

Initializing Wiener-Hammerstein Models Based on Partitioning of the Best Linear Approximation^{*}

J. Sjöberg^{*} J. Schoukens^{**}

^{*} Chalmers University of Technology, Department of Signals & Systems, SE412 96 Gothenburg, Sweden (e-mail: Jonas.Sjoberg@chalmers.se)

^{**} Vrije Universiteit Brussel, Faculty of Engineering, Department of Fundamental Electricity and Instrumentation, Pleinlaan 2, 1050 Brussels, Belgium (e-mail: Johan.Schoukens@vub.ac.be)

Abstract: This paper describes a new algorithm for initializing and estimating Wiener-Hammerstein models. The algorithm makes use of the best linear model of the system which is split in all possible ways into two linear sub-models. For all possible splits, a Wiener-Hammerstein model is initialized which means that a nonlinearity is introduced in between the two sub-models. The linear parameters of this nonlinearity can be estimated using least-squares. All initialized models can then be ranked with respect to their fit. Typically, one is only interested in the best one, for which all parameters are fitted using prediction error minimization. The paper explains the algorithm and the consistency of the initialization is stated. Computational aspects are investigated, showing that in most realistic cases, the number of splits of the initial linear model remains low enough to make the algorithm useful. The algorithm is illustrated on an example where it is shown that the initialization is a tool to avoid many local minima.

Keywords: Wiener-Hammerstein systems, Hammerstein systems, Wiener systems, nonlinear system identification

1. INTRODUCTION

There has always been a need to identify nonlinear systems using measured data. In real life all systems are nonlinear to some extent but linear system theory and linear system identification methods have often been successfully applied. Theory for linear system identification is a fairly mature area, well covered in books like, eg, Ljung (1999); Söderström and Stoica (1989), focusing on time domain methods and Pintelon and Schoukens (2001), focusing on frequency domain methods.

One approach to nonlinear system identification is to work with block-oriented nonlinear models built up by combining blocks which are either linear dynamic or static nonlinear and in this paper the Wiener-Hammerstein (W-H) model is considered. It consists of two linear dynamic blocks and a static nonlinear block in the middle, see Figure 1.

The paper presents a novel consistent algorithm for estimating W-H models. It is based on the Best Linear Approximation (BLA) of the nonlinear system. That means that the algorithm starts with linear identification to obtain the BLA model. Under some mild assumptions, see

^{*} This work is sponsored by the Fund for Scientific Research (FWO-Vlaanderen), the Flemish Government (Methusalem 1), and the Belgian Federal Government (IUAP VI/4). The research was performed during Jonas Sjöberg's sabbatical at Vrije Universiteit Brussel, 2009-2010.

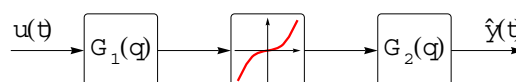


Fig. 1. A Wiener-Hammerstein model structure.

Section 4, the BLA model will be a consistent estimate of the concatenated dynamics of the two linear sub-systems. The core idea of the algorithm is simply to split the BLA model into two sub-models in all possible ways, and to initialize a W-H model with each one of these splits. This can be done quickly with the least squares algorithm so that a large number of splits can be handled fairly fast.

Earlier estimation algorithms for W-H models have been less general, or more complicated than the suggested one. In Crama and Schoukens (2005) an iterative initialization procedure is proposed which requires specially designed periodic excited input signals. This experimental requirement is relaxed in Schoukens et al. (2006). Techniques where the need of initial values is avoided by restricting the allowed model complexity are given in Bershad et al. (2001); Tan and Godfrey (2002, 2003). In Leith et al. (2003) an approach is presented which applies for the case that the linear sub-models can be described as FIR models.

In Wills and Ninness (2009) random, stable, initialization of the two sub-systems is investigated with quite good result. The authors show that there are many local minima, so, typically, the estimation need to be repeated several

times with different starting values to increase the chances to find a model corresponding to a good local minimum. In this sense, the proposed algorithm can be seen as a suggestion on how to obtain a good starting value to increase the chances to end up in a good minimum.

In Lauwers et al. (2009) an alternative algorithm based on splitting the BLA model is given. Instead of considering many splittings, they offer all poles and zeros to both linear models.

The paper is organized as follows. Section 2 details the problem formulation and the proposed algorithm is presented in Section 3. The consistency of the algorithm is shown in Section 4. Section 5 concerns the number of partitions of the initial linear model. In Section 6 the algorithm is illustrated on an example and in Section 7 the paper ends with conclusions.

2. PROBLEM FORMULATION

The problem formulation is divided into three steps, the definition of the model structure, the assumptions on the data, and the definition and computation of the estimate.

2.1 Model structure

The concerned model structure is of W-H type described by

$$\begin{aligned} z(t) &= G_1(q^{-1}, \alpha) u(t) \\ x(t) &= f(\beta, z(t)) \\ \hat{y}(t) &= G_2(q^{-1}, \gamma) x(t) + \nu(t) \end{aligned} \quad (1)$$

where where $\hat{y}(t)$ is the model's prediction of the output $y(t)$, and $G_1(q^{-1}, \alpha)$ and $G_2(q^{-1}, \gamma)$ are linear time invariant transfer functions in the delay operator q^{-1} , and parameterized with α and γ , respectively. The signal $\nu(t)$ is a zero mean stochastic process representing disturbances of the measurements and the function f is a static nonlinearity parameterized with β .

All parameters of the model structure are stored in a common parameter vector

$$\theta = [\alpha, \beta, \gamma]. \quad (2)$$

The first linear part of the model can be described as

$$G_1(q^{-1}, \alpha) = \frac{b_0^1 + b_1^1 q^{-1} + \dots + b_{m_{b_1}}^1 q^{-m_{b_1}}}{1 + a_1^1 q^{-1} + \dots + a_{m_{a_1}}^1 q^{-m_{a_1}}} \quad (3)$$

where $\alpha = [b_0^1, \dots, b_{m_{b_1}}^1, a_1^1, \dots, a_{m_{a_1}}^1]$, and $G_2(q^{-1}, \gamma)$ is described similarly with $\gamma = [b_0^2, \dots, b_{m_{b_2}}^2, a_1^2, \dots, a_{m_{a_2}}^2]$.

The static nonlinearity is described as a basis function expansion

$$\begin{aligned} f(\beta, z) &= \sum_{k=1}^n \beta_k^1 f_k(\beta_k^2, z) \\ \beta &= [\beta^1, \beta^2]^T = [\beta_1^1, \dots, \beta_n^1, \beta_1^2, \dots, \beta_n^2]^T \end{aligned} \quad (4)$$

where f_k are basis functions, and β has been divided into β^1 , which enters linearly in f , and β^2 which enters non-linearly in f . The parameters entering nonlinearly may indicate features like position and width of the basis function and, hence, each β_k^2 can contain several

parameters. With this general description of the static nonlinearity most specific basis function expansions can be described with a specific choice of the basis f_k . If, for example, a polynomial model is chosen, then

$$f(\beta, z) = \beta_0^1 + \beta_1^1 z + \beta_2^1 z^2 + \dots + \beta_n^1 z^n$$

and in this case there are no parameters in β^2 .

To define a model in this model structure, not only the parameters need to be determined but also the orders of the sub-models, and the type of basis function expansion in f .

2.2 Data

For the estimation of the parameters in the model (1) a data set is assumed to be available, $\{u(t), y(t)\}_{t=1}^N$ of N input $u(t)$ and output $y(t)$ samples.

The intermediate variables $z(t)$ and $x(t)$ are not available. For the consistency it is assumed that the input signal $u(t)$ is Gaussian. Otherwise rather mild conditions are needed on the data like the one in Ljung (1978) for the algorithm to be applicable.

2.3 Estimation

A standard prediction error approach is assumed to be used to define the estimate $\hat{\theta}_N$ of the parameter vector θ for the model (1) based on the data set $\{u(t), y(t)\}_{t=1}^N$. It is based on minimizing the prediction error

$$\varepsilon(t, \theta) = y(t) - \hat{y}(t, \theta), \quad (5)$$

the difference between the measured output $y(t)$ and the prediction with (1),

$$\hat{y}(t, \theta) = G_2(q^{-1}, \gamma) f(\beta, G_1(q^{-1}, \alpha) u(t)). \quad (6)$$

This is done by using a criterion of fit

$$V_N(\theta) = \frac{1}{N} \sum_{t=1}^N \varepsilon^2(t, \theta) \quad (7)$$

and then defining the estimate as

$$\hat{\theta}_N = \arg \min_{\theta} V_N(\theta). \quad (8)$$

After that the estimate (8) has been defined, it remains to compute it. This must be done using a gradient based iterative algorithm since the model is not a linear regression model. That is, given a start value $\theta^{(0)}$, iterate

$$\theta^{(i+1)} = \theta^{(i)} - R_i \frac{dV_N(\theta)}{d\theta} \quad (9)$$

until convergence. The matrix R_i is to modify the search direction and step size to assure downhill steps. Depending on how R_i is chosen, (9) describes a wide class of well-known standard algorithms like Gauss-Newton and Levenberg-Marquardt algorithms.

All three blocks of the model structure (1) contain a gain parameter, and two of them are typically fixed in the iterative minimization, eg, b_0^1 and b_0^2 .

In the example in Section 6 a Levenberg-Marquardt algorithm is used to compute the minimization. A software package for the *Mathematica* platform, Sjöberg and Hjalmarsson (2009), has been used which has the advantage

that when the model structure (1) has been defined, all expressions needed in (9) to apply the Levenberg-Marquardt algorithm are calculated automatically. The symbolic features of *Mathematica* are used to automatically generate expressions for the derivatives. The symbolic expressions are also simplified so that, eg, multiple identical expressions are only calculated once.

Typically, $V_N(\theta)$ can have many minima and the success of the minimization depends on the initial estimate $\theta^{(0)}$. The contribution of this work is a novel algorithm to compute this initial estimate.

3. PROPOSED ALGORITHM

The algorithm consists of the following steps.

1. Start with the best linear model $G(q^{-1})$ of the plant.
2. Split the linear model into all possible $G_1(q^{-1})$ and $G_2(q^{-1})$ so that $G(q^{-1}) = G_1(q^{-1})G_2(q^{-1})$.

To do this, the poles and the zeros of the linear model need to be calculated. These are then divided in all possible ways into two sub-models G_1 and G_2 . Depending on prior knowledge of the system, some of the divisions can be excluded. This is discussed in Section 5.

3. For all partitions of the linear model, $\{G_1, G_2\}$, use $u(t)$ and G_1 to decide values for β^2 and then LS to fit the linear parameters, β^1 , in the nonlinearity as initialization.

The position parameters β^2 for the basis functions are decided using the distribution of the input to the nonlinearity $\{z(t) = G_1(q^{-1})u(t)\}_{t=1}^N$.

Minimizing (7) with respect to the parameters β^1 in (4) is straightforward by first writing (6), $\hat{y}(t, \theta)$,

$$\hat{y}(t, \theta) = \sum_{k=1}^n \beta_k^1 G_2(q^{-1}, \gamma) f_k(\beta_k^2, z(t)) = \beta^{1T} \varphi(t) \quad (10)$$

where

$$\varphi^T(t) = [G_2(q^{-1}, \gamma) f_0(\beta_0^2, z(t)), \dots, G_2(q^{-1}, \gamma) f_n(\beta_n^2, z(t))]. \quad (11)$$

Since (10) is a linear regression, the LS estimate is given by

$$\hat{\beta}^1 = \left(\frac{1}{N} \sum_{t=1}^N \varphi(t) \varphi^T(t) \right)^{-1} \frac{1}{N} \sum_{t=1}^N \varphi^T(t) y(t). \quad (12)$$

4. Order the initialized models with respect to their initial fit.

The criterion $V_N(\theta)$, (7) is calculated for all initializations and the models are ranked using this measure.

5. Fit all parameters of the best, or some of the best models.

This means that the minimization algorithm (9) is applied and it is actually not part of the initialization, but the step after the initialization.

4. CONSISTENCY OF THE ALGORITHM

If the system generating the data is within the model structure described by the W-H model (1), and with

some assumption on the input signal, then the consistency of proposed algorithm when the number of data goes to infinity follows almost immediate. To show this the following lemma is used.

Lemma 1. Suppose the input data $u(t)$ is a stationary normal distributed sequence and the output $y(t)$ is obtained by filtering $u(t)$ through a system of form (1) with linear parts G_1^0 and G_2^0 being stable, single input, single output finite order transfer functions, and the nonlinear part f^0 is a continuous function $\mathfrak{R} \rightarrow \mathfrak{R}$. Then best linear approximation (BLA), converge to

$$\kappa G_1^0(q^{-1}) G_2^0(q^{-1}) \quad (13)$$

where κ is a constant which value depends on $u(t)$, f^0 , G_1^0 and G_2^0 .

Proof: See Pintelon and Schoukens (2001).

The essence of this lemma is that the BLA captures the dynamics of the two linear parts and the nonlinear function is approximated with a constant. The miss-match between the true nonlinear function and the constant is captured as noise. This is further described in Schoukens et al. (1998); Enqvist and Ljung (2005); Schoukens et al. (2005).

Given this lemma, it follows that, asymptotically in N , one of the partitioning of the linear models will have the correct dynamics, ie, $\kappa_1 G_1^0$ in the first linear part and $\kappa_2 G_2^0$ in the second one, where $\kappa_1 \kappa_2 = \kappa$. It remains to show consistency in the estimate of the nonlinear function. This is done using a polynomial expansion and Weierstrass's Approximation Theorem for polynomials. The convergence is over an arbitrary bounded interval.

Theorem 1. Suppose the data is generated in the same way as in Lemma 1 with the additional assumption that $G_1^0 G_2^0$ does not contain any zero-pole cancellations.

Assume the algorithm in Section 3 is applied and the nonlinearity is modeled by a polynomial expansion f_n of degree n combined with a saturation to limit the output for large positive and negative values outside a region which grows with the number of data. Then, on any interval $[z_a, z_b]$ for the nonlinear part of the model, the best initialized Wiener-Hammerstein model converges to the true data generating system when $N, n \rightarrow \infty$ under the constraint $n/N \rightarrow 0$.

Proof: See Sjöberg and Schoukens (2011).

The theorem states that the proposed algorithm is sound and it gives the true system description asymptotically. In practical situations the number of data is limited and the theorem motivates the use of the algorithm to obtain the initial parameter estimate before the iterative minimization. There are a number of comments one can make on the theorem.

- The properties of $u(t)$ influence the convergence speed when $N \rightarrow \infty$. Generally, an input signal which excites the system more will accelerate the convergence.
- The consistency is proven for a general continuous nonlinear function but only on an arbitrary chosen interval. A larger interval will typically make the convergence slower in n and then also in N due to the requirement $n/N \rightarrow 0$.

- Instead of a polynomial model for the non-linearity, any basis function expansion can be used.

5. NUMBER OF LINEAR PARTITIONS

An obvious possible disadvantage of the algorithm is that the original linear model can be partitioned in many different ways which leads to many least-squares problems to solve when all W-H models are being initialized. In this section it will be shown that for moderate model orders, this number does not need to be very high.

With m real poles 2^m different splittings into two linear models are possible. If the poles are complex the number of possibilities reduce to $2^{m/2}$ (assuming m to be even). Assuming also m zeros gives equal many possibilities of ways to split them. This gives the following result.

Result 1. A linear system of order m can be partitioned into between 2^m and 2^{2m} different pairs of linear systems. The lower limit hold if all poles and zeros are complex and the upper limit when they are all real.

For a 10th order system this means that there will be between 1024 and $1024^2 > 10^6$ possible splittings of the linear system. The lower limit is feasible for the proposed algorithm but the upper limit would typically be too many possibilities to investigate.

The number of partitions can, however be reduced with prior knowledge or assumptions. In many situations it could make sense to require the two linear parts to be proper. Dividing m poles into two models with k and $m-k$ poles can be done in

$$\frac{m(m-1)\dots(m-k+1)}{1\cdot 2\cdot \dots\cdot k} = \binom{m}{k} \quad (14)$$

ways. For a given k , the number of possible divisions of the zeros are equally many. The total number of proper divisions is given by summing (14) over all k ,

$$\sum_{k=0}^m 2 \binom{m}{k} = 2^{m+1}$$

where the factor 2 comes from the zeros. Hence, we have the following result.

Result 2. A linear system of order m can be partitioned into between $2^{m/2+1}$ and 2^{m+1} different pairs of proper linear systems. The lower limit holds if all poles and zeros are complex and the upper limit when they are all real.

This means that for an original linear plant of order 10 the number of splits becomes between $2^{10/2+1} = 64$ and $2^{11} = 2048$. Note that the two special cases where all the dynamics is either placed in the first or in the second linear sub-system corresponds to either a Wiener model or a Hammerstein model, respectively. Tests of these model structures are hence treated as special cases if the W-H model selection.

Another type of possible assumption could be the order of the two linear sub-system. The number of possible splits is then given directly by (14) and we form this as a result.

Result 3. A linear system of order m can be partitioned into two proper linear parts, one of order k and the other of order $m-k$, in between $2^{\binom{m}{2}}$ and $2^{\binom{m}{k}}$ different ways.

The lower limit is for the case that all poles and zeros are complex and the upper limit is for the case that they are all real.

For the example system of order 10, and a split into a 4th and a 6th order systems this means between $2^{\binom{5}{2}} = 2 \cdot 5!/(2!3!) = 20$ and $2^{\binom{10}{4}} = 2 \cdot 10!/(4!6!) = 420$ different possible splits. The number of possibilities is slightly higher if the 10th order system is to be split into two 5th order systems. However, this can only be done if at least two of the poles and two of the zeros are real. Hence the lower number of possible splits becomes $2 \cdot 2^{\binom{4}{2}} = 24$, where the extra factor 2 is due to the two possibilities how to divide the real poles and zeros. Similarly, the upper limits of possible splits becomes $2^{\binom{10}{5}} = 2 \cdot 10!/(5!5!) = 504$.

Since linear regression problems can be solved very fast, one concludes from the results in this section that the computational burden is reasonable if the linear model stays within limits.

6. EXAMPLE

In this example the proposed algorithm is tested on data generated by a W-H model. The main message is to illustrate that the algorithm has a good chance to avoid bad local minima. The software described in Sjöberg and Hjalmarsson (2009) is used for implementing the example.

A white Gaussian signal with standard deviation 15 is used as input signal to a system with the W-H structure. Figure 2 depicts the poles and zeros of the two linear parts and Figure 3 a) shows the nonlinearity which is between them. The exact mathematical definition of the system is given in Appendix A.

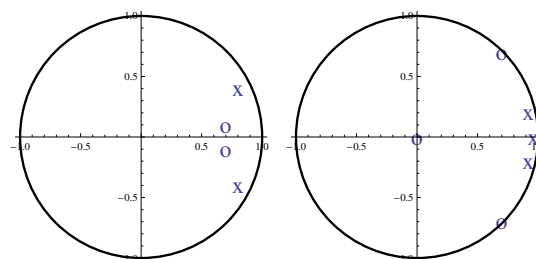


Fig. 2. Poles (x) and zeros (o) of the two linear parts of the true system.

With this true system, 2000 data samples were generated where the output is corrupted with white Gaussian noise with standard deviation 0.1. This gives a signal-to-noise amplitude ratio of 57.

The first step of the algorithm is to estimate a linear model. In the example the search for the best linear model is skipped and the prior knowledge that it should be of 5th order is used. See any standard literature on system identification for strategies to obtain the best linear model, eg Pintelon and Schoukens (2001); Ljung (1999); Söderström and Stoica (1989). In Figure 4 the simulation with the 5th order linear model is shown together with the true output. The poles and zeros of the linear model are shown in Figure 3 b). Two zeros are clearly incorrect compared to

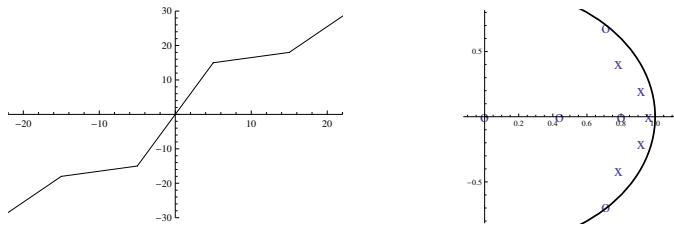


Fig. 3. a) True nonlinearity. b) Poles (x) and zeros (o) of the linear model.

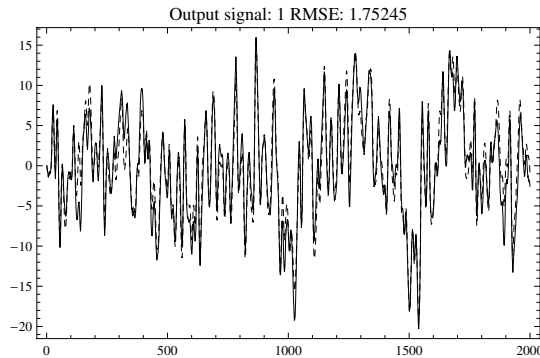


Fig. 4. Output signal (solid) together with the simulated output of the linear model (dashed).

the true positions shown in Figure 2 but the rest of poles and zeros seem to be reasonably well estimated.

The linear model is now partitioned into two linear parts which should be at least first order systems. That means that the prior information of the degree of the two subsystems is not used. This gives 22 partitions and all of these are extended with a first order spline with 8 knots, ie, a local linear function with 9 segments. The positions of the knots are distributed so that each segment contains the same number of data.

Given the division of the linear model and the positions of the knots, the nonlinearity can be initialized according to (12). The models are sorted according to their fit at the initialization and the result is shown in Figure 5 where the bullets show the RMS after initialization. The best initialization has a RMS fit of 1.05 compared to 1.75 for the linear model. The initialization of all 22 models took 35 CPU seconds on an average PC of the time of this paper. The best initialization outperforms the other. In the same

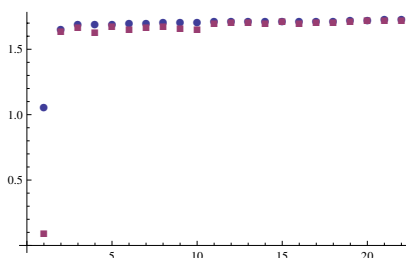


Fig. 5. Bullets: The RMS error of the 22 initialized W-H models. Squares: The obtained RMS after fitting all parameters of the 22 models. The first model is clearly the best initialized model, and it is also the one which improves considerably when the parameters are fitted.

figure the fit after that all parameters have been fitted, is

also shown with squares. The best initialized model is also the one which gives the best fit after that the criterion has been minimized with respect to all parameters. It has an RMS fit of 0.095 which corresponds to the noise level.

In Figure 6 the estimated nonlinearity at initialization and for the final model are shown together with the true nonlinearity.

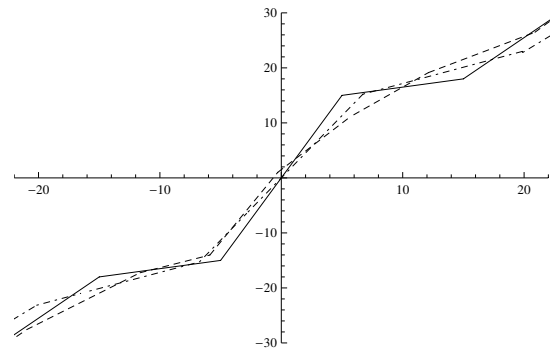


Fig. 6. Solid, true nonlinearity, dashed estimate at the initialization, dashed-dotted after estimating all parameters.

Since poles and zeros cannot move from one linear part to the other, it is clear that the most important issue is to have the right number of poles and zero in each one of them. For this example, out of the 22 partitions, 8 of these have the correct number of poles and zeros in each linear part. That is, all of these could converge to the best model if they would not get caught in any local minima. Figure 7 shows the criterion decrease during the iterative minimization for these 8 models.

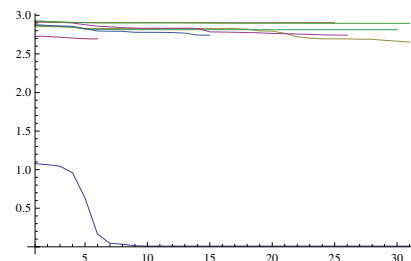


Fig. 7. Criterion decrease for the 8 initializations with correct number of zeros and poles in the linear parts. The ones with incorrect positions of poles and/or zeros either converge very slow or they are caught in bad local minima.

Clearly, the proposed initialization gives a good initialization so that only 10 iterations are needed in the recursive minimization. Some of the other initializations might lead to a good minimum if many more iterations would be applied, but after 30 iterations only a few of them have not yet terminated due to convergence to a local minima, and their decrease is very slow.

7. CONCLUSIONS

A novel algorithm for initializing Wiener-Hammerstein models has been proposed. It starts with a best linear model which is partitioned in all possible ways into two

linear sub-models. Using least squares the nonlinearity between the two linear parts can efficiently be initialized. The initialized models can be ranked, and the one with the best fit is the one with best chances to converge to the global minimum when all parameters are estimated simultaneously.

Consistency of the algorithm has been stated. Finally the algorithm has been illustrated with an example which shows that it can really make a difference.

Appendix A. TRUE WIENER-HAMMERSTEIN SYSTEM

The data used in the example in Section 6 where generated by the Wiener-Hammerstein system with of the following form of the two linear parts

$$G_1(q^{-1}) = \frac{1 - 1.4q^{-1} + 0.5q^{-2}}{1 - 1.6q^{-1} + 0.8q^{-2}} \quad (\text{A.1})$$

and

$$G_2(q^{-1}) = \frac{0.01 - 0.014q^{-1} + 0.0098q^{-2}}{1 - 2.8q^{-1} + 2.6528q^{-2} - 0.850944q^{-3}} q^{-1}. \quad (\text{A.2})$$

The poles and zeros of these transfer functions are depicted in Figure 2. The nonlinearity, depicted in Figure 3 a), is defined as

$$f(z) = \begin{cases} 1.5z + 4.5 & z < -15 \\ 0.3z - 13.5 & -15 < z < -5 \\ 3z & -5 < z < 5 \\ 0.3z + 13.5 & 5 < z < 15 \\ 1.5z - 4.5 & 15 < z \end{cases} \quad (\text{A.3})$$

REFERENCES

- Bershad, N., Celka, P., and McLaughlin, S. (2001). Analysis of stochastic gradient identification of Wiener-Hammerstein systems for nonlinearities with hermite polynomial expansions. *IEEE Trans. Signal Process*, 49, 1060–1071.
- Crama, P. and Schoukens, J. (2005). Computing an initial estimate of a Wiener-Hammerstein system with a random phase multisine excitation. *IEEE Trans. Instrum. Meas*, 54, 117–122.
- Enqvist, M. and Ljung, L. (2005). Linear approximations of nonlinear FIR systems for separable input processes. *Automatica*, 41(3), 459 – 473.
- Lauwers, L., Pintelon, R., and Schoukens, J. (2009). Modelling of Wiener-Hammerstein systems via the Best Linear Approximation. In *Preprint, 15th IFAC Symposium on System Identification Saint-Malo, France*, 1098–1103.
- Leith, D., Leithhead, W., and Murray-Smith, R. (2003). Nonlinear structure identification with application to Wiener-Hammerstein systems. In *Proc. 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands*, 1317–1320.
- Ljung, L. (1978). Convergence analysis of parametric identification methods. *IEEE Transactions on Automatic Control*.
- Ljung, L. (1999). *System Identification: Theory for the User*. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition.
- Pintelon, R. and Schoukens, J. (2001). *System Identification: A Frequency Domain Approach*. IEEE-press, Piscataway.
- Schoukens, J., Dobrowiecki, T., and Pintelon, R. (1998). Parametric and nonparametric identification of linear systems in the presence of nonlinear distortions - a frequency domain approach. *IEEE Transactions on Automatic Control*, 43(2), 176–190.
- Schoukens, J., Pintelon, R., Dobrowiecki, T., and Rolain, Y. (2005). Identification of linear systems with nonlinear distortions. *Automatica*, 41, 451–504.
- Schoukens, J., Pintelon, R., Paduart, J., and Vandersteen, G. (2006). Nonparametric initial estimates for Wiener-Hammerstein systems. In *Preprint, 14th IFAC Symposium on System Identification, Newcastle, Australia*, 778–783.
- Sjöberg, J. and Hjalmarsson, H. (2009). A system, signal, and identification toolbox in mathematica with symbolic capabilities. In *Preprint, 15th IFAC Symposium on System Identification Saint-Malo, France*, 747–751.
- Sjöberg, J. and Schoukens, J. (2011). Initializing Wiener-Hammerstein models based on partitioning of the best linear approximation. Submitted to *Automatica*.
- Söderström, T. and Stoica, P. (1989). *System Identification*. Prentice-Hall International, Hemel Hempstead, Hertfordshire.
- Tan, A. and Godfrey, K. (2002). Identification of Wiener-Hammerstein models using linear interpolation in the frequency domain (LIFRED). *IEEE Transactions on Instrumentation and Measurement*, 51(3), 509–521.
- Tan, A. and Godfrey, K. (2003). Identification of Wiener-Hammerstein models with cubic nonlinearity using LIFRED. In *Proc. 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands*, 1339–1344.
- Wills, A. and Ninness, B. (2009). Estimation of generalised Wiener-Hammerstein systems. In *Preprint, 15th IFAC Symposium on System Identification, Saint-Malo, France*, 1104–1109.