

IMPLICIT ESTIMATION OF WIENER SERIES

Matthias O. Franz, Bernhard Schölkopf
Max-Planck-Institut für biologische Kybernetik
Spemannstr. 38, D-72076 Tübingen, Germany
E-mail: mof,bs@tuebingen.mpg.de
Web: www.tuebingen.mpg.de/mof

Abstract. The Wiener series is one of the standard methods to systematically characterize the nonlinearity of a system. The classical estimation method of the expansion coefficients via cross-correlation suffers from severe problems that prevent its application to high-dimensional and strongly nonlinear systems. We propose an implicit estimation method based on regression in a reproducing kernel Hilbert space that alleviates these problems. Experiments show performance advantages in terms of convergence, interpretability, and system sizes that can be handled.

INTRODUCTION

In system identification, one tries to infer the functional relationship between system input and output from observations of the in- and outgoing signals. If the system is linear, it can be characterized uniquely by measuring its impulse response, for instance by reverse correlation. For nonlinear systems, however, there exists a whole variety of system representations. One of them, the *Wiener expansion*, has found a somewhat wider use in neuroscience and signal processing since its estimation constitutes a natural extension of linear system identification. The coefficients of the Wiener expansion can be estimated by a cross-correlation procedure that is conveniently applicable to experimental data (overview in [5]).

Unfortunately, the estimation of the Wiener expansion by cross-correlation suffers from severe problems: (1) High-dimensional input leads to a prohibitively large number of terms that have to be estimated; (2) the cross-correlation estimator converges very slowly; (3) input signals are restricted to Gaussian noise. In this contribution, we want to overcome these problems by proposing an estimation method based on concepts from regression in *reproducing kernel Hilbert spaces* (RKHS). We will show that the new estimation method is superior to the classical one in terms of convergence, interpretability, and system sizes that can be handled.

In the remainder of this section, we introduce the Wiener expansion and discuss the problems of the cross-correlation procedure. The implicit esti-

mation method is described in Sect. 2, followed by some examples of use in Sect. 3. We conclude in Sect. 4 by briefly discussing the results and possible improvements of the estimation procedure.

Wiener theory of nonlinear systems. A system can be defined mathematically as a rule that assigns an output signal $y(t)$ to an input signal $x(t)$ (t stands for the time variable). This rule can be expressed in the form $y(t) = T[x(t)]$ using the system operator T . Subject to certain restrictions, one can show that a time-invariant T can be expressed as a *Volterra series* consisting of integral operators on the input

$$y(t) = H_0[x(t)] + H_1[x(t)] + H_2[x(t)] + \cdots + H_n[x(t)] + \cdots \quad (1)$$

in which $H_0[x(t)] = h_0 = \text{const.}$ and

$$H_n[x(t)] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h^{(n)}(\tau_1, \dots, \tau_n) x(t - \tau_1) \cdots x(t - \tau_n) d\tau_1 \cdots d\tau_n \quad (2)$$

is the n th-order *Volterra functional* [7]. The functions $h^{(n)}(\tau_1, \dots, \tau_n)$ are the *Volterra kernels* of the system. Throughout this text, we assume that the Volterra kernels are symmetric with respect to permutations of the τ_i . It can be shown that any non-symmetric kernel can be converted into a symmetric kernel without changing the system output [5].

In system identification, we are interested in finding a model of the unknown system in the form of a Volterra series representation that minimizes the mean squared error between the true output and the model output. The classical cross-correlation method for this problem has been introduced by Wiener [8]. Wiener assumed an experimental setting in which the input is controlled by the experimenter such that a convenient class of input functions can be chosen. If the distribution of the input is known then one can choose an input-specific decomposition of the system operator T

$$y(t) = G_0[x(t)] + G_1[x(t)] + G_2[x(t)] + \cdots + G_n[x(t)] + \cdots \quad (3)$$

into operators $G_n[x(t)]$ that are uncorrelated, i.e., orthogonal with respect to the input. The $G_n[x(t)]$ are linear combinations of Volterra operators up to order n . They can be obtained from the original Volterra series by a procedure very similar to Gram-Schmidt orthogonalization [5]. The use of orthogonal operators has the advantage that all operators of the expansion can be estimated independently of each other.

One possible choice is white Gaussian noise with zero mean and known variance A . In this case, the orthogonal expansion in Eq. (3) is called a *Wiener series*, and the n -th order Volterra kernels of the n th-order Wiener functionals, the *Wiener kernels* $k^{(n)}$, can be estimated by

$$k^{(0)} = \overline{y(t)} \quad \text{and} \quad k^{(n)}(\sigma_1, \dots, \sigma_n) = \frac{1}{n!A^n} \overline{y(t)x(t - \sigma_1) \cdots x(t - \sigma_n)} \quad (4)$$

where the bar indicates the average over time [8]. Besides the Wiener kernel, every Wiener functional of degree n contains a varying number of lower order

Volterra operators which can be derived from the Wiener kernel by a procedure described in [5]. It can be shown that any Wiener expansion of finite degree minimizes the mean squared error between the true system output and its Volterra series model [5].

In practical signal processing, one uses a discretized form for a finite sample of data instead of Eq. (2), where the Volterra functionals are expressed as

$$H_n[\mathbf{x}] = \sum_{i_1=1}^m \cdots \sum_{i_n=1}^m h_{i_1 \dots i_n}^{(n)} x_{i_1} \dots x_{i_n}. \quad (5)$$

Here, we assume that the input data is given as a vector $\mathbf{x} = (x_1, \dots, x_m)^\top \in \mathbb{R}^m$. The vectorial data can be generated from any multi-dimensional input or, for instance, by a sliding window on a discretized time series. The discretized n th-order Volterra kernel is given as a finite number of m^n coefficients $h_{i_1 \dots i_n}^{(n)}$. The discretized n th-order Volterra functional is, accordingly, a linear combination of all ordered n th-order monomials of the elements of \mathbf{x} .

Problems of the cross-correlation method. The estimation of the Wiener expansion via cross-correlation poses some serious problems:

1. In practice, the cross-correlations have to be estimated at a finite resolution (cf. the discretized version of the Volterra operator in Eq. (5)). The number of expansion coefficients in Eq. (5) increases with m^n for an m -dimensional input signal and an n th-order Wiener kernel. However, the number of coefficients that actually have to be estimated by cross-correlation is smaller. Since the products in Eq. (5) remain the same when two different indices are permuted, the associated coefficients are equal in symmetric Volterra operators. As a consequence, the required number of measurements is $(n+m-1)!/(n!(m-1)!)$ [5]. Nonetheless, the resulting numbers are huge for higher-order Wiener kernels. For instance, a 5th-order Wiener kernel operating on 16×16 sized image patches contains roughly 10^{12} coefficients, 10^{10} of which would have to be measured individually by cross-correlation. As a consequence, this procedure is not feasible for higher-dimensional input signals.
2. The estimation of cross-correlations requires large sample sizes. Typically, one needs several tens of thousands of input-output pairs before a sufficient convergence is reached. Moreover, the variance of the estimator $\overline{y(t)x(t-\sigma_1) \dots x(t-\sigma_n)}$ in Eq. (4) increases with increasing values of the σ_i [4] such that only operators with relatively small memory can reliably be estimated.
3. The estimation via cross-correlation works only if the input is Gaussian noise with zero mean, not for general types of input.

In this study, we propose a different method for estimating the Wiener expansion that alleviates these practical problems. The method relies on a regression technique taken from the field of kernel methods which is widely used in the machine learning community.

ESTIMATING WIENER SERIES BY REGRESSION IN RKHS

Linear regression in RKHS. Given observations $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$, linear regression tries to estimate y as a function of \mathbf{x} via

$$y = f(\mathbf{x}) = \sum_{j=1}^M \gamma_j \varphi_j(\mathbf{x}), \quad (6)$$

using $\gamma_j \in \mathbb{R}$ and a dictionary of functions φ_j . We will be interested in the case where the dictionary is specified in terms of a kernel function k via $\varphi_j(\mathbf{x}) = k(\mathbf{x}, \mathbf{z}_j)$, using a set of points $\mathbf{z}_1, \dots, \mathbf{z}_M$ (\mathbf{x} and \mathbf{z}_j are from a set \mathcal{X} , say, \mathbb{R}^m). In particular, we consider *positive definite* kernels, i.e., functions k with the property that the *Gram matrix* $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ is positive definite for all choices of the $\mathbf{x}_1, \dots, \mathbf{x}_N$. It can be shown that such kernels admit a representation as a dot product in an associated linear space \mathbb{F} , i.e., there exists a map Φ such that $k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x})^\top \Phi(\mathbf{x}')$. Modulo certain details, \mathbb{F} can be identified with a space of functions

$$f(\mathbf{x}) = \sum_{j=1}^M \gamma_j k(\mathbf{x}, \mathbf{z}_j). \quad (7)$$

This space has the structure of a *reproducing kernel Hilbert space (RKHS)*. By carrying out linear methods in \mathbb{F} , one can obtain elegant solutions for various nonlinear estimation problems (see [6]), examples being Support Vector Machines. Although \mathbb{F} can have infinite dimension,¹ these problems can often be solved efficiently, which is in part due to the so called *representer theorem*. It states the following: suppose c is an arbitrary cost function, Ω is a nondecreasing function on $\mathbb{R}_{>0}$ and $\|\cdot\|_{\mathbb{F}}$ is the norm of the RKHS. If we minimize an objective function

$$c((\mathbf{x}_1, y_1, f(\mathbf{x}_1)), \dots, (\mathbf{x}_N, y_N, f(\mathbf{x}_N))) + \Omega(\|f\|_{\mathbb{F}}), \quad (8)$$

over all functions of the form (7), then an optimal solution² can be expressed as

$$f(\mathbf{x}) = \sum_{j=1}^N a_j k(\mathbf{x}, \mathbf{x}_j), \quad a_j \in \mathbb{R}. \quad (9)$$

In other words, although we did consider functions which were expansions in terms of arbitrary points \mathbf{z}_j (see (7)), it turns out that we can always express the solution in terms of the training points \mathbf{x}_j only. Hence the optimization problem over an arbitrarily large number of M variables γ_j is transformed into one over N variables a_j , where N is the number of training points.

Let us consider the special case where the cost function is the mean squared error, $c((\mathbf{x}_1, y_1, f(\mathbf{x}_1)), \dots, (\mathbf{x}_N, y_N, f(\mathbf{x}_N))) = \frac{1}{N} \sum_{j=1}^N (f(\mathbf{x}_j) - y_j)^2$, and the regularizer Ω is zero. The solution for $\mathbf{a} = (a_1, \dots, a_N)$ is readily

¹Note that with a slight abuse of notation, we nevertheless used the transpose to denote the dot product in that space.

²for conditions on uniqueness of the solution, see [6]

computed by setting the derivative of (8) with respect to the vector \mathbf{a} equal to zero; it takes the form $\mathbf{a} = K^{-1}\mathbf{y}$, hence³

$$y = f(\mathbf{x}) = \mathbf{a}^\top \mathbf{z}(\mathbf{x}) = \mathbf{y}^\top K^{-1} \mathbf{z}(\mathbf{x}), \quad (10)$$

where $\mathbf{z}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), k(\mathbf{x}, \mathbf{x}_2), \dots, k(\mathbf{x}, \mathbf{x}_N))^\top \in \mathbb{R}^N$.

Volterra series as linear operator in RKHS. We now have the prerequisites to develop an alternative approach to estimating the Wiener series expansion. As our first step, we have to convert the Volterra series into a form suitable for regression in RKHS. Our starting point is the discretized version of the Volterra operators from Eq. (5) which is also the base of the classical cross-correlation procedure. The n th-order Volterra operator is a weighted sum of all n th-order monomials of the input vector \mathbf{x} . For $n = 0, 1, 2, \dots$ we define the map ϕ_n as

$$\phi_0(\mathbf{x}) = 1 \quad \text{and} \quad \phi_n(\mathbf{x}) = (x_1^n, x_1^{n-1}x_2, \dots, x_1x_2^{n-1}, x_2^n, \dots, x_m^n) \quad (11)$$

such that ϕ_n maps the input $\mathbf{x} \in \mathbb{R}^m$ into a vector $\phi_n(\mathbf{x}) \in \mathbb{F}_n = \mathbb{R}^{m^n}$ containing all m^n ordered monomials of degree n . Using ϕ_n , we can write the n th-order Volterra operator in Eq. (5) as a scalar product in \mathbb{F}_n ,

$$H_n[\mathbf{x}] = \eta_n^\top \phi_n(\mathbf{x}), \quad (12)$$

with the coefficients stacked into the vector $\eta_n = (h_{1,1,\dots,1}^{(n)}, h_{1,2,\dots,1}^{(n)}, h_{1,3,\dots,1}^{(n)}, \dots)^\top \in \mathbb{F}_n$. Fortunately, the functions ϕ_n constitute a RKHS. It can be easily shown (e.g., [6]) that

$$\phi_n(\mathbf{x}_1)^\top \phi_n(\mathbf{x}_2) = (\mathbf{x}_1^\top \mathbf{x}_2)^n = k_n(\mathbf{x}_1, \mathbf{x}_2). \quad (13)$$

The same idea can be applied to the entire p th-order Volterra series. By stacking the maps ϕ_n into a single map $\phi^{(p)}(\mathbf{x}) = (\phi_0(\mathbf{x}), \phi_1(\mathbf{x}), \dots, \phi_p(\mathbf{x}))^\top$, one obtains a mapping from \mathbb{R}^m into $\mathbb{F}^{(p)} = \mathbb{R} \times \mathbb{R}^m \times \mathbb{R}^{m^2} \times \dots \times \mathbb{R}^{m^p} = \mathbb{R}^M$ with dimensionality $M = \frac{1-m^{p+1}}{1-m}$. The entire p th-order Volterra series can be written as a scalar product in $\mathbb{F}^{(p)}$

$$\sum_{n=0}^p H_n[\mathbf{x}] = (\eta^{(p)})^\top \phi^{(p)}(\mathbf{x}) \quad (14)$$

with $\eta^{(p)} \in \mathbb{F}^{(p)}$. Since $\mathbb{F}^{(p)}$ is generated as a Cartesian product of the single spaces \mathbb{F}_n , the associated scalar product is simply the sum of the scalar products in the \mathbb{F}_n :

$$\phi^{(p)}(\mathbf{x}_1)^\top \phi^{(p)}(\mathbf{x}_2) = \sum_{n=0}^p (\mathbf{x}_1^\top \mathbf{x}_2)^n = k^{(p)}(\mathbf{x}_1, \mathbf{x}_2). \quad (15)$$

Thus, we have shown that the discretized Volterra series can be expressed as a linear operator in a RKHS. A similar approach has been taken by [1] using the inhomogeneous polynomial kernel

$$k_{inh}^{(p)}(\mathbf{x}_1, \mathbf{x}_2) = (1 + \mathbf{x}_1^\top \mathbf{x}_2)^p. \quad (16)$$

³If K is not invertible, K^{-1} denotes the pseudo-inverse of K .

This kernel implies a map ϕ_{inh} into the same space of monomials, but it weights the degrees of the monomials differently as can be seen by applying the binomial theorem to (16).

Implicit Wiener series estimation. As we stated above, the p th-degree Wiener expansion is the p th-order Volterra series that minimizes the squared error if the input is white Gaussian noise with zero mean. This can be put into the regression framework: assume we generate white Gaussian noise with zero mean, feed it into the unknown system and measure its output. Since any finite Volterra series can be represented as a linear operator in the corresponding RKHS, we can find the p th-order Volterra series that minimizes the squared error by linear regression. This, by definition, must be the p th-degree Wiener series since no other Volterra series has this property⁴. From Eqns. (4) and (10), we obtain the following expressions for the implicit Wiener series

$$G_0[\mathbf{x}] = \frac{1}{N} \mathbf{y}^\top \mathbf{1}, \quad \sum_{n=0}^p G_n[\mathbf{x}] = \sum_{n=0}^p H_n[\mathbf{x}] = \mathbf{y}^\top K_p^{-1} \mathbf{z}^{(p)}(\mathbf{x}) \quad (17)$$

where the Gram matrix K_p and the coefficient vector $\mathbf{z}^{(p)}(\mathbf{x})$ are computed using the kernel from Eq. (15) and $\mathbf{1} = (1, 1, \dots)^\top \in \mathbb{R}^N$. Note that the Wiener series and its Volterra functionals are represented only implicitly since we are using the RKHS representation as a sum of scalar products with the training points. Thus, we can avoid the ‘‘curse of dimensionality’’, i.e., there is no need to compute the possibly large number of coefficients explicitly.

The explicit Volterra and Wiener expansions can be recovered from Eq. (17) by collecting all terms containing monomials of the desired order and summing them up. The individual n th-order Volterra operators ($p > 0$) are given implicitly by

$$H_n[\mathbf{x}] = \mathbf{y}^\top K_p^{-1} \mathbf{z}_n(\mathbf{x}) \quad (18)$$

with $\mathbf{z}_n(\mathbf{x}) = ((\mathbf{x}_1^\top \mathbf{x})^n, (\mathbf{x}_2^\top \mathbf{x})^n, \dots, (\mathbf{x}_N^\top \mathbf{x})^n)^\top$. For $p = 0$ the only term is the constant zero-order Volterra operator $H_0[\mathbf{x}] = G_0[\mathbf{x}]$. The coefficient vector $\eta_n = (h_{1,1,\dots,1}^{(n)}, h_{1,2,\dots,1}^{(n)}, h_{1,3,\dots,1}^{(n)}, \dots)^\top$ of the explicit Volterra operator is obtained as

$$\eta_n = \Phi_n^\top K_p^{-1} \mathbf{y} \quad (19)$$

using the design matrix $\Phi_n = (\phi_n(\mathbf{x}_1)^\top, \phi_n(\mathbf{x}_1)^\top, \dots, \phi_n(\mathbf{x}_1)^\top)^\top$.

The individual Wiener functionals can only be recovered by applying the regression procedure twice. If we are interested in the n th-degree Wiener functional, we have to compute the solution for the kernels $k^{(n)}(\mathbf{x}_1, \mathbf{x}_2)$ and $k^{(n-1)}(\mathbf{x}_1, \mathbf{x}_2)$. The Wiener functional for $n > 0$ is then obtained from the difference of the two results as

$$\begin{aligned} G_n[\mathbf{x}] &= \sum_{i=0}^n G_i[\mathbf{x}] - \sum_{i=0}^{n-1} G_i[\mathbf{x}] \\ &= \mathbf{y}^\top \left[K_n^{-1} \mathbf{z}^{(n)}(\mathbf{x}) - K_{n-1}^{-1} \mathbf{z}^{(n-1)}(\mathbf{x}) \right]. \end{aligned} \quad (20)$$

⁴assuming symmetrized Volterra kernels which can be obtained from any Volterra expansion.

The corresponding i th-order Volterra operators of the n th-degree Wiener functional are computed analogously to Eqns. (18) and (19).

Orthogonality. The resulting Wiener functionals must fulfill the orthogonality condition which in its strictest form states that a p th-degree Wiener functional must be orthogonal to all monomials in the input of lower order. Formally, we will prove the following

Theorem 1 *The functionals obtained from Eq. (20) fulfill the orthogonality condition*

$$E[m(\mathbf{x})G_p[\mathbf{x}]] = 0 \quad (21)$$

where E denotes the expectation over the input distribution and $m(\mathbf{x})$ an i th-order monomial with $i < p$.

We will show that this is a consequence of the least squares fit of any linear expansion in a set of basis functions of the form of Eq. (6). In the case of the Wiener and Volterra expansions, the basis functions $\varphi_j(\mathbf{x})$ are monomials of the components of \mathbf{x} .

We denote the error of the expansion as $e(\mathbf{x}) = y - \sum_{j=1}^M \gamma_j \varphi_j(\mathbf{x}_i)$. The minimum of the expected quadratic loss L with respect to the expansion coefficient γ_k is given by

$$\frac{\partial L}{\partial \gamma_k} = \frac{\partial}{\partial \gamma_k} E \|e(\mathbf{x})\|^2 = -2E[\varphi_k(\mathbf{x})e(\mathbf{x})] = 0. \quad (22)$$

This means that, for an expansion of the type of Eq. (6) minimizing the squared error, the error is orthogonal to all base functions used in the expansion.

Now let us assume we know the Wiener series expansion (which minimizes the mean squared error) of a system up to degree $p - 1$. The approximation error is given by the sum of the higher-order Wiener functionals $e(\mathbf{x}) = \sum_{n=p}^{\infty} G_n[\mathbf{x}]$, so $G_p[\mathbf{x}]$ is part of the error. As a consequence of the linearity of the expectation, Eq. (22) implies

$$\sum_{n=p}^{\infty} E[\varphi_k(\mathbf{x})G_n[\mathbf{x}]] = 0 \quad \text{and} \quad \sum_{n=p+1}^{\infty} E[\varphi_k(\mathbf{x})G_n[\mathbf{x}]] = 0 \quad (23)$$

for any φ_k of order less than p . The difference of both equations yields $E[\varphi_k(\mathbf{x})G_p[\mathbf{x}]] = 0$, so that $G_p[\mathbf{x}]$ must be orthogonal to any of the lower order basis functions, namely to all monomials with order smaller than p . \square

For both the regression and the orthogonality of the resulting functionals, the assumption of white Gaussian noise was not required. In practice, this means that we can compute a Volterra expansion of type (17) for any type of input, not just for Gaussian noise. Note, however, that the orthogonality of functionals can be only defined with respect to an input distribution. If we use Eq. (20) for non-Gaussian input the resulting functionals will still be orthogonal, but with respect to the non-Gaussian input distribution. The resulting decomposition of the Volterra series into orthogonal functionals will be different from the Gaussian case. As a consequence, the functionals computed according to Eq. (20) will be different from the Wiener functionals.

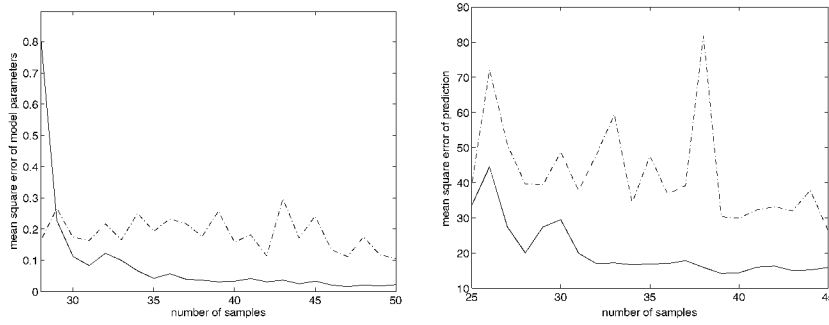


Figure 1: *Left*: Mean squared error between true and estimated coefficients of the second-order Wiener kernel of a correlation-type motion detector for varying number of training samples. *Right*: Mean squared prediction error on test set. The solid line depicts the error of the regression technique, the dash-dot line that of the cross-correlation technique.

EXPERIMENTS

Convergence. In this example, the system to be identified is a simple correlation-type motion detector of the form $y[n] = (\sum_{k=0}^4 h[k]x[n - k])(\sum_{k=0}^4 l[k]x[n + 1 - k])$ with some arbitrary, but fixed high pass $h[k]$ and low pass $l[k]$ of order 5. The data are generated by sliding two windows of width $x[n - 4] \dots x[n]$ and $x[n - 3] \dots x[n + 1]$ over a time series of white, zero-mean Gaussian noise and simultaneously measuring the system output $y[n]$. Finally, we added white, zero-mean Gaussian measurement noise to the signal with a variance of 10% of the signal variance. We applied both estimation methods, cross-correlation and regression, to estimate the 21 free parameters of the second-degree Wiener model $\hat{y} = \sum_{i=0}^2 G_i[\mathbf{x}]$. The modeling error ϵ was measured for the second order kernel using $\epsilon = \sum_{i=1}^5 (\eta_i - h_i)^2$ and averaging ϵ over the 20 trials. We varied the number of training samples to see how the modeling error decreases with the number of samples. For comparison, we also computed the prediction error on an independent test set of 100 examples.

As the result shows (Fig. 1), both the modeling and the prediction error of the regression technique decreases at a significantly faster rate than the cross-correlation method due to the unfavorable properties of the cross-correlation estimator. In fact, a comparable modeling error is only reached at sample sizes that are more than 10 times as large (not contained in the figure).

Reconstruction of a fifth-order nonlinear receptive field. This experiment demonstrates the applicability of the proposed method to high-dimensional input. Our example is the fifth-order system $y = \left(\sum_{k,l=1}^{16} h_{kl} x_{kl} \right)^5$ that acts on 16×16 image patches by convolving them with a receptive field h_{kl} of the same size shown in Fig. 2a before the nonlinearity is applied. We generated 2500 image patches containing uniformly distributed white noise

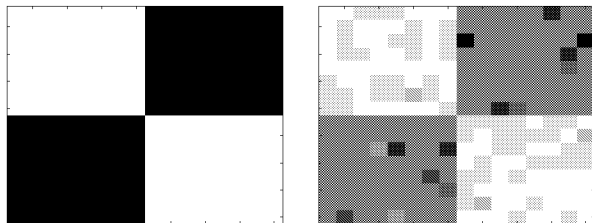


Figure 2: *Left:* 16×16 nonlinear receptive field of the test system; *Right:* Reconstructed receptive field from the fifth-order Volterra kernel by computing a preimage (after 2500 samples).

and computed the corresponding system output to which, as above, we added 10% Gaussian measurement noise. The resulting data was used to estimate the implicit Wiener expansion using the regression procedure. In the classical cross-correlation procedure, this would require the computation of roughly 9.5 billion independent terms for the fifth-order Wiener kernel. Moreover, even for much lower-dimensional problems, it usually takes tens of thousands of samples until a sufficient convergence is reached.

Even if all entries of the fifth-order Wiener kernel were known, it would be still hard to interpret the result in terms of its effect on the input signal. The implicit representation of the Volterra series allows for the use of preimage techniques (e.g., [6]) where one tries to choose a point \mathbf{z} in the input space such that the nonlinearly mapped image in \mathbb{F} , $\phi(\mathbf{z})$, is as close as possible to the representation in RKHS. In the case of the fifth-order Wiener kernel, this amounts to representing $H_5[\mathbf{x}]$ by the operator $(\mathbf{z}^\top \mathbf{x})^5$ with an appropriately chosen preimage $\mathbf{z} \in \mathbb{R}^{256}$. The nonlinear map $z \mapsto z^5$ is invertible, so that we can use the direct technique described in [6] where one applies the implicitly given Volterra operator from Eq. (18) to each of the canonical base vectors of \mathbb{R}^{256} resulting in a 256-dimensional response vector \mathbf{e} . The preimage is obtained as $\mathbf{z} = \sqrt[5]{\mathbf{e}}$. The result in Fig. 2b demonstrates that the original receptive field is already recognizable after using 2500 samples. The example shows that the preimage technique elucidates to which input structures the Volterra-operator is tuned, similar to the classical analysis techniques in linear systems.

CONCLUSION

The benefits of the proposed estimation technique of the Wiener expansions via kernel regression can be summarized as follows:

1. The implicit representation of the Wiener series allows for system identification with high-dimensional input signals. Essentially, this is due to the representer theorem: although a higher order series expansion contains a huge number of coefficients, it turns out that when estimating such a series from a finite sample, the information in the coefficients can be represented more parsimoniously using an example-based implicit representation.

2. Convergence is considerably faster than in the classical procedure because the estimation is done directly on the data. The regression method

omits the intermediate step of estimating cross-correlations which converges very slowly.

3. Preimage techniques reveal the input structures to which the Wiener operators are tuned. The preimage corresponds to a nonlinear receptive field where the input is convolved with a linear filter whose output is fed into a nonlinearity. The present method works only for Volterra kernels of odd order. More general techniques exist, including the case of other kernels and the computation of approximations in terms of several preimages (“reduced sets”) [6]. The latter corresponds to an invariant subspace of the Volterra operator (cf. [3]).

4. The method works also for non-Gaussian input. In particular, uniform noise turned out to lead to better results than Gaussian noise since its values are bounded. The Gaussian distribution sometimes produces very large values which are extremely amplified by the higher order monomial terms.

From the point of view of learning theory, the proposed estimation method has the drawback that the regularization term in the objective function (8) is currently set to zero in order to preserve the orthogonality property of the resulting Wiener functionals. This may possibly lead to a degraded generalization performance and an increased sensitivity to noise. Currently, we are investigating regularization techniques [2], e.g., penalizing the RKHS-norm of the solution (as it is done in Support Vector Machines). The resulting Volterra model can still be converted to a Wiener series by applying the orthogonalization procedure described in [5]. In future work, we plan to explore iterative estimation techniques to accommodate large sample sizes.

Acknowledgments. The ideas presented in this paper have greatly profited from discussions with G. Bakır, M. Kuss, and C. Rasmussen.

REFERENCES

- [1] T. J. Dodd and R. F. Harrison, “A new solution to Volterra series estimation,” in **CD-Rom Proc. 2002 IFAC World Congress**, 2002.
- [2] M. O. Franz, Y. Kwon, C. E. Rasmussen and B. Schölkopf, “Semi-supervised kernel regression using whitened function classes,” in **Proc. DAGM04, 26th Pattern Recognition Symposium**, subm., 2004.
- [3] A. Hyvärinen and P. Hoyer, “Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces,” **Neural Computation**, vol. 12, pp. 1705 – 1720, 2000.
- [4] A. Papoulis, **Probability, random variables and stochastic processes**, Boston: McGraw-Hill, 1991.
- [5] M. Schetzen, **The Volterra and Wiener theories of nonlinear systems**, Malabar: Krieger, 1989.
- [6] B. Schölkopf and A. J. Smola, **Learning with kernels**, Cambridge, MA: MIT Press, 2002.
- [7] V. Volterra, “Sopra le funzioni che dipendono de altre funzioni,” in **Rend. R. Accademia dei Lincei 2° Sem.**, 1887, pp. 97 – 105, 141 – 146, and 153 – 158.
- [8] N. Wiener, **Nonlinear problems in random theory**, New York: Wiley, 1958.