Modelling and identification of MIMO non linear communication channel

NABIHA SAIDI, OKBA TAOUALI AND HASSANI MESSAOUD

Unité de Recherche ATSI
Ecole Nationale d’Ingénieurs de Monastir
Rue Ibn El Jazzar 5019 Monastir ; Tel : +(216) 73 500511, Fax : +(216) 73 500 514, TUNISIA
saidinabiha@yahoo.fr; taoualiokba@yahoo.fr; hassani.messaoud@enim.rnu.tn

Abstract: - During the last decade, Volterra Model has been considered as the most usual and popular way to handle non linear systems. As an alternative to this modelling strategy the last few years has registered the birth of a new modelling technique developed on a particular Hilbert Space the kernel of which is reproducing. This space known as Reproducing Kernel Hilbert Space (RKHS) uses the statistical learning theory to provide an RKHS model. This paper proposes the modelling of a non linear Multi Input Multi Output (MIMO) communication channel with two modelling techniques reduced MIMO Volterra model and RKHS MIMO. The performances of both models are evaluated and the results were successful.

Key-Words: - Modelling, MIMO, RKHS, communication channel

1 Introduction

Till while ago Volterra [3], [4] models still the most usual and popular way to describe non linear systems behaviour as it provides a model linear with respect to its parameters. This positive feature is disqualified by the model complexity which may become cumbersome if the system non linearity hardness is increased. A way to reduce the parameter number is based on expanding the Volterra kernels on Generalized Orthonormal Basis (GOB) [4], [5] and [6]. The complexity reduction depends on the choice of the basis parameter structure such as poles and truncating order. The reduced model is named GOB Volterra model in SISO case and GOB MIMO Volterra model [1] in MIMO case. The last few years has registered the birth of a new modelling technique of non linear systems. This technique, developed on a particular Hilbert Space, known as Reproducing Kernel Hilbert Space (RKHS) uses the statistical learning theory (SLT) to provide a RKHS model. Contrary to Volterra model the model complexity is independent of the non linearity degree and the system memory. In [2] the SISO system modelling problem has been investigated and the MISO case has been processed in [10]. The MIMO case has been processed in [11]. In this paper we focus on the modelling and identification of a non linear MIMO communication channel for this we use two techniques of modelling such as: MIMO RKHS model and GOB MIMO Volterra model. The modelling of a non linear MIMO communication channel is confined to section 4.

2 Modelling of MISO and MIMO process in RKHS

2.1 Reproducing Kernel Hilbert Space (RKHS)

Let X be a given space and let H a Hilbert space of functions defined on X. This space is dotted with the scalar product $\langle \cdot, \cdot \rangle_H$. Consider the function $K : X \times X \to \mathbb{R}$ such as

$$K(x,t) = \langle f,K_x(t) \rangle = f(x)$$

is a function of the space $H$.

2.2 Statistical Learning Theory (SLT)

The Statistical Learning Theory [10] aims to develop a model of non linear system from a set of data $D = \{(x^{(0)},y^{(0)}),\ldots,(x^{(N)},y^{(N)})\}$ and to evaluate the error risks associated to the resulting model.

We call learning the procedure which enables to select, from the set of observations $D$, the function $f \in H$ that describes as close as possible the relation between any process input/output couple $(x^{(i)}, y^{(i)})$ even it doesn’t belong to $D$. To determine the optimal function, the SLT proposes to minimize the functional risk associated to
the chosen function \( f \in H \). This risk \( R(f) \) is given by:

\[
R(f) = \int_{x,y} P(x,y) V(y,f(x)) \, dx \, dy
\]  

(3)

Where \( P(x,y) \) is the probability associated to the input/output couple \((x,y)\) and \( V(y,f(x)) \) is a cost function which evaluates the error between the process output \( y \) and its estimation \( f(x) \). In practice \( P(x,y) \) is unknown and we minimise the empirical risk \( R_{\text{emp}}(f) \) instead of \( R(f) \), with

\[
R_{\text{emp}}(f) = \frac{1}{N} \sum_{i=1}^{N} V\left(y^{(i)}, f\left(x^{(i)}\right)\right)
\]  

(4)

However the minimisation of \( R_{\text{emp}}(f) \) in the space \( H \) may lead to an over fitting of the given function so that its generalization to new data isn’t assured. To solve this problem Vapnik [12] proposes to adopt the (Structural Risk Minimisation: SRM) which can be settled by amending the empirical risk by a function evaluating the complexity of the given model. To do so we minimise instead of the empirical risk, the criterion

\[
D(f) = \frac{1}{N} \sum_{i=1}^{N} V\left(y^{(i)}, f\left(x^{(i)}\right)\right) + \frac{\lambda}{2} \|f\|_{H}^2
\]  

(5)

where \( \lambda \) allows to tune the compromise between the empirical risk minimization and the generalization ability. The minimization of criterion (5) on an arbitrary function space can be a hard task however this can be handled when this space is an RKHS.

Based on the representer theorem [13] the optimal function \( f_{\text{opt}} \) which minimizes \( D(f) \) can be written as:

\[
f_{\text{opt}}(x) = \sum_{i=1}^{N} a_i K\left(x^{(i)}, x\right)
\]  

(6)

Where \( a_i, i = 1,...,N \) are the model parameters. The norm of the function \( f \) is then

\[
\|f\|_{H}^2 = \left(\sum_{i=1}^{N} a_i K\left(x^{(i)}, x^{(i)}\right)\right)_{H} = \sum_{i,j=1}^{N} a_i a_j K\left(x^{(i)}, x^{(j)}\right)
\]  

(7)

### 2.3 Modelling of a MISO and MIMO Process in RKHS space

When a MISO process is considered, the input \( x \) is a d-dimensional vector and \( y \) is a scalar. An important particular case of cost function \( V \) is the square error:

\[
V\left(y^{(i)}, f\left(x^{(i)}\right)\right) = \left(y^{(i)} - f\left(x^{(i)}\right)\right)^2
\]  

(8)

In this case the minimization of the criterion (5) in the RKHS space leads to the solution given by (6) where:

\[
a_i = \sum_{j=1}^{N} \Psi_{i,j} y^{(j)}
\]  

(9)

With \( \Psi_{i,j} \) the \( i,j \) th component of the matrix \( \Psi \in \mathbb{R}^{N \times N} \)

\[
\Psi = (G + \lambda N I)^{-1}
\]  

(10)

And the matrix \( G \in \mathbb{R}^{N \times N} \) is such that:

\[
G_{ij} = K\left(x^{(i)}, x^{(j)}\right), i,j = 1,...,N
\]  

(11)

Or in matrix form:

\[
A = (G + \lambda N I)^{-1} Y, A = (a_1, a_2, ..., a_N)^T \text{ and } Y = (y_1, y_2, ..., y_N)^T
\]

In the MIMO case the process output is a p-dimensional vector, we consider the network of kernel functions illustrated by Figure 1.

Figure 1. Network of kernels functions for the MIMO modelling

The MIMO process is considered as a set of MISO processes modelled in RKHS space as above. To decrease the model complexity, all the MISO output are linear combinations of the same kernel components and with different parameters.

The output of the \( q^{th} \) MISO model is:
\begin{align}
y_q &= \sum_{i=1}^{N} a_i^q K(x^{(i)}, x) \quad \text{for} \quad q = 1, ..., p \\
y_q &= A_q^T H(x) \quad q = 1, ..., p
\end{align}

Where:
\begin{equation}
H(x) = [H_1(x), ..., H_N(x)] \in \mathbb{R}^N
\end{equation}

With
\begin{equation}
H_i(x) = K(x^i, x), i = 1, ..., N
\end{equation}

\begin{equation}
A_q = [a_1^q, ..., a_N^q]^T, \quad q = 1, ..., p
\end{equation}

The output vector \( Y_p \) is then given by:
\begin{equation}
Y_p = \begin{pmatrix}
y_1 \\
\vdots \\
y_p \\
\end{pmatrix} = \begin{pmatrix}
a_1^1 a_1^2 \cdots a_1^N \\
a_1^2 a_1^1 \cdots a_1^2 \\
\vdots \\
a_1^p a_1^{p-1} \cdots a_1^1 \\
\end{pmatrix} K(x_1, x) \\
\begin{pmatrix}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{pmatrix} \\
\begin{pmatrix}
a_1^p a_1^{p-1} \cdots a_1^1 \\
\vdots \\
\vdots \\
\vdots \\
\end{pmatrix} \end{pmatrix} \begin{pmatrix}
y_p \\
\vdots \\
y_p \\
\end{pmatrix} (17)
\end{equation}

Or from (14) and (16)
\begin{equation}
Y_p = \begin{pmatrix}
A_q^T \\
A_q^T \\
\vdots \\
A_q^T \\
\end{pmatrix} H(x) (18)
\end{equation}

### 3 GOB MIMO Volterra model

#### 3.1 MIMO Volterra model

The MIMO system can be considered as a set of Multi Input Single Output (MISO) sub systems. Thus the modelling of the MIMO System is equivalent to the modelling of its sub systems. Let a MIMO system with \( n \) inputs and \( S \) outputs, each subsystem output \( y_i(k) \) can be developed on Volterra series as:
\begin{equation}
y_i(k) = h_0^i + \sum_{j=1}^{L} \sum_{k_1=1}^{n} \sum_{k_1=0}^{\infty} \cdots \sum_{k_s=0}^{\infty} h_{j_1 \cdots j_s}(m_{1} \cdots m_{s}) \prod_{s=1}^{n} u_j, (k-m_j) \quad (19)
\end{equation}

\( h_{j_1 \cdots j_s}(m_{1} \cdots m_{s}) \) is the Volterra Kernel of \( i \) th order corresponding to the sub system the output of which is \( y_i(k) \) and \( h_0^i \) is the statistical characteristic corresponding to \( y_i(k) \).

And the parameter number is:
\begin{equation}
n_{MIMO} = 1 + \sum_{i=1}^{n} \frac{L-1+i}{(M-1)!} * S \quad (20)
\end{equation}

#### 3.2 GOB MIMO Volterra model

As mentioned in the introduction, because of the linearity of Volterra model with respect to its parameters, we proceed in this paragraph to the decomposition of Volterra kernels of the Volterra model on the Generalised Orthogonal Base GOB [1]. This decomposition will be done on the same GOB basis. The output of the model \( y_i(k) \) is then written as:
\begin{equation}
y_i(k) = h_0^i + \sum_{j=1}^{L} \sum_{k_1=1}^{n} \sum_{k_1=0}^{\infty} \cdots \sum_{k_s=0}^{\infty} g_{s,j_1 \cdots j_s} \prod_{s=1}^{n} y_{j_s}^i, (k-m_j) \quad (21)
\end{equation}

With \( x_{s,j_l}^i(k) \) is the filtered input given by:
\begin{equation}
x_{s,j_l}^i(k) = \sum_{m=0}^{\infty} b_{s,j_l^i}(i) u_j(k-m) \quad l = 1 \ldots n \quad (22)
\end{equation}

\( b_{s,j_l} \) is the function of the GOB the Z-transform of which is:
\begin{equation}
B_s(z) = \sqrt[1]{1 - |\xi_s|} \prod_{s=0}^{s=0} \left( 1 - \frac{\xi_s}{z - \xi_s} \right) \quad (23)
\end{equation}

\( \xi_s \) is the pole of order \( k \), \( |\xi_s| < 1 \)

And \( g_{s,j_1 \cdots j_s} \) are the coefficients of the kernels on the GOB base.

Taking into account the truncating order \( K \) of the GOB base, relation (21) can be written in triangular form as the following:
\begin{equation}
y_i(k) = h_0^i + \sum_{j=1}^{L} \sum_{k_1=1}^{n} \sum_{k_1=0}^{\infty} \cdots \sum_{k_s=0}^{\infty} g_{s,j_1 \cdots j_s} \prod_{s=1}^{n} y_{j_s}^i, (k-m_j) \quad (24)
\end{equation}

\( y_i(k) \)}
The total number of parameters is:

\[
1 + \sum_{i=1}^{P} n^i \frac{(K-1+i)!}{(K-1)!i!} \text{S}
\]  

(25)

**Pole optimisation:**

The complexity reduction is then very depending on the choice of poles that characterize the GOB. Poles can be optimized by means of a gradient-type algorithm [4] and [8] Or to use the experimental method developed in SISO case by [7] for selecting the GOB poles with an iterative procedure this method select poles that minimize Normalized Mean Square Error (NMSE) between the real output and the output of the model. In this paper we adopt this method for optimizing poles of GOB MIMO Volterra model.

\[
\xi_{j, opt} = \text{Arg min } EQ^j \left( p_j \right); \quad j = 1, \ldots, P
\]  

(26)

\[
EQ^j \left( p_j \right) = \frac{\sum_{k=0}^{N} \left[ y_s(k) - \hat{y}_s(k) \right]^2}{\sum_{k=0}^{N} \left[ y_s(k) \right]^2}
\]  

(27)

Where \( y_s(k) \) and \( \hat{y}_s(k) \) are the real output and the output of the model respectively.

**4 Application**

Consider a MIMO non linear communication channel characterised by the number of sources (users) and the number the received antenna. This channel can be modelled by a MIMO Volterra given by:

\[
y_s(k) = h'_0 + \sum_{i=1}^{P} \sum_{j_1=1}^{n} \sum_{j_2=1}^{n} \cdots \sum_{m_{P-1}=1}^{m_{P-1}} \sum_{m_0=0}^{m_0} \cdots \sum_{m_{P-1}=0}^{m_{P-1}} h'_{j_1,j_2,\ldots,j_p} (m_1,\ldots,m_p) \\
\times \prod_{i=1}^{P} u_{j_i} (k-m_i) + v_s(k)
\]  

(28)

Where \( y_s \) (s = 1, \ldots, S) is the signal received by the \( s^{th} \) antenna at time instant \( k \), \( P \) is the non linearity order of the channel and \( M \) is the channel memory. \( h'_{j_1,j_2,\ldots,j_p} (m_1,\ldots,m_p) \) are the kernel coefficients of the \( s^{th} \) subchannel and \( v_s(k) \) is the additive white Gaussian noise to the \( s^{th} \) antenna, it is assumed that the noise components are zero mean.

![MIMO non linear communication channel](image)

**Figure 2: MIMO non linear channel**

Consider the non linear Multiple Input Multiple Output MIMO Volterra channel [14] described by:

\[
y_1(k) = 2 u_1(k) + 0.4 u_1(k-1) + 0.08 u_2(k-2) + 2 u_2(k) u_1(k-1) + 0.2 u_1(k) u_1(k-1) + 0.2 u_2(k) u_1(k-1) + 0.1 u_1^2(k-1) + v_1(k)
\]

\[
y_2(k) = u_1(k) + 0.3 u_1(k-1) + 0.09 u_2(k-2) + 0.01 u_1^2(k-1) + 0.01 u_2^2(k-1) + v_2(k)
\]  

(29)

Where \( u_1 \in \{-1, 1\} \) and \( u_2 \in \{-2, 2\} \) are the channel inputs, \( y_1 \) and \( y_2 \) are its outputs and \( v_1 \) and \( v_2 \) are additive white noise.

The NMSE between the output of the channel \( y_s(k) \) and the estimated output \( \hat{y}_s \)

\[
\text{NMSE}(s) = \frac{\sum_{k=0}^{N} (y_s(k) - \hat{y}_s(k))^2}{\sum_{k=0}^{N} (y_s(k))^2}
\]  

(30)

Where \( y_s \) and \( \hat{y}_s \) are respectively the real and the estimated output of the channel. The additive noise evaluated by the signal to noise ratio SNR(s) for the \( s^{th} \) output of the channel.

\[
\text{SNR}(s) = \frac{\sum_{k=0}^{N} (y_s(k) - \bar{y}_s)^2}{\sum_{k=0}^{N} (v_s(k) - \bar{v}_s)^2}
\]  

(31)

With \( N_m \) the observation number, \( \bar{y}_s \) and \( \bar{v}_s \) are the mean values of the \( s^{th} \) channel output \( y_s(k) \) of and the \( s^{th} \) noise value \( v_s(k) \) respectively.

**4.1 Modelling in RKHS space**

To build the RKHS model we use the polynomial Kernel...
\[ K(x, x') = (1 + \langle x, x' \rangle)^p \]  
\[ (32) \]

Where \( p = 2 \). The regularisation term is \( \lambda = 5 \times 10^{-9} \)

In the identification phase we use a training set of 250 inputs/outputs and in the validation phase 120 new inputs/outputs are used to evaluate the performance of the resulting RKHS model.

- **First output of the channel**:

Figure 3 plots the first output of the MIMO non linear channel we notice a concordance between the RKHS model output and the process output in the validation phase. The NMSE in validation is 7.12%.

![Figure 3: Validation of the first output](image)

- **Second output of the channel**

In Figure 4 we plot the second output of the channel we notice a concordance between the RKHS model output and the process output in the learning phase and this concordance remains excellent in the validation phase. The NMSE validation is 5.75%.

![Figure 4: Validation of the second output](image)

**4.2 GOB MIMO Volterra model:**

This MIMO non linear channel can be modelled by a GOB MIMO Volterra model with \( P = 2 \) and a truncating order \( K = 2 \). The total parameter number of the reduced model is 34 and the number of poles is 4.

- **First output of the channel**:

The optimal poles for the first subsystem optimised by the minisation of the NMSE, are \( \xi_{01} = 0.1 \) and \( \xi_{11} = -0.1 \). We plot in Figure 5 the validation of the first output of the channel and the output of the model; we note the concordance between both outputs.

![Figure 5: Validation of the first output](image)

The NMSE for the first subsystem is 6.08%.

- **Second output of the channel**

The optimal poles for the second subsystem are \( \xi_{02} = 0.3 \) and \( \xi_{12} = 0.9 \). In Figure 6 we plot the second output of the channel and the output of the model we note the concordance between both outputs. The NMSE is 8.3%.

![Figure 6: Validation of the first output](image)

Results are summarized in Table 1
We can conclude that this MIMO non linear communication channel can be modelled by a GOB MIMO Volterra model, this model is characterized by a minimum number of parameter but we must first optimize poles of the GOB which is a hard task and take high calculation time.

This channel can be modelled also modelled by a MIMO RKHS model, this model is characterized by a high number of parameter to be identified. From Table 1 we conclude that NMSE for both models is comparable.

5 Conclusion
This paper has dealt with the study of two non linear MIMO system modelling techniques the reduced Volterra model and the MIMO RKHS model. The complexity of GOB MIMO Volterra model depends on degree of non linearity and in the truncating order and for MIMO RKHS model the complexity depends only in the number of observations. These models have been tested for modelling of a non linear MIMO numerical communication channel and results are successful.

References:
[14]: F. Yangwang, J. Licheng and P. Jin, “MIMO Volterra filter equalization using Pth-order inverse approach“ 0-7803-6293-4/00/$10.00 02000 IEEE.