COMPUTATIONAL ALGORITHM FOR NONPARAMETRIC MODELLING OF NONLINEARITIES IN HAMMERSTEIN SYSTEMS

Przemysław Śliwiński and Zygmunt Hasiewicz
Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology
W. Wyspiańskiego 27, Wrocław, Poland

Keywords: Hammerstein system, Non-parametric identification, Orthogonal expansions, Regression estimation, Computational algorithm.

Abstract: In the paper a fast computational routines for identification algorithms for recovering nonlinearities in Hammerstein systems based on orthogonal series expansions of functions are proposed. It is ascertained that both, convergence conditions and convergence rates of the computational algorithms are the same as their much less computationaly attractive 'theoretic' counterparts. The generic computational algorithm is derived and illustrated by three examples based on standard orthogonal series on interval, *viz.* Fourier, Legendre, and Haar systems. The exemplary algorithms are presented in a detailed, ready-to-implement, form and examined by means of computer simulations.

1 INTRODUCTION

Recursive routines for nonparametric identification are of interest for practitioners mainly because the *recursive formulas*, involving only the last estimate value and/or the current measurements, are much simpler and much less computationally demanding than their closed-form counterparts, and hence, they seem to be more suitable for applications with limited computational capabilities (*e.g.* in power constrained mobile and/or remote devices).

The advantages of the recursive *orthogonal series* identification algorithms presented here may thus be of importance for a wide range of prospective users, since Hammerstein systems (*i.e.* the cascades of nonlinear static element followed by the linear dynamics; Fig. 1) are a popular modelling tool in many fields, see (Giannakis and Serpedin, 2001); *e.g.* in biocybernetics: (Westwick and Kearney, 2001; Dempsey and Westwick, 2004; Kukreja et al., 2005), chemistry: (Eskinat et al., 1991), control: (Lin, 1994; Zi-Qiang, 1993; Zhu and Seborg, 1994), and in economy: (Capobianco, 2002).

In the paper the new fast routine for a generic orthogonal series algorithm modelling a nonlinear characteristic in Hammerstein systems is proposed and three examples, employing representative orthogonal bases on intervals, are presented. Namely, the following algorithms are provided in a unified and ready-to-implement form:

- the *Fourier* trigonometric,
- the *Legendre* polynomial, and
- the *Haar* wavelet algorithm.

Nonparametric estimates[1] are well known for their flexibility. They allow to model virtually any nonlinearity – be it continuous or not – exploiting the measurement set only, see *e.g.* (Härdle, 1990; Györfi et al., 2002). Application of *orthogonal series*, in particular, enables evaluation of the estimates values in arbitrary points and at any stage of the identification process (in contrast to kernel-based recursive algorithms when the estimation points need to be set beforehand; see *e.g.* (Greblicki and Pawlak, 1989)).

2 REFERENCE ALGORITHM

The Hammerstein system under consideration is described by the discrete-time input-output equation

$$y_k = \sum_{i=0,1,\ldots} \lambda_i m(x_{k-i}) + z_k \qquad (1)$$

where $m(x)$ is the system nonlinearity, $\{\lambda_i\}$ is the impulse response of the dynamic subsystem, and z_k is

[1]The term 'nonparametric' refers to the *a priori* knowledge which is at ones disposal rather to the form of the resulting algorithm.

116 Śliwiński P. and Hasiewicz Z. (2009).
COMPUTATIONAL ALGORITHM FOR NONPARAMETRIC MODELLING OF NONLINEARITIES IN HAMMERSTEIN SYSTEMS.
In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics - Signal Processing, Systems Modeling and Control*, pages 116-123
DOI: 10.5220/0002207301160123

the external, additive noise. The standard nonparametric assumptions are imposed on the system characteristics, input signals and external noise; *cf.* (Greblicki and Pawlak, 1994; Greblicki and Pawlak, 2008; Śliwiński et al., 2009):

1. An input signal, $\{x_k\}$, and an external noise, $\{z_k\}$, are second-order random stationary processes. They are mutually independent and the latter is a zero-mean process. The input $\{x_k\}$ is white and has density, $f(x)$, strictly positive in the identification interval, say a standard unit interval, $[0,1]$.
2. A nonlinear characteristic of the static system, $m(x)$, has ν derivatives.
3. A linear dynamic subsystem is asymptotically stable. Its impulse response, $\{\lambda_i\}$, $i=0,1,\ldots$, is unknown.
4. A set, $\{(x_l,y_l)\}$, $l=1,2,\ldots,k,\ldots$ of the system input and output measurements is available.

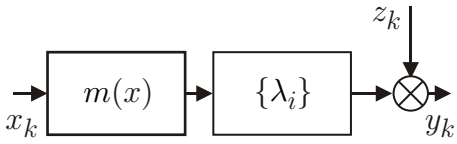

Figure 1: The identified Hammerstein system.

Remark 1. *Due to a composite structure of Hammerstein systems, only a scaled and shifted version of the characteristic $m(x)$ of the static block,* i.e. *the nonlinearity $\mu(x)=am(x)+b$, where $a=\lambda_0\neq 0$, $b=Em(x_1)\sum_{i=1}^{\infty}\lambda_i$, can at most be recovered from the input-output measurements. Indeed, the following holds,* cf. *(1) and (Greblicki and Pawlak, 2008):*

$$\begin{aligned} E(y_k|x_k=x) &= \lambda_0 m(x)+Ez_k \\ &\quad +E\sum_{i=1,\ldots}\lambda_i m(x_{k-i}) \\ &= \lambda_0 m(x)+b \end{aligned}$$

and to recover the genuine $m(x)$ in general case, one needs an additional a priori information about the nonlinearity, e.g. *its value in some points.*

The reference algorithm construction starts with the observation that any square integrable function in the unit interval $[0,1]$ may be represented by the orthogonal series (expansion):

$$\mu(x)=\sum_{m=0}^{\infty}\alpha_m\phi_m(x) \tag{2}$$

where $\{\phi_m\}$, $m=0,1,\ldots$ is a proper orthonormal basis on the interval $[0,1]$, and where

$$\alpha_m=\langle\phi_m,\mu\rangle=\int_0^1\phi_m(x)\mu(x)\,dx \tag{3}$$

are the expansion (generalized Fourier) coefficients associated with ϕ_m's. Let $\mu_{\mathfrak{m}}(x)$ be an $\mathfrak{m}$-term approximation (cut-off) of $\mu(x)$, that is, let (*cf.* (2))

$$\mu_{\mathfrak{m}}(x)=\sum_{m=0}^{\mathfrak{m}}\alpha_m\phi_m(x). \tag{4}$$

Due to the completeness of the basis $\{\phi_m\}$ we have

$$\int_0^1[\mu(x)-\mu_{\mathfrak{m}}(x)]^2\,dx\to 0 \text{ as } \mathfrak{m}\to\infty$$

for virtually any $\mu(x)$; *cf.* (2). Moreover, due to orthogonality of $\{\phi_m\}$, the approximation accuracy grows with the increasing number of approximation terms, $\mathfrak{m}$, as the approximation error is $\sum_{m=\mathfrak{m}+1}^{\infty}\alpha_m^2$.

Assume now that for any k, the earlier and present measurements $\{(x_l,y_l)\}$, $l=1,\ldots,k$, are sorted (ordered) increasingly with respect to the input values x_l. Then, the orthogonal series reference algorithm may have the following natural form (*cf.* (4))

$$\bar{\mu}_{\mathfrak{m}}(x)=\sum_{m=0}^{\mathfrak{m}}\bar{\alpha}_m\phi_m(x) \tag{5}$$

where (*cf.* (3) and see (Greblicki and Pawlak, 1994; Greblicki and Pawlak, 2008))

$$\bar{\alpha}_m=\sum_{l=1}^{k}y_l\int_{x_{l-1}}^{x_l}\phi_m(u)\,du \tag{6}$$

are estimates of the true expansion coefficients α_m (with $x_0=0$). The following theorem describes the limit properties of the reference algorithm:

Theorem 1. *If the number $\mathfrak{m}$ of terms in (5),* i.e. *the number of the estimated coefficients $\bar{\alpha}_m$ in the algorithms, increases with the measurements number k so that*

$$\mathfrak{m}\to\infty \text{ and } \mathfrak{m}/k\to 0 \text{ as } k\to\infty,$$

then

$$E\int_0^1[\mu(x)-\bar{\mu}_{\mathfrak{m}}(x)]^2\,dx\to 0 \text{ as } k\to\infty.$$

Moreover, for Fourier and Legendre series the algorithm attains, for $\mathfrak{m}=\left\lfloor k^{1/(2\nu+1)}\right\rfloor$, the best possible asymptotic convergence rate, i.e. *for any $\varepsilon>0$, it holds for them that*

$$E\int_{\varepsilon}^{1-\varepsilon}[\mu(x)-\bar{\mu}_{\mathfrak{m}}(x)]^2\,dx=O\left(k^{-2\nu/(2\nu+1)}\right)$$

while the convergence of the Haar series algorithm achieves, for $\mathfrak{m}=\lfloor k^{1/3}\rfloor$, the asymptotic rate

$$E\int_0^1[\mu(x)-\bar{\mu}_{\mathfrak{m}}(x)]^2\,dx=O\left(k^{-2/3}\right)$$

for any $\nu=1,2,\ldots$.

Proof. The proofs of the theorem for the algorithms with Fourier trigonometric and Legendre polynomial bases can be found in (Greblicki and Pawlak, 1994) and in (Greblicki and Pawlak, 2008). The proof for Haar wavelet algorithm is in (Greblicki and Śliwiński, 2002). □

Remark 2. *Using sorted measurements results in a non-quotient form of the identification algorithms. Such a form (achieved at a moderate cost of keeping the measurement data sorted) can be seen as superior to the alternate quotient-form estimates from the stability and numerical error standpoint, especially, when the number of measurement data is small or moderate (see (Śliwiński, 2009a; Śliwiński, 2009b) for* on-line *and* e.g. *(Greblicki, 1989; Pawlak and Hasiewicz, 1998; Hasiewicz, 1999; Hasiewicz, 2001; Hasiewicz et al., 2005) for off-line quotient orthogonal series algorithms). The orthogonal series algorithm (5)-(6) were presented in (Greblicki and Pawlak, 1994).*

3 COMPUTATIONAL (FAST) ALGORITHM

In a view of Theorem 1, the algorithm (5)-(6) possesses desirable theoretical properties. It however seems not to be computationally attractive for the following two reasons:

- calculating coefficient estimates needs integration, and
- updating the estimates, in case when the new measurement data appear, requires repeating the whole computation routine (6) 'right from scratch'.

Our goal is therefore to make the algorithm computationally efficient without sacrificing its prominent properties. Namely, the abovementioned numeric shortcomings maybe circumvented by:

- avoiding explicit integration in favor of subtraction, and
- providing a computation formula for recursive updating of coefficients estimates.

The goal is accomplished in the following fast generic routine. The first step is elementary – we simply apply here *The First Fundamental Theorem of Calculus* to get the *integration-free* counterpart of the estimate in (6)

$$\bar{\alpha}_m = \sum_{l=1}^{k} y_l \left[\Phi_m(x_l) - \Phi_m(x_{l-1})\right] \tag{7}$$

where $\Phi_m(x)$ are *the indefinite integrals* for the basis functions $\phi_m(x)$. The second step is described in the following proposition (being a generalization of the result presented in (Śliwiński et al., 2009) for wavelets).

Proposition 2. *Let $\bar{\alpha}_m^{(k)}$ denote the estimate of the expansion coefficient α_m obtained for k measurements. Given the ordered sequence, $\{(x_1,y_1),\ldots,(x_l,y_l),(x_{l+1},y_{l+1}),\ldots,(x_k,y_k)\}$, assume that for the new, $(k+1)$th measurement pair, (x_{k+1},y_{k+1}), it holds that $x_l < x_{k+1} < x_{l+1}$. Then,* (i) *the new pair is inserted between (x_l,y_l) and (x_{l+1},y_{l+1}) to maintain the ascending order of the updated measurement set, and* (ii) *the following recurrence formula should be applied to update the coefficient estimates*

$$\begin{aligned}\bar{\alpha}_m^{(k+1)} &= \bar{\alpha}_m^{(k)} + (y_{k+1} - y_{l+1}) \times \\ &\quad \times \left[\Phi_m(x_{k+1}) - \Phi_m(x_l)\right]\end{aligned} \tag{8}$$

with the initial values $\bar{\alpha}_m^{(0)} = 0$, and with the initial measurements set $\{(0,0),(1,0)\}$.

Proof. The proof is immediate. To derive the recurrence formula (8), it suffices to subtract the estimate in (7), computed for k, from the one obtained for $k+1$ measurements. □

Below we present three examples showing how to implement Fourier, Legendre, and Haar orthogonal systems in the general identification routine (5)-(8).

3.1 Fourier Trigonometric Series

Since sequence of trigonometric functions

$$\sqrt{1/2\pi}, \left\{\sqrt{1/\pi}\cos(mu), \sqrt{1/\pi}\sin(mu)\right\}$$

constitutes, for $m = 1,2,\ldots$, an orthogonal basis on the interval $[-\pi,\pi]$; *cf.* (Szego, 1974; Greblicki and Pawlak, 2008), thus for our identification interval, $[0,1]$, we need $\phi_0(x) = 1$ and

$$\begin{aligned}\phi_{2m-1}(x) &= \sqrt{2}\sin((2m-1)\pi x) \\ \phi_{2m}(x) &= \sqrt{2}\cos(2m\pi x)\end{aligned}$$

for $m = 1,2,\ldots$. From the above we immediately obtain $\Phi_0(x) = x$ and

$$\begin{aligned}\Phi_{2m-1}(x) &= -\tfrac{\kappa}{2m-1}\cos((2m-1)\pi x) \\ \Phi_{2m}(x) &= \tfrac{\kappa}{2m}\sin(2m\pi x)\end{aligned}$$

for $m = 1,2,\ldots$ and $\kappa = \sqrt{2}/\pi$.

3.2 Legendre Polynomial Series

The Legendre polynomials can be defined recursively as

$$p_{m+1}(x) = \frac{2m+1}{m+1} x p_m(x) + \frac{m}{m+1} p_{m-1}(x)$$

for $m = 1, 2, \ldots$ with $p_0(x) = 1$, $p_1(x) = x$; *cf.* (Szego, 1974; Greblicki and Pawlak, 2008). They form an orthonormal basis on the interval $[-1, 1]$ with the weighting function $\sqrt{(2m+1)/2}$. In our algorithm, for the unit interval we thus need a slightly reformulated

$$\phi_m(x) = \sqrt{2m+1}\, p_m(2x-1).$$

The following recurrence formula for primitives of Legendre polynomials holds (see the derivation in Proposition 3 in Appendix A).

$$P_{m+1}(x) = \kappa_m \left(x^2 - 1\right) p_m(x) + K_m P_{m-1}(x)$$

for $m = 1, 2, \ldots$, where $\kappa_m = (2m+1)/((m+1)(m+2))$, $K_m = m(m-1)/((m+1)(m+2))$ and $P_0(x) = x + 1$, $P_1(x) = \left(x^2 - 1\right)/2$. Eventually, we have

$$\Phi_m(x) = \frac{\sqrt{2m+1}}{2} P_m(2x-1).$$

3.3 Haar Wavelet Series

To construct Haar wavelet basis one needs two functions, the father and mother Haar wavelets:

$$\varphi(x) = I_{0 \le x < 1}(x) \text{ and } \psi(x) = \varphi(2x) - \varphi(2x-1)$$

and translations and dilations of the latter, *i.e.*

$$\psi_{kl} = 2^{k/2} \psi\left(2^k x - l\right)$$

where the indices k, l run through the ranges $1, \ldots,$ and $0, \ldots, 2^k - 1$, respectively; *cf. e.g.* (Wojtaszczyk, 1997). In our case the identification interval, $[0, 1]$, is the native one for Haar system and we can directly take

$$\phi_0(x) = \varphi(x) \text{ and } \phi_m(x) = \psi_{kl}(x)$$

for $m = 1, 2, \ldots$, where

$$k = \lfloor \log_2 m \rfloor \text{ and } l = m \bmod 2^k \tag{9}$$

and where $x \bmod y = x - y \cdot \lfloor x/y \rfloor$ denotes standard modulus function.

Since, in fact, the father wavelet, $\varphi(x)$, is merely a box function, then the primitives of basis functions, $\phi_m(x)$, are simply

$$\Phi_0(x) = x I_{0 \le x < 1}(x) + I_{x \ge 1}(x)$$

and

$$\Phi_m(x) = \frac{1}{\sqrt{2^{k+1}}} \left[\Phi_0\left(2^{k+1}x - l\right) - \Phi_0\left(2^{k+1}x - (l+1)\right)\right]$$

for $m = 1, 2, \ldots$ with k, l dependent on m and defined as in (9).

4 COMPUTATIONAL COMPLEXITY ANAYSIS

In what follows we compare the computational complexities of both the *reference* and the proposed *fast computational* versions.

4.1 Reference Algorithm

In the reference algorithm implementation one can naturally distinguish two phases with the main routine (5)-(6) preceded by sorting of the measurement sequence. The latter, employing a fast sorting algorithm (*e.g. quick sort*, *heap sort*; *cf.* (Knuth, 1998)), needs $O(k \log k)$ operations.

A naive implementation of the main routine (5)-(6) requires $O(\mathfrak{m}\eta)$ operations in (5) and $O(k\iota)$ operations in (6), where $O(\eta)$ is the cost of evaluating of $\phi_m(x)$ and where $O(\iota)$ is the cost of calculating of the definite integral for $\phi_m(x)$. The overall cost is therefore $O(\mathfrak{m}\eta \cdot k\iota)$. In a view of Theorem 1 this reads $O\left(k^{1+1/(2q+1)} \cdot \eta\iota\right)$. In case of the *Fourier* and *Haar* algorithms we have $O(\eta) = O(1)$. In the *Legendre* algorithm computing $\phi_m(x)$ (*i.e.* a polynomial of order m) takes m operations. The cost $O(\iota)$ of computing integrals (since the indefinite integrals for $\phi_m(x)$ are known) is the same.

Table 1: Complexities of a direct implementation of the reference algorithms.

Algorithm	Cost
Fourier	$O\left(k^{\frac{2}{2q+1}(q+1)}\right)$
Legendre	$O\left(k^{\frac{2}{2q+1}(q+2)}\right)$
Haar	$O\left(k^{\frac{4}{3}}\right)$

4.2 Fast Algorithm

Using the above naive implementation results in cost of at least $O\left(k^{2(q+1)/2q+1}\right)$ operations for every single new measurement to be added. Our algorithm (5), (8) substantially reduces this complexity. First, searching for the pairs (x_l, y_l) and (x_{l+1}, y_{l+1}) in the measurement sequence (employing *e.g.* a standard binary search algorithm) requires $O(\log k)$ operations; *cf.* (Knuth, 1998). Computing the updated value of $\bar{\mu}_{\mathfrak{m}}(x)$ requires another $O(\mathfrak{m}(k))$ operations. The overall cost of the *Fourier* and *Legendre* algorithms (in the latter the recurrence formulas (10), (11) are

used) is therefore of order $O(\log k) + O(\mathfrak{m}(k)) = O(\sqrt[2q+1]{k})$.

In case of the *Haar* algorithm, this cost can further be reduced to the order $O(\log k)$ after observation that, due to compactness of Haar functions supports, only $O(\log k)$ terms of are involved in computations of (5), (8). Indeed, using wavelet 'natural' scale-translation notation (*cf.* (9)), one can easily ascertain that, for each scaling index $n = 0, \ldots, \lfloor \log_2 \mathfrak{m}(k) \rfloor$, at most one function $\psi_{nl}(x)$ is non-zero – the one with translation index $l = \lfloor 2^n x \rfloor$. The computation phase of the *Haar* algorithm requires thus only $O(\log k)$ operations for $\mathfrak{m} = \lfloor \sqrt[3]{k} \rfloor$.

Table 2: Complexities of fast implementation of the reference algorithms.

Algorithm	Cost
Fourier	$O\left(k^{\frac{1}{2q+1}}\right)$
Legendre	$O\left(k^{\frac{1}{2q+1}}\right)$
Haar	$O(\log k)$

5 NUMERICAL EXPERIMENTS

The first two examples, the Fourier and Legendre algorithms, possess the same asymptotic behavior while the last, the Haar one, is slightly slower for smooth nonlinearities (*i.e.* for $\nu = 2, 3, \ldots$). However, as we will see in the following numerical experiments, this fact does not necessarily hold true for sample sizes being small or moderate.

To this end, the following piecewise-[smooth|linear|constant] characteristics, referred further to as the *root*, *ramp*, and *step* functions, respectively, were considered in the interval $[0, 1]$:

$$m(x) = \begin{cases} \sqrt[3]{u} \\ 2\left(u + \frac{1}{2}\right) I\left(u + \frac{1}{2}\right) + 2I\left(u - \frac{1}{2}\right) - 1 \\ I(u+1) - I(u) \end{cases}$$

where $u = 2x - 1$ and $I(x)$ is the abbreviated notation of the box function $I_{0 \le x < 1}(x)$. The number $\mathfrak{m}$ of estimate components, *i.e.* of coefficients in the algorithms, was governed by *the practical rule*, according to which $\mathfrak{m} = \lfloor \sqrt[3]{k} \rfloor$; *cf.* (Greblicki and Pawlak, 2008; Hasiewicz et al., 2005). The input $\{x_l\}$ was uniformly distributed over $[0, 1]$, and the (infinite) impulse response of the dynamic part was $\lambda_i = 2^{-i}$, $i = 0, 1, \ldots$ (thus we had exactly $\mu(x) = m(x)$ for all three nonlinearities, *cf.* Remark 1); the external zero-mean uniform noise was set to make $\max |z_l| / \max |m(x)| = 10\%$. Numerically computed MISE error served as

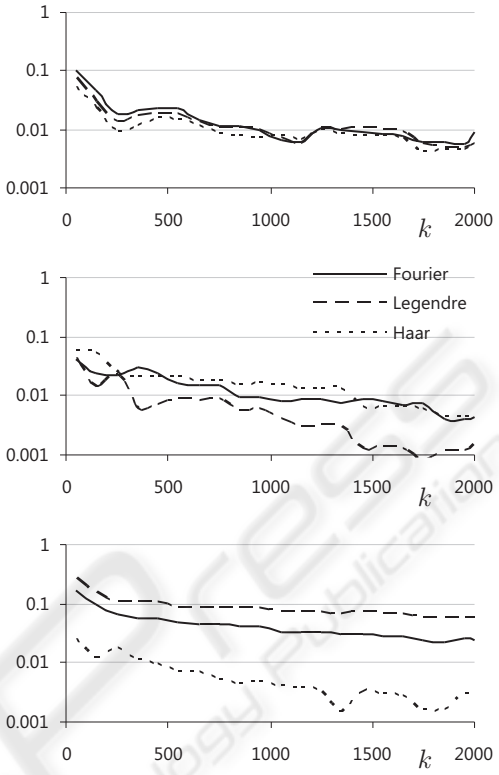

Figure 2: The algorithms errors for three test nonlinearities: **a)** root, **b)** ramp, and **c)** step one.

the indicator of algorithms accuracy (computed in slightly narrowed interval, $[\varepsilon, 1 - \varepsilon], \varepsilon = 0.1$, in order to avoid the boundary effect affecting Fourier algorithm (*cf.* Figs 2a and 3)).

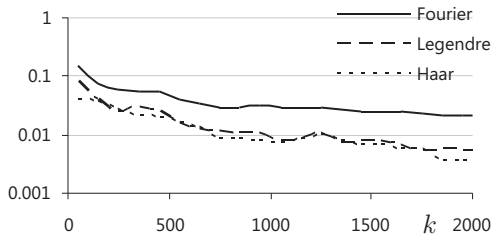

Figure 3: Boundary effect illustration.

The experiments unveil that algorithms offer similar accuracy for the root function. Slightly better performance of Legendre algorithm in case of ramp function and the Haar algorithm in case of step function can both be attributed to similarity of their basis functions to the respective nonlinearities. Nevertheless, the Haar wavelet algorithm – achieving the similar results and being much faster – can be pointed out as the most effective across the whole experiment.

6 FINAL REMARKS

The new class of fast routines for nonparametric identification algorithms recovering the nonlinearity in Hammerstein systems has been proposed. Preserving all the asymptotic properties of their off-line origins, the new algorithms offer much more computationally efficient formulas. Comparing the algorithm properties one can draw the following conclusions:

- *Fourier* algorithm is fast but prone to boundary effect,
- *Legendre* algorithms is the slowest but free boundary problems, finally
- *Haar* algorithm is fast but do not perform well in case of smooth nonlinearities (like the *Fourier* and *Legendre* do).

Remark 3. *Owing to the beneficial features pointed out above it is not a serious disadvantage that all measurement data need to be kept in our algorithm. This – admittedly idiosyncratic feature – is a consequence of both the form of the initial off-line version of the algorithm (6) and the random nature of the input data;* cf. *(Śliwiński et al., 2009). Moreover, the measurement set needs to be maintained only during the synthesis of the estimate. In the implementation step, all k measurements can be rid off and only* $\mathfrak{m}$ *coefficients (with* $\mathfrak{m}$ *being a significantly smaller number than k) have to be stored. Observe also that in all nonparametric algorithms, be them kernel or k – NN algorithms, see* e.g. *(Györfi et al., 2002; Greblicki and Pawlak, 2008), the measurements need to be kept as well in order to allow computing the estimate value in arbitrary point.*

That the measurements need to be kept in nonparametric modelling is rather typical as the measurements are essentially the only source of the information about the system/phenomenon. This problem is addressed in (Śliwiński, 2009a; Śliwiński, 2009b) where the quotient form wavelet algorithm is proposed. It is shown there that – on the one hand side – getting rid of the measurements allows the algorithms to be asymptotically equivalent to those possessing all the data, but – on the other – reveals that for small and moderate measurements number such algorithm perform worse.

Finally, we would like to emphasize that the simplicity of the proposed computational algorithm should be seen as an advantage for the practitioners as it allows a straightforward implementation (*cf.* the Appendix).

REFERENCES

Capobianco, E. (2002). Hammerstein system representation of financial volatility processes. *The European Physical Journal B - Condensed Matter*, 27:201–211.

Dempsey, E. and Westwick, D. (2004). Identification of Hammerstein models with cubic spline nonlinearities. *IEEE Transactions on Biomedical Engineering*, 51(2):237–245.

Eskinat, E., Johnson, S. H., and Luyben, W. L. (1991). Use of Hammerstein models in identification of non-linear systems. *American Institute of Chemical Engineers Journal*, 37:255–268.

Giannakis, G. B. and Serpedin, E. (2001). A bibliography on nonlinear system identification. *Signal Processing*, 81:533–580.

Greblicki, W. (1989). Nonparametric orthogonal series identification of Hammerstein systems. *International Journal of Systems Science*, 20:2355–2367.

Greblicki, W. and Pawlak, M. (1989). Recursive nonparametric identification of Hammerstein systems. *Journal of the Franklin Institute*, 326(4):461–481.

Greblicki, W. and Pawlak, M. (1994). Dynamic system identification with order statistics. *IEEE Transactions on Information Theory*, 40:1474–1489.

Greblicki, W. and Pawlak, M. (2008). *Nonparametric system identification*. Cambridge University Press, New York.

Greblicki, W. and Śliwiński, P. (2002). Non-linearity estimation in Hammerstein system based on ordered observations and wavelets. In *Proceedings 8th IEEE International Conference on Methods and Models in Automation and Robotics – MMAR 2002*, pages 451–456, Szczecin. Institute of Control Engineering, Technical University of Szczecin.

Györfi, L., Kohler, M., A. Krzyżak, and Walk, H. (2002). *A Distribution-Free Theory of Nonparametric Regression*. Springer-Verlag, New York.

Härdle, W. (1990). *Applied Nonparametric Regression*. Cambridge University Press, Cambridge.

Hasiewicz, Z. (1999). Hammerstein system identification by the Haar multiresolution approximation. *International Journal of Adaptive Control and Signal Processing*, 13(8):697–717.

Hasiewicz, Z. (2001). Non-parametric estimation of nonlinearity in a cascade time series system by multiscale approximation. *Signal Processing*, 81(4):791–807.

Hasiewicz, Z., Pawlak, M., and Śliwiński, P. (2005). Nonparametric identification of non-linearities in block-oriented complex systems by orthogonal wavelets with compact support. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 52(1):427–442.

Knuth, D. E. (1998). *The Art of Computer Programming. Volume 3. Sorting and searching*. Addison-Wesley Longman Publishing Co., Inc, Boston, MA.

Kukreja, S., Kearney, R., and Galiana, H. (2005). A least-squares parameter estimation algorithm for switched Hammerstein systems with applications to the VOR. *IEEE Transactions on Biomedical Engineering*, 52(3):431–444.

Lin, S.-K. (1994). Identification of a class of nonlinear deterministic systems with application to manipulators. *IEEE Transactions on Automatic Control*, 39(9):1886–1893.

Pawlak, M. and Hasiewicz, Z. (1998). Nonlinear system identification by the Haar multiresolution analysis. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 45(9):945–961.

Śliwiński, P. (2009a). On-line wavelet estimation of nonlinearities in Hammerstein systems. Part I - algorithm and limit properties. *IEEE Transactions on Automatic Control*. Submitted for review.

Śliwiński, P. (2009b). On-line wavelet estimation of nonlinearities in Hammerstein systems. Part II - small sample size properties. *IEEE Transactions on Automatic Control*. Submitted for review.

Śliwiński, P., Rozenblit, J., Marcellin, M. W., and Klempous, R. (2009). Wavelet amendment of polynomial models in nonlinear system identification. *IEEE Transactions on Automatic Control*, 54(4):820–825.

Szego, G. (1974). *Orthogonal Polynomials*. American Mathematical Society, Providence, R.I., 3rd edition.

Westwick, D. T. and Kearney, R. E. (2001). Separable least squares identification of nonlinear Hammerstein models: Application to stretch reflex dynamics. *Annals of Biomedical Engineering*, 29(8):707–718.

Wojtaszczyk, P. (1997). *A Mathematical Introduction to Wavelets*. Cambridge University Press, Cambridge.

Zhu, X. and Seborg, D. E. (1994). Nonlinear predictive control based on Hammerstein system. *Control Theory Applications*, 11:564–575.

Zi-Qiang, L. (1993). Controller design oriented model identification method for Hammerstein system. *Automatica*, 29:767–771.

APPENDIX

Recursion in Legendre Polynomials

The well known recurrence relation between Legendre polynomials of adjacent orders (see *e.g.* (Szego, 1974)), *i.e.*:

$$(m+1)\,p_{m+1}(x) = (2m+1)\,x p_m(x) + m p_{m-1}(x)$$

allows convenient generation of increasing order elements of polynomial orthogonal basis

$$p_{m+1}(x) = \tfrac{2m+1}{m+1} x p_m(x) + \tfrac{m}{m+1} p_{m-1}(x) \qquad (10)$$

for $m = 1,2,\ldots$, given $p_0(x) = 1$ and $p_1(x) = x$.

In the following proposition we show that the similar relation holds for primitive functions of these polynomials.

Proposition 3. *Let $P_m(x) = \int_{-1}^{x} p_m(u)\,du$. The following recurrence relation holds*

$$\begin{aligned} P_{m+1}(x) &= \tfrac{(2m+1)\left(x^2-1\right)}{(m+1)(m+2)} p_m(x) \\ &\quad + \tfrac{m(m-1)}{(m+1)(m+2)} P_{m-1}(x) \end{aligned} \qquad (11)$$

for $m = 1,2,\ldots$ and with

$$P_0(x) = x+1 \text{ and } P_1(x) = \tfrac{1}{2}\left(x^2-1\right).$$

Proof. We will give only a sketch of the proof as it involves elementary (yet a bit tedious) calculations. Integrating both sides of the formula in (10) yields

$$P_{m+1}(x) = \tfrac{2m+1}{m+1} \int_{-1}^{x} u p_m(u)\,du - \tfrac{m}{m+1} P_{m-1}(x) \qquad (12)$$

Employing now integration by parts and another known recursive formula:

$$\left(1-x^2\right) p'_m(x) = m\left[p_{m-1}(x) - x p_m(x)\right]$$

we get

$$\int_{-1}^{x} u p_m(u)\,du = \tfrac{x^2-1}{m+2} p_m(x) + \tfrac{m}{m+2} P_{m-1}(x)$$

which applied to (12) yields (11), and (after substitution `m := m - 1`) the formula used in subsequent C++ implementation. □

Code Samples

The following C++ implementations of the presented recursive formulas prove not to be much more intricate than their mathematical origins in (10):

```
template <typename T> struct p
{
 T operator()(T const &x, size_t m)const
 {
    T const _2_ = T(2), _1_ = T(1);
    if(m == 0) return _1_;
    if(m == 1) return  x ;
    p<T> const lp;
    return ((_2_*m-_1_)*x*lp(x, m-1)
             -  (m-_1_)*lp(x, m-2))/m;
 }
};
```

and in (11), respectively:

```
template <typename T> struct P
{
 T operator()(T const &x, size_t m)const
 {
   T const _2_ = T(2), _1_ = T(1);
   if(m == 0) return    x + _1_;
   if(m == 1) return (x*x - _1_)/_2_;
```

```
    p<T> const  lp;
    P<T> const lpi;
    return ( (_2_*m-_1_)*(x*x-_1_)
            * lp(x, m - 1)
            + (m - _1_) * (  m - _2_)
            * lpi(x, m - 2))
              /(m * (m + _1_));
  }
};
```