[2] A. Juditsy, 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, 1995.
[3] S. Chen and S. A. Billings, "Orthogonal least squares methods and their application to nonlinear system identification," *Int. J. Control*, vol. 50, pp. 1873–1896, 1989.
[4] ——, "Representations of nonlinear systems: The NARMAX model," *Int. J. Control*, vol. 49, pp. 1013–1032, 1989.
[5] G.-O. Glentis, P. Koukoulas, and N. Kalouptsidis, "Efficient algorithms for Volterra system identification," *IEEE Trans. Signal Process.*, vol. 47, no. 11, pp. 3042–3057, Nov. 1999.
[6] L. Piroddi and W. Spinelli, "An identification algorithm for polynomial NARX models based on simulation error minimization," *Int. J. Control*, vol. 76, no. 17, pp. 283–295, 2001.
[7] F. Previdi and M. Lovera, "Identification of a class of nonlinear parametrically varying models," *Int. J. Adapt. Control Signal Process.*, vol. 17, pp. 33–50, 2003.
[8] J. Schoukens, J. G. Nemeth, P. Crama, Y. Rolain, and R. Pintelon, "Fast approximate identification of nonlinear systems," *Automatica*, vol. 39, no. 7, pp. 1267–1274, 2003.
[9] R. Murray-Smith and T. A. Johansen, *Multiple Model Approaches to Modeling and Control*. London, U.K.: Taylor and Francis, 1997.
[10] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, 1999.
[11] J. P. Jones and S. Billings, "Mean levels in nonlinear analysis and identification," *Int. J. Control*, vol. 58, no. 5, pp. 1033–1052, 1993.
[12] I. Leontaritis and S. Billings, "Input output parametric models for non linear systems," *Int. J. Control*, vol. 41, pp. 303–344, 1985.
[13] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. New York: Wiley, 1980.
[14] W. Rugh, *Nonlinear System Theory: The Volterra/Wiener Approach*. Baltimore, MD: Johns Hopkins Univ. Press, 1981.
[15] S. Billings and K. Tsang, "Spectral analysis for nonlinear systems, part I: Parametric nonlinear spectral analysis," *Mech. Syst. Signal Process.*, vol. 3, no. 4, pp. 319–339, 1989.
[16] S. Boyd, L. Chua, and C. Desoer, "Analytical foundations of Volterra series," *IMA J. Math. Control Inform.*, vol. 1, pp. 243–282, 1984.
[17] S. Billings, S. Chen, and M. Korenberg, "Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator," *Int. J. Control*, vol. 49, pp. 2157–2189, 1989.
[18] L. Aguirre and S. Billings, "Improved structure selection for nonlinear models based on term clustering," *Int. J. Control*, vol. 62, pp. 569–587, 1995.

Kernel Based Partially Linear Models and Nonlinear Identification

Marcelo Espinoza, Johan A. K. Suykens, and Bart De Moor

***Abstract*—In this note, we propose partially linear models with least squares support vector machines (LS-SVMs) for nonlinear ARX models. We illustrate how full black-box models can be improved when prior information about model structure is available. A real-life example, based on the Silverbox benchmark data, shows significant improvements in the generalization ability of the structured model with respect to the full black-box model, reflected also by a reduction in the effective number of parameters.**

***Index Terms*—Kernels, least squares support vector machine (LS-SVM), nonlinear system identification, partially linear models.**

I. Introduction

The objective of nonlinear system identification is to estimate a relation between inputs $u(t)$ and outputs $y(t)$ generated by an unknown dynamical system. Let $z(t)$ be the regression vector corresponding to the output $y(t)$ in a NARX model, $z(t) = [y(t-1); \ldots; y(t-p); u(t); u(t-1); \ldots; u(t-q)]$. The nonlinear system identification task is to estimate a function g such that $y(t) = g(z(t)) + e(t)$, where $e(t)$ is a disturbance term. Under the nonlinear black box setting, the function g can be estimated by using different techniques (artificial neural networks [6], (least-squares) support vector machines ((LS)-SVM) [14], [15], wavelets [18], basis expansions [12], splines [16], polynomials [7], etc., involving a training and validation stage [14]. The one-step-ahead prediction is simply $\hat{y}(t) = \hat{g}(z(t))$ using the estimated $\hat{g}$; simulation n-steps ahead can be obtained by iteratively applying the prediction equation replacing future outputs by its predictions [8], [12]. In this note, we focus on the case where the nonlinearity in the model does not apply over all the inputs, rather a subset of it, leading to the identification of a partially linear model [5], [10], [13]. We propose the partially linear LS-SVM (PL-LSSVM) [2] to improve the performance of an existing black-box model when there is evidence that some of the regressors in the model are linear. Consider, e.g., a system for which the true model can be described as $y(t) = 0.5y(t-1)^3 - 0.5y(t-2) + 0.3u(t) - 0.2u(t-1) + e(t)$. For this model, the regression vector is defined by $z(t) = [y(t-1); y(t-2); u(t); u(t-1)]$. If a linear parametric model is estimated, it may have a misspecification error because the nonlinear term may not be known a priori to the user. However, by moving to a full nonlinear black-box technique, the linear parts are not identified as such, yet as a part of the general nonlinear model. In this case, a nonlinear black-box model will show a better performance than a full linear model, but it has a complexity which may be

Manuscript received May 28, 2004; revised March 15, 2005 and May 30, 2005. Recommended by Guest Editor L. Ljung. This work was supported by grants and projects for the Research Council K. U. Leuven (GOA-Mefisto 666, GOA-Ambiorics, several Ph.D./Postdocs and Fellow grants), the Flemish Government (FWO: Ph.D./Postdocs Grants, Projects G.0240.99, G.0407.02, G.0197.02, G.0211.05, G.0141.03, G.0491.03, G.0120.03, G.0452.04, G.0499.04, ICCoS, ANMMM; AWI; IWT: Ph.D. Grants, GBOU (McKnow, Soft4s), the Belgian Federal Government (Belgian Federal Science Policy Office: IUAP V-22; PODO-II (CP/01/40), the EU (FP5-Quprodis; ERNSI, Eureka 2063-Impact; Eureka 2419-FLiTE) and Contracts Research/Agreements (ISMC/IPCOS, Data4s, TML, Elia, LMS, IPCOS, Mastercard).

The authors are with the Department of Electrical Engineering ESAT-SCD, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium (e-mail: marcelo.espinoza@esat.kuleuven.be; johan.suykens@esat.kuleuven.be; bart.demoor@esat.kuleuven.be).

Digital Object Identifier 10.1109/TAC.2005.856656

0018-9286/$20.00 © 2005 IEEE

Authorized licensed use limited to: Katholieke Universiteit Leuven. Downloaded on October 14, 2008 at 07:31 from IEEE Xplore. Restrictions apply.

larger than required. The identification of a partially linear model may not only improve the model complexity and generalization ability for this case, but it also may enhance the interpretability of the nonlinear component [4]. We illustrate considerable improvements by following this methodology on the Silverbox benchmark problem. This note is organized as follows. Section II contains some preliminaries. Section III presents the derivation of the PL-LSSVM for use with NARX models. Section IV applies the PL-LSSVM as a tool for empirical model structure identification to different examples.

II. Preliminaries

Let $z(t)$ be the regression vector corresponding to the output $y(t)$ in a NARX model. Let $\mathcal{Z} = \{\boldsymbol{x} : \boldsymbol{x}$ is a component of the vector $z(t)\}$. Define an arbitrary partition $\mathcal{Z} = \mathcal{Z}^a \cup \mathcal{Z}^b$ with $\mathcal{Z}^a \cap \mathcal{Z}^b = \emptyset$. Define a vector $\boldsymbol{z}^a(t)$ with regressors $\boldsymbol{x} \in \mathcal{Z}^a$, and a vector $\boldsymbol{z}^b(t)$ with regressors $\boldsymbol{x} \in \mathcal{Z}^b$. The superscript a (respectively, b) represents the subset of regressors that enters linearly (respectively, nonlinearly) into the model. The original regression is partitioned as $z(t) = [z^a(t); \boldsymbol{z}^b(t)]$. Let $\mathcal{M}_0$ denote the model class containing all models estimated by a nonlinear black-box model over the full set of regressors with orders (p, q) of the form

$$y(t) = g(\boldsymbol{z}(t)) + e(t). \tag{1}$$

We denote by $\mathcal{M}_a$ the model class containing all models of the form

$$y(t) = \boldsymbol{\beta}^T \boldsymbol{z}^a(t) + g(\boldsymbol{z}^b(t)) + e(t). \tag{2}$$

Class $\mathcal{M}_a$ is a more restrictive class of models than $\mathcal{M}_0$. The condition $\mathcal{Z}^a \cap \mathcal{Z}^b = \emptyset$ is required, as pointed out in [13], because when using the same regressor in both components the linear coefficient $\boldsymbol{\beta}$ is not uniquely represented without any further constraints on g. For example, the system $y(t) = u(t-1)^3 + u(t-1)$ can be identified as $y(t) = g(u(k-1)) + \rho u(k-1)$ with $g(x) = x^3 + (1-\rho)x$ for any ρ because g is, in principle, allowed to contain a linear term itself.

III. Partially Linear LS-SVM for System Identification

Consider the model $y(t) = \boldsymbol{\beta}^T \boldsymbol{z}^a(t) + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}^b(t)) + c + e(t)$ for $t = 1, \ldots, N$, where $\boldsymbol{z}^a(t) \in \mathbb{R}^{N_a}$, $\boldsymbol{\beta} \in \mathbb{R}^{N_a}$, $\boldsymbol{z}^b(t) \in \mathbb{R}^{N_b}$, and c is a constant (bias) term. The $e(t)$ values are assumed to be i.i.d. random errors with zero mean and constant (finite) variance. The nonlinear mapping $\boldsymbol{\varphi} : \mathbb{R}^{N_b} \to \mathbb{R}^{N_h}$ is called the feature map from the input space to the so-called feature space (of dimension N_h which can be possibly infinite). In the area of support vector machines and kernel-based learning this feature map is used in relation to a Mercer kernel [14], [15]. As with standard LS-SVM, a constrained optimization problem is formulated (primal problem)

$$\begin{aligned} \min_{\boldsymbol{w},c,e(t),\boldsymbol{\beta}} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \gamma\frac{1}{2}\sum_{t=1}^{N} e(t)^2 \\ \text{s.t. } & y(t) = \boldsymbol{\beta}^T \boldsymbol{z}^a(t) + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}^b(t)) + c + e(t), \\ & t = 1, \ldots, N \end{aligned} \tag{3}$$

where γ is a regularization constant.

Lemma I: Given the problem (3) and a positive–definite kernel function $K : \mathbb{R}^{N_b} \times \mathbb{R}^{N_b} \to \mathbb{R}$, the solution to (3) is given by the dual problem

$$\left[\begin{array}{c|c|c} \boldsymbol{\Omega} + \gamma^{-1}\mathbf{I} & \mathbf{1} & \boldsymbol{Z} \\ \hline \mathbf{1}^T & 0 & \mathbf{0}_{1\times N_a} \\ \hline \boldsymbol{Z}^T & \mathbf{0}_{N_a\times 1} & \mathbf{0}_{N_a\times N_a} \end{array}\right] \left[\begin{array}{c} \boldsymbol{\alpha} \\ \hline c \\ \hline \boldsymbol{\beta} \end{array}\right] = \left[\begin{array}{c} \boldsymbol{y} \\ \hline 0 \\ \hline \mathbf{0}_{N_a\times 1} \end{array}\right] \tag{4}$$

where $\boldsymbol{Z} = [\boldsymbol{z}^a(1)^T; \boldsymbol{z}^a(2)^T; \ldots; \boldsymbol{z}^a(N)^T] \in \mathbb{R}^{N\times N_a}$ is the matrix of linear regressors; $\boldsymbol{y} = [y(1); y(2); \ldots; y(N)]$, and $\boldsymbol{\Omega}$ is the kernel matrix with $\boldsymbol{\Omega}_{i,j} = K(\boldsymbol{z}^b(i), \boldsymbol{z}^b(j)) = \boldsymbol{\varphi}(\boldsymbol{z}^b(i))^T \boldsymbol{\varphi}(\boldsymbol{z}^b(j))$ for $i, j = 1, \ldots, N$.

Proof: Consider the Lagrangian of problem (3) $\mathcal{L}(\boldsymbol{w}, c, e(t), \boldsymbol{\beta}; \alpha_t) = (1/2)\boldsymbol{w}^T\boldsymbol{w} + \gamma(1/2)\sum_{t=1}^{N} e(t)^2 - \sum_{t=1}^{N} \alpha_t[\boldsymbol{\beta}^T \boldsymbol{z}^a(t) + \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{z}^b(t)) + c + e(t) - y(t)]$ where the $\alpha_t \in \mathbb{R}$ are the Lagrange multipliers. The conditions for optimality $((\partial\mathcal{L})/(\partial\boldsymbol{w}) = 0, (\partial\mathcal{L})/(\partial c) = 0, (\partial\mathcal{L})/(\partial e(t)) = 0, (\partial\mathcal{L})/(\partial\boldsymbol{\beta}) = 0, (\partial\mathcal{L})/(\partial\alpha_t) = 0)$ are given by $\boldsymbol{w} = \sum_{t=1}^{N} \alpha_t \boldsymbol{\varphi}(\boldsymbol{z}^b(t)), \sum_{t=1}^{N} \alpha_t = 0, \alpha_t = \gamma e(t) (t = 1, \ldots, N), \sum_{t=1}^{N} \boldsymbol{z}^a(t)\alpha_t = 0$ and $y(t) = \boldsymbol{\beta}^T \boldsymbol{z}^a(t) + \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{z}^b(t)) + c + e(t) (t = 1, \ldots, N)$.

By elimination of $\boldsymbol{w}$ and $e(t)$, we obtain $y(t) = \boldsymbol{\beta}^T \boldsymbol{z}^a(t) + \sum_{k=1}^{N} \alpha_k K(\boldsymbol{z}^b(k), \boldsymbol{z}^b(t)) + c + (\alpha_t)/(\gamma)$ with the application of Mercer's theorem [15], $\boldsymbol{\varphi}(\boldsymbol{z}^b(k))^T\boldsymbol{\varphi}(\boldsymbol{z}^b(t)) = K(\boldsymbol{z}^b(k), \boldsymbol{z}^b(t))$ with a positive definite kernel K. Building the kernel matrix $\boldsymbol{\Omega}_{i,j} = K(\boldsymbol{z}^b(i), \boldsymbol{z}^b(j))$ and writing the equations in matrix notation gives the final system (4). □

Remark 1 [Kernel Functions and Their Corresponding Feature Map $\boldsymbol{\varphi}(\cdot)$]: Any positive definite kernel can be chosen. Some typical choices are: $K_{\mathrm{lin}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T\boldsymbol{x}_j$ (linear kernel); $K_{\mathrm{pol}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^T\boldsymbol{x}_j + r)^d$ (polynomial of degree d, with $r > 0$ a tuning parameter); $K_{\mathrm{RBF}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2/\sigma^2)$ (RBF kernel, where σ is a tuning parameter). On the other hand, the mapping $\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{x}$ for $\boldsymbol{x} \in \mathbb{R}^n$ gives the linear kernel; the mapping $\boldsymbol{\varphi}(x) = [1; \sqrt{2}x; x^2]$ for $x \in \mathbb{R}$ gives the polynomial kernel of degree 2. The mapping for the RBF kernel has been shown to be infinite dimensional [15]. The feature map should not be explicitly known in general. Taking a positive–definite kernel guarantees the existence of the feature map.

Remark 2 [Primal and Dual Expressions]: The estimated model becomes $\hat{y}(t) = \hat{\boldsymbol{\beta}}^T \boldsymbol{z}^a(t) + \sum_{k=1}^{N} \alpha_i K(\boldsymbol{z}^b(k), \boldsymbol{z}^b(t)) + c$, expressed in terms of the dual variables α_t, which corresponds to a nonparametric regression [5] on the nonlinear component that contains N coefficients and kernel evaluations. Even for the case of a linear kernel with linear regression in primal space in (3), the expression $\sum_{k=1}^{N} \alpha_k K_{\mathrm{lin}}(\boldsymbol{z}^b(k), \boldsymbol{z}^b(t)) = \sum_{k=1}^{N} \alpha_k \boldsymbol{z}^b(k)^T \boldsymbol{z}^b(t)$ is a nonparametric regression. Working in dual space, therefore, leads to a nonparametric representation of the model. On the other hand, working explicitly with the $\boldsymbol{\varphi}(\cdot)$ mapping in primal space, provides parametric framework where the solution is expressed directly in terms of the coefficient vectors $\boldsymbol{\beta}, \boldsymbol{w}$.

Remark 3 [Estimation in Primal Space and Large Scale Problems]: It is possible to obtain a finite-dimensional approximation $\hat{\boldsymbol{\varphi}}(\cdot)$ of the nonlinear mapping from any kernel matrix $\boldsymbol{\Omega}$ by means of the Nyström technique [17]. The ith component of $\hat{\boldsymbol{\varphi}}(\cdot)$ evaluated at $\boldsymbol{x}$ is obtained as $\hat{\boldsymbol{\varphi}}_i(\boldsymbol{x}) = (N)/(\sqrt{\lambda_i})\sum_{k=1}^{N} u_{ki} K(\boldsymbol{x}_k, \boldsymbol{x})$ where u_{ki} is the kth element of the ith eigenvector of $\boldsymbol{\Omega}$, and λ_i is the $i-$th

TABLE I
Test Set Performance of the Models for Systems 1 and 2

		Mean Squared Error (Test data)	
System	Evaluation	Black-Box model	Partially Linear Model
System 1	Prediction	1.44×10^{-4}	1.08×10^{-4}
	Simulation	2.01×10^{-4}	1.31×10^{-4}
System 2	Prediction	7.66×10^{-4}	2.50×10^{-4}
	Simulation	0.0019	0.0011

Authorized licensed use limited to: Katholieke Universiteit Leuven. Downloaded on October 14, 2008 at 07:31 from IEEE Xplore. Restrictions apply.

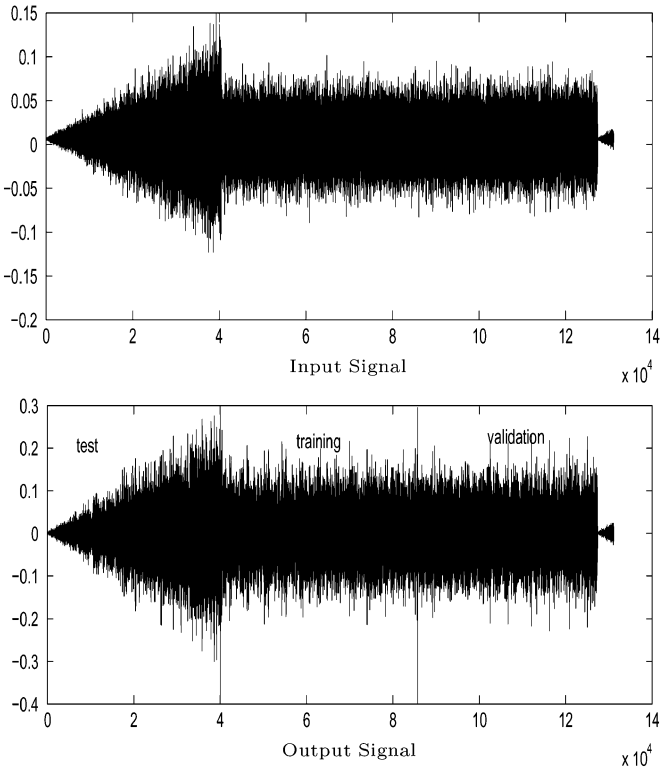


Fig. 1. Input and output signal for the Silverbox data set, showing the regions for training, validation and testing.

eigenvalue. Working in primal space is practical when the number of datapoints N is large and the system in (4) becomes too large for computational storage because it leads to a sparse representation. In this case a smaller matrix $\boldsymbol{\Omega}_M$ of dimensions $M \times M$ with $M \ll N$ computed from a fixed subset of M datapoint provides the starting point to build the approximation $\hat{\boldsymbol{\varphi}}(\cdot)$. This leads to the fixed-size LS-SVM formulation [14]. When working in primal space it is possible to apply all classical parametric techniques for system identification [8].

Remark 4 [Equivalent Smoother Matrix and Effective Number of Parameters]: It is useful to write the vector of predictors from a model in the form $\hat{\boldsymbol{y}} = \mathcal{S}\boldsymbol{y}$, where $\mathcal{S}$ is the smoother matrix. The effective number of parameters (degrees-of-freedom) is given by the trace of $\mathcal{S}$ [9]. For the case of a fully black-box model estimated with LS-SVM, the smoother matrix takes the form $\mathcal{S}_F = \boldsymbol{\Omega}(\boldsymbol{\Omega} + \gamma^{-1}I)^{-1}$ with $\boldsymbol{\Omega}_{i,j} = K(z(i), z(j))$. For the PL-LSSVM model estimated in dual space (4), the smoother matrix takes the form $\mathcal{S}_D = \mathcal{S} - \mathcal{S}\mathcal{W} + \mathcal{W}$ with $\mathcal{S} = \boldsymbol{\Omega}(\boldsymbol{\Omega} + \gamma^{-1}I)^{-1}, \mathcal{W} = \boldsymbol{Z}(\boldsymbol{Z}^T(I - \mathcal{S})\boldsymbol{Z})^{-1}\boldsymbol{Z}^T(I - \mathcal{S})$ and $\boldsymbol{\Omega}_{i,j} = K(z^b(i), z^b(j))$. For the PL-LSSVM estimated using a fixed-size approximation in primal space $\hat{\boldsymbol{\varphi}}(z^b(t)) \in \mathbb{R}^M$ computed from a subsample of size M, let $\hat{\boldsymbol{\Phi}} = [\hat{\boldsymbol{\varphi}}(\boldsymbol{z}^b(1))^T; \ldots; \hat{\boldsymbol{\varphi}}(\boldsymbol{z}^b(N))^T] \in \mathbb{R}^{N \times M}$, the smoother matrix is given by

$$\mathcal{S}_P = [\,\hat{\boldsymbol{\Phi}} \quad \boldsymbol{Z}\,] \left(\begin{bmatrix} \hat{\boldsymbol{\Phi}}^T\hat{\boldsymbol{\Phi}} & \hat{\boldsymbol{\Phi}}^T\boldsymbol{Z} \\ \boldsymbol{Z}^T\hat{\boldsymbol{\Phi}} & \boldsymbol{Z}^T\boldsymbol{Z} \end{bmatrix} + \Lambda \right)^{-1} \begin{bmatrix} \hat{\boldsymbol{\Phi}}^T \\ \boldsymbol{Z}^T \end{bmatrix}$$

with

$$\Lambda = \left[\begin{array}{c|c} \gamma^{-1}\mathbf{I} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right]$$

and $\mathbf{I}$ the identity matrix of dimension M. This smoother matrix is used with respect to generalized cross validation [16].

IV. Examples With NARX Models

A. Model Selection

First, a full black-box model $M_0 \in \mathcal{M}_0$ is estimated with LS-SVM, using cross-validation for hyperparameter tuning and order selection. Then, a PL-LSSVM model $M_1 \in \mathcal{M}_a$ is defined using some initial partition over the regressors. If this new model M_1 has a better performance than M_0, measured as the cross-validation mean squared error (CV-MSE), it is assumed that M_1 is closer to the true structure of the system. The process is repeated over an arbitrary selection of linear regressors, looking for the model $M \in \mathcal{M}_a$ with the best performance. Although this may require a search over all combinations of linear and nonlinear regressors [3], practical insights from the user can reduce the search.

B. Example 1: The True Model Is Known

Consider the following two systems:

$$\begin{aligned} \text{System 1}: y(t) &= (0.5y(t-1))^3 - 0.5y(t-2) \\ &\quad + 0.3u(t) - 0.2u(t-1) + e(t) \\ \text{System 2}: y(t) &= 0.5y(t-1)^3 - 0.5y(t-2)u(t) \\ &\quad + 0.3u(t) - 0.2u(t-1) + e(t). \end{aligned}$$

On each system, the input sequence $u(t)$ is generated from a $N(0,1)$; the error term $e(t)$ is i.i.d. $N(0, 0.001)$. Both systems were generated up to $N = 1000$ datapoints. The first 350 datapoints were discarded to avoid any transient effect; the training was done with the next 400 datapoints; the last 250 points were used for final testing. System 1 contains $[y(t-2); u(t); u(t-1)]$ as a linear input. System 2 considers only $u(t-1)$ as linear due to the condition $\mathcal{Z}^a \cap \mathcal{Z}^b = \emptyset$. First, the overall order of the system is identified using a full black-box NARX model M_0 trained with LS-SVM. The evolution of the CV_MSE(M_0) over different model orders for System 1, where for simplicity we assume $p = q + 1$, gives an optimum at $p = 2$, which corresponds to the regression vector defined as $z(t) = [y(t-1); y(t-2); u(t); u(t-1)]$. The same situation happens for System 2. Once the regression vector has been identified, a sequential application of (3) is performed. We see that even if the original M_0 model was estimated optimally, it is still possible to improve the performance by using a PL-LSSVM correctly specified. For System 1, we have CV_MSE(M_0) $= 2.32 \times 10^{-4}$, which is reduced to CV_MSE(M_1) $= 1.20 \times 10^{-4}$; for System 2, we have CV_MSE(M_0) $= 2.31 \times 10^{-4}$ reduced to CV_MSE(M_1) $= 1.67 \times 10^{-4}$. The effective number of parameters, measured as the trace of the equivalent smoother matrix [9] for System 1 is reduced from 41 to 5.5. For System 2, it is reduced from 39 to 24. We then compute the one-step-ahead error and the simulation error [8] for the original black-box model M_0 and the PL-LSSVM model M_1. The results are reported in Table I. In addition, the model that minimizes the CV-MSE correctly identifies the linear coefficients. For System 1 we obtain $\hat{\boldsymbol{\beta}} = [-0.4996; 0.3002; -0.2002]$. For System 2, $\hat{\boldsymbol{\beta}} = 0.1996$.

C. Example 2: The Silverbox Data

The real-life nonlinear dynamical system that was used in the NOLCOS 2004 Special Session benchmark [11] consists of a sequence of 130 000 inputs and outputs measured from a real physical system, shown in Fig. 1 with the definition of the data that was used for training and final testing. The final test consists on computing a simulation for the first 40 000 datapoints (the " head of the arrow"), which requires the models to generalize on a region of larger amplitude than the one used for training. A full black-box LS-SVM model reached excellent levels of performance [1] using 10 lags of inputs and

Authorized licensed use limited to: Katholieke Universiteit Leuven. Downloaded on October 14, 2008 at 07:31 from IEEE Xplore. Restrictions apply.

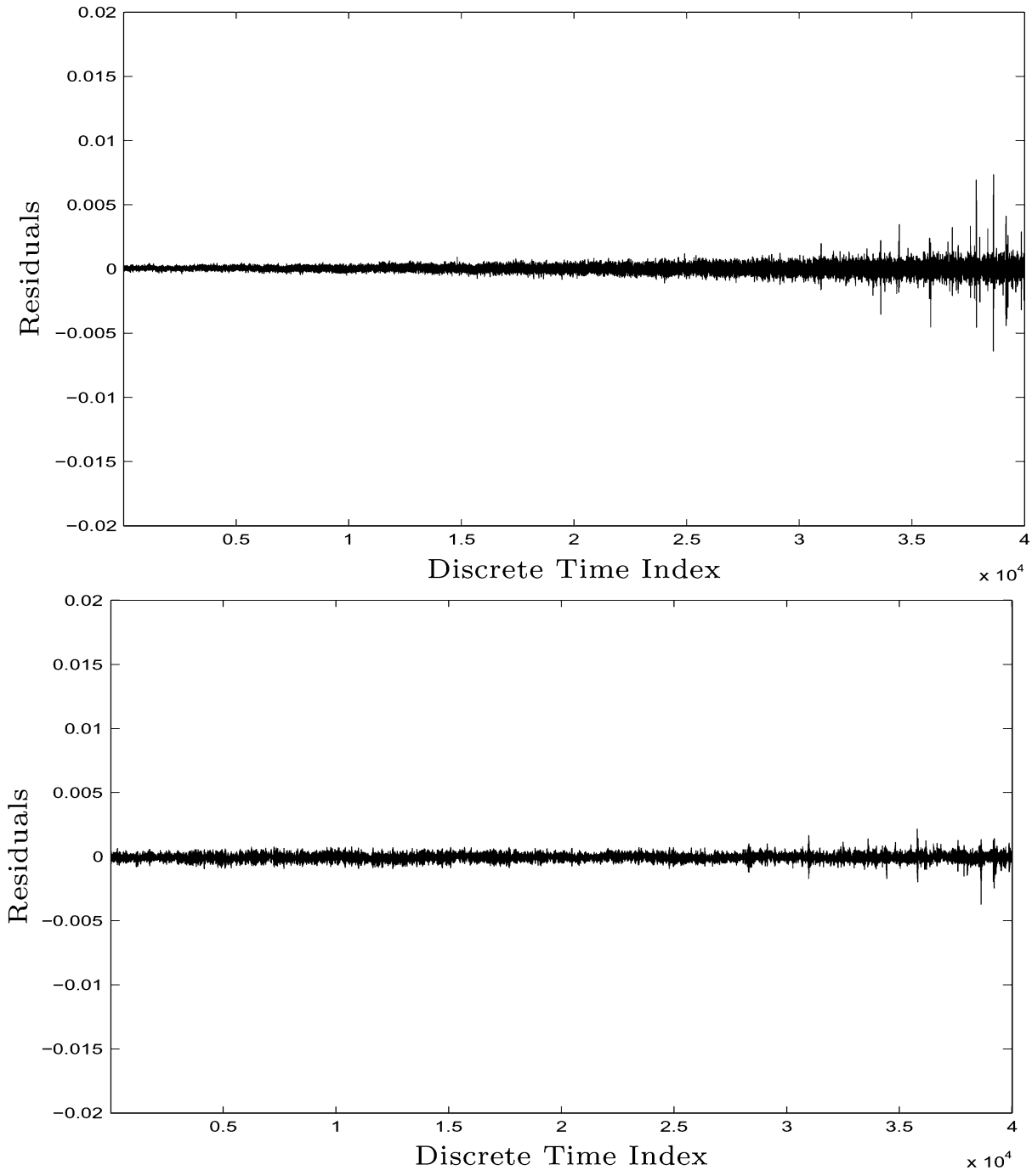


Fig. 2. Simulation errors in the test region of the Silverbox data. Full black-box LS-SVM model (Top) and PL-LSSVM (Bottom).

outputs, obtaining a root mean squared error (RMSE) of 3.24×10^{-4} in simulation mode. Fig. 2 shows the errors obtained in simulation mode when testing the generalization performance of the model in the " head of the arrow" for the full nonlinear black-box LS-SVM. Now the objective is to check if the knowledge of the existence of linear regressors can improve further on the simulation performance. A partially linear model using $p = q = 10$ is formulated using past and current inputs as linear regressors

$$y(t) = \boldsymbol{\beta}^T[u(t); u(t-1); u(t-2); \ldots; u(t-p)] + \boldsymbol{w}^T\boldsymbol{\varphi}([y(t-1); y(t-2); \ldots; y(t-p)]) + e(t)$$

and estimated with PL-LSSVM. Due to the large sample size, a fixed-size PL-LSSVM in primal space is used. It improves the simulation performance over the full black-box model, as it is shown in Fig. 2. Table II shows a comparison between both models in terms of their in-sample accuracy, their validation performance, the simulation accuracy and the model complexity. By imposing a linear structure the simulation root

TABLE II
PERFORMANCE COMPARISON BETWEEN THE MODELS FOR THE SILVERBOX DATA IN TERMS OF RMSE FOR VALIDATION, TESTING AND SIMULATION

	Black-Box model	Partially Linear Model
RMSE_{val}	1.05×10^{-4}	0.45×10^{-4}
$\text{RMSE}_{\text{test}}$	1.75×10^{-4}	0.57×10^{-4}
RMSE_{sim}	3.24×10^{-4}	2.71×10^{-4}
N_{eff}	490	190

Authorized licensed use limited to: Katholieke Universiteit Leuven. Downloaded on October 14, 2008 at 07:31 from IEEE Xplore. Restrictions apply.

TABLE III
SIMULATION ERRORS FOR THE SILVERBOX DATA, OVER THE FULL TEST SET (CASE I) AND ONLY FOR THE LAST 10 000 POINTS OF THE TEST SET (CASE II)

Case	Indicators	Black-Box model	Partially Linear Model
Case I	$\max(\lvert e_i\rvert)$	0.0081	0.0037
	$\text{mean}(\lvert e_i\rvert)$	2.30×10^{-4}	2.02×10^{-4}
	$\text{RMSE}(e_i)$	3.24×10^{-4}	2.71×10^{-4}
Case II	$\max(\lvert e_i\rvert)$	0.0081	0.0037
	$\text{mean}(\lvert e_i\rvert)$	3.72×10^{-4}	2.31×10^{-4}
	$\text{RMSE}(e_i)$	5.86×10^{-4}	3.34×10^{-4}

mean squared error decreases to 2.7×10^{-4}. Moreover, when considering only the last 10 000 points of the test data, the improvement is more important, as shown in Table III. Using the full black-box model, the maximum absolute error is 0.0081, which is reduced to 0.0037 with the PL-LSSVM. The mean absolute error for the full black-box model is 2.3×10^{-4}; for the partially linear model, it is 2.02×10^{-4}. The effective number of parameters is reduced from 490 to 190.

V. CONCLUSION

In this note, we illustrated that it is possible to use a partially linear model with least squares support vector machines to successfully identify a model containing a linear part and a nonlinear component, with better performance results than a full nonlinear black-box model. The structured model may show a better generalization ability, and a reduced effective number of parameters, than a full nonlinear black-box model. In the real-life example of the Silverbox benchmark data, an existing nonlinear black-box model can be further improved by imposing a linear structure, as it is illustrated in the simulation performance.

REFERENCES

[1] M. Espinoza, K. Pelckmans, L. Hoegaerts, J. A. K. Suykens, and B. De Moor, "A comparative study of LS-SVM's applied to the silverbox identification problem," in *Proc. 6th IFAC Symp. Nonlinear Control Systems (NOLCOS)*, 2004, pp. 513–518.

[2] M. Espinoza, J. A. K. Suykens, and B. De Moor, "Partially linear models and least squares support vector machines," in *Proc. 43rd IEEE Conf. Decision and Control*, 2004, pp. 3388–3393.

[3] J. Gao, "Semiparametric nonlinear time series model selection," *J. R. Statist. Soc. B*, vol. 66, pp. 321–336, 2004.

[4] S. R. Gunn and J. S. Kandola, "Structural modeling with sparse kernels," *Mach. Learn.*, vol. 48, pp. 137–163, 2002.

[5] W. Härdle, H. Liang, and J. Gao, *Partially Linear Models*. Heidelberg, Germany: Physica-Verlag, 2000.

[6] S. Haykin, *Neural Networks, a Comprehensive Foundation*. New York: Macmillan, 1994.

[7] D. Lindgren, "Projection techniques for classification and identification," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 2005.

[8] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, 1987.

[9] C. L. Mallows, "Some comments on C_p," *Technometrics*, vol. 15, pp. 661–675, 1973.

[10] P. M. Robinson, "Root n-consistent semiparametric regression," *Econometrica*, vol. 56, no. 4, pp. 931–954, 1988.

[11] J. Schoukens, G. Nemeth, P. Crama, Y. Rolain, and R. Pintelon, "Fast approximate identification of nonlinear systems," *Automatica*, vol. 39, no. 7, 2003.

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

[13] P. Speckman, "Kernel smoothing in partial linear models," *J. R. Statist. Soc. B*, vol. 50, 1988.

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

[15] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[16] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA: SIAM, 1990.

[17] C. K. I. Williams and M. Seeger, "Using the nystrom method to speed up kernel machines," in *Proc. NIPS 2000*, vol. 13, V. Tresp, T. Leen, and T. Dietterich, Eds., Vienna, Austria, 2000.

[18] Y. Yu and W. Lawton, "Wavelet based modeling of nonlinear systems," in *Nonlinear Modeling: Advanced Black Box Techniques*, J. A. K. Suykens and J. Vandewalle, Eds. Norwell, MA: Kluwer, 1998, pp. 119–148.

Model Quality in Identification of Nonlinear Systems

Mario Milanese and Carlo Novara

***Abstract*—In this note, the problem of the quality of identified models of nonlinear systems, measured by the errors in simulating the system behavior for future inputs, is investigated. Models identified by classical methods minimizing the prediction error, do not necessary give "small" simulation error on future inputs and even boundedness of this error is not guaranteed. In order to investigate the simulation error boundedness (SEB) property of identified models, a Nonlinear Set Membership (NSM) method recently proposed by the authors is taken, assuming that the nonlinear regression function, representing the difference between the system to be identified and a linear approximation, has gradient norm bounded by a constant γ. Moreover, the noise sequence is assumed unknown but bounded by a constant ε. The NSM method allows to obtain validation conditions, useful to derive "validated regions" within which to suitably choose the bounding constants γ and ε. Moreover, the method allows to derive an " optimal" estimate of the true system. If the chosen linear approximation is asymptotically stable (a necessary condition for the SEB property), in the present note a sufficient condition on γ is derived, guaranteeing that the identified optimal NSM model has the SEB property. If values of γ in the validated region exist, satisfying the sufficient condition, the previous results can be used to give guidelines for choosing the bounding constants γ and ε, additional to the ones required for assumptions validation and useful for obtaining models with "low" simulation errors. The numerical example, representing a mass-spring-damper system with nonlinear damper and input saturation, demonstrates the effectiveness of the presented approach.**

***Index Terms*—Identification, nonlinear systems, Set Membership, simulation error, stability.**

I. INTRODUCTION

Consider a nonlinear dynamic system of the form

$$y_{t+1} = f_o(w_t) = f_o(x_t, v_t) \tag{1}$$

where: $w_t = (x_t, v_t), x_t = [y_t \dots y_{t-n+1}]^T, v_t = [u_t^1 \dots u_{t-n_1+1}^1 \dots u_t^q \dots u_{t-n_q+1}^q]^T, y_t, u_t^1, \dots, u_t^q \in \mathbb{R}, m = n + n_u, n_u = \sum_{i=1}^{q} n_i, f_o : \mathbb{R}^m \to \mathbb{R}$, and $t = 0, 1, 2, \dots$.

Suppose that the function f_o is not known, but a set of noise corrupted measurements $\widetilde{y}_t$ and $\widetilde{v}_t$, of y_t and $v_t, t = 0, 1, 2, \dots, T$ is available. Then, the aim is to find an estimate $\widehat{f}$ of f_o such that the simulation error for future input sequence is "small."

Most of identification methods in the literature (see, e.g., [1]–[4]) consider that f_o belongs to a finitely parametrized set of functions

Manuscript received May 28, 2004; revised March 15, 2005 and June 1, 2005. Recommended by Guest Editor L. Ljung.

The authors are with the Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy (e-mail: mario.milanese@polito.it; carlo.novara@polito.it).

Digital Object Identifier 10.1109/TAC.2005.856657

0018-9286/$20.00 © 2005 IEEE

Authorized licensed use limited to: Katholieke Universiteit Leuven. Downloaded on October 14, 2008 at 07:31 from IEEE Xplore. Restrictions apply.