Filter Bank Realizations of Volterra Kernels

Vadim O. Kudryavtsev

Institute of Engineering Cybernetics of the National Academy of Sciences of Belarus
6, Surganov St., 220012, Minsk, Belarus
Phone: (375) 17 284-20-06, Fax: (375) 17 231 84 03,
E-mail: vadkudr@yahoo.com

Abstract

A great many of processes in a nature are nonlinear, so their modeling requires an embedding of nonlinear parts into the model structure. One of the popular approaches to the nonlinear system modeling are Volterra series. Unfortunately, already the second order Volterra kernel requires high amount of coefficients for its identification and therefore a large number of computations for its realization. This paper shows several possibilities of reducing the amount of computations based on matrix decomposition of a second order Volterra kernel, and presents both mathematically accurate algorithms and approximate ones. Presented algorithms are in the form of filter bank structure. They can be used for highest order Volterra kernel realization as well.

Keywords: Nonlinear Modeling; Volterra Series; Filter Banks; Eigen Value Decomposition; Fast Algorithms; Finite Fields.

1. Introduction

Increasing requirements to analyzing and modeling natural processes compel the researcher looking for various methods of identifying nonlinearities in objects under consideration and effective realizing of its models. One of the popular approaches in this area are Volterra series because of their kernel coefficients linearity in a respect to input and output signals and also due to the possibility of using a big arsenal of algorithms designed for linear signal processing.

Unfortunately, already the second order Volterra kernel requires high amount of coefficients for its identification and therefore a large number of computations for its realization. A number of works was devoted to developing simplified methods of Volterra kernel realization. They required less coefficients for identification process (and hence higher convergence of the process in spite of nonlinearity between kernel coefficients and input-output signals) and less number of nonlinear operations (such as multiplications) in realization by some increasing of error modeling [1, 2].

As can be seen, the most of the works are related to the classic Volterra kernel representation and are particular cases of it. On the other hand these methods are similar to the filter bank structures, i.e. one of the realization steps is filtering in parallel filters.

These considerations lead to thoughts that it should be at least one method generalizing all the previous approaches. In this paper an approach present is based on the diagonalization methods of a matrix of second Volterra kernel operator, such as LDL-decomposition and eigen value decomposition. One of them is well suited for efficient implementation of the second Volterra kernel operator and other gives a good possibility to remove some calculations allowing a minimal error. Methods presented can be also used to design approaches for highest order Volterra kernel realization.

2. Volterra series and their realizations.

The Volterra series expansion is a causal and time-invariant nonlinear operation with the discrete time representation:

\[
y_n = h_0 + \sum_{m_0=0}^{\infty} h_{m_0} x_{n-m_0} + \sum_{m_1=0}^{\infty} \sum_{m_0=m_1}^{\infty} h_{m_1,m_0} x_{n-m_1} x_{n-m_0} + \cdots,
\]

where \( h_0 \) is a constant term (or Volterra kernel of order zero), \( h_{m_0} \) is impulse response of discrete linear filter (or Volterra kernel of first order), \( h_{m_1,m_0} \) and \( h_{m_2,m_1,m_0} \) are Volterra kernels of second and third order, respectively.

As can be seen from expansion (1), the filter output is still linear in the coefficients. This fact simplifies an identification procedure and allows applying many algorithms known from linear signal processing to Volterra filtering by use of a nonlinearly extended signal vector.

But a number of coefficients for defining Volterra kernels grows catastrophically fast with increasing of their orders (in spite of their symmetry). And here...
appears necessity of truncation of Volterra kernels or some other kind of redundancy eliminating.

Consider calculation of the term in expansion (1) containing Volterra kernel of a second order:

\[
y^{(2)}_n = \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} h_{n,m_1,m_2} x_{n-m_1} x_{n-m_2}
\]  \( (2) \)

It's easy to see, from equation (2) that computation of the sample \( y^{(2)}_n \) can be expressed as multiplication of a matrix by vectors of input signal:

\[
y^{(2)}_n = x^T V_2 x,
\]  \( (3) \)

where \( V_2 = [h_{n,m_1,m_2}] \), \( m_1, m_2 = 0 \ldots N - 1 \) is the second order discrete Volterra matrix operator,

\[
x = [x_{n-N} \ldots x_{n-N+1} \ldots x_n]
\]

is input signal vector.

Consider decomposition of the second order Volterra symmetrical matrix operator in the form:

\[
V_2 = ADA^T,
\]  \( (4) \)

where \( D \) is diagonal matrix,

\( A \) is a square matrix with \( \text{rank}(A) = \text{rank}(V_2) \). This decomposition is not new and was used for example in [3] for LDL-decomposition of the matrix \( V_2 \).

Such decomposition allows realization of a second order Volterra operator in the form of a number of parallel linear filters with squarers and multipliers following them (fig. 1). Advantage of such representation is that we can apply a big arsenal of fast spectral algorithms to the computation of linear filters.

3. Ways for complexity reducing

One of the modern algorithms of linear filter realization is based on a filter bank structure [4]. This method stays from complexity point of view between spectral algorithms and direct realization method. With appropriate choice of parameters of algebraic field combined with a choice of number of channel developer always can select optimal hardware realization variant.

If we use such kind of filter realizations (spectral or "semispectral" methods, based on rectangular transforms), it is possible to use common preaddition or "semispectral" methods, based on rectangular realization variant.

If computations are performed in a finite field, it is always possible to do it, because of equality \( a^2 + b^2 = a^2 + (−1)b^2 = (a + √−1b)(a − √−1b) \). By appropriate choice of field characteristic we can provide that multiplication on \( √−1 \) in this field can be performed only with shifts. For example, in field with characteristic \( q = 2^2 \) \( + 1 \) \( √−1 = √2^2 = 2 \) -1.


To this moment we have considered methods for exact Volterra kernel realization. Note, however, that there is always certain estimation error by system identification. And hence we can drop several coefficients in the decomposition scheme by the condition that error between output signals of model and the real system not exceed the estimation error. This can be done in the following way.

Consider matrix \( A \) from equation (4). Columns of this matrix corresponds to the impulse responses of the filters in fig. 1. It is evident, that we can make columns of the matrix \( A \) to have norm of unity by appropriate division of each column by its norm and in order to preserve the equality in expression (4) we have to do multiplication of each diagonal element of the matrix \( D \) by square of norm of the corresponding column. Suppose now that input signal has uniform power spectral density. In this case variance of output signals of each filter are the same. And it is evident, that output error is determined by the sum of coefficients \( d_i \) of ignored channels.

In the case of input signal with given power spectral density \( S(f) \) output processes of squarers are proportional to the variances on outputs of each filter

\[
\sigma_i^2 = \frac{1}{2π} \int_{-1/2}^{1/2} S(f) |A_i(f)|^2 df,
\]  \( (5) \)

where \( A_i(f) = \sum_{n=0}^{N-1} a_{n,i} \exp(-j2πfn) \) is frequency response of \( i \)-th channel. It is evident, that a minimal error is obtained by the truncation of channels corresponding to the smallest values \( \sigma_i^2 d_i \).
When the power spectral density of the input data is unknown, the best way is nevertheless to cutoff the branches with small coefficients $d_j$.

The special kind of matrix diagonalization is eigen value decomposition

$$V_2 = U_0 D_0 U_0^T.$$  \hfill (6)

where $U_0$ is unitary matrix. Another example of decomposition of the second order matrix kernel is LDL-decomposition, described in [3].

In fact there are infinitely many decomposition variants for the second order matrix kernel. It is easy to see by noting that matrix $A$ can be expressed as a product of unitary and diagonal matrices following in sequence. But the following theorem states that arbitrary matrix $A$ in equation (4) can be expressed by finite product of matrices.

**Theorem.** Any matrix $A$ in equation (4) can be written as product of two unitary and two diagonal matrices:

$$A = U_0 D_0^{1/2} U D^{-1/2}.$$  \hfill (7)

**Proof.** The statement (7) will be evident if we expand the decomposition (6) in the following way:

$$V_2 = (U_0 D_0^{1/2} U D^{-1/2}) D (D^{-1/2} U^T D_0^{1/2} U_0^T).$$

End of proof.

Note that diagonal matrices $D_0^{1/2}$ and $D^{-1/2}$ and unitary matrix $U$ have their entries in the form of fully real or fully imaginary numbers. To ensure matrix $A$ to be completely real, numbers of positive and negative values in matrices $D_0$ and $D$ have to be the same, i.e. ratio of positive and negative factors $d_j$ in fig.1 is the same independently of variant of Volterra kernel diagonalization.

Consider now relationship between values of matrices $D_0$ and $D$. As above let us require for columns of matrix $A$ to be have norm of unity. Regarding that and expression (7) we can write this condition in the next form:

$$\text{diag}(A^T A) = [1,1,\ldots,1].$$ \hfill (8)

Substituting $A$ by right hand side of the expression (7) we get:

$$\text{diag}(D^{-1/2} U^T D U D^{1/2}) = [1,1,\ldots,1].$$ \hfill (9)

The last equation can be reduced to

$$\text{diag}(D_0) Q = \text{diag}(D),$$ \hfill (10)

where $Q$ is an orthostochastic matrix, i.e. one that is obtained from a unitary matrix $U$ by replacing each element $s_{i,j} = |d_{i,j}|^2$. Properties of orthostochastic matrices are described in [7]. One of them is that the sum of the entries in any row or column of $Q$ is unity. Hence, the sum of the entries of matrices $D_0$ and $D$ is the same.

If unitary matrix $U$ is completely real then expression (10) says that vector $\text{diag}(D_0)$ majorizes vector $\text{diag}(D)$ in the sense that in ordering $d_0(0) \geq d_0(1) \geq \ldots \geq d_0(N)$ and $d(0) \geq d(1) \geq \ldots \geq d(N)$ these vectors obey the property that $\sum_{i=0}^p d_0(n) \geq \sum_{i=0}^p d(n)$ for all $P = 0,1,\ldots,N-1$, with equality holding when $P = N - 1$ [7]. This result means that eigen value decomposition is optimal for the channel truncation process described above if all eigen values of the second order Volterra matrix operator are of the same sign (compare with the proof of Karhunen-Loeve transform optimality described in [6]).

Consider representation of a matrix operator $V_2$ as a sum of two matrices:

$$V_2 = V_2^+ + V_2^- = U_0 D_0 U_0^T U_0 D_0^T U_0^T,$$ \hfill (11)

where $V_2^+$ and $V_2^-$ have lower rank, and $V_2^+$ has only all positive eigen values of $V_2$ and $V_2^-$ has only all negative eigen values of $V_2$. Then for two these matrices separately eigen value decomposition is optimal, but from (11) it is evident that optimal decomposition for the general case of eigen values of mixed signs should be determined from the condition for squares of eigen values. We can conclude therefore that is for the general case of the second order Volterra kernel eigen value decomposition is suboptimal.

Further possibility of complexity reducing is based on a reconstruction of matrix operator after the truncation process. We can decompose reconstructed second order matrix operator $V_2' = U_0 D_0' U_0'^T$, where $D_0'$ has zeros in place of eigen values corresponding to ignored channels, by LDL-decomposition algorithm $V_2' = LDL^T$, where $L$ is lower triangular matrix. Note that due to the rank decreasing of the matrix operator $V_2'$, number of nonzero columns in matrix $L$ is equal to rank of the matrix $V_2'$, so the number of filters in the structure in fig.1 keep reduced. Because $L$ is lower triangular matrix, length of filters are constantly decreasing, so complexity of filter realization is reduced.

Further complexity reducing consists in possibility of combining channels in blocks and using for each block common preadditions for filters in the block.
5. Conclusion

In this paper various variants of a second order Volterra kernel realization are presented. Combining proposed methods developer can find optimal variant for realization of a second order Volterra kernel based either on exact algorithms or on approximate ones (with reduced number of operations).

Besides there are still a set of open questions: 1) is it possible to decimate channel sequences for eliminating redundancy as it was done in conventional linear filter bank structure [5]; 2) what is optimal combining law for filters corresponding to matrix L in LDL-decomposition; 3) what decomposition is optimal for combined filter computation in all channels or in block of channels.

References


Fig. 1. Realization of a second Volterra kernel by means of filter bank.

Fig. 2. Reducing number of multiplications.