State-parameter Dependency Estimation of Stochastic Time Series using Data Transformation and Parameterization by Support Vector Regression

Elvis Omar Jara Alegria, Hugo Tanzarella Teixeira and Celso Pascoli Bottura

Semiconductors, Instruments, and Photonics Department, School of Electrical and Computer Engineering,
State University of Campinas - UNICAMP, Av. Albert Einstein, N. 400 - LE31 - CEP 13081-970, Sao Paulo, Brazil

Keywords: Time Series Identification, State-dependent Parameter, Support Vector Regression.

Abstract: This position paper is about the identification of the dependency among parameters and states in regression models of stochastic time series. Conventional recursive algorithms for parameter estimation do not provide good results in models with state-dependent parameters (SDP) because these may have highly non-linear behavior. To detect this dependence using conventional algorithms, we are studying some data transformations that we implement in this paper. Non-parametric relationships among parameters and states are obtained and parameterized using support vector regression. This way we look for a final non-linear structure to solve the SDP identification problem.

1 INTRODUCTION

The regression models with SDP, called SD-ARX (Priestley, 1988) or quasi-ARX (Hu et al., 2001), are always non-linear due to the product between the regressor function and the SDP. Young proposed a limited and approximated but useful solution to SDP estimation (Young, 2006). It is based in a ascendant temporal data reordering to simplify the estimation process. This is based on the fact that if this state-parameter dependence exists, then both should react the same manner to this data reordering. Young has shown that when the estimated parameters are returned to the normal temporal order, then the state-parameter dependence shape turned evident. Of course, the nature of the reordering will affect the estimation and the ascending order is not always appropriate. A criterion to determine a good sorting could be a function of the minimum variance estimates. But then is it possible to find other data transformations as well as data sorting that could simplify the model response and satisfy the criteria of minimum variance estimates?

The present paper proposes data transformations for SDP estimation. Such as Young's data sorting can simplify the rapid variation, the data transformations is currently studied by us and find to simplify even more the estimation process by bringing the data to a constant value in the new data transformed space.

After to data transformation step, a non-parametric relationship between model parameters and model states is obtained. Various functions approximators are used to parametrize those relationships technics. Widely used function approximators are multilayer perceptrons, radial basis functions and fuzzy models (Nørgaard et al., 2000). Usually the identification of non-linear models is computationally expensive due the fact that the optimization problem is usually non-linear and non-convex. In addition, the designer must trade off the expressiveness of the architecture with the need to maintain computational tractability.

To address these issues, we propose a approximation architecture based on the idea of support vector regression (SVR) (Smola and Schölkopf, 2004). The SVR solution is calculated via a convex program which has a unique optimal solution, and being a kernel-based method, SVR can handle very large numbers of basis functions in a computationally tractable way.

2 SDP IDENTIFICATION

In this paper, the regression linear model with SDP parameter is expressed by equation (1).

$$y(k) = \mathbf{z}^T(k)\rho(k) + e(k); \qquad e(k) = N(0, \sigma^2) \quad (1)$$

where,

$$\mathbf{z}^T(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & \cdots & -y(k-n) \\ u(k-\delta) & \cdots & u(k-\delta-m) \end{bmatrix}$$

$$\rho(k) = \begin{bmatrix} a_1\{\mathcal{X}(k)\} & a_2\{\mathcal{X}(k)\} & \cdots & a_n\{\mathcal{X}(k)\} \\ b_0\{\mathcal{X}(k)\} & \cdots & b_m\{\mathcal{X}(k)\} \end{bmatrix}^T$$

$$\rho(k) = \begin{bmatrix} \rho_1\{\mathcal{X}(k)\} & \rho_2\{\mathcal{X}(k)\} & \cdots & \rho_{n+m+1}\{\mathcal{X}(k)\} \end{bmatrix}^T$$

The regression vector $\mathbf{z}(k) \in \mathbb{R}^{n+m+1}$ is composed of the measurement $y(k) \in \mathbb{R}$ and of the exogenous input $u(k) \in \mathbb{R}$. The SDP $\rho_i \in \mathbb{R}$, $i = 1, 2, ...n+m+1$, related to the parameters a_i or b_i, are assumed to be functions of some variable in a non-minimal state vector $\mathcal{X}^T(k) = \begin{bmatrix} \mathbf{z}^T(k) & \mathbf{U}^T(k) \end{bmatrix}$. Here $\mathbf{U}(k)$ is a vector of other variables that may affect the relationship between these two primary variables, ρ_i and $\mathcal{X}(k)$, e.g. a regressors combination. Also, δ is a pure time delay on the input variable and $e(k)$ is a zero mean, white noise. To simplify, let's suppose that each SDP is a function of only one state variable.

The Simple Fixed Interval Smoothing (SFIS) is a simple but useful estimation algorithm to solve 1. This can be obtained by a combination of the recursive estimate with *forward* data and *backward* data, i.e. with data from sample k to N and from N to k respectively. To allow the TVP in each one of these cases, an exponential windows past (EWP) with a forgetting factor α is used (Alegria, 2015). An optimal value of α can be obtained by the hyper-parameters optimization (Young, 2011). This is the difference between the SFIS and the optimal FIS algorithms.

2.1 Young's Reordering of Data

In this paper the temporal data reordering (TDR) is considered as a temporal data transformation, where only the time ordering is affected. It is yield to simplify the estimation process decreasing the rapid variations on the state values $x(k)$. In this case there are two data observational spaces; the untransformed space and the transformed space. Fig. 1 shows both kinds of observation spaces. In the transformation process, it's important to save an original index vector, because this will be necessary to come back to the original temporal order space or untransformed space.

2.2 Two Proposals of Data Transformations

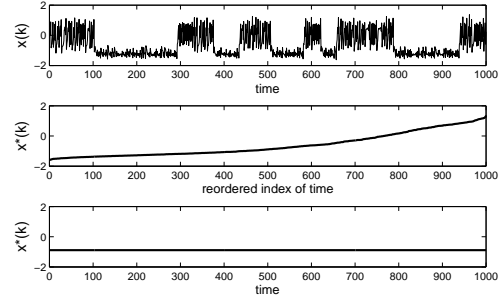Data transformation refers to the application of a known deterministic mathematical function to each

Figure 1: Time series in untransformed space (up), in temporal data reordering TDR (middle) and using data transformation to the mean DTM (below).

point in the data set, i.e. each data point of $\mathcal{X}^j = \begin{bmatrix} x_1^j & x_2^j & \ldots & x_N^j \end{bmatrix}$ is replaced with the transformed value $x_i^* = f(x_i)$, where $f(.)$ is an appropriate mathematical function and $(*)$ represents the transformed space (Dolby, 1963). To return to the untransformed or original data, the inverse function $x_i = f^{-1}(x_i^*)$ is used. The particularity of our proposed transformations is that each point in the data has a different transformation in order to obtain a smoothed transformed value. Differently to the Young's data reordering, our transformed data is ideally smoothed. Where, by definition, ideal smoothing is when $x_i^* = \psi$, $i = 1, 2, ..., N$, where ψ is a constant value. Due to this ideal smoothing, the transformed SDP will also be constant, because of the dependence between the SDP and the state $\mathcal{X}^j$.

A simple linear function $f(x_i) = \beta_i x_i + \theta_i$; $\beta \in \mathbb{R}$, $\theta \in \mathbb{R}$, $i = 1, 2, ..., N$, and its respective inverse function $f^{-1}(x_i^*) = (x_i^* - \theta_i)\beta_i^{-1}$ are used for the data transformations. The function parameters vectors $\beta = \begin{bmatrix} \beta_1 & \beta_2 & \ldots & \beta_N \end{bmatrix}$ and $\theta = \begin{bmatrix} \theta_1 & \theta_2 & \ldots & \theta_N \end{bmatrix}$ are calculated in order to obtain a constant value for $x_i^* = \psi$, e.g. fixing $\beta_i = 1$ then $\theta_i = x_i^* - x_i$. Two different options, equals to the mean value $\psi = E\{\mathcal{X}_j(k)\}$ and to zero $\psi = 0$, are proposed. Fig. 1 (down) shows the data transformation for the model (19), shown in the numerical example, when $\psi = E\{\mathcal{X}_j(k)\}$ is tested.

3 SUPPORT VECTOR REGRESSION

This section provides a basic overview of support vector regression; for more details, see (Smola and Schölkopf, 2004). The objective of the SVR problem is to learn a function.

$$f(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) \quad (2)$$

(a) A tube with radius ε is fitted to the data for a one dimensional linear regression problem.

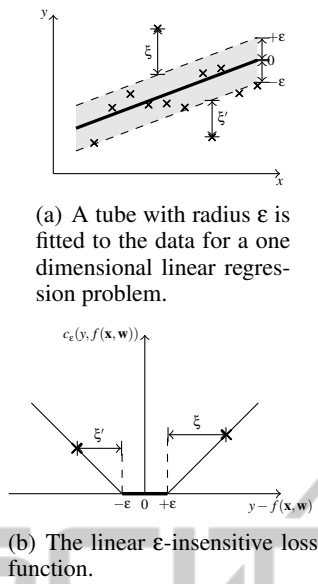(b) The linear ε-insensitive loss function.

Figure 2: The soft margin loss setting for a linear SVR.

that gives a good approximation of a given set of training data $\{\mathbf{x}_i, y_i\}_{i=1}^n$ where $\mathbf{x}_i \in \mathbb{R}^m$ and $y_i \in \mathbb{R}$ is the observed output, $\{\varphi_j(\mathbf{x})\}_{j=1}^{m_b}$ is a set of nonlinear basis functions that maps an input space into a feature space, the parameter vector $\mathbf{w} \in \mathbb{R}^{m_b}$ is unknown. The problem is to compute estimates of $\mathbf{w}$ which minimizes the norm $||\mathbf{w}||^2 = \mathbf{w}^\mathsf{T}\mathbf{w}$. We can write this problem as a convex optimization problem:

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} \\ \text{subject to:} \quad & y_i - \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) \leq \varepsilon \\ & \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - y_i \leq \varepsilon \\ & i = 0, 1, \dots, n \end{aligned} \quad (3)$$

The support vector (SV) method was first developed for pattern recognition. To generalize the SV algorithm to the regression case, an analog of the soft margin is constructed in the space of the observed output y by using Vapnik's ε-insensitive loss function (Vapnik, 1995) described by

$$c(\mathbf{x}, y, f(\mathbf{x})) := |y - f(\mathbf{x})|_\varepsilon := \max\{0, |y - f(\mathbf{x})| - \varepsilon\} \quad (4)$$

Fig. 2 depicts the situation graphically. Only the points outside the shaded region contribute to the cost insofar, as the deviations are penalized in a linear fashion.

Now, we can transform the optimization problem (3) by introducing slack variables, denoted by ξ_i, ξ_i'.

Hence we arrive at the formulation

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} + C\sum_{i=0}^n (\xi_i + \xi_i') \\ \text{subject to:} \quad & y_i - \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) \leq \varepsilon + \xi_i \\ & \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - y_i \leq \varepsilon + \xi_i' \\ & \xi_i, \xi_i' \geq 0 \\ & i = 0, 1, \dots, n \end{aligned} \quad (5)$$

where, the regularization term $\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w}$ penalizes model complexity, and C is a non-negative weight which determines how much prediction errors which exceed the threshold value ε are penalized.

The minimization problem (5) is difficult to solve when the number n is large. To address these issue, one can solve the primal problem through its dual, which can be formulated finding a saddle point of the associated Lagrange function (Vapnik, 1995)

$$\begin{aligned} & L(\mathbf{w}, \xi, \xi', \alpha, \alpha', \beta, \beta') \\ & = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=0}^n (\xi_i + \xi_i') - \sum_{i=0}^n (\beta_i \xi_i + \beta_i' \xi_i') \\ & + \sum_{i=0}^n \alpha_i (y_i - \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - \varepsilon - \xi_i) \\ & + \sum_{i=0}^n \alpha_i' (\mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - y_i - \varepsilon - \xi_i') \end{aligned} \quad (6)$$

which is minimized with respect to $\mathbf{w}$, ξ_i and ξ_i' and maximized with respect to Lagrange multipliers $\alpha_i, \alpha_i', \beta_i, \beta_i' \geq 0$. It fallows from the saddle point condition that the partial derivatives of L with respect to the primal variables $(\mathbf{w}_i, \mathbf{w}_0, \xi_i, \xi_i')$ have to vanish for optimality.

$$\partial_\mathbf{w} L = \mathbf{w} - \sum_{i=1}^n (\alpha_i - \alpha_i')\mathbf{x}_i = 0, \quad (7)$$

$$\partial_{\xi_i} L = C - \alpha_i - \beta_i = 0, \quad (8)$$

$$\partial_{\xi_i'} L = C - \alpha_i' - \beta_i' = 0, \quad (9)$$

$$i = 1, \dots, n$$

$$\partial_{\mathbf{w}_0} L = \sum_{i=0}^n (\alpha_i - \alpha_i') = 0 \quad (10)$$

Substituting (7)–(10) into (6) yields the dual optimization problem.

$$\begin{aligned} \text{maximize:} \quad & -\frac{1}{2}\sum_{i,j=0}^n (\alpha_i - \alpha_i')(\alpha_j - \alpha_j')\varphi(\mathbf{x}_i)^\mathsf{T}\varphi(\mathbf{x}_j) \\ & + \sum_{i=0}^n (\alpha_i - \alpha_i')y_i - \varepsilon\sum_{i=0}^n (\alpha_i + \alpha_i') \\ \text{subject to:} \quad & \sum_{i=0}^n (\alpha_i - \alpha_i') = 0 \\ & 0 \leq \alpha_i, \alpha_i' \leq C \\ & i = 0, 1, \dots, n \end{aligned}$$

$$(11)$$

In deriving (11) we already eliminated the dual variables β_i, β_i' through conditions (8) and (9). Eq. (7) can be rewritten as follows

$$\mathbf{w} = \sum_{i=1}^{n} (\alpha_i - \alpha_i') \varphi(\mathbf{x}_i) \qquad (12)$$

The corresponding Karush-Kuhn-Tucker (KKT) complementarity conditions are

$$\alpha_i(y_i - \mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - \varepsilon - \xi_i) = 0 \qquad (13)$$
$$\alpha_i'(\mathbf{w}^\mathsf{T}\varphi(\mathbf{x}_i) - y_i - \varepsilon - \xi_i') = 0 \qquad (14)$$
$$\xi_i\xi_i' = 0, \alpha_i\alpha_i' = 0 \qquad (15)$$
$$(\alpha_i - C)\xi_i = 0, (\alpha_i' - C)\xi_i' = 0 \qquad (16)$$
$$i = 0, 1, \ldots, n$$

From (13) and (14) it follows that the Lagrange multipliers may be nonzero only for $|y_i - f(\mathbf{x}_i)| \geq \varepsilon$; i.e., for all samples inside the ε-tube (the shaded region in Fig. 2(a)) the α_i, α_i' vanish. This is because when $|y_i - f(\mathbf{x}_i)| < \varepsilon$ the second factor in (13) and (14) is nonzero, hence α_i, α_i' must be zero for the KKT conditions to be satisfied. Therefore we have a sparse expansion of $\mathbf{w}$ in terms of $\mathbf{x}_i$ (we do not need all $\mathbf{x}_i$ to describe $\mathbf{w}$). The samples that come with nonvanishing coefficients are called support vectors. Thus substituting (12) into (2) yields the so-called support vector expansion

$$f(x) = \sum_{i=0}^{n_{sv}} (\alpha_i - \alpha_i')\varphi(\mathbf{x}_i)^\mathsf{T}\varphi(\mathbf{x}) \qquad (17)$$

where n_{sv} is the number of support vectors. Now, a final note must be made regarding the basis function vector $\varphi(\mathbf{x})$. In (11) and (17) it appear only as inner products. This is important, because in many cases a kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^\mathsf{T}\varphi(\mathbf{x}_j)$ can be defined whose evaluation avoids the need to explicitly calculate the vector $\varphi(\mathbf{x})$. This is possible only if the kernel function satisfy the Mercer's condition, for more details see (Schölkopf and Smola, 2001).

3.1 Deriving the State-dependent Parameter Model

Suppose that after the SDP estimation algorithm we have obtained a relationship between a parameter ρ_j and a state $\mathcal{X}^j(k)$. Now, we can obtain a parametric model of this relationship using the SV method. Thus, we can rewrite (17) as

$$\hat{\rho}_j(k) = \sum_{l=0}^{n_{sv}} (\alpha_l - \alpha_l')K(\mathbf{z}_{j,l}, \mathbf{z}_j) \qquad (18)$$

where the Lagrange multipliers α, α' and the number of support vectors n_{sv} are obtained solving the optimization problem (11). Note that the inner product already was replaced by a kernel function.

4 NUMERIC EXAMPLE

Let the SDP model for pure stochastic time series be:

$$y(k) = \mathbf{z}^T(k)\rho(k) + e(k) \qquad (19)$$

where

$$\mathbf{z}^T(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & \cdots & -y(k-n) \end{bmatrix}$$
$$\rho(k) = \begin{bmatrix} a_1\{\mathcal{X}(k)\} & a_2\{\mathcal{X}(k)\} & \cdots & a_n\{\mathcal{X}(k)\} \end{bmatrix}^T$$

The model (19) is a special case of (1). For this specific example, let's study the cosine map model:

$$y(k) = \cos(2.8y(k-1)) + 0.3y(k-2) + e(k), (20)$$
$$y(k) = a_1(k) + a_2(k)y(k-2) + e(k)$$

where $e(k)$ is a zero mean, white noise; the parameter $a_1(k) = \cos(2.8y(k-1))$ is a SDP that depends on the state $\mathcal{X}^1 = y(k-1)$ and the parameter $a_2(k) = 0.3$ is a constant.

This example initially focuses on the nonparametric estimation step of the SDP algorithm shown above. First, using temporal data reordering and an optimal FIS estimation (TDR-FIS) of the CAPTAIN toolbox (Taylor et al., 2007), the Young's algorithm is tested. Next, we only change the optimal FIS by SFIS estimation (TDR-SFIS). Finally our two proposed transformations are tested using SFIS estimation, to the mean (DTM-SFIS) and zero value (DTZ-SFIS), and the results are discussed.

4.1 SDP Estimation using TDR with FIS and SFIS

The data was generated based on the equation (20). Later, the data was sorted to allow the application of the optimal FIS algorithm. The estimation based on TDR-FIS was obtained the CAPTAIN toolbox (Taylor et al., 2007). This is based on temporal data reordering and on the optimal FIS estimation algorithm. The Fig. 3 shows both parameter dependency estimations; the first dependency is well detected. For the second we can conclude that it does not exist, at least for the analyzed state $\mathcal{X}^1 = y(k-1)$.

In the case of estimation based on TDR-SFIS, the result shown in Fig. 4 is obtained in a faster way, but the result is not as good as the previous one.

4.2 SDP Estimation using DTM and DTZ with SFIS

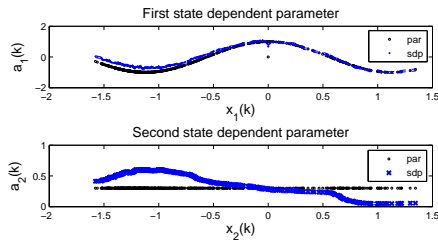First, the estimation is based on DTM-SFIS, i.e doing $\psi = E\{\mathcal{X}_j(k)\}$. Fig. 5 shows the SDP estimation

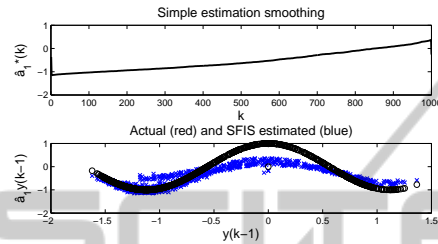Figure 3: SDP-FIS estimation using the CAPTAIN (blue) and reference (black).

Figure 4: SDP-SFIS estimation of $\hat{a}_1^*(k)$ using temporal data reordering (up). Actual a_1 (down-black) and estimated $\hat{a}_1$ (down-blue).
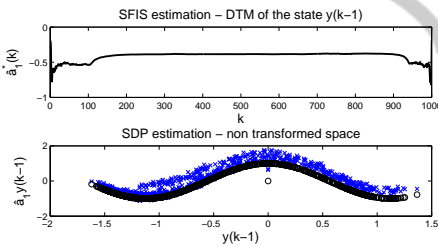

Figure 5: SFIS estimation of $\hat{a}_1^*(k)$ using DTM (up-black). Actual (down-black) and SDP estimated (down-blue).

results in the transformed space (up) and the dependence among the transformed parameter $\hat{a}_1^*$ and his respective state-dependent $y(k-1)$ (down). Comparing these results with the previous two, we can conclude, for this specific example, that the DTM-SFIS case is better than the TDR-SFIS shown in Fig. 4. But it's still not as good as the first one TDR-FIS estimation shown in Fig. 3.

Fig. 6 shows the results for the DTZ-SFIS case when the state values $y(k-1)$ are transformed to zero: $\psi = 0$. It's easy to appreciate that the obtained result is better than the previous one shown in Fig. 5, for $\psi = E\{X_i(k)\}$. Note that using $\psi = 0$ the SDP estimation in the transformed space (up) $\hat{a}_1^*$ is equal to zero such as the initial state reordering $y(k-1)$ in the transformed space. It is an obvious similarity because when both parameters $a_1\{y(k-1)\}$ and $b_0\{u(k)\}$ are zero, in the equation model (19), then the measure $y(k)$ also will be zero in terms of least squares. Then the parameter $\hat{a}_1$ is equals to $y(k)$ and discovering the dependence
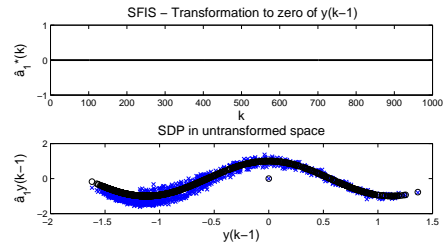

Figure 6: Estimation based on DTZ-SFIS of $\hat{a}_1^*(k)$ using $\psi = 0$ (up-green). Actual (down-black) and SFIS estimated (down-blue).

Table 1: Identification results for numeric example using standard SVR models and SVR models for state-dependent parameters after data transformation.

Method	ε	NSV	MSE $(y - \hat{y})$	MSE $(a_1 - \hat{a}_1)$	
				Test set	Real value
Standard SVR	0.1	613	0.1729	-	-
	0.2	443	0.1778	-	-
	0.3	386	0.1781	-	-
	0.4	257	0.1845	-	-
TDR	0.1	343	0.1560	0.0212	0.1662
	0.2	87	0.1914	0.0219	0.2238
	0.3	63	0.2253	0.0655	0.2796
	0.4	15	0.3341	0.0483	0.3970
DTM	0.1	345	0.2621	0.0336	0.2379
	0.2	193	0.3471	0.0369	0.3090
	0.3	123	0.3595	0.0655	0.3190
	0.4	47	0.3581	0.0515	0.3290
DTM-unbiased	0.1	345	0.0431	-	0.0761
	0.2	193	0.0230	-	0.0743
	0.3	123	0.0099	-	0.0955
	0.4	47	0.0272	-	0.1347
DTZ	0.1	407	0.0344	0.0380	0.0718
	0.2	152	0.0281	0.0376	0.0842
	0.3	92	0.0201	0.0444	0.1011
	0.4	51	0.0297	0.0547	0.1413

between the SDP $\hat{a}_1$ and the state $y(k-1)$ consists in discovering the phase plane between $y(k)$ and the state $y(k-1)$.

The results obtained after SVR application are summarized in Table 1. The model complexity was chosen as $C = 400$ by cross-validation tests for all cases. For comparison, the standard SVR was also applied to problem, where (20) was directly approximate using (17). The accuracy ε, the number of support vectors (NSV) and the mean-square errors (MSE) of the prediction error and the model parameters are shown. In all examples, kernel functions were taken as gaussian kernels, i.e.:

$$K(\mathbf{z}_{j,l}, \mathbf{z}_j) = \exp\left(-\frac{1}{2\sigma^2}||\mathbf{z}_{j,l} - \mathbf{z}_j||^2\right) \qquad (21)$$

where $\sigma = 0.5$. In all cases, the data set were divided in a training data set consisting of 700 data points and a test data set consisting of 300 data points.

We can see in Fig. 7, although the SVR method provides a good fit for covalidation test set, they do not correspond with the actual behavior of the relationship between the parameter and the state, com-
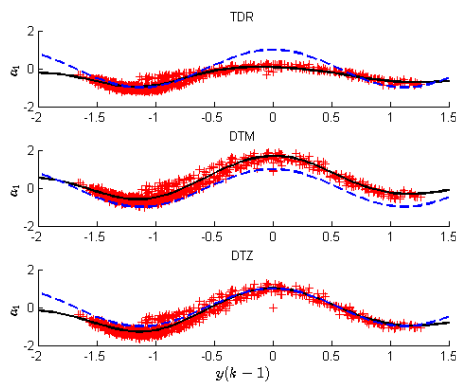
Figure 7: SVR parameterization for $C = 400$ and $\varepsilon = 0.2$ (black solid lines). The real state-parameter relationship (blue dotted lines).

promising the good performance in the estimation of the output.

In DTM case the state values $y(k-1)$ are transformed to its mean, for $\psi = E\{X^i(k)\}$. When the data are reordered some bias appears, as we can see in second plot in Fig. 7. The bias can be subtracted by adding the difference between the mean of the reordered data and the original data. Now DTM-unbiased and DTZ methods results in similar performances to estimate the output, see Table 1.

5 CONCLUSIONS

Parameter estimation methods based on recursive least square are inefficient when the model presents SDP. It is because the parameter and his respective state vary very fast. Young shows an approximated solution based on temporal data reordering and fixed interval smoothing TDR-FIS. This TDR smooths the state and the associated parameter. Both signals, state and parameter, are assumed smoothed since the dependence exists. Numerical example shows that TDR is efficient only with an optimal FIS algorithm instead of an SFIS, see Fig. 3 and Fig. 4. Unlike the TDR, our two proposed transformation brings the data to a constant value. The first brings the state to the mean and the other to zero. These methods are called DTM and DTZ respectively. The supposed constant parameter in the transformed space allow the use of SFIS algorithm instead of an optimal FIS.

Numerical example shows that DTM and DTZ both with SFIS estimation algorithm are equally efficient to detect dependence on the SDP. SVR methods was employed to parametrize and to obtain an error measure for each transformation case. Based on the result of Table 1, we can conclude that DTM-SFIS

and DTZ-SFIS are equally efficient to detect dependence among state and parameter and both are better than the estimation using TDR-SFIS. Based on numerical complexity or computational cost, we also can conclude that the DTZ is better than DTM because a state is converted to zero and it simplifies a lot the algorithm. A final and more accurate model structure was obtained using parametrization based on SVR on the equation 18. Although our results are goods, other examples to demonstrate our practical proposal are necessary, e.g. implement our algorithm in models with exogenous inputs or multi-state parameter dependency.

ACKNOWLEDGEMENTS

REFERENCES

Alegria, E. J. (2015). Off-line state-dependent parameter models identification using simple fixed interval smoothing. In *12th International Conference on Informatics in Control, Automation and Robotics ICINCO*.

Dolby, J. L. (1963). A quick method for choosing a transformation. *Technometrics*, 5:317–325.

Hu, J., Kumamura, K., and Hirasawa, K. (2001). A quasi-ARMAX approach to modeling of nonlinear systems. *International Journal of Control*, 74:1754–1766.

Nørgaard, M., Ravn, O., Poulsen, N. K., and Hansen, L. K. (2000). *Neural Networks for Modelling and Control of Dynamic Systems*. Springer Verlag, London.

Priestley, M. (1988). Non-linear and non-stationary time series analysis. *Academic Press, London*.

Schölkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.

Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14:199–222.

Taylor, C., Pedregal, D., Young, P., and Tych, W. (2007). Environmental time series analysis and forecasting with the captain toolbox. *Environmental Modelling & Software*, 22(6):797–814.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. Springer-Verlag, New York.

Young, P. (2006). Nonlinear system modeling based on non-parametric identification and linear wavelet stimation of sdp models. *45th IEEE Conference on Decision & Control*, pages 2523–2528.

Young, P. C. (2011). *Recursive Estimation and Time-Series Analysis*. Springer-Verlag, 2 edition.