Subspace Identification of Hammerstein Systems using Least Squares Support Vector Machines

Ivan Goethals, Kristiaan Pelckmans, Johan A.K. Suykens, Bart De Moor

K.U. Leuven, ESAT-SCD-SISTA

Kasteelpark Arenberg 10

B-3001 Leuven (Heverlee), Belgium

Tel: 32/16/32 86 58 - Fax: 32/16/32 19 70

ivan.goethals@esat.kuleuven.ac.be

kristiaan.pelckmans@esat.kuleuven.ac.be

johan.suykens@esat.kuleuven.ac.be

bart.demoor@esat.kuleuven.ac.be

Abstract

This paper presents a method for the identification of multi-input/multi-output (MIMO) Hammerstein systems for the goal of prediction. The method extends the N4SID (Numerical algorithms for Subspace State Space System IDentification) linear subspace identification algorithm, mainly by rewriting the oblique projection in the N4SID algorithm as a set of componentwise Least Squares Support Vector Machines regression problems. The linear model and static nonlinearities follow from a low rank approximation of a matrix obtained from this regression problem.

Keywords: Subspace identification, Hammerstein models, Least Squares Support Vector Machines.

1 Introduction

Throughout the last few decades, the field of linear modeling has been explored to the level that most linear identification problems can be solved efficiently with fairly standard and well known tools. Extensions to nonlinear systems are often desirable but in general much harder from a practical as well as a theoretical perspective. In many situations Hammerstein systems are seen to provide a good trade-off between the complexity of general non-linear systems and interpretability of linear dynamical systems (see e.g. [1]). They have been used e.g. for modeling biological processes [13, 32], chemical processes [7], and in signal processing applications [21]. Hammerstein models have also been shown to be useful for control problems (as e.g. in [14]).

Identification of Hammerstein systems has been explored from different perspectives. Published approaches mainly differ in the way the static nonlinearity is represented and in the type of optimization problem that is finally obtained. Known approaches include the expansion of the nonlinearity as a sum of (orthogonal or non-orthogonal) basis functions [16, 17, 15], the use of a finite number of cubic spline functions as presented in [6], piecewise linear functions [28] and neural networks [12]. Regardless of the parameterization scheme that is chosen, the final cost function will involve cross-products between parameters describing the static nonlinearity and those describing the linear dynamical system. Employing a maximum likelihood criterion results in a non-convex optimization problem where global convergence is not guaranteed [20]. Hence, in order to find a good optimum for these techniques, a proper initialization is often necessary [5].

Different approaches were proposed in the literature to overcome this difficulty. These result in convex methods which generate models of the same, or almost the same quality as their nonconvex counterparts. Unfortunately, convexity is either obtained by placing heavy restrictions on the input sequence (e.g. whiteness) and the nonlinearity under consideration [2] or by using a technique known as overparameterization [3, 1]. In the latter, one replaces

every cross-product of unknowns by new independent parameters resulting in a convex but overparameterized method. In a second stage the obtained solution is projected onto the Hammerstein model class using a singular value decomposition. A classical problem with the overparameterization approach is the increased variance of the estimates due to the increased number of unknowns in the first stage.

In [8, 9] it was seen that by combining ideas from the overparameterization approach with concepts of Least Squares Support Vector Machines (LS-SVMs), a Hammerstein ARX (AutoRegressive with eXogeneous inputs) identification algorithm was obtained which outperforms existing overparameterization approaches, mostly due to the effect of regularization. LS-SVMs [24, 23] are reformulations to the standard Support Vector Machines (SVM). SVM's [29, 11, 19] and related methods constitute a powerful methodology for solving problems in linear and nonlinear classification, function approximation and density estimation and also stimulated new results in kernel based methods in general. They have been introduced on the interplay between learning theory, statistics, machine learning, neural networks and optimization theory.

A drawback with the method introduced in [8] is that the ARX model class is a rather restricted model class and is for instance not suitable to describe systems involving output noise. To this extent, identification algorithms based on state-space models are in many cases preferable. In this paper we study the extension of the linear N4SID subspace identification algorithm to Hammerstein systems. It will be shown that by using the concept of componentwise LS-SVM regression, the state reconstruction step in classical identification algorithms can readily be extended to Hammerstein systems. The linear system and static nonlinearity are recovered in a second step.

The outline of this paper is as follows: In Section 2 the N4SID subspace algorithm for linear systems is reviewed briefly. Section 3 extends the N4SID algorithm towards a non-linear setting using a variation on the theme of LS-SVMs. Section 4 presents some illustrative examples for SISO and MIMO systems and relates the presented algorithm to

existing approaches. A brief introduction into LS-SVM regression and component-wise LS-SVM regression is provided in appendices A and B.

As a general rule in this paper, lowercase symbols will be used to denote column vectors. Uppercase symbols are used for matrices. Elements of matrices and vectors are selected using Matlab standards, e.g. $A(i,j)$ denotes the ij^{th} entry of a matrix A, and $A(:,i)$ symbolizes the i^{th} column of the same matrix. Estimates for a parameter x will be denoted by $\hat{x}$. The symbol $\triangleq$ is used for definitions.

2 The N4SID algorithm for linear subspace identification

The subspace algorithm considered in this paper is the so-called N4SID algorithm, which is part of the set of combined deterministic-stochastic subspace algorithms as presented in [26, 27]. We consider systems of the form

$$\left\{ \begin{array}{rcl} x_{t+1} & = & Ax_t + Bu_t + \nu_t, \\ y_t & = & Cx_t + Du_t + v_t, \end{array} \right. \tag{1}$$

with $u_t \in \mathbb{R}^m$ and $y_t \in \mathbb{R}^l$ the input and output at time t, $x_t \in \mathbb{R}^n$ the state, and $\nu_t \in \mathbb{R}^n$ and $v_t \in \mathbb{R}^l$ zero mean white Gaussian noise vector sequences with covariance matrix:

$$E\left\{ \begin{bmatrix} \nu_p \\ v_p \end{bmatrix} \begin{bmatrix} \nu_q^T & v_q^T \end{bmatrix} \right\} = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \delta_{pq}.$$

Given observed sequences $\{(u_t, y_t)\}_{t=0}^{N}$, N4SID identification algorithms are concerned with finding an estimate for the model order n of the system (1), estimates for the matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{l\times n}$, $D \in \mathbb{R}^{l\times m}$ up to a similarity transformation, and the noise covariance matrices $Q \in \mathbb{R}^{n\times n}$, $S \in \mathbb{R}^{n\times l}$, and $R \in \mathbb{R}^{l\times l}$.

Block Hankel matrices play an important role in these algorithms. The input block

Hankel matrices are defined as:

$$U_{0|2i-1} \triangleq \begin{bmatrix} u_0 & u_1 & u_2 & \dots & u_{j-1} \\ u_1 & u_2 & u_3 & \dots & u_j \\ \vdots & \vdots & \vdots & & \vdots \\ u_{i-1} & u_i & u_{i+1} & \dots & u_{i+j-2} \\ u_i & u_{i+1} & u_{i+2} & \dots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & u_{i+3} & \dots & u_{i+j} \\ \vdots & \vdots & \vdots & & \vdots \\ u_{2i-1} & u_{2i} & u_{2i+1} & \dots & u_{2i+j-2} \end{bmatrix} \in \mathbb{R}^{(2i)m \times j}$$

$$\triangleq \begin{bmatrix} U_{0|i-1} \\ U_{i|2i-1} \end{bmatrix} \triangleq \begin{bmatrix} U_p \\ U_f \end{bmatrix} \triangleq \begin{bmatrix} U_{0|i} \\ U_{i+1|2i-1} \end{bmatrix} \triangleq \begin{bmatrix} U_p^+ \\ U_f^- \end{bmatrix},$$

with i and j user defined indices such that $2i + j - 1 = N$. The output block Hankel matrices $Y_p, Y_f \in \mathbb{R}^{il \times j}$, $Y_p^+ \in \mathbb{R}^{(i+1)l \times j}$ and $Y_f^- \in \mathbb{R}^{(i-1)l \times j}$ are defined in a similar way. Finally,

$$W_p \triangleq \begin{bmatrix} U_p \\ Y_p \end{bmatrix}, \quad W_p^+ \triangleq \begin{bmatrix} U_p^+ \\ Y_p^+ \end{bmatrix},$$

are introduced as the past input/output block Hankel matrices and

$$\Gamma_i \triangleq \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{i-1} \end{bmatrix},$$

as the so-called extended observability matrix of order i. Defining $A^*/_{B^*}C^*$ as the oblique projection of the row space of a matrix A^* into the row space of a matrix C^* along the row space of a matrix B^*: $A^*/_{B^*}C^* = L_{C^*}C^*$ whereby

$$A^* = L_{B^*}B^* + L_{C^*}C^* + L_{B^{*\perp},C^{*\perp}} \begin{bmatrix} B^* \\ C^* \end{bmatrix}^{\perp},$$

the main reasoning behind N4SID subspace algorithms follows from the fact that under the assumptions that:

1. the process noise ν_t and measurement noise v_t are uncorrelated with the input u_t,
2. the input u_t is persistently exciting of order $2i$, i.e. the input block Hankel matrix $U_{0|2i-1}$ is of full rank,
3. the sample size goes to infinity: $j \to \infty$,
4. the process noise ν_t and the measurement noise are not identically zero,

the following relation holds:

$$Y_f/_{U_f} W_p = \Gamma_i \widetilde{X}_i,$$

with $Y_f/_{U_f} W_p$ the so-called oblique projection of the future outputs Y_f onto the past data W_p along the future inputs U_f, which can be written explicitly as [27]

$$Y_f/_{U_f} W_p = (Y_f - Y_f U_f^\dagger U_f)(W_p - W_p U_f^\dagger U f)^\dagger W_p$$

where $\widetilde{X}_i$ can be shown to correspond to an estimate for the state in (1), resulting from a bank of non-steady state Kalman filters [26]. Hence, the order of the system and a realization of the state can be obtained from a singular value decomposition of $\mathcal{O}_i$. Once the state is known, extraction of A, B, C and D is straightforward. Without going into further theoretical details of the N4SID algorithm (interested readers are referred to [25, 26, 27]), we summarize here a practical N4SID algorithm that will be used towards the Hammerstein model extension:

1. Calculate the oblique projections of the future outputs along the future inputs onto the past:

$$\begin{cases} \mathcal{O}_i &= Y_f/_{U_f} W_p, \\ \mathcal{O}_{i+1} &= Y_f^-/_{U_f^-} W_p^+, \end{cases} \tag{2}$$

where $\mathcal{O}_i \in \mathbb{R}^{il\times j}$ and $\mathcal{O}_{i+1} \in \mathbb{R}^{(i+1)l\times j}$. This projection can be implemented using a least squares algorithm as follows:

$$\begin{aligned}(\widehat{L}_u, \widehat{L}_y) &= \underset{L_u, L_y}{\arg\min} \left\| \begin{bmatrix} L_u & L_y \end{bmatrix} \begin{bmatrix} \begin{bmatrix} U_p \\ U_f \end{bmatrix} \\ Y_p \end{bmatrix} - Y_f \right\|_F^2, \\ (\widehat{L}_u^-, \widehat{L}_y^-) &= \underset{L_u^-, L_y^-}{\arg\min} \left\| \begin{bmatrix} L_u^- & L_y^- \end{bmatrix} \begin{bmatrix} \begin{bmatrix} U_p^+ \\ U_f^- \end{bmatrix} \\ Y_p^+ \end{bmatrix} - Y_f^- \right\|_F^2.\end{aligned}$$

Estimates for $\mathcal{O}_i$ and $\mathcal{O}_{i+1}$ are then obtained as follows:

$$\begin{aligned}\mathcal{O}_i &= \widehat{L}_u(:, 1:im)U_p + \widehat{L}_y Y_p, \\ \mathcal{O}_{i+1} &= \widehat{L}_u^-(:, 1:(i+1)m)U_p^+ + \widehat{L}_y^- Y_p^+.\end{aligned}$$

2. Calculate the SVD of the oblique projection $\mathcal{O}_i$, determine the order by inspecting the singular values and partition the SVD accordingly to obtain U_1 and S_1:

$$\mathcal{O}_i = USV^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$$

3. Determine the extended observability matrices Γ_i and Γ_{i-1} from:

$$\Gamma_i = U_1 S_1^{1/2}, \quad \Gamma_{i-1} = \Gamma_i(1 : l(i-1), :). \tag{3}$$

4. Determine estimates for the state sequences from the equations

$$\begin{cases} \widetilde{X}_i &= \Gamma_i^\dagger \mathcal{O}_i, \\ \widetilde{X}_{i+1} &= \Gamma_{i-1}^\dagger \mathcal{O}_{i+1}. \end{cases}$$

5. Extract estimates for A, B, C and D from:

$$\begin{bmatrix} \widetilde{X}_{i+1} \\ Y_{i|i} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \widetilde{X}_i \\ U_{i|i} \end{bmatrix} + \begin{bmatrix} \rho_\nu \\ \rho_v \end{bmatrix}.$$

by minimizing the least-squares residuals ρ_ν and ρ_v.

6. Determine estimates for Q, S and R from

$$\begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} = \frac{1}{j} \begin{bmatrix} \rho_\nu \\ \rho_v \end{bmatrix} \begin{bmatrix} \rho_\nu^T & \rho_v^T \end{bmatrix}.$$

The extension of this approach towards the identification of a Hammerstein system mainly concentrates on steps 1 and 5 where one uses the technique of componentwise LS-SVMs [18] instead.

3 Extending the N4SID algorithm towards identification of Hammerstein models

In this section, the linear N4SID algorithm will be extended to the identification of Hammerstein systems making use of the concept of overparameterization in an LS-SVM framework. Equation (1) is transformed into a Hammerstein system by introducing a static nonlinearity $f : \mathbb{R}^m \to \mathbb{R}^m$ which is applied to the inputs:

$$\begin{cases} x_{t+1} = Ax_t + Bf(u_t) + \nu_t, & \forall t = 0, \ldots, N-1 \\ y_t = Cx_t + Df(u_t) + v_t, & \forall t = 0, \ldots, N-1. \end{cases} \tag{4}$$

Inputs and outputs $\{(u_t, y_t)\}_{t=0}^{N-1}$, are assumed to be available. The sequences of process and measurement noise $\{\nu_t\}_{t=0}^{N-1}$ and $\{v_t\}_{t=0}^{N-1}$ follow the same statistics as outlined in Section 2. We define the matrix operator Φ as an operator on a block Hankel matrix and a nonlinear function ρ on $\mathbb{R}^m$ which applies $\rho(\cdot)$ to every block matrix Z_i in Z and stacks the results in the original Hankel configuration:

$$\Phi_{\mathcal{F}} \left(\begin{bmatrix} Z_1 & Z_2 & \ldots & Z_p \\ Z_2 & Z_3 & \ldots & Z_{p+1} \\ \vdots & \vdots & & \vdots \\ Z_q & Z_q+1 & \ldots & Z_{p+q-1} \end{bmatrix} \right) = \begin{bmatrix} \mathcal{F}(Z_1) & \mathcal{F}(Z_2) & \ldots & \mathcal{F}(Z_p) \\ \mathcal{F}(Z_2) & \mathcal{F}(Z_3) & \ldots & \mathcal{F}(Z_{p+1}) \\ \vdots & \vdots & & \vdots \\ \mathcal{F}(Z_q) & \mathcal{F}(Z_q+1) & \ldots & \mathcal{F}(Z_{p+q-1}) \end{bmatrix} \in \mathbb{R}^{qm \times p}.$$

3.1 Overparameterization for the oblique projection $\mathcal{O}_i$

The oblique projection $\mathcal{O}_i = Y_f/_{U_f} W_p$ can be calculated from estimates for L_u, L_y and f obtained by minimizing the residuals E of the following equation [27]:

$$Y_f = \begin{bmatrix} L_u & L_y \end{bmatrix} \begin{bmatrix} \Phi_f(U_{0|2i-1}) \\ Y_p \end{bmatrix} + E, \tag{5}$$

in a least-squares sense. This can be rewritten as

$$Y_f(s,t) = L_y(s,:)Y_p(:,t) + \sum_{h=1}^{2i} L_u(s,(h-1)m+1:hm) f(u_{h+t-2}) + E(s,t), \tag{6}$$

for $s = 1,\ldots,il$ and $t = 1,\ldots,j$. Once estimates for $\widehat{L}_u$, $\widehat{L}_y$ and $\hat{f}$ occuring in equations (5) and (6) are obtained, the oblique projection is calculated as:

$$\mathcal{O}_i(s,t) = \widehat{L}_y(s,:)Y_p(:,t) + \sum_{h=1}^{i} \widehat{L}_u(s,(h-1)m+1:hm) \hat{f}(u_{h+t-2}), \tag{7}$$

for $s = 1,\ldots,il$ and $t = 1,\ldots,j$. Note that in (6) and (7), products between parameter matrices L_u and L_y and the static non-linearity f appear which are hard to incorporate in an optimization problem. In order to deal with the resulting non-convexity, we apply the concept of overparameterization (see Appendix B) by introducing a set of functions $g_{h,s} : \mathbb{R}^m \to \mathbb{R}$ such that [1]:

$$g_{h,s} \triangleq c_{h,s}^T f, \quad \text{s.t.} \quad c_{h,s}^T = L_u(s,(h-1)m+1:hm), \ \forall h = 1,\ldots,2i, \ s = 1,\ldots,il. \tag{8}$$

With these new functions we obtain a generalization to (6) and (7):

$$Y_f(s,t) = L_y(s,:)Y_p(:,t) + \sum_{h=1}^{2i} g_{h,s}(u_{h+t-2}) + E(s,t), \tag{9}$$

$$\mathcal{O}_i(s,t) = \widehat{L}_y(s,:)Y_p(:,t) + \sum_{h=1}^{i} \hat{g}_{h,s}(u_{h+t-2}), \tag{10}$$

for $s = 1,\ldots,il$ and $t = 1,\ldots,j$. Note that (9) is now linear in the functions $g_{h,s} : \mathbb{R}^m \to \mathbb{R}$. The central idea behind the algorithm presented in this paper is that $\widehat{L}_y$ and estimates for

the functions $g_{h,s}$ in (9-10) can be determined from data using the concept of componentwise LS-SVM regression as presented in Appendix B.

Let the kernel function be defined as $K : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ such that $K(u_p, u_q) = \varphi(u_p)^T \varphi_k(u_q)$ for all $p, q = 0, \ldots, N-1$ and the kernel matrix $\Omega \in \mathbb{R}^{N \times N}$ such that $\Omega(i,j) = K(u_{i-1}, u_{j-1})$ for all $i = 1, \ldots, N,\ j = 1, \ldots, N$. Substituting $g_{h,s}$ for the primal model $w_{h,s}^T \varphi$ (see Appendix B) in (9) results in

$$\begin{aligned} Y_f(s,t) &= L_y(s,:)Y_p(:,t) + \sum_{h=1}^{2i} w_{h,s}^T \varphi(u_{h+t-2}) + E(s,t), \\ &\quad \forall s = 1, \ldots, li,\ \ t = 1, \ldots, j. \end{aligned} \tag{11}$$

As argued in Appendix B, the expansion of a non-linear function as the sum of a set of non-linear functions is not unique, e.g.

$$\left(w_1^T \varphi_1(u)\right) + \left(w_2^T \varphi_2(u)\right) = \left(w_1^T \varphi_1(u) + \delta\right) + \left(w_2^T \varphi_2(u) - \delta\right),$$

for all $\delta \in \mathbb{R}$. It was seen that this problem can be avoided by including centering constraints of the form

$$\sum_{t=0}^{N-1} f(u_t) = 0. \tag{12}$$

This constraint can always be applied since for any constant δ_u, and any function $\underline{f} : \mathbb{R}^m \to \mathbb{R}^m$ such that $f = \underline{f} + \delta_u$ there exists a state transformation $\xi_t = \Psi(x_t)$ with $\Psi : \mathbb{R}^n \to \mathbb{R}^n$ and a constant δ_y such that (4) is transformed as follows:

$$\begin{cases} \xi_{t+1} = A\xi_t + Bf'(u_t) + \nu_t, \\ y_t - \delta_y = C\xi_t + Df'(u_t) + v_t, \end{cases} \tag{13}$$

with $\xi_t \in \mathbb{R}^n$ and $\delta_y \in \mathbb{R}^l$ defined as

$$\begin{cases} \xi_t &= \Psi(x_t) = x_t - (I - A)^{-1} B \delta_u, \\ \delta_y &= \left(C(I - A)^{-1} B + D\right) \delta_u. \end{cases}$$

Hence, the constraint (12) can be applied provided that a new parameter δ_y is added to the model, transforming (11) into

$$\begin{aligned} Y_f(s,t) + [1_i \otimes \delta_y](s) &= L_y(s,:)(Y_p(:,t) + 1_i \otimes \delta_y) + \sum_{h=1}^{2i} w_{h,s}^T \varphi(u_{h+t-2}) + E(s,t), \\ & \forall s = 1, \ldots, li, \;\; t = 1, \ldots, j, \end{aligned}$$

where $\otimes$ denotes the matrix kronecker product. Through the equality $w_{h,s}^T \varphi = g_{h,s} = c_{h,s}^T f$ for all $h = 1, \ldots, 2i, \quad s = 1, \ldots, il$, the constraint (12) amounts to

$$\sum_{t=0}^{N-1} w_{h,s}^T \varphi(u_t) = 0, \;\; \forall h, s.$$

The LS-SVM primal problem is then formulated as a constrained optimization problem:

$$\min_{w_{h,s}, L_y, E, \delta_y} \mathcal{J}(w_{h,s}, L_y, E, \delta_y) = \frac{1}{2} \sum_{s=1}^{il} \sum_{h=1}^{2i} w_{h,s}^T w_{h,s} + \frac{\gamma}{2} \sum_{s=1}^{il} \sum_{t=1}^{j} E(s,t)^2,$$

$$\text{s.t.} \begin{cases} Y_f(s,t) + [1_i \otimes \delta_y](s) = L_y(s,:)(Y_p(:,t) + 1_i \otimes \delta_y) & (a) \\ \quad + \sum_{h=1}^{2i} w_{h,s}^T \varphi(u_{h+t-2}) + E(s,t), \quad \forall s = 1, \ldots, il, \; t = 1, \ldots, j, & (14) \\ \sum_{t=0}^{N-1} w_{h,s}^T \varphi(u_t) = 0, \qquad \forall h = 1, \ldots, 2i, s = 1, \ldots, li. & (b) \end{cases}$$

Lemma 3.1. *Given the primal problem (14), estimates for L_y and δ_y follow from the dual system:*

$$\left[\begin{array}{c|c|c|c} 0 & 0 & 1^T & 0 \\ \hline 0 & 0 & Y_p & 0 \\ \hline 1 & Y_p^T & \mathcal{K}_p + \mathcal{K}_f + \gamma^{-1} I & \mathcal{S} \\ \hline 0 & 0 & \mathcal{S}^T & \mathcal{T} \end{array}\right] \left[\begin{array}{c} \overline{d} \\ \hline L_y^T \\ \hline \mathcal{A} \\ \hline \mathcal{B} \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 0 \\ \hline Y_f^T \\ \hline 0 \end{array}\right],$$

where $\overline{d} = (1_i \otimes I_l - L_y(1_i \otimes I_l))\, \delta_y$, 1_j is a column vector of length j with elements 1, $\mathcal{T} = I_{2i} \times 1_N^T \Omega^1 1_N$, $\mathcal{S}_q = \sum_{t=1}^{N} \Omega(t,q)$ and

$$\mathcal{A} = \begin{bmatrix} \alpha_{1,1} & \alpha_{2,1} & \ldots & \alpha_{li,1} \\ \alpha_{1,2} & \alpha_{2,2} & \ldots & \alpha_{li,2} \\ \vdots & \vdots & & \vdots \\ \alpha_{1,j} & \alpha_{2,j} & \ldots & \alpha_{li,j} \end{bmatrix}, \mathcal{B} = \begin{bmatrix} \beta_{1,1} & \beta_{1,2} & \ldots & \beta_{1,li} \\ \beta_{2,1} & \beta_{2,2} & \ldots & \beta_{2,li} \\ \vdots & \vdots & & \vdots \\ \beta_{2i,1} & \beta_{2i,2} & \ldots & \beta_{2i,li} \end{bmatrix}, \mathcal{S} = \begin{bmatrix} \mathcal{S}_1 & \mathcal{S}_2 & \ldots & \mathcal{S}_{2i} \\ \mathcal{S}_2 & \mathcal{S}_3 & \ldots & \mathcal{S}_{2i+1} \\ \vdots & \vdots & & \vdots \\ \mathcal{S}_j & \mathcal{S}_{j+1} & \ldots & \mathcal{S}_N \end{bmatrix}.$$

The matrices $\mathcal{K}_p \in \mathbb{R}^{j\times j}$ and $\mathcal{K}_f \in \mathbb{R}^{j\times j}$ have elements:

$$\begin{aligned}\mathcal{K}_p(p,q) &= \sum_{h=1}^{i} K(u_{h+p-2}, u_{h+q-2}),\\ \mathcal{K}_f(p,q) &= \sum_{h=i+1}^{2i} K(u_{h+p-2}, u_{h+q-2}),\end{aligned}$$

for all $p,q = 1,\ldots,j$. Estimates for the $g_{h,s}$ in (9) are given as:

$$\hat{g}_{h,s} : \mathbb{R}^m \to \mathbb{R} : u^* \to \sum_{t=1}^{j} \alpha_{s,t} K(u_{h+t-2}, u^*) + \beta_{h,s} \sum_{t=0}^{N-1} K(u_t, u^*), \ \ \forall h, s. \tag{15}$$

Proof. This directly follows from the Lagrangian

$$\begin{aligned}\mathcal{L}(w,d,L_y,E;\alpha,\beta,d_s) = \mathcal{J}(w,E) - \sum_{h=1}^{2i}\sum_{s=1}^{li} \beta_{h,s}\left\{\sum_{t=0}^{N-1} w_{h,s}^T \varphi(u_t)\right\} - \\ \sum_{s=1}^{li}\sum_{t=1}^{j} \alpha_{s,t}\left\{L_y(s,:)Y_p(:,t) + \sum_{h=1}^{2i} w_{h,s}^T\varphi(u_{h+t-2}) + d_s + E(s,t) - Y_f(s,t)\right\},\end{aligned}$$

with $\overline{d} = \begin{bmatrix} d_1^T & \ldots & d_l^T \end{bmatrix}^T$ by taking the conditions for optimality $\frac{\partial \mathcal{L}}{\partial w_{h,s}} = 0$, $\frac{\partial \mathcal{L}}{\partial L_y(s,:)} = 0$, $\frac{\partial \mathcal{L}}{\partial E(s,t)} = 0$, $\frac{\partial \mathcal{L}}{\partial d_s} = 0$, $\frac{\partial \mathcal{L}}{\partial \alpha_{s,t}} = 0$, $\frac{\partial \mathcal{L}}{\partial \beta_{h,s,k}} = 0$, $\frac{\partial \mathcal{L}}{\partial d_s} = 0$ and after elimination of the primal variables $w_{h,s}$ and E. □

Combining the results from Lemma (3.1) with equation (10), we have

$$\begin{aligned}\mathcal{O}_i &= \sum_{h=1}^{i} \left(\begin{bmatrix} \Phi_{\hat{g}_{h,1}} U_{h|h} \\ \Phi_{\hat{g}_{h,2}} U_{h|h} \\ \vdots \\ \Phi_{\hat{g}_{h,2li}} U_{h|h} \end{bmatrix} \right) + \widehat{L}_y \left(Y_p - 1_{li} 1_{li}^T \otimes \hat{\delta}_y \right) \\ &= \mathcal{A}^T \mathcal{K}_p + \mathcal{B}_p^T \mathcal{S}_p^T + \widehat{L}_y \left(Y_p - (1_i 1_j^T) \otimes \hat{\delta}_y \right),\end{aligned} \tag{16}$$

with $\mathcal{B}_p = \mathcal{B}(1:i,:)$ and $\mathcal{S}_p = \mathcal{S}(:,1:i)$.

3.2 Calculating the oblique projection $\mathcal{O}_{i+1}$

The calculation of $\mathcal{O}_{i+1}$ is entirely equivalent to that of $\mathcal{O}_i$. Without further proof, we state that $\mathcal{O}_{i+1}$ is obtained as:

$$\mathcal{O}_{i+1} = (\mathcal{A}^-)^T(\mathcal{K}_p^+)^T + (\mathcal{B}_p^-)^T(\mathcal{S}_p^-)^T + L_y^-\left(Y_p^+ - 1_{(i+1)}1_j^T \otimes \delta_y\right) \tag{17}$$

with $\mathcal{K}_p^+(p,q) = \sum_{h=1}^{i+1} K(u_{h+p-2}, u_{h+q-2})$ for all $p, q = 1, \ldots, j$, and

$$\begin{aligned} \mathcal{B}_p^- &= \mathcal{B}^-(1:i+1,:), \\ \mathcal{S}_p^- &= \mathcal{S}^-(:,1:i+1). \end{aligned}$$

$\mathcal{A}^-, \mathcal{B}^-$ and L_y^- follow from:

$$\left[\begin{array}{c|c|c|c} 0 & 0 & 1^T & 0 \\ \hline 0 & 0 & Y_p^+ & 0 \\ \hline 1 & (Y_p^+)^T & \mathcal{K}_p + \mathcal{K}_f + \gamma^{-1}I & \mathcal{S} \\ \hline 0 & 0 & \mathcal{S}^T & \mathcal{T} \end{array}\right] \left[\begin{array}{c} \overline{d^-} \\ \hline (L_y^-)^T \\ \hline \mathcal{A}^- \\ \hline \mathcal{B}^- \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 0 \\ \hline (Y_f^-)^T \\ \hline 0 \end{array}\right],$$

with

$$\overline{d^-} = \left((1_{(i-1)} \otimes I_l) - L_y^-(1_{(i+1)} \otimes I_l)\right)\delta_y.$$

3.3 Obtaining estimates for the states

The state sequences $\widetilde{X}_i$ and $\widetilde{X}_{i+1}$ can now be determined from $\mathcal{O}_i$ and $\mathcal{O}_{i+1}$ in line with what is done in the linear case discussed in Section 2. These state sequences will be used in a second step of the algorithm to obtain estimates for the system matrices and the non-linearity f. Note that in the linear case, it is well known that the obtained state sequences $\widetilde{X}_i$ and $\widetilde{X}_{i+1}$ can be considered as the result of a bank of non steady state Kalman filters working in parallel on the columns of the block-Hankel matrix W_p [27]. In the Hammerstein case, and if f were known, this relation would still hold provided that W_p is replaced by $\left[\Phi_f(U_p)^T \quad Y_p^T\right]^T$. However, an estimate $\hat{f}$ for f based on a finite amount of data will in

general be subject to approximation errors [29]. As the classical results for the bank of linear Kalman filters are not applicable if the inputs $\hat{f}(u_t)$ to the linear model are not exact the obtained states $\widetilde{X}_i$ and $\widetilde{X}_{i+1}$ can no longer be seen as the result of a bank of Kalman filters working on $\begin{bmatrix}\Phi_{\hat{f}}(U_p)^T & Y_p^T\end{bmatrix}^T$. Despite the loss of this property, it will be illustrated in the examples that the proposed method outperforms existing Hammerstein approaches such as approaches based on NARX models (Nonlinear AutoRegressive with eXogeneous inputs) and N4SID identification algorithms with an expansion in Hermite polynomials.

3.4 Extraction of the system matrices and the non-linearity f

The linear model and static non-linearity are estimated from:

$$(\widehat{A},\widehat{B},\widehat{C},\widehat{D},\hat{f}) = \underset{A,B,C,D,f}{\arg\min} \left\| \begin{bmatrix} \widetilde{X}_{i+1} \\ Y_{i|i} - \delta_y \end{bmatrix} - \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \widetilde{X}_i \\ \Phi_f\left(U_{i|i}\right) \end{bmatrix} \right\|_F^2 . \tag{18}$$

It will be shown in this subsection that this least-squares problem can again be written as an LS-SVM regression problem. Denoting

$$\mathcal{X}_{i+1} = \begin{bmatrix} \widetilde{X}_{i+1} \\ Y_{i|i} - \delta_y \end{bmatrix}, \ \Theta_{AC} = \begin{bmatrix} A \\ C \end{bmatrix}, \ \Theta_{BD} = \begin{bmatrix} B \\ D \end{bmatrix}, \tag{19}$$

and replacing $\Theta_{BD}(s,:)f$ by $\omega_s^T\varphi$, where again an expansion of a product of scalars and non-linear functions is written as a linear combination of non-linear functions, we have:

$$\mathcal{X}_{i+1} = \Theta_{AC}\widetilde{X}_i + \begin{bmatrix} \omega_1^T \\ \omega_2^T \\ \vdots \\ \omega_{n+l}^T \end{bmatrix} \Phi_\varphi(U_{i|i}) + E,$$

with E the residuals of (18). The resulting LS-SVM primal problem can be written as

$$\min_{\omega_s,E,\Theta_{AC}} \mathcal{J}(\omega,E) = \frac{1}{2}\sum_{s=1}^{n+l}\omega_s^T\omega_s + \frac{\gamma_{BD}}{2}\sum_{s=1}^{n+l}\sum_{t=1}^{j}E(s,t)^2,$$

$$\text{s.t.}\begin{cases} \mathcal{X}_{i+1}(s,t) = \Theta_{AC}(s,:)\tilde{X}_i(:,t) + \omega_s^T\varphi(u_{i+t-1}),\ \forall s=1,\ldots,li,\ t=1,\ldots,j, & (a)\\ \sum_{t=0}^{N-1}\omega_s^T\varphi(u_t) = 0, \qquad\qquad \forall s=1,\ldots,li, & (b)\end{cases}$$

where γ_{BD} denotes a regularization constant which can be different from the γ used in Subsection 3.1.

Lemma 3.2. *Estimates for A and C in Θ_{AC} are obtained from the following dual problem*

$$\left[\begin{array}{c|c|c} 0 & \widetilde{X}_i & 0 \\ \hline \widetilde{X}_i^T & \mathcal{K}_{BD}+\gamma_{BD}^{-1}I & \mathcal{S}_{BD} \\ \hline 0 & \mathcal{S}_{BD}^T & \mathcal{T}_{BD}\end{array}\right]\left[\begin{array}{c} \Theta_{AC}^T \\ \hline \mathcal{A}_{BD} \\ \hline \mathcal{B}_{BD}\end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \mathcal{X}_{i+1}^T \\ \hline 0\end{array}\right], \tag{20}$$

whereby $\omega_s = \sum_{t=1}^{j}\alpha_{s,t}\varphi(u_{i+t-1}) + \beta_s\sum_{t=0}^{N-1}\varphi(u_t)$ for all $s=1,\ldots,n+l$, $\mathcal{K}_{BD}(p,q) = K(u_{i+p-1},u_{i+q-1})$ for all $p,q=1,\ldots,j$, $\mathcal{B}_{BD} = \begin{bmatrix}\beta_1 & \beta_2 & \ldots & \beta_{n+l}\end{bmatrix}$, $\mathcal{T}_{BD} = 1_N^T K 1_N$, and

$$\mathcal{A}_{BD} = \begin{bmatrix}\alpha_{1,1} & \alpha_{2,1} & \ldots & \alpha_{n+l,1}\\ \alpha_{1,2} & \alpha_{2,2} & \ldots & \alpha_{n+l,2}\\ \vdots & \vdots & & \vdots \\ \alpha_{1,j} & \alpha_{2,j} & \ldots & \alpha_{n+l,j}\end{bmatrix}, \mathcal{S}_{BD} = \begin{bmatrix}\mathcal{S}_{i+1}\\ \mathcal{S}_{i+2}\\ \vdots \\ \mathcal{S}_{i+j}\end{bmatrix}.$$

Proof. This follows directly from the Lagrangian

$$\mathcal{L} = \mathcal{J}(\omega,E) - \sum_{s=1}^{n+l}\beta_s\left\{\sum_{t=0}^{N-1}\omega_s\varphi(u_t)\right\} - \sum_{s=1}^{n+l}\alpha_{s,t}\left\{\mathcal{X}_{i+1}(s,t) - \Theta_{AC}(s,:)\widetilde{X}_i(:,t) - \omega_s^T\varphi(u_{i+t-1})\right\},$$

by taking the conditions for optimality $\frac{\partial\mathcal{L}}{\partial\omega_s}=0$, $\frac{\partial\mathcal{L}}{\partial E}=0$, $\frac{\partial\mathcal{L}}{\partial\Theta_{AC}}=0$, $\frac{\partial\mathcal{L}}{\partial\alpha_{s,t}}=0$, $\frac{\partial\mathcal{L}}{\partial\beta_s}=0$, and after elimination of the primal variables ω_s and E. □

By combining the results from Lemma 3.2 with (18) and (19), we have:

$$\Theta_{BD}\begin{bmatrix} f(u_0) & f(u_1) & \dots & f(u_{N-1}) \end{bmatrix} = \mathcal{A}_{BD}^T \Omega(i+1 \,:\, i+j,:) + \mathcal{B}_{BD}^T \sum_{t=1}^{N} \Omega(t,:). \quad (21)$$

Hence, estimates for B, D in Θ_{BD} and the non-linearity f can be obtained from a rank m approximation of the right hand side of (21), for instance using a singular value decomposition. This is a typical step in overparameterization approaches [1] and amounts to projecting the results for the overparameterized model as used in the estimation onto the class of Hammerstein models.

3.5 Practical implementation

Following the discussion in the previous sections, the final algorithm for Hammerstein N4SID subspace identification can be summarized as follows:

1. Find estimates for the oblique projections $\mathcal{O}_i$ and $\mathcal{O}_{i+1}$ from (16) and (17).

2. Find estimates for the state following the procedure outlined in Subsection 3.3.

3. Obtain estimates for A, C, $\mathcal{A}_{BD}$ and $\mathcal{B}_{BD}$ following the procedure outlined in Subsection 3.4.

4. Obtain estimates for B, D en f from a rank-m approximation of (21).

It should be noted at this point that given the fact that regularization is inherently present in the proposed identification technique, lack of persistency of excitation will not lead to any numerical problems. However, in order to ensure that all aspects of the linear system are properly identified, persistency of excitation of $f(u)$ of at least order $2i$ is desired (see also Section 2). Persistency of excitation of $f(u)$ can for some nonlinear functions f be expressed as a condition on the original inputs u but the relation is certainly not always straightforward (see for instance [30] for a discussion on this issue).

Furthermore, it is important to remark that the estimate of the static nonlinearity will only be reliable in regions where the input density is sufficiently high.

4 Illustrative examples

In this section, the presented algorithm is compared to the Hammerstein ARX approach presented in [8] and classical subspace Hammerstein identification algorithms involving overparameterization in orthogonal basis functions. Two properties of the Hammerstein N4SID subspace approach will thereby be highlighted:

- The greater flexibility that comes with the use of state-space models over more classical Hammerstein ARX approaches.
- The superior performance of the introduced algorithm over existing overparameterization approaches for Hammerstein subspace identification.

4.1 Comparison between Hammerstein N4SID and Hammerstein ARX

Consider the following system which belongs to the Hammerstein class of models:

$$A(z)(y+\nu) = B(z)f(u) + e, \tag{22}$$

with A and B polynomials in the forward shift operator z where $B(z) = z^6 + 0.8z^5 + 0.3z^4 + 0.4z^3$, $A(z) = (z - 0.98e^{\pm i})(z - 0.98e^{\pm 1.6i})(z - 0.97e^{\pm 0.4i})$. Let $f : \mathbb{R} \to \mathbb{R}$ such that $f(u) = \text{sinc}(u)u^2$ be the static nonlinearity. A dataset was generated from this system where $u_t \sim \mathcal{N}(0, 2)$ is a white Gaussian noise sequence for $t = 0, \ldots, N-1$ with $N = 1000$ and $\{e_t\}_{t=0}^{N-1}$ is a sequence of Gaussian white noise with a level of 10% of the level of the nonlinearity $f(u)$.

The measurement noise terms ν_t were chosen to be zero mean Gaussian white noise such that a signal to noise ratio of 10 was obtained at the output signal. The Hammerstein N4SID subspace identification algorithm as derived in Section 3 was used to extract the linear model and the static nonlinearity f from the dataset described above. The number of block-rows i in the Block Hankel matrices was set to 10 which is a common choice in subspace identification algorithms [27]. An advantage of the N4SID algorithm is that the model order, 6 in this case, follows automatically from the spectrum of the SVD. The hyper-parameters in the LS-SVM N4SID algorithm were selected as $\sigma = 0.1, \gamma = 1000, \gamma_{BD} = 10$ by validation on an independent validation set. The resulting linear system and static nonlinearity are displayed in Figure 1.

As a comparison, the results of the LS-SVM ARX estimator [8] are also displayed in Figure 1. For the ARX-estimator, the number of poles and zeros were assumed to be fixed a priori. Two hyper-parameters (the regularization constant and the bandwidth of the RBF kernel) which need to be set in this method were chosen in accordance with the choices reported in [8]. Note that although the Hammerstein ARX method performed very well for this example in the absense of output noise (see the examples in [8]), its performance deteriorates in the presence of output noise as evidenced by the poor fit in Figure 1.

This highlights one of the main advantages of the use of subspace identification methods [27] over more classical ARX procedures, namely that they allow for the successful estimation of a much wider class of linear systems. Note on the other hand that if the true system fits well into the more restricted ARX framework, use of the latter is to be prefered [8].

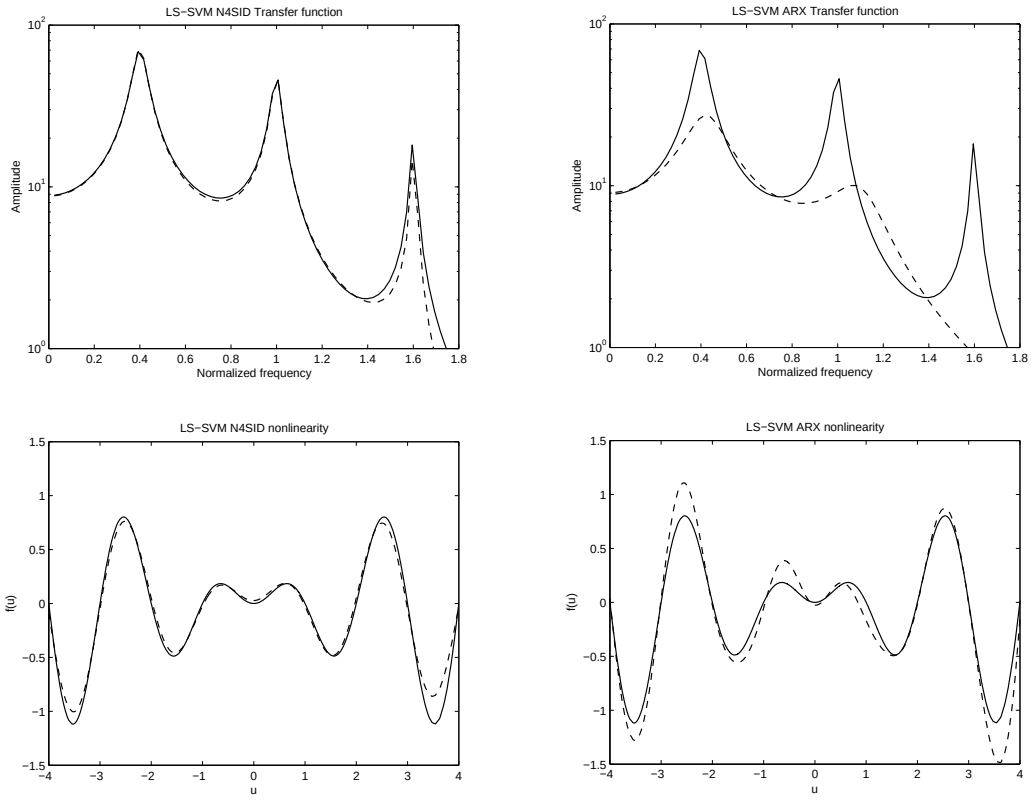


Figure 1: *True transfer function (solid) and estimated ones (dashed) for the LS-SVM N4SID subspace algorithm (top-left) and the LS-SVM ARX algorithm (top-right), as estimated from a sequence of 1000 input/output measurements on a simulated system, with the addition of 10% output noise. The true nonlinearities (solid) and estimated ones (dashed) are displayed below the transfer functions, for the N4SID case (lower-left), and the ARX-case (lower-right).*

4.2 Comparison with classical subspace overparameterization approaches

As mentioned before, a classical approach to Hammerstein system identification is to expand the static nonlinearity in a set of orthogonal or non-orthogonal basis-functions [17]. The same idea can be applied to subspace algorithms [15]. Once a set of basis-functions is considered, the one-dimensional input is transformed into a higher-dimensional input vector which contains the coefficients of the expansion of $f(u)$ in its basis. The classical N4SID subspace algorithm as outlined in Section 2 is thereafter applied. The linear system and static nonlinearities can be obtained from the obtained matrices B and D (see [15] for a detailed procedure).

This example will adopt the common choice of the Hermite polynomials as a basis. The best results on the dataset with output noise were obtained when selecting 7 Hermite polynomials with orders ranging from 0 to 6. The obtained linear system corresponding to this choice of basis functions is displayed in Figure 2. Note the rather poor performance of this method, compared to the LS-SVM N4SID algorithm. This can largely be attributed to the fact that the performance of subspace algorithms degrades as the number of inputs increases, certainly if these inputs are highly correlated [4]. This as a result of a bad conditioning of the matrices U_p and U_f as the number of rows increases and these rows get more correlated. For the 0^{th} order Hermite polynomial (which is a constant) this is certainly the case but also when leaving out this polynomial, condition numbers of 10^5 and higher are encountered. This problem does not occur in the N4SID LS-SVM algorithm as the latter features an inherently available regularization framework. An additional advantage is the flexibility one gets by plugging in an appropriate kernel and the fact that if localized kernels are used, no specific choices have to be made for their locations. The locations follow directly from the formulation of costfunctions as (14).

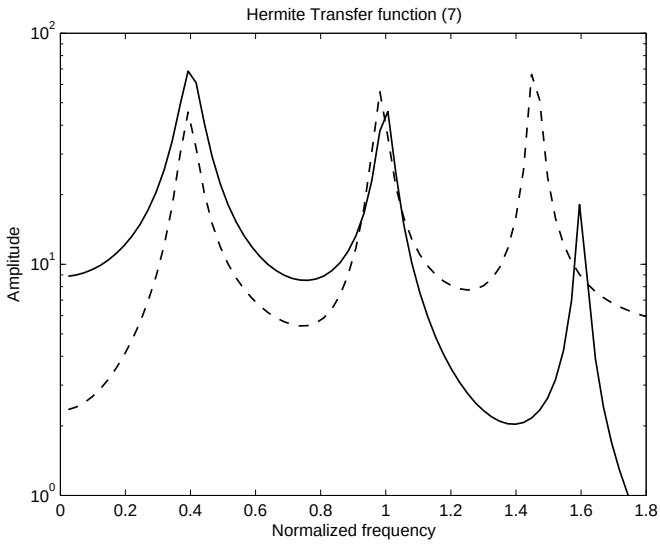


Figure 2: *True transfer function (solid) and the estimated one (dashed) for the Hermite N4SID subspace algorithm as estimated from a sequence of 1000 input/output measurements on a simulated system, with the addition of 10% output noise.*

5 Conclusions

In this paper, a method for the identification of Hammerstein systems was presented based on the well-known N4SID subspace identification algorithm. The basic framework of the N4SID algorithm is largely left untouched, except for the ordinary least squares steps which are replaced by a set of componentwise LS-SVM regressions. The proposed algorithm was observed to be able to extract the linear system and the static nonlinearity from data, even in the presence of output noise.

Acknowledgements. This research work was carried out at the ESAT laboratory of the Katholieke Universiteit Leuven. It is supported by grants from several funding agencies and sources: Research Council KU Leuven: Concerted Research Action GOA-Mefisto 666 (Mathematical Engineering), IDO (IOTA Oncology, Genetic networks), several PhD/postdoc & fellow grants; Flemish Government: Fund for Scientific

Research Flanders (several PhD/postdoc grants, projects G.0407.02 (support vector machines), G.0256.97 (subspace), G.0115.01 (bio-i and microarrays), G.0240.99 (multilinear algebra), G.0197.02 (power islands), G.0499.04 (robust statistics), G.0211.05 (nonlinear identification), research communities ICCoS, ANMMM), AWI (Bil. Int. Collaboration Hungary/ Poland), IWT (Soft4s (softsensors), STWW-Genprom (gene promotor prediction), GBOU-McKnow (Knowledge management algorithms), Eureka-Impact (MPC-control), Eureka-FLiTE (flutter modeling), several PhD grants); Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-10-29 (2002-2006) (2002-2006): Dynamical Systems and Control: Computation, Identification & Modelling), Program Sustainable Development PODO-II (CP/40: Sustainibility effects of Traffic Management Systems); Direct contract research: Verhaert, Electrabel, Elia, Data4s, IPCOS. JS is an associate professor and BDM is a full professor at K.U.Leuven Belgium, respectively.

References

[1] E.W. Bai. An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems. *Automatica*, 4(3):333–338, 1998.

[2] E.W. Bai. A blind approach to Hammerstein model identification. *IEEE Transactions on Signal Processing*, 50(7):1610–1619, 2002.

[3] F.H.I. Chang and R. Luus. A noniterative method for identification using the Hammerstein model. *IEEE Transactions on Automatic Control*, 16:464–468, 1971.

[4] A. Chiuso and G. Picci. On the ill-conditioning of subspace identification with inputs. *Automatica*, 40(4):575–589, 2004.

[5] P. Crama and J. Schoukens. Initial estimates of Wiener and Hammerstein systems using multisine excitation. *IEEE Transactions on Measurement and Instrumentation*, 50(6):1791–1795, 2001.

[6] E.J. Dempsey and D.T. Westwick. Identification of Hammerstein models with cubic spline nonlinearities. *IEEE Transactions on Biomedical Engineering*, 51:237–245, 2004.

[7] E. Eskinat, S.H. Johnson, and W.L. Luyben. Use of Hammerstein models in identification of nonlinear systems. *AIChE Journal*, 37(2):255–268, 1991.

[8] I. Goethals, K. Pelckmans, J.A.K. Suykens, and B. De Moor. Identification of MIMO Hammerstein models using least squares support vector machines. Technical Report 04-45, ESAT-SISTA, K.U.Leuven (Leuven Belgium), *Accepted for publication in Automatica*, 2004.

[9] I. Goethals, K. Pelckmans, J.A.K. Suykens, and B. De Moor. NARX identification of Hammerstein models using least squares support vector machines. In *Proceedings of*

the 6th IFAC Symposium on Nonlinear Control Systems (NOLCOS 2004), Stuttgart, Germany, pages 507–512, Sep. 2004.

[10] G.H. Golub and C.F. Van Loan. *Matrix Computations*. John Hopkins University Press, 1989.

[11] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, Heidelberg, 2001.

[12] A. Janczak. Neural network approach for identification of Hammerstein systems. *International Journal of Control*, 76(17):1749–1766, 2003.

[13] M.J. Korenberg. Recent advances in the identification of nonlinear systems: minimum-variance approximation by hammerstein models. In *Proceedings IEEE EMBS*, volume 13, pages 2258–2259, 1991.

[14] Z.H. Lang. Controller design oriented model identification method for Hammerstein systems. *Automatica*, 29:767–771, 1993.

[15] T. McKelvey and C. Hanner. On identification of Hammerstein systems using excitation with a finite number of levels. *Proceedings of the* 13^{th} *International Symposium on System Identification (SYSID2003)*, pages 57–60, 2003.

[16] K.S. Narendra and P.G. Gallman. An iterative method for the identification of nonlinear systems using the Hammerstein model. *IEEE Transactions on Automatic Control*, 11:546–550, 1966.

[17] M. Pawlak. On the series expansion approach to the identification of Hammerstein systems. *IEEE Transactions on Automatic Control*, 36:736–767, 1991.

[18] K. Pelckmans, I. Goethals, J. De Brabanter, J.A.K. Suykens, and B. De Moor. Componentwise least squares support vector machines. chapter in *Support Vector Machines: Theory and Applications,* L. Wang (Ed.), Springer, 2005, *in press*.

[19] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[20] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P. Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.

[21] J.C. Stapleton and S.C. Bass. Adaptive noise cancellation for a class of nonlinear dynamic reference channels. *IEEE Transactions on Circuits and Systems*, CAS-32:143–150, Feburari 1985.

[22] J.A.K. Suykens, G. Horvath, S. Basu, C. Micchelli, and J. Vandewalle, editors. *Advances in Learning Theory: Methods, Models and Applications*, volume 190 of *NATO Science Series III: Computer & Systems Sciences*. IOS Press Amsterdam, 2003.

[23] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[24] J.A.K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9:293–300, 1999.

[25] P. Van Overschee and B. De Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29(3):649–660, 1993.

[26] P. Van Overschee and B. De Moor. A unifying theorem for three subspace system identification algorithms. *Automatica,* Special Issue on Trends in System Identification, 31(12):1853–1864, 1995.

[27] P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers, 1996.

[28] T.H. van Pelt and D.S. Bernstein. Nonlinear system identification using Hammerstein and nonlinear feedback models with piecewise linear static maps - part i: Theory. *Proceedings of the American Control Conference (ACC2000)*, pages 225–229, 2000.

[29] V.N. Vapnik. *Statistical Learning Theory*. Wiley and Sons, 1998.

[30] M. Verhaegen and D. Westwick. Identifying MIMO Hammerstein systems in the context of subspace model identification methods. *International Journal of Control*, 63:331–349, 1996.

[31] G. Wahba. *Spline models for Observational data*. SIAM, 1990.

[32] D. Westwick and R. Kearney. Identification of a hammerstein model of the stretch reflex EMG using separable least squares. In *Proceedings of the World Congress on Medical Physics and Biomedical Engineering, Chicago*, 2000.

A LS-SVM function estimation

Let $\{(x_i, y_i)\}_{i=0}^N \subset \mathbb{R}^d \times \mathbb{R}$ be a set of independently and identically distributed (i.i.d.) input/output training data with input samples x_i and output samples y_i. Consider the static regression model $y_i = f(x_i) + e_i$ where where $f : \mathbb{R}^d \to \mathbb{R}$ is an unknown real-valued smooth function and $\{e_i\}_{i=1}^N$ are i.i.d. (uncorrelated) random errors with $E[e_i] = 0$, $E[e_i^2] = \sigma_e^2 < \infty$. Originating from the research on classification algorithms, Support Vector Machines (SVMs) and other kernel methods have been used for the purpose of estimating the nonlinear f. The following model is assumed

$$f(x) = w^T \varphi(x) + b,$$

where $\varphi(x) : \mathbb{R}^d \to \mathbb{R}^{n_H}$ denotes a potentially infinite ($n_H = \infty$) dimensional feature map which doesn't have be known explicitly. In the following paragraph we will see how the feature map can be induced in an implicit way by the adoption of a proper kernel function. The regularized cost function of the Least Squares SVM (LS-SVM) is given as

$$\begin{aligned} \min_{w,b,e} \mathcal{J}(w,e) &= \frac{1}{2} w^T w + \frac{\gamma}{2} \sum_{i=1}^{n} e_i^2, \\ \text{s.t. } y_i &= w^T \varphi(x_i) + b + e_i, \quad i = 1, \ldots, N. \end{aligned}$$

The relative importance between the smoothness of the solution and the data fitting is governed by the scalar $\gamma \in \mathbb{R}_0^+$ referred to as the regularization constant. The optimization performed corresponds to ridge regression (see e.g. [10]) in feature space. In order to solve the constrained optimization problem, the Lagrangian $\mathcal{L}(w, b, e; \alpha) = \mathcal{J}(w, e) - \sum_{i=1}^N \alpha_i \{w^T \varphi(x_i) + b + e_i - y_i\}$ is constructed, with $\alpha_i \in \mathbb{R}$ the Lagrange multipliers. After application of the conditions for optimality: $\frac{\partial \mathcal{L}}{\partial w} = 0$, $\frac{\partial \mathcal{L}}{\partial b} = 0$, $\frac{\partial \mathcal{L}}{\partial e_i} = 0$, $\frac{\partial \mathcal{L}}{\partial \alpha_i} = 0$, the following set of linear equations is obtained:

$$\left[\begin{array}{c|c} 0 & 1_N{}^T \\ \hline 1_N & \Omega + \gamma^{-1} I_N \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline y \end{array}\right], \tag{23}$$

where $y = \begin{bmatrix} y_1 & \dots & y_N \end{bmatrix}^T$, $1_N = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^T$, $\alpha = \begin{bmatrix} \alpha_1 & \dots & \alpha_N \end{bmatrix}^T$, $\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j), \ \forall i, j = 1, \dots, N$, with K a positive definite Mercer kernel function. Note that in order to solve the set of equations (23), the feature map φ does never have to be defined explicitly. Only its inner product, a positive definite kernel, is needed. This is called the kernel trick [29, 19]. For the choice of the kernel $K(\cdot, \cdot)$, see e.g. [19]. Typical examples are the use of a linear kernel $K(x_i, x_j) = x_i^T x_j$, a polynomial kernel $K(x_i, x_j) = (\tau + x_i^T x_j)^d$ of degree d or the RBF kernel $K(x_i, x_j) = \exp(-\|x_i - x_j\|_2^2/\sigma^2)$ where σ denotes the bandwidth of the kernel. The resulting LS-SVM model can be evaluated at a new point $x_* \in \mathbb{R}^d$ as

$$\hat{f}(x_*) = \sum_{i=1}^{N} \hat{\alpha}_i K(x_*, x_i) + \hat{b},$$

where $(\hat{b}, \hat{\alpha})$ is the solution to (23).

LS-SVMs are reformulations to the original SVMs employed for tasks in classification [24], regression [23] and provides primal-dual optimization formulations to the algorithm of kernel Principal Component Analysis (KPCA), kernel Partial Least Squares (KPLS) and kernel Canonical Correlation Analysis (KCCA) and others [23]. By the use of the least squares criterion and the use of equality instead of inequality constraints, the estimation typically boils down to the solution of a set of linear equations or eigenvalue problems instead of the optimization of quadratic programming problems [24, 23, 22]. In [8], the task of identifying a Hammerstein model using an LS-SVM based approach of the nonlinearity and an ARX system was considered.

B Hammerstein FIR models using LS-SVMs

The extension of LS-SVMs towards the estimation of additive models was studied in [18]. It was applied towards the identification of Hammerstein ARX models in [8]. We review briefly the basic steps as they will reoccur in the presented technique. Let $\{(u_t, y_t)\}_{t=0}^{N-1} \subset$

$\mathbb{R} \times \mathbb{R}$ be a (SISO) sequence of observations. Consider a Hammerstein FIR model of order $L \in \mathbb{N}_0$.

$$y_t = \sum_{l=0}^{L-1} c_l f\left(u_{t-l}\right) + e_t = \sum_{l=0}^{L-1} c_l \left(w^T \varphi\left(u_{t-l}\right) + b\right) + e_t \quad \forall t = L-1, \ldots, N-1, \tag{24}$$

where $c = (c_1, \ldots, c_l)^T \in \mathbb{R}^l$ is a vector. Whenever both w, b as well as c are unknown, the simultaneous estimation of those parameters is known to be a hard problem. Following [1], an overparameterization technique can be adopted. Consider in the first stage the identification of the parameters w_l, b_l and c of the slightly broader model

$$y_t = \sum_{l=0}^{L-1} f_l(u_{t-l}) + e_t = \sum_{l=0}^{L-1} \left(w_l^T \varphi\left(u_{t-l}\right) + b_l\right) + e_t \quad \forall t = L-1, \ldots, N-1, \tag{25}$$

where $w_l \in \mathbb{R}^{N_h^l}$ and $b_l \in \mathbb{R}$ for all $l = 0, \ldots, L-1$. A necessary and sufficient condition for the restriction of (25) to the Hammerstein class (24) can be written as the rank constraint

$$\left[f_0(u) \ \ldots \ f_{L-1}(u)\right] = f(u)c^T \quad \forall u \in \mathbb{R}. \tag{26}$$

It becomes clear that the right hand side occuring in the term (25) has a non-unique representation as one can always add (and substract) a row-vector $\delta \in \mathbb{R}^{1\times L}$ to the nonlinear function $f_L : \mathbb{R} \to \mathbb{R}^{1\times L}$ such that $f_L(u) = \begin{bmatrix} f_0(u) & \ldots & f_{L-1}(u) \end{bmatrix} \in \mathbb{R}^{1\times L}$ and $\sum_{l=1}^{L} \delta(l) = 0$. This follows from the following relation

$$\left(f_L(u) + \delta\right) 1_L = f_L(u) 1_L \quad \forall u \in \mathbb{R}.$$

However, this operation does not preserve the constraint (26) if $\delta \neq 0_L$. As a bias term b can be found such that $y_t = f(u_{t-L}, \ldots, u_t) + b$ and $E[f_l(u)] = 0$, the nonlinear functions can be centered around zero without loss of generality. Then a necessary linear condition for (26) becomes

$$E\left[f_0(u) \ldots f_{L-1}(u)\right] = E\left[f(u)\right] c^T = 0_L^T, \tag{27}$$

or using the empirical counterpart

$$\sum_{t=0}^{N-1} f_l(u_t) = \sum_{t=0}^{N-1} w_l^T \varphi(u_t) = 0 \quad \forall l = 0, \ldots, L-1, \tag{28}$$

which are referred to as the centering constraint.

The overparameterization procedure amounts to first obtaining estimates of the model class (25) subject to the centering constraints (28) and afterwards projecting the result onto the Hammerstein class by calculating a rank one approximation of the estimate using a Singular Value Decomposition.

The primal-dual derivation can be summarized as follows [8]

Lemma B.1. *Consider the primal estimation problem*

$$\begin{aligned} \underset{w,e,b}{\arg\min} \; &= \frac{\gamma}{2}\sum_{t=L-1}^{N-1} e_t^2 + \frac{1}{2}\sum_{l=0}^{L-1} w_l^T w_l \\ s.t. \quad &\begin{cases} \sum_{l=0}^{L-1} w_l^T \varphi\left(u_{t-l}\right) + b + e_t - y_t & \forall t = L-1,\ldots,N-1 \\ \sum_{t=0}^{N-1} w_l^T \varphi(u_t) = 0 & \forall l = 0\ldots,L-1. \end{cases} \end{aligned} \tag{29}$$

Let $\Omega_l \in \mathbb{R}^{N-L\times N-L}$ be a positive definite matrix defined as $\Omega_{l,ij} = \varphi(u_{i-l})^T\varphi(u_{j-l})$ for all $i,j = L-1,\ldots,N-1$ and $l = 0,\ldots,L-1$. Let $\Omega = \sum_{l=0}^{L-1}\Omega_l$ be the kernel matrix and let $\mathcal{S} \in \mathbb{R}^{(N-L)\times L}$ such that $\mathcal{S}(:,l) = \Omega_{(l-1)}1_{N-L}$ for all $l = 0,\ldots,L-1$.The solution is uniquely characterized by the following dual problem

$$\left[\begin{array}{c|c|c} 0 & 0_L^T & 1_{N-L}^T \\ \hline 0_L & 0_{L\times L} & \mathcal{S}^T \\ \hline 1_{N-L} & \mathcal{S} & \Omega + \gamma^{-1}I_{N-L} \end{array}\right]\left[\begin{array}{c} b \\ \hline \beta \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 0_L \\ \hline y \end{array}\right], \tag{30}$$

where $\alpha \in \mathbb{R}^{N-L}$ and $\beta = (\beta_1,\ldots,\beta_L)^T \in \mathbb{R}^L$ denote the Lagrange multipliers to the constraints in (29). The estimate can be evaluated in a new datapoint $u \in \mathbb{R}$ as

$$\hat{f}_l(u) = \sum_{t=L-1}^{N-1}\hat{\alpha}K(u_t,u) + \hat{\beta}_l\sum_{t=L-1}^{N-1}K(u_t,u)$$

and $\hat{f}(u_t) = \sum_{l=0}^{L-1}\hat{f}_l(u_{t-l+1}) + \hat{b}$ where $\hat{\alpha},\hat{\beta}$ and $\hat{b}$ are the solution to (30).