# Linear Parametric Noise Models for Least Squares Support Vector Machines

Tillmann Falck, Johan A.K. Suykens and Bart De Moor

***Abstract*—In the identification of nonlinear dynamical models it may happen that not only the system dynamics have to be modeled but also the noise has a dynamic character. We show how to adapt Least Squares Support Vector Machines (LS-SVMs) to take advantage of a known or unknown noise model. We furthermore investigate a convex approximation based on overparametrization to estimate a linear auto regressive noise model jointly with a model for the nonlinear system. Considering a noise model can improve generalization performance.**

**We discuss several properties of the proposed scheme on synthetic data sets and finally demonstrate its applicability on real world data.**

## I. Introduction

The objective in system identification [1] of nonlinear systems [2], [3] is to estimate a model for a dynamical system from observational data. In linear as well as in nonlinear systems, model structures are of particular interest as they are crucial for the flexibility of the model to explain data. In nonlinear systems NARX and NFIR structures are most used as the corresponding estimation problems are linear in the parameters. Then the estimation is convex, if a conex objective is used. Generalizations of more advanced model structures like ARMAX or Box-Jenkins (BJ) to nonlinear systems exist but even in a linear setting the identification is a non convex problem. In this paper we consider NARX models extended by a linear ARMA model for the noise. This structure is depicted in Figure 1. We will denote this hybrid structure as ARMA-NARX. Note that in a NARMAX model the estimated noise is used as an additional input to the nonlinear system and thus can have nonlinear dynamics. The ARMA-NARX model is simply tailored towards colored noise instead of assuming a white spectrum as in NARX models.

We consider two cases:

- In the first case we assume that the noise model is known. This information can be easily integrated into the estimation problem and can improve the performance of the resulting model. This approach has already been explored in [4]. In [4] the noise model is tuned as hyperparameters of the nonlinear model, if it is not known a priori. In this part, we restrict ourselves to generalize the results from AR to ARMA models.
- The second case jointly estimates an AR noise model and the NARX part. This is a nonconvex, nonlinear problem. The main contribution of this paper is to propose a convex relaxation to this problem. This complements [4] with an effective way to obtain estimates for unknown noise models.

Tillmann Falck, Johan Suykens and Bart De Moor are with the SCD group of the Department of Electrical Engineering (ESAT), Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven (Heverlee), Belgium. Email: {tillmann.falck,johan.suykens,bart.demoor}@esat.kuleuven.be

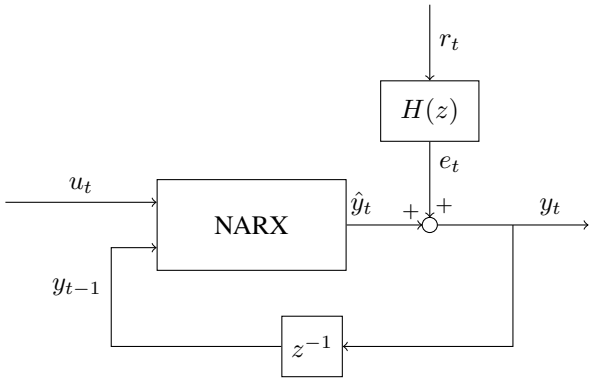



Fig. 1: Block diagram of a nonlinear model ($D_u = 0$, $D_y = 1$) consisting of a NARX part and a linear noise model. Here denoted as ARMA-NARX.

The relaxation is based on the overparametrization technique [11], [12]. It was introduced for a special class of structured nonlinear systems called Hammerstein systems. The idea is to relax non-convex bilinear products by replacing them with new independent variables. This leads to a convex formulation and in the context of identification of Hammerstein systems using LS-SVMs has been successfully applied in [13], [14].

To model the nonlinear system we employ Least Squares Support Vector Machines (LS-SVMs) which is based on the methodology of Support Vector Machines (SVMs) [5], [6]. Both belong to the class of kernel based models, which also includes e.g. Splines [7] and Gaussian Processes [8]. In LS-SVM the inequality constraints of SVMs are replaced by equality constraints and the $L_1$-loss on the residuals by the sum of squares. For regression problems this has the advantage that it can be solved by a linear system instead of a QP. Disadvantages of this scheme are the non-sparse solution and no inherent robustness. Especially for large scale data sets sparsity can be obtained by approximating the feature map on a subsample and then solving the primal problem. This is called Fixed-Size LS-SVM [9]. If needed robustness can be achieved by reweighting the residuals [10].

This paper is structured as follows. In Section II we show how to integrate a known ARMA noise model with a LS-SVM based nonlinear model. The joint convex estimation of an AR(P) noise model with the nonlinear model based on overparametrization is covered in Section III. Experimental results on synthetic as well as real data illustrating the pro-

posed scheme are given in IV. Finally the paper is concluded in Section V.

## II. Incorporating linear noise models in LS-SVMs

### A. LS-SVM regression

Consider observational data $\{\boldsymbol{x}_t, y_t\}_{t=1}^N$ with $\boldsymbol{x}_t \in \mathbb{R}^D$ and $y_t \in \mathbb{R}$ where $\boldsymbol{x}_t = [y_{t-1}, \ldots, y_{t-D_y}, u_t, \ldots, u_{t-D_u}]^T$ and $D = D_u + D_y$. $u_t$, $y_t$ are respectively input and output measurements of a nonlinear system $\mathcal{S}$ and $t$ denotes the time index. Then a nonlinear dynamical model for $\mathcal{S}$ can be estimated using least squares support vector machine (LS-SVM) regression [9]

$$\begin{aligned} &\min_{\boldsymbol{w},b,e_t} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\sum_{t=1}^N e_t^2 \\ &\text{subject to} \\ &\quad y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \ldots, N \end{aligned} \tag{P-0}$$

with the feature map $\boldsymbol{\varphi} : \mathbb{R}^D \to \mathbb{R}^{n_h}$ and the parameter vector $\boldsymbol{w} \in \mathbb{R}^{n_h}$. Here we use equation numbers of the form (P-$x$) and (D-$x$) to denote a primal optimization problem and its dual. Note that not all dual formulations are given explicitly. The predictive model for (P-0) is given by $\hat{y} = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}) + b$. The dual model is

$$\hat{y} = \sum_{t=1}^N \alpha_t K(\boldsymbol{x}_t, \boldsymbol{x}) + b \tag{1}$$

respectively. It can be derived using Lagrangian duality and by applying the kernel trick to replace inner products of the feature map with a the kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j)$.

Least squares estimation is optimal for Gaussian white noise $e_t$. The problem as stated above is convex in its parameters $\boldsymbol{w}$, $e_t$ and $b$ and can be solved as a linear system. To obtain a model with good generalization performance model selection is crucial. The model is defined the regularization parameter $\gamma$ and possibly one or more parameters of the feature map/kernel function. The selection of model parameters is usually done by a validation criterion e.g. cross-validation [15]. In this paper we will discuss noise models, i.e. situations where $e_t$ is not white. We show how to incorporate an a priori known linear noise model for $e_t$ to improve the estimation and a propose scheme to estimate a noise model.

### B. Parametric noise models

Define the backshift operator in time $z^{-1}$ as $z^{-1}e_t = e_{t-1}$. Consider a minimum phase noise model for $e_t$ in transfer function form $A(z)e_t = B(z)r_t$ where $r_t$ is a white noise sequence, $A(z) = 1 + a_1z^{-1} + \cdots + a_Pz^{-P}$ and $B(z) = b_0 + b_1z^{-1} + \cdots + b_Qz^{-Q}$. For the sake of simplicity we will assume for the rest of the paper that $P = Q$. Assuming that $A(z)$ and $B(z)$ are known apriori, the optimal LS-SVM based model is obtained by solving

$$\begin{aligned} &\min_{\boldsymbol{w},b,e_t,r_t} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\sum_{t=P+1}^N r_t^2 \\ &\text{subject to} \\ &\quad y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = P+1, \ldots, N \\ &\quad A(z)e_t = B(z)r_t, \qquad\quad t = P+1, \ldots, N. \end{aligned} \tag{P-1}$$

If $A(z)$ and $B(z)$ are not known a priori, they can be seen as additional hyper-parameters to the problem, which have to be selected for example by cross-validation. In case $A(z)$ and $B(z)$ correspond to the true noise model the residual $r_t$ is white and the formulation becomes optimal. The values of $r_t$ and $e_t$ for $t = 1, \ldots, P$ are the initial conditions for the noise model and are assumed to be zero.

The constraint $A(z)e_t = B(z)r_t$ for $t = P+1, \ldots, N$ with zero initial conditions can be written in matrix notation as $\boldsymbol{Ae} = \boldsymbol{Br}$ with

$$\boldsymbol{A} = \begin{bmatrix} 1 & & & & \\ a_1 & 1 & & & \\ a_2 & a_1 & 1 & & \\ \ddots & & \ddots & & \\ & a_P & \cdots & & 1 \end{bmatrix} \in \mathbb{R}^{N\times N},$$

$$\boldsymbol{B} = \begin{bmatrix} b_0 & & & & \\ b_1 & b_0 & & & \\ b_2 & b_1 & b_0 & & \\ \ddots & & \ddots & & \\ & b_P & \cdots & & b_0 \end{bmatrix} \in \mathbb{R}^{N\times N},$$

$\boldsymbol{e} = [e_1, \ldots, e_N]^T$ and $\boldsymbol{r} = [r_1, \ldots, r_N]^T$.

*Proposition 1:* Solving (P-0) with weighted residuals [10] $\boldsymbol{e}^T\boldsymbol{De}$ and $\boldsymbol{D} = \boldsymbol{A}^T\boldsymbol{B}^{-T}\boldsymbol{B}^{-1}\boldsymbol{A}$ instead of $\boldsymbol{e}^T\boldsymbol{e}$ is equivalent to solving (P-1) with zero initial conditions.

The solution to the weighted problem is given by the linear system

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{D}^{-1} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix} \tag{D-1}$$

in terms of the dual variables $\boldsymbol{\alpha}$ and with $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

*Proof:* For $b_0 \neq 0$ the matrix $\boldsymbol{B}$ is invertible, therefore the noise model can be rewritten as $\boldsymbol{r} = \boldsymbol{B}^{-1}\boldsymbol{Ae}$. Substitution of this relation into the objective function of (P-1) yields the weighting matrix $\boldsymbol{D}$. Note that $\boldsymbol{A}$ is also invertible and thus $\boldsymbol{D}$ as well. This is needed for the solution in the dual domain.

Deriving the dual system relies on Lagrangian duality and the kernel trick $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j)$. For details consult [10]. ■

*Remark 1:* The nonlinear model $y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t$ can also be solved for $e_t$. Then substitution into $A(z)e_t =$

$B(z)r_t$ yields a new combined modeling equation

$$y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + \sum_{k=1}^{P} a_k \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + \bar{b} - \sum_{k=1}^{P} a_k y_{t-k} + B(z)r_t \quad (2)$$

with $\bar{b} = b(1 + \sum_{k=1}^{P} a_k)$. This relation can be written more compactly using matrix notation as $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\Phi}^T\boldsymbol{w} + b\boldsymbol{A}\boldsymbol{1} + \boldsymbol{B}\boldsymbol{r}$ where $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\varphi}(\boldsymbol{x}_N)]$. Then the dual system for the problem (P-1), with (2) replacing the model constraints, is

$$\begin{bmatrix} \boldsymbol{A}\boldsymbol{\Omega}\boldsymbol{A}^T + \gamma^{-1}\boldsymbol{B}\boldsymbol{B}^T & \boldsymbol{A}\boldsymbol{1} \\ \boldsymbol{1}^T\boldsymbol{A}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}\boldsymbol{y} \\ 0 \end{bmatrix}. \quad \text{(D-1}')$$

Note that no explicit inverse is needed in this formulation. The kernel matrix is replaced by an equivalent kernel matrix $\boldsymbol{A}\boldsymbol{\Omega}\boldsymbol{A}^T$. In [4] also the model is expressed in terms of an equivalent kernel matrix $K_{eq}(\boldsymbol{x}_j, \boldsymbol{x}_i) = \sum_{k,l=0}^{P} a_k a_l K(\boldsymbol{x}_{i-k}, \boldsymbol{j} - \boldsymbol{l})$ with $a_0 = 1$. The model can then be evaluated at a new point by

$$\hat{y}_t = f(y_{t-1}, \ldots, y_{t-P}, \boldsymbol{x}_t, \ldots, \boldsymbol{x}_{t-P}) = \sum_{n=P+1}^{N} \alpha_t K_{eq}(\boldsymbol{x}_n, \boldsymbol{x}_t) + \bar{b} - \sum_{k=1}^{P} a_k y_{t-k}. \quad (3)$$

For a model with good generalization performance model selection is needed. If the parameters $a_p$ and $b_q$ of the noise model are not known apriori, they have to be included in the model selection. In that case the regularization parameter $\gamma$, parameters of the kernel function and the noise model coefficients have to be tuned according to a validation scheme. This is computationally very demanding for all but very low order noise models. In the next section we propose a convex relaxation, that is able to estimate the noise model coefficients jointly with the parameters of the nonlinear model $\boldsymbol{w}$ and $b$.

## III. Estimation of parametric noise models

In the following we will only consider purely autoregressive models of order $P$ ($AR(P)$ noise models) i.e. $B(z) = 1$. This simplifies the estimation problem as the nonconvex product of unknowns $b_q r_{t-q}$ do not have to be considered. It also simplifies the prediction as the sequence $r_t$ does not have to estimated.

### A. Primal model

Therefore we consider the problem of jointly estimating the nonlinear model and a linear parametric noise model. This is formalized in the following nonconvex optimization problem

$$\begin{aligned} \min_{\boldsymbol{w},b,a_k,e_t,r_t} \; & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=P+1}^{N} r_t^2 \\ \text{subject to} \; & y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \qquad t = 1, \ldots, N, \\ & e_t = \sum_{k=1}^{P} a_k e_{t-k} + r_t, \qquad t = P+1, \ldots, N. \end{aligned} \quad \text{(P-2)}$$

The nonconvexity is due to the bilinear term $a_k e_{t-k}$. Based on the idea of overparametrization [11], [12] we propose a convex approximation to (P-2). In a first step eliminate $e_t$ as done in (2). In the new expression the nonconvexity is now contained in bilinear terms $a_k\boldsymbol{w}$. The idea of overparametrization is to replace these terms by new variables $\boldsymbol{w}_k = a_k\boldsymbol{w}$, $k = 1, \ldots, P$. For ease of notation and clarity we will additionally define $\boldsymbol{w}_0 = a_0\boldsymbol{w}$ where $a_0 = 1$. Then a convex relaxation to (P-2) can be written as

$$\begin{aligned} \min_{\boldsymbol{w}_k,\bar{b},a_k,r_t} \; & \frac{1}{2}\sum_{k=0}^{P} \boldsymbol{w}_k^T\boldsymbol{w}_k + \frac{1}{2}\gamma \sum_{t=P+1}^{N} r_t^2 \\ \text{subject to} \; & \sum_{k=0}^{P} \boldsymbol{w}_k^T\boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + \bar{b} + r_t \\ & \qquad = y_t + \sum_{k=1}^{P} a_k y_{t-k}, \qquad t = P+1, \ldots, N. \end{aligned} \quad \text{(P-3)}$$

To fully recover the original problem in (P-2) a rank constraint $\mathrm{rank}([\boldsymbol{w}_0, ..., \boldsymbol{w}_P]) = 1$ on the newly introduced variables would have to be included in the problem. This rank constraint captures the nonconvexity of (P-2) and the convex approximation is then achieved by simply dropping it from the problem. For LS-SVMs this technique has been successfully applied for the identification of Hammerstein systems in [13], [14].

### B. Solution in dual domain

In support vector machines the feature map $\boldsymbol{\varphi}$ is implicitly defined by a positive semidefinite kernel function $K(\cdot, \cdot)$. Depending on the choice of the kernel function the feature map can be infinite dimensional, as it is the case for the widely used RBF kernel $K_{\mathrm{RBF}}(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\|\boldsymbol{x}-\boldsymbol{y}\|^2/\sigma^2)$. To obtain a finite dimensional solution in terms of the kernel function the Lagrange dual is computed and the kernel trick is applied to replace inner products of the feature map $\boldsymbol{\varphi}(\boldsymbol{x})^T\boldsymbol{\varphi}(\boldsymbol{y})$ by the kernel function $K(\boldsymbol{x}, \boldsymbol{y})$. The final solution is formalized in the following Lemma.

*Lemma 1:* The solution of (P-3) in the dual is given by

$$\begin{bmatrix} \sum_{k=0}^{P} \boldsymbol{\Omega}_k + \frac{1}{\gamma}\boldsymbol{I} & \boldsymbol{Y} & \boldsymbol{1} \\ \boldsymbol{Y}^T & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{1}^T & \boldsymbol{0} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{a} \\ \bar{b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{0} \\ 0 \end{bmatrix} \quad \text{(D-3)}$$

with $(\boldsymbol{\Omega}_k)_{ij} = K(\boldsymbol{x}_{i-k}, \boldsymbol{x}_{j-k})$, $P+1 \leq i, j \leq N$ where $(\boldsymbol{\Omega}_k)_{ij}$ is the $ij$-th element of $\boldsymbol{\Omega}_k$. Furthermore $\boldsymbol{\alpha}$ are the Lagrange multipliers corresponding to the equality

constraints, $\boldsymbol{y}_k = [y_{P+1-k}, \ldots, y_{N-k}]^T$ for $k = 0, \ldots, P$ and $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_P]$. In the following we will use $\hat{\boldsymbol{a}}_{\mathrm{LS}} = [1, \hat{\boldsymbol{a}}^T]^T$ to denote the estimate for $a_k$ resulting from (D-3).

*Proof:* The Lagrangian for (P-3) is

$$\mathscr{L}(\boldsymbol{w}_k, \bar{b}, a_k, r_t, \boldsymbol{\alpha}) = \frac{1}{2}\sum_{k=0}^{P} \boldsymbol{w}_k^T \boldsymbol{w}_k + \frac{1}{2}\gamma \sum_{t=P+1}^{N} r_t^2 - \sum_{t=P+1}^{N} \alpha_t \left( \sum_{k=0}^{P} \boldsymbol{w}_k^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + \bar{b} + r_t - \sum_{k=1}^{P} a_k y_{t-k} - y_t \right). \quad (4)$$

Taking the Karush-Kuhn-Tucker (KKT) conditions [16] for optimality one obtains

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{w}_k} : \quad \boldsymbol{w}_k = \sum_{t=P+1}^{N} \alpha_t \boldsymbol{\varphi}(\boldsymbol{x}_{t-k}), \quad k = 0, \ldots, P, \quad (5a)$$

$$\frac{\partial \mathscr{L}}{\partial \bar{b}} : \quad \sum_{t=P+1}^{N} \alpha_t = 0, \quad (5b)$$

$$\frac{\partial \mathscr{L}}{\partial r_t} : \quad \gamma r_t = \alpha_t, \quad t = P+1, \ldots, N, \quad (5c)$$

$$\frac{\partial \mathscr{L}}{\partial a_k} : \quad \sum_{t=P+1}^{N} \alpha_t y_{t-k} = 0, \quad k = 1, \ldots, P, \quad (5d)$$

$$\frac{\partial \mathscr{L}}{\partial \alpha_t} : \quad y_t = \sum_{k=0}^{P} \boldsymbol{w}_k^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + \bar{b} + r_t - \sum_{k=1}^{P} a_k y_{t-k}, \quad t = P+1, \ldots, N. \quad (5e)$$

Substitution of (5a) and (5c) into $\partial \mathscr{L} / \partial \alpha_t = 0$ yields

$$\sum_{k=0}^{P} \sum_{n=P+1}^{N} \alpha_n K(\boldsymbol{x}_{n-k}, \boldsymbol{x}_{t-k}) + \bar{b} + \alpha_t = y_t + \sum_{k=1}^{P} a_k y_{t-k}$$

after applying the kernel trick $K(\boldsymbol{x}_{n-k}, \boldsymbol{x}_{t-k}) = \boldsymbol{\varphi}(\boldsymbol{x}_{n-k})^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-k})$. Expressing this, (5b) and (5d) in matrix notation yields (D-3). ∎

*Remark 2:* Evaluating the overparametrized model in terms of the dual variables $\boldsymbol{\alpha}$ and primal variables $\bar{b}$ and $\{a_k\}_{k=1}^P$ is done using the one step ahead predictor

$$\hat{y}_t = f(y_{t-1}, \ldots, y_{t-P}, \boldsymbol{x}_t, \ldots, \boldsymbol{x}_{t-P}) = \sum_{n=P+1}^{N} \alpha_n \sum_{k=0}^{P} K(\boldsymbol{x}_{n-k}^{\mathrm{train}}, \boldsymbol{x}_{t-k}) + \bar{b} - \sum_{k=1}^{P} a_k y_{t-k}. \quad (6)$$

### C. Projection onto true model class

The model obtained from (D-3) is only an approximation for the AR(P)-LS-SVM model stated in (P-2). The overparametrized model needs to be projected to recover the AR(P) model structure. The approximation stems from dropping the rank-1 constraint on $\boldsymbol{W} = [\boldsymbol{w}_0, \ldots, \boldsymbol{w}_P]$. In presence of the rank-1 constraint the solution could be expressed as the outer product $\boldsymbol{W} = \boldsymbol{w}\boldsymbol{a}^T$. For the popular Gaussian RBF kernel this matrix has an infinite number of rows. Therefore consider the matrix $\boldsymbol{W}^T\boldsymbol{W}$ and its eigenvalue decomposition $\boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{V}\boldsymbol{S}^2\boldsymbol{V}^T$ to recover the rank-1 structure. Using the KKT conditions (5a) for $\boldsymbol{w}_k$ the columns of $\boldsymbol{W}$ can be expressed in terms of the dual variables $\boldsymbol{\alpha}$. Then the finite dimensional matrix can be computed by applying the kernel trick on all elements of the matrix. This yields

$$\boldsymbol{W}^T\boldsymbol{W} = \begin{bmatrix} \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{00}\boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{0P}\boldsymbol{\alpha} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{P0}\boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{PP}\boldsymbol{\alpha} \end{bmatrix} \quad (7)$$

with $(\boldsymbol{\Omega}_{kl})_{ij} = K(\boldsymbol{x}_{i-k}, \boldsymbol{x}_{j-l})$ for $k, l = 0, \ldots, P$ and $i, j = P+1, \ldots, N$.

Now let $s_0 \geq s_1 \geq \cdots \geq s_P \geq 0$ be the orderd sequence of singular values of $\boldsymbol{W}$ such that $\boldsymbol{S} = \mathrm{diag}(s_0, \ldots, s_P)$ and denote the corresponding right singular vectors by $\boldsymbol{v}_k$. Both can be obtained from the eigenvalue decomposition of $\boldsymbol{W}^T\boldsymbol{W}$. Then an estimate for $\boldsymbol{a}$ can be obtained as $\hat{\boldsymbol{a}}_{\mathrm{SVD}} = \boldsymbol{v}_0/(\boldsymbol{v}_0)_0$. Using this estimate a complete model can be estimated by solving (P-1).

*Remark 3:* The projection onto the class of AR(P)-LS-SVM is incomplete as two independent estimates for $a_k$ are obtained, one following directly from the solution of the dual system (D-3) and the other from the rank one approximation of $\boldsymbol{W}$ as outlined in this section. Therefore we compare the performance of an AR(P)-LS-SVM model based on both estimates in the experimental section.

**Algorithm 1** Overparametrized model (OVER)

**Training:**

1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (D-3) to obtain estimates for $\boldsymbol{\alpha}$, $b$ and $\boldsymbol{a}$

**Prediction:**

Estimates are generated according to (6)

**Algorithm 2** AR(P) model with direct estimate for the noise model (DIRECT)

**Training:**

1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (D-3) to obtain estimates for $\boldsymbol{\alpha}$, $b$ and $\boldsymbol{a}$, denote the estimate for $\boldsymbol{a}$ by $\hat{\boldsymbol{a}}_{\mathrm{LS}}$
3. compute final model by solving (P-1) given $\hat{\boldsymbol{a}}_{\mathrm{LS}}$

**Prediction:**

Estimates are generated according to (3)

This results in three possible algorithms to obtain a predictive model. The first possibility is described in Algorithm 1 and uses the overparametrized model for projections. The second model uses the direct estimate for $\boldsymbol{a}$ to estimate an AR(P) model as explained in Alg. 2. Finally another AR(P) model can be obtained by using the estimate for $\boldsymbol{a}$ obtained from the projection. This is outlined in Alg. 3.

**Algorithm 3** AR(P) model with projection based estimate for the noise model (SVD)

**Training:**

1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (D-3) to obtain estimates for $\boldsymbol{\alpha}$, $b$ and $\boldsymbol{a}$
3. compute $\boldsymbol{W}^T\boldsymbol{W}$ according to (7)
4. $\sigma_0, \boldsymbol{v}_0 \leftarrow$ largest eigenvalue and eigenvector of $\boldsymbol{W}^T\boldsymbol{W}$
5. $\hat{\boldsymbol{a}}_{\mathrm{SVD}} \leftarrow \boldsymbol{v}_0/(\boldsymbol{v}_0)_0$
6. compute final model by solving (P-1) given $\hat{\boldsymbol{a}}_{\mathrm{SVD}}$

**Prediction:**

Estimates are generated according to (3)

## IV. Numerical Experiments

All simulations are implemented in Python using Numpy[1]. The RBF kernel is used for all considered models. Model selection is performed using an independent validation set. The regularization parameter $\gamma$ and the kernel bandwidth $\sigma$ are selected using grid search. Performance measures are reported on independent test sets in both cases.

For the synthetic examples we compare the nonlinear systems given in [4].

1) $y_t = f_1(u_t) = 0.2(1 - 6u_t + 36u_t^2 - 53u_t^3 + 22u_t^5) + e_t$ with $u_t$ uniformly distributed on $[-0.5, 1.3]$ and
2) $y_t = f_2(y_{t-1}) = \mathrm{sinc}(y_{t-1}) + e_t$.

The noise term $e_t$ is generated with a linear AR(P) noise model according to

$$e_t = \sum_{p=1}^{P} a_k r_{t-k} + r_t.$$

We consider models of order $P = 2p$ with $p$ pairs of conjugate complex poles on the unit disc and gain one. The excitation signal $r_t$ is white Gaussian noise with standard deviation $\sigma_r = 0.3$.

### A. Model Order Selection

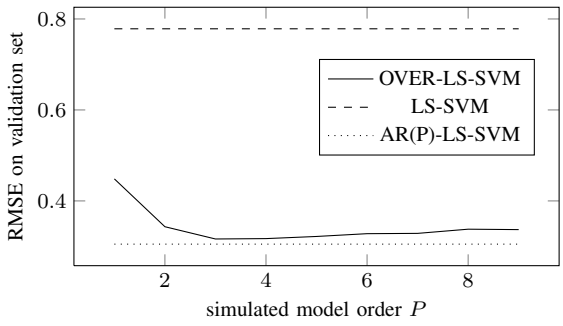
In Figure 2 the validation performance of an over-parametrized model is shown as a function of the model order $P$. The two particular examples are generated for $f_1$ and show that a model order can be selected based on the validation performance. Yet it is not necessarily the case that the true model order is revealed. From our simple experiments it seems that the model order tends to be under estimated.

### B. Correlation of Estimated Parameters with True Noise Model

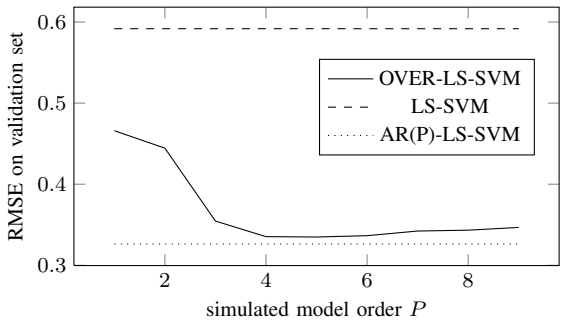
Solving (D-3) we obtain $\hat{\boldsymbol{a}}_{\mathrm{LS}}$ as an estimate for $a_k$. Projecting the model as described in Section III-C yields a second estimate for $a_k$ which we denote as $\hat{\boldsymbol{a}}_{\mathrm{SVD}}$. To assess the quality of the overparametrized model we investigate several quantities

1) angle between estimates $\sphericalangle(\hat{\boldsymbol{a}}_{\mathrm{LS}}, \hat{\boldsymbol{a}}_{\mathrm{SVD}})$,

[1] http://www.scipy.org




(a) true model order $P = 4$




(b) true model order $P = 8$

Fig. 2: Validation performance as a function of the noise model order $P$. Tested for $f_1$. The solid line is the validation performance of an overparametrized model of order $P$. The dashed line gives the performance of an AR(P)-LS-SVM model with the true noise model while the dotted line indicates the performance of a standard LS-SVM model.

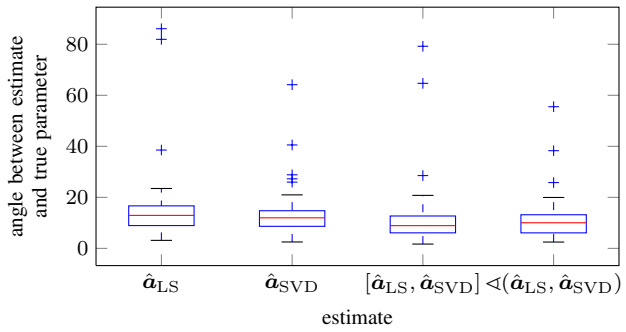
2) angle between true parameters and plane spanned by estimates $\sphericalangle(\boldsymbol{a}, [\hat{\boldsymbol{a}}_{\mathrm{LS}}, \hat{\boldsymbol{a}}_{\mathrm{SVD}}])$ and
3) individual angles between true parameters and estimates $\sphericalangle(\boldsymbol{a}, \hat{\boldsymbol{a}}_{\mathrm{LS}})$, $\sphericalangle(\boldsymbol{a}, \hat{\boldsymbol{a}}_{\mathrm{SVD}})$.

50 Monte Carlo simulations with different realizations of the noise model for orders $P = 4$ and $P = 8$ are shown Figures 3 and 4. The former depicts results for $f_1$ while the latter shows results obtained with $f_2$. Especially for the lower order models the correlation of the different quantities are mostly below 10 degrees. Even for a model order of $P = 8$ for a lot of runs the correlations are still in a meaningful range. It seems that with the overparametrized formulation, the true noise model coefficients cannot be recovered. Yet the approximation is good enough to obtain predictive models that significantly outperform standard LS-SVM as shown in the next section.

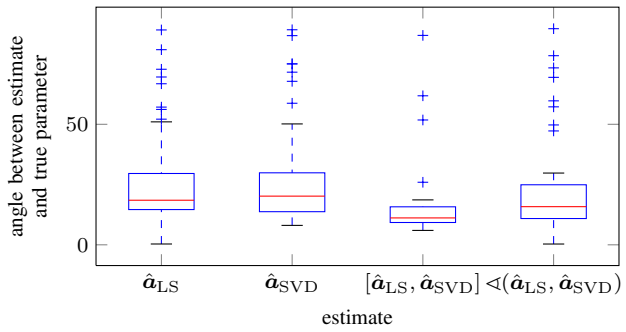
### C. Performance of Projected Models

We consider the same experiments as in the previous section but now compare prediction performances of

1) standard LS-SVM without noise model,
2) AR(P)-LS-SVM given the true noise model (AR(P)),
3) overparametrized LS-SVM (OVER, Alg. 1),
4) AR(P)-LS-SVM with $\hat{\boldsymbol{a}}_{\mathrm{LS}}$ estimates (DIRECT, Alg. 2),
5) AR(P)-LS-SVM with $\hat{\boldsymbol{a}}_{\mathrm{SVD}}$ estimates (SVD, Alg. 3).

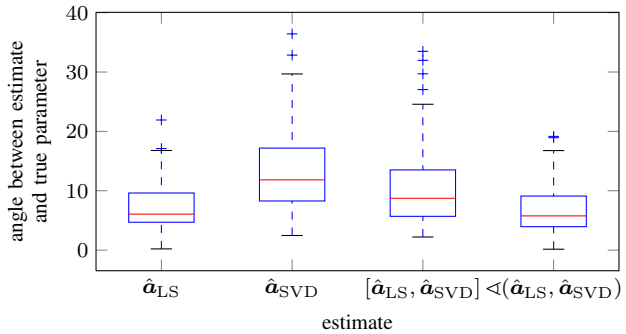



(a) true model order $P = 4$

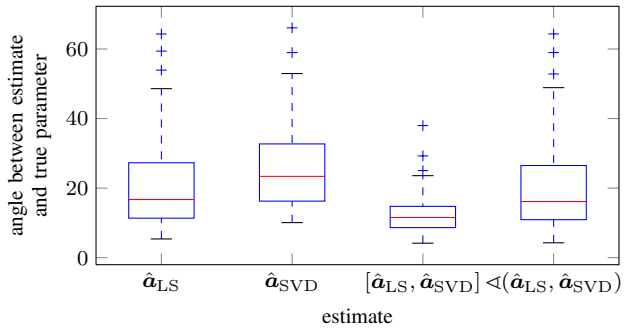



(b) true model order $P = 8$

Fig. 3: Correlation of true noise model parameters $\boldsymbol{a}$ with estimates $\hat{\boldsymbol{a}}_{\mathrm{LS}}$ and $\hat{\boldsymbol{a}}_{\mathrm{SVD}}$ based on 50 Monte Carlo simulations for $f_1$.

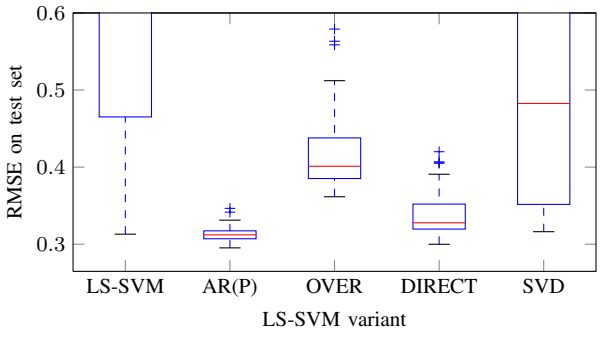



(a) true model order $P = 4$

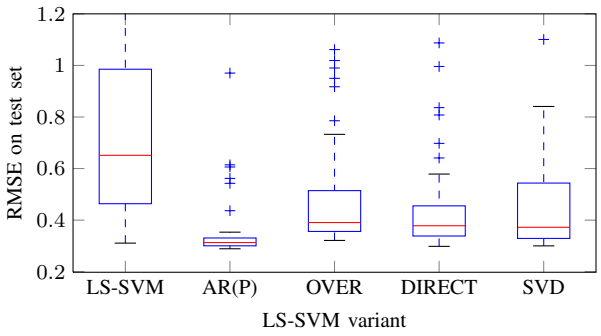



(b) true model order $P = 8$

Fig. 4: Correlation of true noise model parameters $\boldsymbol{a}$ with estimates $\hat{\boldsymbol{a}}_{\mathrm{LS}}$ and $\hat{\boldsymbol{a}}_{\mathrm{SVD}}$ based on 50 Monte Carlo simulations for $f_2$.




(a) nonlinearity $f_1$




(b) nonlinearity $f_2$

Fig. 5: Performance of different model structures (cf. Sec. IV-C) evaluated for different nonlinearities in 50 Monte Carlo runs. The true noise model order is $P = 8$.

Results for Monte Carlo simulations are shown in Figures 5.

We can observe that the AR(P)-LS-SVM significantly outperforms standard LS-SVMs in a lot of cases. The overparametrized model is much better than LS-SVMs but does not perform as well as AR(P)-LS-SVM with the true parameters. For the projected model we observe that the estimate obtained by (D-3) is much more reliable than the obtained by the rank one approximation. In most cases the projected model slightly outperforms the overparametrized model.

### *D. Projection Quality*

For the models evaluated in the previous sections, we can also analyse the quality of the projection step. As a measure to asses how close $\boldsymbol{W}$ is to rank-1 we propose $s_0/\sqrt{\sum_{k=0}^{P} s_k^2}$ the ratio of the largest singular value over the energy in all singular values. Thus a value close to one in figure 6 corresponds to a matrix that is close to rank one. We can conclude that most of the energy is successfully concentrated in the largest singular value.

### *E. Real Data*

We consider the second data set from the ESTSP08 benchmark [17]. The data set has one variable and contains 1300 hourly measurements of internet traffic in an academic network. We train a standard LS-SVM model with $\boldsymbol{x}_t = [y_{t-1}, \ldots, y_{t-m}]^T$ for $m = 1, \ldots, 30$ and select $m = 17$ as

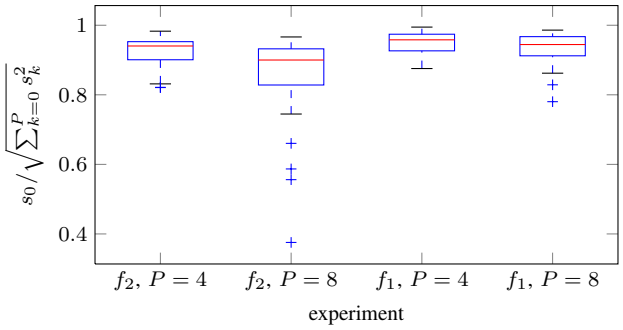



Fig. 6: Quality measure for the rank of $\boldsymbol{W}$. Values close to one indicate a solution dominated by the largest singular value. Results are given for both nonlinear models and different noise model orders and compared for 50 Monte Carlo simulations.

TABLE I: Test performance for ESTSP08 [17] data set 2.

| model order $P$ | RMSE on test set |
|---|---|
| 0 (LS-SVM) | 0.2195 |
| 6 | 0.2156 |
| 8 | 0.2103 |
| 10 | 0.2095 |
| 12 | 0.2073 |
| 14 | 0.1992 |

the order with the smallest validation error. Table I compares the performance on the last 10% of the data. These have not been used for estimating or selecting the model. It can be seen that the performance on this independent test set can be improved by considering a noise model. An open problem that requires further work is model order selection for the noise model. In [18] cross-validation is considered for the case of correlated errors. [19] uses auto- and crosscorrelations to select candidate noise models. The latter approach is more likely to lead to sparse models.

## V. Conclusions

We showed how to integrate a noise model with LS-SVM based models and that doing so is beneficial in the presence of colored noise. For the case that the noise model is not known apriori we proposed a novel convex relaxation based on overparametrization to solve the otherwise non-convex problem. This makes it viable to identify high order noise models without a significant increase in computational complexity. The identified coefficients of the noise model clearly deviate from the true parameters. Nevertheless the prediction capability of the identified models is superior to standard LS-SVM and can. in some cases, come close to the performance of a model given the true noise model parameters. Finally we demonstrated the applicability on two real world data sets.

## Acknowledgements

Research supported by Research Council KUL: GOA AMBioRICS, GOA MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects G.0452.04 (new quantum algorithms), G.0499.04 (Statistics), G.0211.05 (Nonlinear), G.0226.06 (cooperative systems and optimization), G.0321.06 (Tensors), G.0302.07 (SVM/Kernel), G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine); research communities (ICCoS, ANMMM, MLDM); G.0377.09 (Mechatronics MPC); IWT: PhD Grants, McKnow-E, Eureka-Flite+, SBO LeCoPro, SBO Climaqs, POM; Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011); EU: ERNSI; FP7-HD-MPC (INFSO-ICT-223854), COST intelliCIS, EMBOCOM; Contract Research: AMINAL; Other: Helmholtz: viCERP, ACCM, Bauknecht, Hoerbiger.

Johan Suykens is a professor and Bart De Moor is a full professor at the Katholieke Universiteit Leuven, Belgium.

## References

[1] L. Ljung, *System identification: Theory for the User*. Prentice Hall PTR Upper Saddle River, NJ, USA, 1999.

[2] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjoberg, and Q. Zhang, "Nonlinear black-box models in system identification: Mathematical foundations," *Automatica*, vol. 31, pp. 1725–1750, December 1995.

[3] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, pp. 1691–1724, December 1995.

[4] M. Espinoza, J. A. K. Suykens, and B. De Moor, "LS-SVM Regression with Autocorrelated Errors," in *Proc. of the 14th IFAC Symposium on System Identification (SYSID)*, Newcastle, Australia, March 2005, pp. 582–587.

[5] V. N. Vapnik, *Statistical Learning Theory*. John Wiley & Sons, 1998.

[6] B. Schölkopf and A. J. Smola, *Learning with Kernels*. MIT Press Cambridge, Mass, 2002.

[7] G. Wahba, *Spline Models for Observational Data*. SIAM, 1990.

[8] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. Springer, 2006.

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

[10] J. A. K. Suykens, J. De Brabanter, L. Lukas, and J. Vandewalle, "Weighted least squares support vector machines: robustness and sparse approximation," *Neurocomputing*, vol. 48, no. 1-4, pp. 85–105, 2002.

[11] F. H. I. Chang and R. Luus, "A noniterative method for identification using Hammerstein model," *IEEE Transactions on Automatic Control*, vol. 16, no. 5, pp. 464–468, 1971.

[12] E.-W. Bai, "An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems," *Automatica*, vol. 34, no. 3, pp. 333–338, 1998.

[13] I. Goethals, K. Pelckmans, J. A. K. Suykens, and B. De Moor, "Subspace identification of Hammerstein systems using least squares support vector machines," *IEEE Transactions on Automatic Control*, vol. 50, pp. 1509–1519, October 2005.

[14] T. Falck, K. Pelckmans, J. A. K. Suykens, and B. De Moor, "Identification of Wiener-Hammerstein Systems using LS-SVMs," in *Proceedings of the 15th IFAC Symposium on System Identification (SYSID 2009)*, Saint-Malo, France, 2009, pp. 820–825.

[15] D. J. C. MacKay, "Comparison of approximate methods for handling hyperparameters," *Neural Computation*, vol. 11, pp. 1035–1068, 1999.

[16] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[17] A. Lendasse, T. Honkela, and O. Simula, "European symposium on time series prediction," *Neurocomputing*, to appear, 2010.

[18] K. De Brabanter, J. De Brabanter, J. A. K. Suykens, and B. De Moor, "Kernel Regression with Correlated Errors," in *Proceedings of the 11th Symposium on Computer Applications in Biotechnology*, Leuven, Belgium, 2010.

[19] M. Espinoza, J. A. K. Suykens, R. Belmans, and B. De Moor, "Electric Load Forecasting - Using kernel based modeling for nonlinear system identification," *IEEE Control Systems Magazine*, vol. 27, pp. 43–57, 2007.