a_{km}, b_{km})\right) + \sum_{k,l}^{M}\sum_{i,m}^{N} w_{jl} w_{jk} c_{li} c_{km} F_{lk}(a_{li}, b_{li}, a_{km}, b_{km}) \quad (89)$$

Application to the case study:

It can be easily shown that:

$$E\left(e_j^2(n)\right) = {\sigma_0}^2 + \sum_{i}^{M} h_{ji}^2 G_{ii} - 2\sum_{k}^{M} h_{jk} w_{jk}\left(\sum_{m=1}^{N} c_{km} K_k(a_{km}, b_{km})\right) + \sum_{k}^{M} w_{jk}^2 \sum_{i,m}^{N} c_{ki}^2 F_{kk}(a_{ki}, b_{ki}, a_{km}, b_{km}). \quad (90)$$

The 1^{st} term of $E\left(e_j^2(n)\right)$ represents the noise power, the 2^{nd} term is the signal power of the j^{th} MIMO output, the 3^{rd} term is the sum of the individual contributions of the neurons weighed by W and H weights, the 4^{th} term represents the sum of the coupling terms between

neurons inside the same block weighed by W. Note that since the inputs are Zero-mean and independent, there are no coupling terms between neurons in different blocks (as in Eq. (89)).

The total MSE is then expressed as:

$$E\left(\|e(n)\|^2\right) = L{\sigma_0}^2 + \sum_{j,i}^{M} h_{ji}^2 G_{ii} - 2\sum_{j,k}^{M} h_{jk} w_{jk}\left(\sum_{m=1}^{N} c_{km} K_k(a_{km}, b_{km})\right) + \sum_{j,k}^{M} w_{jk}^2 \sum_{i,m}^{N} c_{ki}^2 F_{kk}(a_{ki}, b_{ki}, a_{km}, b_{km}). \tag{91}$$

Case of frozen NN weights:

It is interesting to see the behavior of the MSE in the case where the NN weights are frozen.

In this case we have:

$$\zeta_0 = E\left(\|e_{W_0}(n)\|^2\right) = L\sigma_0^2 + E\left(\|Hg(X(n)) - W_0 NN(X(n))\|^2\right). \tag{92}$$

Here the minimum MSE depends on the noise floor and on the NN approximation error of the nonlinearities. It is clear from this equation and from Section 3.2 that, if the NN blocks ideally identify the nonlinearities (to within scale factors), then ζ_0 reduces to the noise floor.

The MSE can be written as a function of ζ_0 as:

$$E\left(\|e(n)\|^2\right) = E\left(\|e_{W_0}(n) - (W(n) - W_0)NN(X(n))]\|^2\right) = E\left(\|e_{W_0}(n) - V(n)X(n)\|^2\right) \tag{93}$$

The steady state MSE is in this case:

$$E\left(\|e(\infty)\|^2\right) = \zeta_0 + \sum_{j=1}^{L} tr\left(R_{NN(X)NN(X)} K_{V_j V_j}(\infty)\right). \tag{94}$$

The misadjustment can be derived similarly as in Sections 3.1 and 3.2. We obtain a similar equation as (53), by replacing R_{ZZ} by $R_{NN(X)NN(X)}$. The equation can not be simplified further.

It is interesting to notice, however, that if the NN blocks perfectly identify the nonlinearities and if the conditions above equation (60) are fulfilled, then:

$$\Delta(\infty) = \sum_{j=1}^{L} tr\left(R_{NN(X)NN(X)} K_{V_j V_j}(\infty)\right) = \mu \frac{\sigma_0^2 \sigma_g^2 ML}{1 - \mu\sigma_g^2(M+2)} \tag{95}$$

Simulation examples

In this section we present some simulation results which are applied to the case study described in Section 2.3. In these simulations, we have considered a 2×2 MIMO system (i.e., $M = L = 2$). For the parameterized nonlinearities we have chosen α_1=α_2=1, β_1=1, β_2=2. Unless otherwise specified, the inputs are uncorrelated Zero-mean white Gaussian processes with σ_{xi} = 1. In the simulations, the unknown combining matrix was fixed and was taken as $H = \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix}$. For example, in a MIMO communication system, H can be seen as the propagation matrix between 2 transmitting antennas and 2 receiving antennas.

Linear adaptation

For the linear adaptation case (Section 3.1), the adaptive system is composed of a 2x2 matrix W. For the noise we have taken σ_0 = 0.001. The mean weight recursions and the MSE transient behaviors (Figures 6, 7) have been estimated over 20 Monte Carlo (MC) simulations and compared to the theoretical derivations (Equations (19) and (41)). This chosen number of MC simulations shows excellent fit between the Theory and MC estimations which confirms the validity of the different assumptions made. A larger number of MC simulations allows a better smoothing of the curves, but the conclusions remain the same.

Matrix W converges to a scaled version of H: $W_0 = \begin{bmatrix} 0.3536 & 0.0577 \\ 0.1061 & 0.1925 \end{bmatrix} = \mathrm{H} \quad \mathrm{U},\ U = \begin{bmatrix} 0.3536 & 0.0000 \\ 0.0000 & 0.1925 \end{bmatrix}$. Note the typical behavior of the LMS algorithm: A time constant controls the transient part of the learning curve

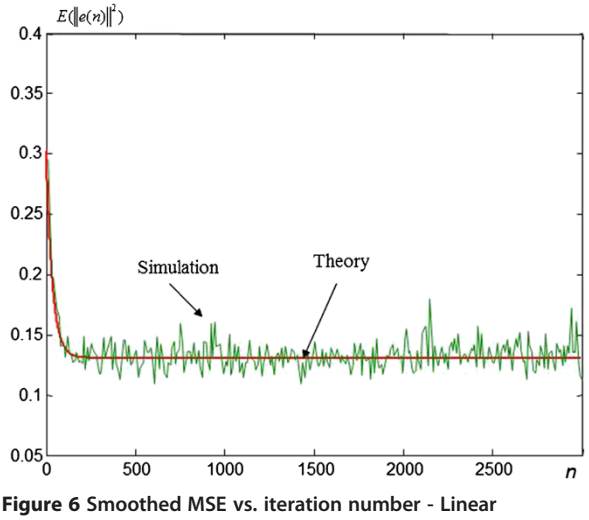

Figure 6 Smoothed MSE vs. iteration number - Linear adaptation case.

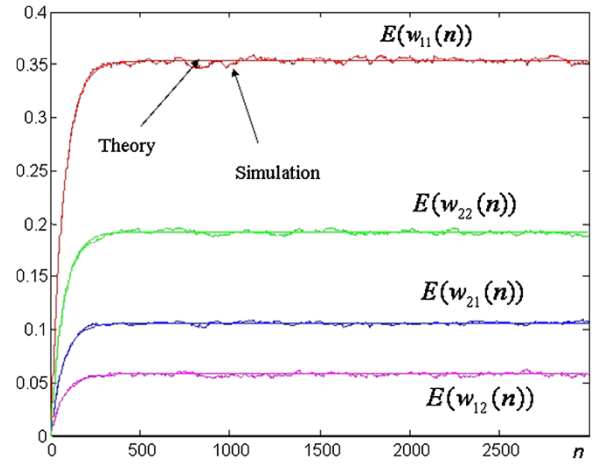

Figure 7 Mean weight transient behavior - Linear adaptation case.

and the mean weight curve. This is fundamentally different from the full NN system learning which is governed by several time constants and presents plateau regions (Section 5.2). It should be noted that the steady state MSE is high because of the error caused by the fact that the nonlinearities are not approximated (actually they are modeled by the identity function) (Equation (25)).

MSE surface for the full NN algorithm

We move now to the study of the full NN algorithm. In this simulation, we have taken N = 3 neurons in each of the two NN blocks. The learning rate was fixed to μ =0.0045. Figure 8 shows the MSE surface (i.e., Eq. (90), with no time dependence) as a function of w_{11} and w_{12} (the other parameters were fixed). It is clear that the MSE is quadratic in w_{11} and w_{21}. It presents a single global minimum (as shown in Equations (76 and 84)).

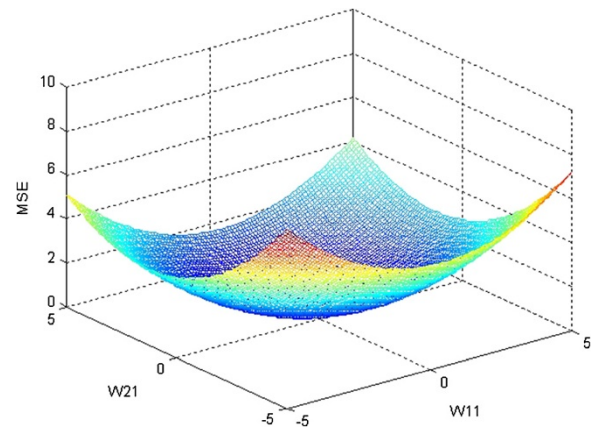

Figure 8 MSE surface, *MSE* (w_{11}, w_{21}).

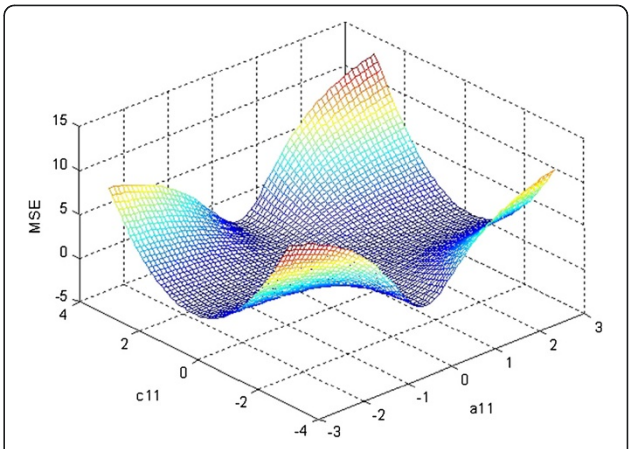

Figure 9 MSE surface, *MSE* (a_{11}, c_{21}).

Figure 9 (resp. Figure 10) shows the MSE surface as a function of a_{11} and c_{11} (resp. a_{11} and a_{12}) (the other parameters were fixed). It can be noted the flat areas (plateau regions) around the minima of the MSE surface. This explains the slow evolution of the NN weights when the algorithm gets close to its convergence point.

The MSE evolution during the learning process (Equation (90)) has been compared to 20 MC estimations (Figure 11a): The theory shows very good fit with the simulation results. In the Figure, we notice that the MSE presents several phases (each phase is controlled by a time constant) which end by a plateau phase where the MSE decreases very slowly. This is a typical behavior of the backpropagation algorithm [1] which is fundamentally different from that of the linear adaptation scheme (Figure 6). Here the MSE error is much smaller. This is expected, since here the additional MSE error due to the nonlinearities (Eq. 92) is highly reduced because our NN

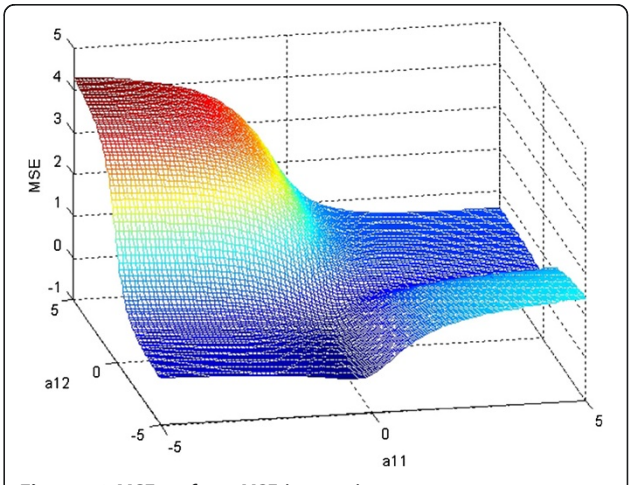

Figure 10 MSE surface, *MSE* (a_{11}, a_{12}).

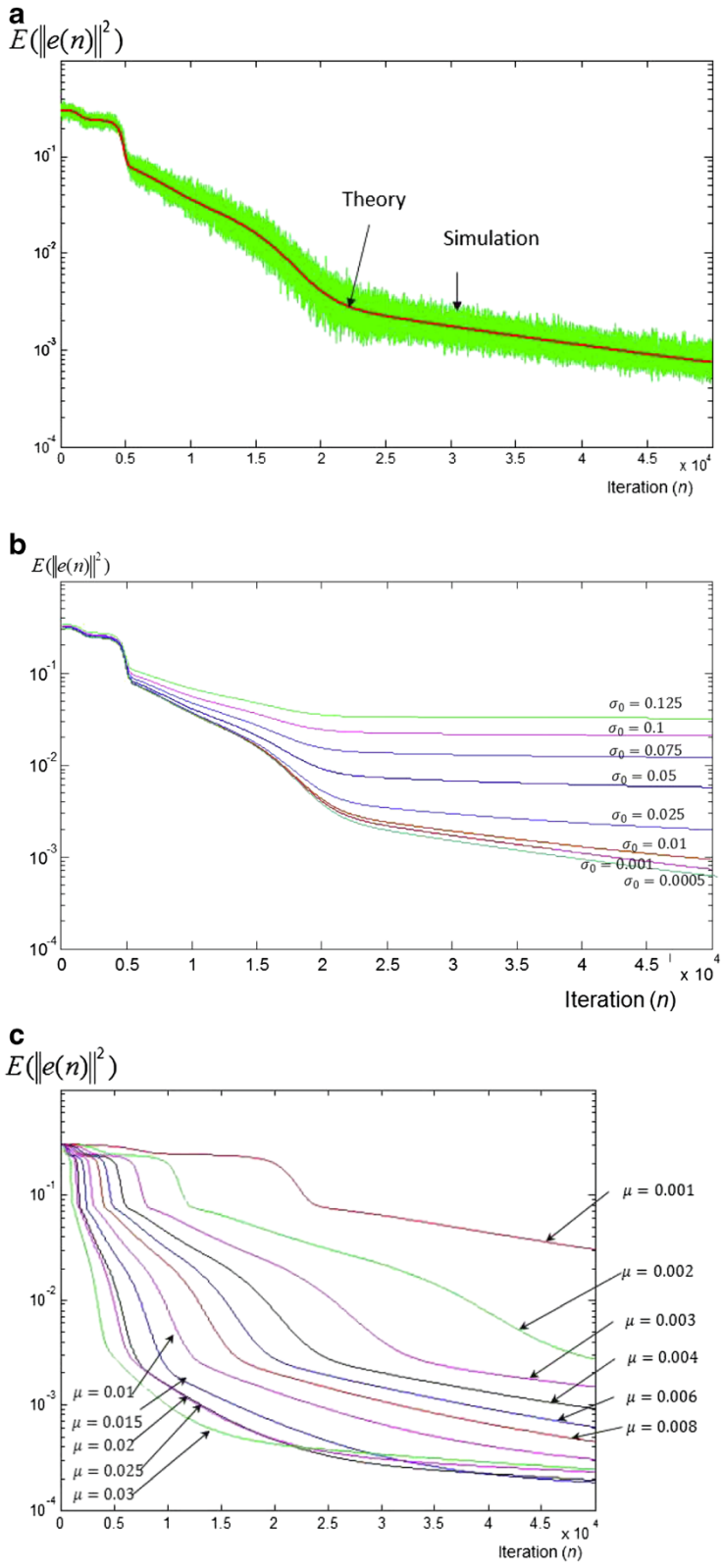

Figure 11 a: MSE during the learning process (theory and simulation), σ_0=0.001 and μ=0.0045. b: MSE for different values of noise variance, the learning rate was μ=0.0045 (theoretical results). **c** : MSE for different values of learning rate μ (the noise variance was set to σ_0=0.001), theoretical results.

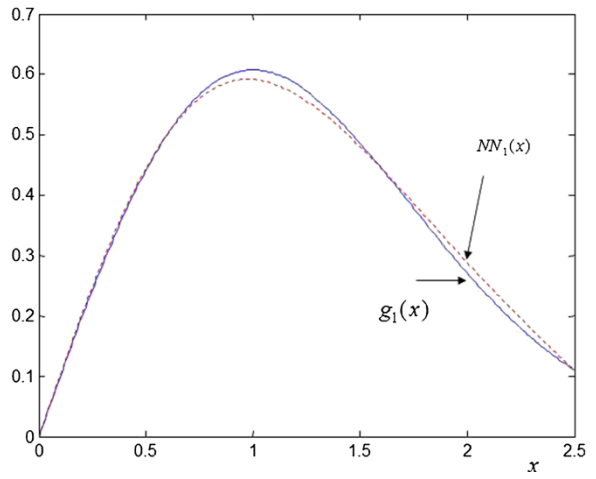

Figure 12 Identification of g_1 (x), by the 1st NN block.

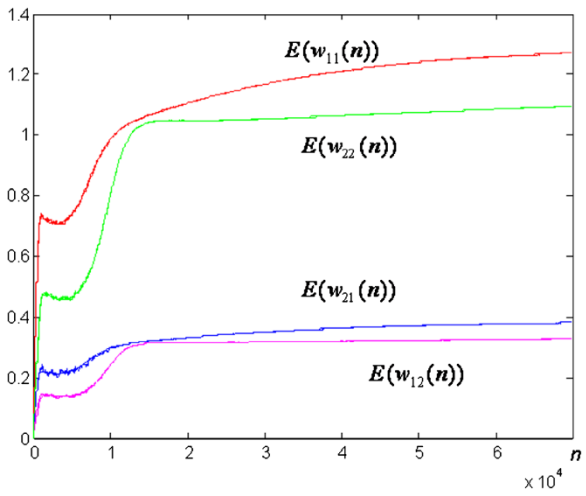

Figure 14 Mean weight behavior: Matrix W (theory and simulation).

blocks have correctly identified the unknown nonlinearities (Figure 12, 13). Here we are in a situation close to that of Section 3.2 (Equations 51-52).

Figure 11b shows the MSE evolution during the learning process for different values of the noise variance σ_0. It can be seen that, as the noise variance decreases, the MSE decreases. However, below a certain value of σ_0 (here σ_0=0.0005), the MSE curves are almost identical. This is because in this case, the weight misadjustment error (for the linear part) and the nonlinear approximation error (of the nonlinear memoryless part) are much higher than the error caused by the presence of noise (see Eqs. 92-93).

Figure 11c investigates the influence of the learning rate μ. It can be seen that as μ increases (up to μ=0.002), the algorithm is faster and the MSE is lower at the end of the simulation time. However, for μ>0.002, as μ increases, the algorithm is faster at the beginning of the learning process, but the MSE is higher at the end of the simulation time. This is due to the misadjustment error which is higher for higher μ (see, e.g., Eq. 95).

Mean weight transient behavior for the full NN algorithm

Here we keep the system described in Section 5.2. The mean weight recursions for the linear combiner W and the two NN blocks are shown in Figure 14, 15, 16 for both theory and MC estimations. The theoretical and estimated curves are indistinguishable. This confirms the validity of the different assumptions made in Sections 4.1 and 4.2.

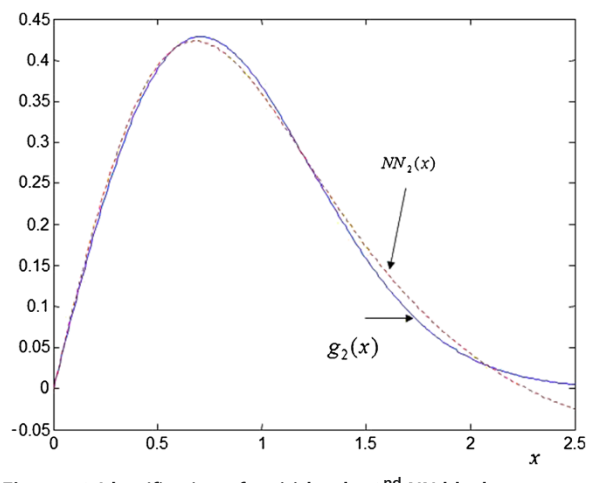

Figure 13 Identification of g_2 (x) by the 2nd NN block.

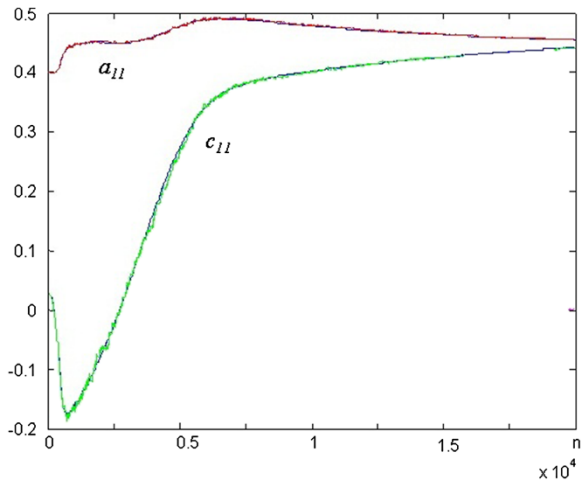

Figure 15 Mean weight behavior: First NN block (theory and simulation).

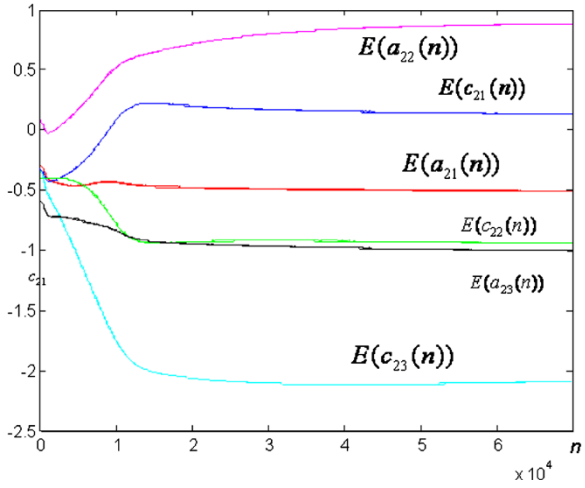

Figure 16 Mean weight behavior: Second NN block (theory and simulation vs. iterations).

Notice that, in Figure 14, W weights have a fast evolution at the beginning of the learning process (with values approaching $H \times U(n)$ where U is a diagonal matrix). They then evolve slowly till the end of the learning process. The slow evolution is justified by the plateau regions presented by the MSE surface. At the end of this simulation, matrix U was close to a diagonal matrix: $U_{Sim} = \begin{bmatrix} 1.2702 & 0.0001 \\ 0.0003 & 1.0946 \end{bmatrix}$, (and $U_{Theory} = \begin{bmatrix} 1.270 & 0 \\ 0 & 1.095 \end{bmatrix}$). This result is expected since the inputs are uncorrelated (Equations 84-85).

Figure 12, 13 show that functions $g_1(x)$ and $g_2(x)$ have been correctly identified by the corresponding NN blocks (the NN functions are normalized by the scaling factors γ_1=1/1.2702, γ_1=1/1.0946, respectively).

Impact of correlated inputs

In the simulations below we study the impact of correlated inputs. The input signal vector is chosen here as a 2D Gaussian process with covariance matrix of the form $R_{XX} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$. The number of neurons in each NN block was taken as N = 5 neurons. The learning rate μ=0.0075. We have run several simulations for different values of the cross correlation. Note that (ρ=0) corresponds to independent inputs, and (ρ=1) corresponds to the same input (i.e., x_1=x_2). The values of matrix U and the MSE after n=210^5 iterations are shown in Table 1. It can be seen from Table 1 that, in practice, matrix U remains very close to a diagonal matrix, even for high correlation between inputs. This indicates that the system is capable of correctly identifying the nonlinearities even when the inputs are highly correlated. The identification performances for the cases (ρ=0.6) and (ρ=0.99) are illustrated in Figures 17, 18 and 19, 20, respectively. As expected, the MSE increases as the correlation between inputs increases (Table 1). When ρ=1 (i.e., the two inputs are the same) the system is capable of correctly identifying the overall MIMO input–output transfer function. However, in this case, it is not capable of separating the nonlinearities (as U is not diagonal). The reason is that in this case, the system is seen by the learning algorithm as a 1x2 SIMO system which has several equivalent structures. Figure 21 shows an example of two equivalent structures. Therefore, the adaptive system is structurally not able to separate the nonlinearities. It is worth to note that for the case (ρ=0.999), the inputs look like noisy versions of each other (i.e., this is equivalent to a 1x2 system identification problem with noisy inputs). Thus, the MSE for the case (ρ=0.999) is larger than the MSE for the case (ρ=1).

Conclusion and future work

The paper provides a statistical analysis of NN modeling and identification of a class of nonlinear MIMO systems. The study investigates the MSE error, mean weight behavior, stationary points, misadjustment error, and stability conditions. The unknown system is composed of a set of single-input memoryless nonlinearities followed by a combining matrix. The NN model is composed of a set of single-input memoryless NN blocks followed by an adaptive linear combiner. The paper is supported with simulation results which show good agreement between the theoretical recursions and MC simulations. Future work will focus on 3 research directions. The first will explore the theoretical findings in order to express the effect of the number of neurons on the transient and steady state behavior of the algorithm. The second research axis will investigate the

Table 1 Effect of correlated inputs

	ρ=0	ρ=0.6	ρ=0.9	ρ=0.99	ρ=0.999	ρ=1 (same input)
$E(U(n))$, for $n=2\times10^5$	$\begin{bmatrix} -1.228 & 0.0000 \\ 0.0004 & -1.066 \end{bmatrix}$	$\begin{bmatrix} -1.2245 & 0.0004 \\ 0.001 & -1.1331 \end{bmatrix}$	$\begin{bmatrix} -1.23 & 0.001 \\ 0.0025 & -1.12 \end{bmatrix}$	$\begin{bmatrix} -1.24 & 0.019 \\ 0.023 & -1.08 \end{bmatrix}$	$\begin{bmatrix} -0.324 & -1.07 \\ -1.002 & -0.03 \end{bmatrix}$	$\begin{bmatrix} -0.023 & -1.24 \\ -1.14 & -0.15 \end{bmatrix}$
MSE(n)	10^{-4}	1.25 10^{-4}	1.5 10^{-4}	1.75 10^{-4}	9 10^{-4}	1.7 10^{-4}

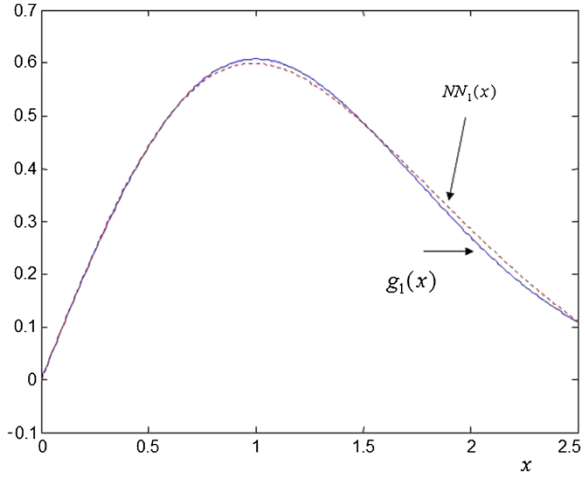

Figure 17 Identification of g_1 (x), case ρ=0.6.

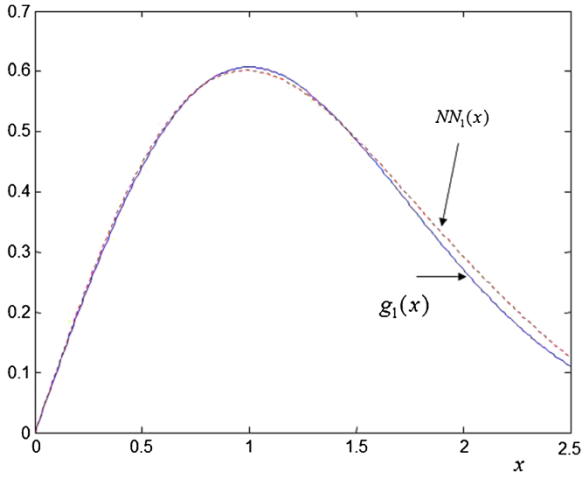

Figure 19 Identification of g_1 (x), case ρ=0.99.

case where matrix H is time-varying and/or with memory (this may have applications, for example, in adaptive control of nonlinear dynamical MIMO systems). Finally, we will study the algorithm behavior and performance for specific inputs (such as space-time coded signals used in wireless communications and their impact on the system capacity).

Endnotes

This work has been supported by The Natural Sciences and Engineering Research Council of Canada (NSERC).

The time index of the weights has been omitted from the right hand side of the equations to make them easier to read.

Appendix I

1) Calculation of F_{kk}

Let x_1 and x_2 be two zero-mean Gaussian variables such that $\sigma^2_{x_1} = \sigma^2_{x_2} = \sigma^2_x$ *and* $E(x_1x_2) = \rho$

Therefore, $F_{kk}(a_{ki}, b_{ki}, a_{km}, b_{km}) = E(f(a_{ki}x_k(n)+ b_{ki})f(a_{km}x_k(n) + b_{km}))_{\rho=\sigma^2_x}$.

Using Price's theorem we have:

$E\left[\frac{\partial^2 f(a_{ki}x_1+b_{ki})f(a_{km}x_2+b_{km})}{\partial x_1 \partial x_2}\right] = \frac{\partial E[f(a_{ki}x_1+b_{ki})f(w_{km}x_2+b_{km})]}{\partial \rho}$.

Let $U(\rho) = E\left[\frac{\partial^2 f(a_{ki}x_1+b_{ki})f(w_{km}x_2+b_{km})}{\partial x_1 \partial x_2}\right]$,

Then: $E[f(a_{ki}x_1 + b_{ki})f(a_{km}x_2 + b_{km})]_{\rho=\sigma^2_x} -$

$E[f(a_{ki}x_1 + b_{ki})f(a_{km}x_2 + b_{km})]_{\rho=0} = \int_0^{\sigma^2_x} U(\rho)d\rho$.

Thus, using the un-correlation criteria between x_1 and x_2 for ρ=0 , we have: $E[f(a_{ki}x_1 + b_{ki})f(a_{km}x_2+ b_{km})]_{\rho=0} = E[f(a_{ki}x_1 + b_{ki})]E[f(a_{km}x_2 + b_{km})]$. Thus:

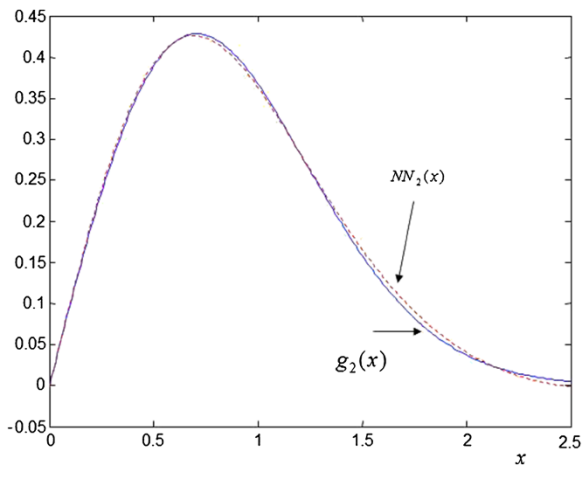

Figure 18 Identification of g_2 (x), case ρ=0.6.

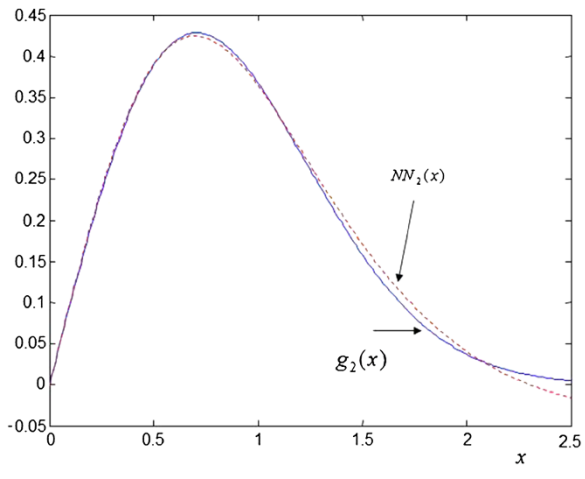

Figure 20 Identification of g_2 (x), case ρ=0.99.

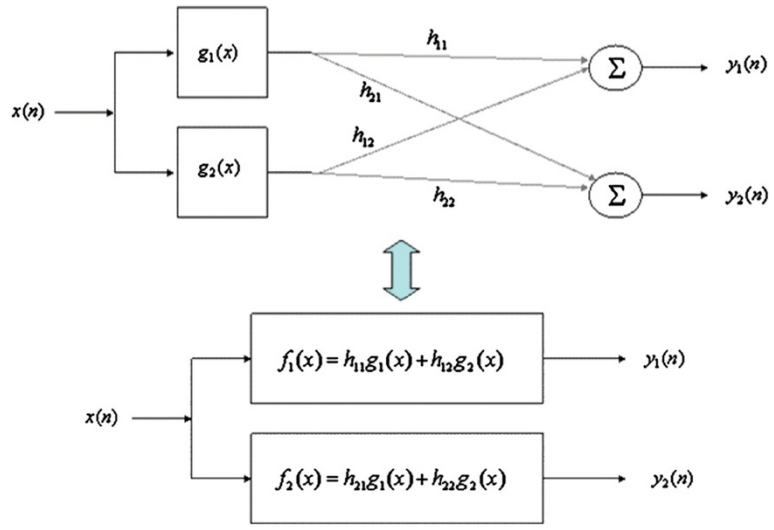

Figure 21 Example of two equivalent structures when the inputs are the same.

$F(a_{ki}, b_{ki}, a_{km}, b_{km}) = E[f(a_{ki}x_1 + b_{ki})]E[f(a_{km}x_2 + b_{km})] + \int_0^{\sigma_x^2} U(\rho)d\rho$.

We have: $U(\rho) = E\left[\frac{\partial^2 f(a_{ki}x_1+b_{ki})f(a_{km}x_2+b_{km})}{\partial x_1 \partial x_2}\right] = \frac{2}{\pi}a_{ki}a_{km}\frac{1}{2\pi|R|^{\frac{1}{2}}}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} e^{-\frac{1}{2}(a_{ki}x_1+b_{ki})^2} e^{-\frac{1}{2}(a_{km}x_2+b_{km})^2} e^{-\frac{1}{2}X^t R^{-1}X}dx_1dx_2$ where $X=[x_1\ x_2]^t$ *and* $R=\begin{bmatrix}\sigma_x^2 & \rho\\ \rho & \sigma_x^2\end{bmatrix}$.

Combining the terms in the exponentials and completing the squares, the integrals can be calculated: $U(\rho) = \frac{2}{\pi}\frac{a_{ki}a_{km}}{\sqrt{1+\sigma_x^2a_{ki}^2+\sigma_x^2a_{km}^2+(\sigma_x^4-\rho^2)a_{ki}^2a_{km}^2}} \times \exp\left[\frac{1}{2}\left[-b_{ki}^2 - b_{km}^2 + \frac{1}{a_{ki}^2+\frac{\sigma_x^2}{\sigma_x^2-\rho^2}}\left[b_{ki}^2a_{ki}^2 + \frac{\left(b_{ki}a_{ki}\left(a_{ki}^2+\frac{\sigma_x^2}{\sigma_x^2-\rho^2}\right)+b_{km}a_{km}\frac{\rho}{\sigma_x^2-\rho^2}\right)^2}{\left(a_{ki}^2+\frac{\sigma_x^2}{\sigma_x^2-\rho^2}\right)\left(a_{km}^2+\frac{\sigma_x^2}{\sigma_x^2-\rho^2}\right)a_{km}^2-\frac{\rho^2}{\sigma_x^2-\rho^2}}\right]\right]\right]$.

Note that in the biasless case (i.e. all the bias terms are set to 0) this expression reduces to:

$$U(\rho) = \frac{2}{\pi}\frac{a_{ki}a_{km}}{\sqrt{1+\sigma_x^2a_{ki}^2+\sigma_x^2a_{km}^2+(\sigma_x^4-\rho^2)a_{ki}^2a_{km}^2}}.$$

The integral is then simple to calculate:

$$\int_0^{\sigma_x^2}U(\rho)d\rho=\frac{2}{\pi}\sin^{-1}\left(\frac{a_{ki}a_{km}\sigma_x^2}{\sqrt{1+\sigma_x^2a_{ki}^2+\sigma_x^2a_{km}^2+\sigma_x^4a_{ki}^2a_{km}^2}}\right)$$

In the other hand, since $E[f(w_kx_1)] = E[f(w_ix_2)] = 0$, then:

$$F(a_{ki}, a_{km}, 0, 0) = \frac{2}{\pi}\sin^{-1}\left(\frac{a_{ki}a_{km}\sigma_x^2}{\sqrt{1+\sigma_x^2a_{ki}^2+\sigma_x^2a_{km}^2+\sigma_x^4a_{ki}^2a_{km}^2}}\right).$$

When the bias terms are not set to 0, a Taylor series expansion on the bias terms can be used in order to avoid the calculation of the integral.

2) Calculation of K

$$K_k(a_{km}, b_{km}) = E[g_k(x_k)f(a_{km}x_k + a_{km})] = \frac{1}{\sqrt{2\pi}}\frac{1}{\sigma_x}\int_{-\infty}^{+\infty}\alpha_k x e^{-\frac{\beta_k x^2}{2}}e^{-\frac{x^2}{2\sigma_x^2}}\int_0^{a_{km}x+b_{km}}e^{-\frac{u^2}{2}}du\,dx.$$

The inside integral can be eliminated by integrating by parts on variable x.

The integral is then evaluated by combining the terms in the exponentials and completing the squares. This yields:

$$K_k(a_{km}, b_{km}) = \sqrt{\frac{2}{\pi}}\frac{\alpha_k}{\frac{1}{\sigma_x^2}+\beta_k}\frac{a_{km}}{\sqrt{\sigma_x^2(a_{km}^2+\beta_k)+1}} \times \exp\left(\frac{-b_{km}^2}{2}\left(1-\frac{\sigma_x^2}{1+\sigma_x^2(\beta_k+a_{km}^2)}\right)\right).$$

Again, a Taylor series expansion can be used to simplify this expression.

Note that in the biasless case we have:

$$K_k(a_{km}, 0) = \sqrt{\frac{2}{\pi}}\frac{\alpha}{\frac{1}{\sigma_x^2}+\beta_k}\frac{a_{km}}{\sqrt{\sigma_x^2(a_{km}^2+\beta_k)+1}}$$

Appendix II

The derivatives needed to compute the different recursions are expressed as follows:

$$\frac{\partial K_k(a_{km},0)}{\partial a_{km}} = \sqrt{\frac{2}{\pi}}\frac{\alpha\sigma_x^2}{\left(\sigma_x^2\left(a_{km}^2+\beta_k\right)+1\right)^{\frac{3}{2}}}, \frac{\partial K_k(a_{km},b_{km})}{\partial a_{km}} = \frac{\partial K_k(a_{km},0)}{\partial a_{km}} - \sqrt{\frac{2}{\pi}}\alpha_k\ \ \sigma_x^2\frac{b_{km}^2}{2}\frac{1+\sigma_x^2\beta_k-2\sigma_x^2a_{km}^2}{\left(1+\sigma_x^2\left(\beta_k+a_{km}^2\right)\right)^{\frac{5}{2}}}$$

$$\frac{\partial F_{kk}(a_{ki},0,a_{ki},0)}{\partial a_{ki}} = \frac{4}{\pi}\frac{\sigma_x^2a_{ki}}{\left(\sigma_x^2a_{ki}^2+1\right)\sqrt{1+2\sigma_x^2a_{ki}^2}},$$

$$\frac{\partial F_{kk}(a_{ki},a_{ki},b_{ki},b_{ki})}{\partial a_{ki}} = \frac{\partial F(a_{ki},0,a_{ki},0)}{\partial a_{ki}} - \frac{2}{\pi}b_{ki}^2\sigma_x^2a_{ki}\frac{5\sigma_x^2\,a_{ki}^2+3}{\left(1+\sigma_x^2a_{ki}^2\right)^2\left(1+2\sigma_x^2a_{ki}^2\right)^{\frac{3}{2}}}$$

$$\frac{\partial F(a_{ki},0,a_{km},0)}{\partial a_{ki}} = \frac{2}{\pi}\frac{\sigma_x^2a_{km}}{\left(\sigma_x^2a_{ki}^2+1\right)\sqrt{1+\sigma_x^2\left(a_{ki}^2+a_{km}^2\right)}}$$

$$\frac{\partial F(a_{ki},a_{km},b_{ki},b_{km})}{\partial a_{ki}} = \frac{\partial F(a_{ki},a_{ki},0,0)}{\partial a_{ki}} - \frac{1}{\pi}b_{ki}^2\frac{\sigma_x^2\,a_{km}}{\left(1+\sigma_x^2a_{ki}^2\right)\sqrt{1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)}}\left(\frac{1+\sigma_x^2a_{km}^2}{1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)}-\frac{2\sigma_x^2a_{ki}^2}{1+\sigma_x^2a_{ki}^2}\right)$$

$$-\frac{1}{\pi}a_{km}^2\frac{\sigma_x^2\,a_{km}}{\left(1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)\right)^{\frac{3}{2}}} - b_{ki}b_{km}\frac{2}{\pi}\frac{\sigma_x^2\,a_{ki}}{\left(1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)\right)^{\frac{3}{2}}}$$

$$\frac{\partial F(a_{ki},a_{km},b_{ki},b_{km})}{\partial b_{km}} = -\frac{2}{\pi}b_{ki}\frac{\sigma_x^2\,a_{ki}a_{km}}{\left(1+\sigma_x^2a_{ki}^2\right)\sqrt{1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)}} - b_{km}\frac{2}{\pi}\frac{1}{\sqrt{1+\sigma_x^2\left(a_{km}^2+a_{ki}^2\right)}}$$

Appendix III

$$\begin{aligned}K_{V_jV_j^t}(n+1) = {} & K_{V_j^tV_j^t}(n) - 2\mu R_{XX}K_{V_jV_j}(n)\\ & - 2\mu K_{V_jV_j}(n)R_{XX}\\ & + 2\mu E\left[e_{W_{0j}}(n)XV_j^t(I-2\mu XX^t)\right]\\ & + 2\mu E\left[e_{W_{0j}}(n)XV_j^t(I-2\mu XX^t)\right]^t\\ & + 4\mu^2E\left[XX^tK_{V_jV_j}XX^t\right]\\ & + 4\mu^2E\left[e_{W_{0j}}^2(n)XX^t\right]\end{aligned} \tag{96}$$

The calculations are similar to [9] Appendix, the main difference is that here we deal with a multi-dimensional input. Therefore, we will follow the same methodology as in [9].

Following [10] the expectation before the last one can be calculated as:

$$\begin{aligned}E\left[XX^tK_{V_jV_j}(n)XX^t\right] = {} & tr\left(R_{XX}K_{V_jV_j}(n)\right)R_{XX}\\ & + 2R_{XX}K_{V_jV_j}(n)R_{XX}.\end{aligned} \tag{97}$$

The first expectation is expressed as:

$$\begin{aligned}&E\left[e_{W_{0j}}(n)XV_j^t(I-\mu XX^t)\right]\\ &\quad = E\left[e_{W_{0j}}(n)XV_j^t\right] - \mu E\left[e_{W_{0j}}(n)XV_j^tXX^t\right]\end{aligned} \tag{98}$$

The first term is Zero (orthogonality principle). The second term is:

$$\begin{aligned}E\left[e_{W_{0j}}(n)XV_j^tXX^t\right] = {} & E\Big[\Big(H_j^tg_j(x_j)+N_j(n)\\ & - W_{0j}^tX(n)\Big)XV_j^tXX^t\Big]\end{aligned} \tag{99}$$

The middle term is Zero (Zero-mean white noise), the last expectation is:

$$\begin{aligned}E\left[W_{0j}^tX(n)XV_j^tXX^t\right] = {} & E\left[X(n)XW_0V_j^t(n)XX^t\right]\\ & \approx tr\Big(R_{XX}W_0E\Big(V_j^t(n)\Big)\Big)R_{XX}\\ & + 2R_{XX}W_0E\Big(V_j^t(n)\Big)R_{XX}\end{aligned} \tag{100}$$

The first expectation in Eq. (99) $E\left[H_j^tg(X)XV_j^t(n)XX^t\right]$ $\approx E\left[H_j^tg(X)XE\Big(V_j^t(n)\Big)XX^t\right]$ involves the nonlinearity $g_j(x_j)$ and should be evaluated explicitly.

The remaining expectation in (96) is: $E\left[e_{oj}^2(n)XX^t\right]$.

$$E\left[e_{oj}^2(n)XX^t\right] = E\left[\Big(H_j^tg(X) - W_{0j}^tX\Big)^2XX^t\right] + \sigma_0^2R_{XX} \tag{101}$$

We have

$$E\left[\left(W_{0j}^t X\right)^2 XX^t\right] = E\left[XX^t W_{0j}^t W_{0j}^t XX^t\right] = tr\left(R_{XX} W_{0j}^t W_{0j}^t\right) R_{XX} + 2R_{XX} W_{0j}^t W_{0j}^t R_{XX} \quad (102)$$

The first term in (102) is:

$$E\left[\left(H_j^t g(X)\right)^2 XX^t\right] = E\left[H_j H_j^t g(X) g^t(X) XX^t\right] = E\left[g(X) g^t(X) H_j H_j^t XX^t\right]$$

(96) is then expressed as:

$$\begin{aligned} K_{V_j V_j^t}(n+1) &= K_{V_j^t V_j^t}(n) - 2\mu R_{XX} K_{V_j V_j}(n) \\ &- 2\mu K_{V_j V_j}(n) R_{XX} + 4\mu^2 \left(-E\left[H_j^t g(X) XE\left(V_j^t(n)\right) XX^t\right]\right. \\ &\left. + tr\left(R_{XX} W_0 E\left(V_j^t(n)\right)\right) R_{XX} + R_{XX} W_0 E\left(V_j^t(n)\right) R_{XX}\right) \\ &+ 4\mu^2 \left(-E\left[H_j^t g(X) XE\left(V_j^t(n)\right) XX^t\right]\right. \\ &\left. + tr\left(R_{XX} W_0 E\left(V_j^t(n)\right)\right) R_{XX} + R_{XX} W_0 E\left(V_j^t(n)\right) R_{XX}\right)^t \\ &+ 4\mu^2 \left(tr\left(R_{XX} K_{V_j V_j}(n)\right) R_{XX} + 2R_{XX} K_{V_j V_j}(n) R_{XX}\right) \\ &+ 4\mu^2 \left(E\left[g(X) g^t(X) H_j H_j^t XX^t\right] + \sigma_0^2 R_{XX}\right. \\ &\left. - tr\left(R_{XX} W_0 t W_0^t\right) R_{XX} - 2R_{XX} W_{0j}^t W_{0j}^t R_{XX}\right) \end{aligned}$$

Competing interests
The author declares that he has no competing interests.

Received: 6 December 2011 Accepted: 13 July 2012
Published: 21 August 2012

References

1. S. Haykin, *Neural Networks: A Comprehensive Foundation* (Prentice Hall, 1999)
2. Y. Gao, M. Er, Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems. IEEE Trans. Fuzzy Systems, 462–476 (2003)
3. S.S. Ge, C. Wang, Adaptive neural control of uncertain nonlinear MIMO systems. IEEE Trans. Neural Networks **15**, 674–692 (2004)
4. M. Ibnkahla, A. Al-Hinai, Adaptive modeling and identification of nonlinear MIMO channels using neural networks, in *Adaptive Signal Processing in Wireless Communications*, ed. by M. Ibnkahla (CRC Press, Boca Raton, FL, USA, 2008)
5. K.S. Narendra, K. Parthasarathy, Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Networks **1**, 4–27 (1990)
6. H. Xu, P. Ioannou, Robust adaptive control for a class of MIMO nonlinear systems with guaranteed error bounds. IEEE Trans. Automatic Control, 718–742 (2003)
7. S. Amari, Mathematical foundations of neurocomputing. Proc. IEEE **78**(9), 1443–1463 (September 1990)
8. N.J. Bershad, M. Ibnkahla, F. Castanié, Statistical analysis of a two-layer back propagation algorithm used for modeling non linear memoryless channels: The single neuron case. IEEE Trans. Signal Processing **45**(3), 747–756 (March 1997)
9. N. Bershad, P. Celka, J.M. Vesin, Stochastic analysis of gradient adaptive identification of nonlinear systems with memory for Gaussian data and noisy input and output measurements. IEEE Trans. Signal Processing **47**(3), 675–689 (March 1999)
10. S. Haykin, *Adaptive Filter Theory* (Prentice Hall, 1996)
11. M. Ibnkahla, N.J. Bershad, J. Sombrin, F. Castanié, Neural network modeling and identification of non linear channels with memory: Algorithms, applications and analytic models. IEEE Trans. Signal Processing **46**, 5 (1998)
12. M. Ibnkahla, Statistical analysis of neural network modeling and identification of nonlinear channels with memory. IEEE Trans. Signal Processing, 1508–1517 (2002)
13. J. Shynk, S. Roy, Convergence properties and stationary points of a perceptron learning algorithm. Proc. IEEE **70**, 1599–1604 (Oct. 1990)
14. J.G. Taylor, *Mathematical Approaches to Neural Networks* (North-Holland, Amsterdam, 1993)
15. H. White, learning in artificial neural networks: A statistical perspective. Neural Comput. **1**, 425–464 (1989)
16. N. Bershad, P. Celka, J.M. Vesin, Analysis of stochastic gradient tracking of time-varying polynomial Wiener systems. IEEE Trans. Signal Processing **48**(6), 1676–1686 (June 2000)
17. H. Bolcskei, MIMO systems: Principles and trends, in *Signal Processing for. Mobile Communications Handbook*, ed. by M. Ibnkahla, 12th edn. (CRC Press, 2004)
18. T. Javornik, G. Kandus, S. Plevel, G. White, A. Burr, V-BLAST algorithm performance in non-linear channel. IEEE Computer as a Tool Conference **1**, 183–187 (September 2003)
19. G. Poitau, A. Kouki, Impact of realistic amplification models on dynamic VBLAST optimization. Proc. Vehicular Technology Conference Spring, 894–897 (2004)
20. S. Woo, D. Lee, K. Kim, H. Hur, C. Lee, J. Laskar, Combined effects of RF impairments in the future IEEE 802.11n WLAN systems. Proc. IEEE Vehicular Technology Conference Spring **2**, 1346–1349 (May 2005)
21. S. Yang, J. Xi, X. Mu, Decision aided joint compensation of clipping noise and nonlinearity for MIMO-OFDM systems. Proc. IEEE International Symposium on Communications and Information Technology (ISCIT) **1**, 725–728 (2005)
22. S. Yang, J. Xi, F. Wang, X. Mu, H. Kobayashi, Decision aided compensation of residual frequency offset for MIMO-OFDM systems with nonlinear channel. Proc. International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 113–116 (2005)
23. A. Sulyman, M. Ibnkahla, Performance of MIMO systems with antenna selection over nonlinear fading channels. IEEE Journal in Selected Topics in Signal Processing **2**, 159–170 (April 2008)
24. A. Sulyman, M. Ibnkahla, Performance analysis of nonlinearly amplified M-QAM signals in MIMO channels. European Transactions in Communications **19**(1), 15–22 (January 2008)
25. J. Pedro, S. Maas, A comparative overview of microwave and wireless power amplifier behavioral modeling approaches. IEEE Trans. Microwave Theory and Techniques **53**(4), 1150–1163 (2005)
26. A. Saleh, Frequency-independent and frequency–dependent nonlinear models of TWT amplifiers. IEEE Trans. Communications **29**, 11 (1981)

doi:10.1186/1687-6180-2012-179
Cite this article as: Ibnkahla: Stochastic analysis of neural network modeling and identification of nonlinear memoryless MIMO systems. *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:179.