Nonlinear Adaptive Estimation by Kernel Least Mean Square with Surprise Criterion and Parallel Hyperslab Projection along Affine Subspaces Algorithm

Angie Forero and Celso P. Bottura
School of Electrical and Computer Engineering, University of Campinas, Brazil

Keywords: Adaptive Nonlinear Estimation, Machine Learning, Kernel Algorithms, Kernel Least Mean Square, Surprise Criterion, Projection along Affine Subspaces.

Abstract: In this paper the algorithm KSCP (KLMS with Surprise Criterion and Parallel Hyperslab Projection Along Affine Subspaces) for adaptive estimation of nonlinear systems is proposed. It is based on the combination of: - the reproducing kernel to deal with the high complexity of nonlinear systems; -the parallel hyperslab projection along affine subspace learning algorithm, to deal with adaptive nonlinear estimation problem; - the kernel least mean square with surprise criterion that uses concepts of likelihood and bayesian inference to predict the posterior distribution of data, guaranteeing an appropriate selection of data to the dictionary at low computational cost, to deal with the exponential growth of the dictionary, as new data arrives. The proposed algorithm offers high accuracy estimation and high velocity of computation, characteristics that are very important in estimation and tracking online applications.

1 INTRODUCTION

Machine Learning algorithms for nonlinear estimation have been widely exploited in the last years. The supervised machine learning algorithms are capable of producing function approximations and signal predictions only from inputs and reference signals, using past information that has been learned and different kinds of optimization and statistical techniques. The high complexity of nonlinear systems has represented a great challenge, but since the apparition of the Mercer Kernel theorem (Aronszajn, 1950), the kernel properties have played an important role in adaptive learning methods (Kivinen et al., 2004), (Scholkopf and Smola, 2001), enabling the use of well-developed linear techniques in nonlinear problems through the Reproducing Kernel Hilbert Space (RKHS). Many methods have been proposed in the area of kernel adaptive learning such as Kernel Least Mean Square (KLMS) (Liu et al., 2008), Kernel Recursive Least Square (KRLS) (Engel et al., 2004), Kernel Affine Projection Algorithms (KAPA) (Slavakis et al., 2008), Extended Kernel Recursive Least Squares (EX-KRLS) (Haykin et al., 1997). Among these methods the Affine Projection Algorithm from Ozeki and Umeda (Ozeki, 2016) a generalization and improvement of the Normalized LMS algorithm, is characterized by a low complexity like the Least Mean Square (LMS) algorithm and has faster convergence than the Normalized LMS algorithm. However, one of its disadvantage is that as more delayed inputs are used its velocity decreases. Yukawa and Takizawa (Takizawa and Yukawa, 2015) proposed a more evolved version of Affine Projection Methods that uses ideas of projection-along-subspace and parallel projection to reduce the complexity and therefore increase the velocity of performance.

A well-known problem of kernel adaptive filtering methods is the exponential growth of the dictionary: the set of past information learned and stored. As new data arrives, incorporating the new data into the dictionary requires an adequate policy to avoid unnecessary calculations with an increasing number of variables, which results in a problem of high computational cost and low velocity performance, making these algorithms unsuitable for online applications. Many heuristic selection criteria for the dictionary have been proposed like: Approximate Linear Dependency (ALD) (Engel et al., 2004) and Novelty Criterion (Platt, 1991). To solve this problem of dictionary control we propose a solution based on the surprise criterion of Liu (Liu et al., 2009a), as it offers a solid mathematical criterion that evaluates the true significance of the new sample and if it will contribute

Forero, A. and Bottura, C.
Nonlinear Adaptive Estimation by Kernel Least Mean Square with Surprise Criterion and Parallel Hyperslab Projection along Affine Subspaces Algorithm.
DOI: 10.5220/0006867803620368
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018) - Volume 1*, pages 362-368
ISBN: 978-989-758-321-6

to the learning system, deciding whether the new data should be taken or discarded.

The surprise criterion calculates the probability of the posterior distribution of data given the past information learned, using log likelihood and Bayesian inference to evaluate in a certain sense, how related or known the new data can be for the learning system.

The surprise criterion for the conventional Kernel affine projection algorithm represents a high computational cost in the sense that it requires calculating the inversion of the Gram matrix at every instant of time, when only the calculation of the approximation of the prediction variance is needed to decide whether the new data are going to be inserted or not into its dictionary. For this reason, we propose to use the KLMS and the Surprise Criterion which have a low computational cost (it will be shown in section 2.2.1), in fact there are two different problems: 1) establish with some mathematical criterion the dictionary control and 2) calculate the approximation of the nonlinear function. We develop an algorithm which establishes the dictionary control based on ideas of the KLMS-Surprise Criterion algorithm and approximates the nonlinear objective function based on ideas of Parallel hyperslab projection along affine subspace algorithm, achieving with this, a method for adaptive nonlinear estimation with high velocity of performance, high accuracy and low complexity. These results will be shown through a computational experiment and be compared with some other methods. In section 2 the main ideas are discussed, in section 3 the algorithm is presented, the computational experiment is shown in section 4 and finally the conclusions are presented in section 5.

2 MAIN IDEAS

2.1 Problem Definition and Notation

Consider the problem of adaptive estimation of a nonlinear system f, where the input data $u \in \mathbb{U}$ and the signal reference d arrive at each instant of time.

To enable the use of the linear techniques, we deal with the nonlinear estimation in the Reproducing Kernel Hilbert Space (RKHS) where the inputs belong to the domain $\mathbb{U}$ and are mapped into a high-dimensional feature space $\mathbb{F}$. The mapping will be done by the nonlinear function $\varphi(\cdot)$, such that $\varphi : \mathbb{U} \to \mathbb{F}$. Then, with the transformed input $\varphi(u)$, we are able to apply a linear algorithm in order to obtain the estimate of the nonlinear function.

Let $\psi(\cdot)$ be a real function of the Hilbert Space $\mathcal{H}$, there exists a continuous, symmetric, positive-definite function $u_i \to k(u_i,u_j)$ such that $k : \mathbb{U}\text{x}\mathbb{U} \to \mathbb{R}$ associated with it; where $k(\cdot,u_j)$ is the kernel function evaluated in u_j, satisfying the reproducing property

$$\psi(u_j) = <\psi(\cdot),k(\cdot,u_j)>_{\mathcal{H}} \tag{1}$$

From this property of the reproducing kernel and the transformed input, we get:

$$k(u_i,u_j) = <\varphi(u_i),\varphi(u_j)>_{\mathcal{H}} \tag{2}$$

Due to this reproducing kernel property, it is possible to evaluate the kernel by using the inner product operation in the feature space. As the algorithm is formulated in terms of inner products, there is no need to develop calculations in the high-dimensional feature space, a great advantage of the kernel properties.

To obtain the best estimation ψ of the nonlinear system f, a least squares approach for this nonlinear regression problem, looks for the determination of a function $\psi(\cdot)$ that minimizes the sum of squared errors between the reference signals and the output estimator, given by

$$\psi(\cdot) = \sum_{i=1}^{N} h_i k(\cdot,u_i) \tag{3}$$

where h_i is a coefficient of $k(\cdot,u_i)$ at time i.

2.2 Dictionary Control

Before calculating the optimal error solution to estimate iteratively the nonlinear function f, we must decide if the incoming data are significant and therefore they should be learned and inserted in the dictionary or if they should be discarded by being insignificant. For doing this we use as a rule for dictionary control the Surprise Criterion of Liu (Liu et al., 2010), that gives the uncertainty amount of the new data with respect to the current knowledge of the learning system and is defined as follows: Surprise $S_{\mathcal{D}(i)}$ is the negative log likelihood of the new data given the current dictionary $\mathcal{D}_i$:

$$S_{\mathcal{D}_i}(u_{i+1},d_{i+1}) = -log\, p(u_{i+1},d_{i+1}|\mathcal{D}_i) \tag{4}$$

where $p(u_{i+1},d_{i+1}|\mathcal{D}_i)$ is the posterior distribution of $\{u_{i+1},d_{i+1}\}$.

In this sense, if the probability of occurrence of $\{u_{i+1},d_{i+1}|\mathcal{D}_i\}$ is large, it means that the new data are known by the learning system and therefore there is no need to be learned; in the other case, if the probability is small, it means that the new data are unknown by the learning system and they should be learned or they are ”abnormal”, which indicates that they can come from the errors or perturbations.

By using Bayesian inference and assuming all the inputs with a normal distribution, the posterior probability density $p(u_{i+1},d_{i+1}|\mathcal{D}_i$ can be evaluated by

$$\begin{aligned} p(u_{i+1}, d_{i+1}|\mathcal{D}_i) &= p(d_{i+1}|u_{i+1}, \mathcal{D}_i)p(u_{i+1}|\mathcal{D}_i) \\ &= \frac{1}{\sqrt{2\pi}\sigma_{i+1}} exp\left(-\frac{(d_{i+1}-\bar{d}_{i+1})^2}{2\sigma_{i+1}^2}\right) p(u_{i+1}|\mathcal{D}_{i+1}) \end{aligned} \tag{5}$$

and

$$\begin{aligned} S_{i+1} &= -log[p(u_{i+1}, d_{i+1}|\mathcal{D}_{i+1})] \\ &= log\sqrt{2\pi} + log\,\sigma_{i+1} \\ &+ \frac{(d_{i+1}-\bar{d}_{i+1})^2}{2\sigma_{i+1}^2} - log[p(u_{i+1}|\mathcal{D}_i)] \end{aligned} \tag{6}$$

Assuming that the distribution $p(u_{i+1})$ is uniform, the equation (6) can be simplified to

$$S_{i+1} = log\,\sigma_{i+1} + \frac{(d_{i+1}-\bar{d}_{i+1})^2}{2\sigma_{i+1}^2} \tag{7}$$

From the equation above, we can observe that the calculation of the surprise can be deduced directly from the variance σ_{i+1}^2 and the error $e = (d_{i+1} - \bar{d}_{i+1})$.

2.2.1 Kernel Least Mean Square with Surprise Criterion (KLMS-SC)

For kernel adaptive filtering, the KLMS algorithm offers a great advantage in terms of simplicity. The calculation of the prediction variance is simplified by using a distance measure M, an approximation that selects the nearest inputs in the dictionary to estimate the total distance and defined as:

$$M = min_{\forall b \forall c_j \in \mathcal{D}_i} \|\varphi(u_{i+1}) - b\varphi(c_j)\| \tag{8}$$

where c_j is an element of the dictionary and b is a coefficient. Solving the equation 8 we get

$$v_{i+1} = \lambda + k(u_{i+1}, u_{i+1}) - max_{\forall c_j \in \mathcal{D}_i} \frac{k^2(u_{i+1}, c_j)}{k(c_j, c_j)} \tag{9}$$

where v_{i+1} denotes the prediction variance and λ is a regularization parameter. For more details of KLMS-Surprise Criterion see (Liu et al., 2010).

In contrast, if the Kernel Affine Projection Algorithm with surprise criterion were used, we should solve the following problem:

$$M = min_{\forall b \forall n_j \in \mathcal{D}_i} \|\varphi(u_{i+1}) - \sum_{j=1}^{K} b_j \varphi(n_j)\| \tag{10}$$

that results in

$$v_{i+1} = \lambda + k(u_{i+1}, u_{i+1}) - \mathbf{h}_u^T[\mathbf{G}_n + \lambda\mathbf{I}]^{-1}\mathbf{h}_u \tag{11}$$

where $\mathbf{h}_u = [k(u_{i+1}, c_1), \ldots, k(u_{i+1}, c_n)]^T$, $\mathbf{I}$ is the identity matrix and $\mathbf{G}_n$ is the Gram matrix.

From the equation above, we see that the calculation of the prediction variance includes the inversion of the Gram matrix $\mathbf{G}$ for each instant of time. This is very computationally expensive and is not always possible to calculate the inversion of the Gram Matrix. However, we only need to calculate the prediction variance and the prediction error of the new data with respect to the learned system, to decide if the new data are going to be inserted to the dictionary or not. This is independent of the calculation process of the approximation of the gradient direction and of the estimation of the nonlinear objective function. For this reason we are able to establish one method for selection of data (dictionary control) and another method for the nonlinear function estimation, and we can take the advantage of the versatility of the Affine Projection Algorithms. Thus, we propose an algorithm that uses the KLMS and the Surprise Criterion for dictionary control and the parallel hyperslab projection along affine subspace for gradient direction calculation that we call KSCP.

2.3 Parallel Hyperslab Projection along Affine Subspace (ϕ-PASS)

Let ψ_{i+1} be the next estimate of the current estimate ψ_i obtained by its affine projection onto a hyperplane of optimal solutions Π_i defined by

$$\Pi_i := \{\psi; d_i - \langle\psi, k(\cdot, u_i)\rangle = 0\} \tag{12}$$

We suppose that ψ_i belongs to the dictionary $\mathcal{D}_i$ for estimation updating. As it is possible that $\psi_i \notin \mathcal{D}_i$ for the dictionary control, the solution we adopt is the parallel hyperslab projection along affine subspace (ϕ-PASS) by Takizawa and Yukawa (Takizawa and Yukawa, 2013). It is based on projection-along-subspace, where the intersection of the dictionary space $\mathcal{D}_i$ with the hyperplane Π_i is calculated and the current estimate ψ_i is projected onto the intersection $\mathcal{D}_i \cap \Pi_i$, that is equivalent to the projection of ψ_i onto Π_i along $\mathcal{D}_i$.

Thus, once the KLMS with surprise criterion algorithm has established that the new data should be learned, the estimate of the nonlinear function is calculated using the Φ-PASS, enabling in all cases that the estimate be updateable. Another advantage of the Φ-PASS algorithm is the idea of parallel projections, where using the p most recent measurements $(u_l, d_l)_{l \in \mathcal{L}_i}$, (where $\mathcal{L}_i := (i, i-1, ..., i-p+1)$ is the index set of data) at each time and accommodating into the hyperplanes $\Pi_{l \in \mathcal{L}_i}$, the current estimate ψ_i is projected onto these hyperplanes in parallel, and the direction of the next estimate is obtained in the average point of the projections. This approach improves the velocity of convergence and the noise robustness.

3 THE KSCP ALGORITHM

The proposed algorithm KSCP (KLMS with Surprise Criterion and Parallel Hyperslab Projection along Affine Subspace) first establishes a dictionary control which guarantees that only the significant data are going to be used for the estimation, allowing an appropriate use of resources to avoid the unnecessary calculations with redundant data or data from errors or perturbations, improving the calculation velocity of the algorithm at low computational cost using ideas of kernel least mean square. This characteristic is very important in online applications. For calculating the gradient direction and the approximation of the nonlinear function at each instant of time in the adaptive way, we use the projection along affine subspace and parallel projection approaches, achieving high accuracy and fast convergence.
The algorithm works in the following way:

When $\{u_{i+1}, d_{i+1}\}$ arrives, the inputs are transformed using the kernel Gaussian function:

$$k(u_i, u_j) = exp(-\gamma \|u_i - u_j\|^2) \tag{13}$$

wherewith we can use the affine projection technique into the transformed inputs for the treatment of the nonlinear system.

With the result of the kernel evaluation in (13), construct

$$\mathbf{x}_{i+1} = [k(u_{i+1}, c_1), \cdots, k(u_{i+1}, c_j)]^T \tag{14}$$

where c_i is a dictionary element at time i.

The output is calculated using the estimative ψ_i obtained by the parallel hyperslab projection along affine subspace. It should be noted that if the dictionary control establishes that the data should not be learned, there is no need to update the estimative and the last calculated value of the estimative will be taken. The output is obtained by

$$\hat{f}_i(u_{i+1}) = \mathbf{x}_{i+1}^T \psi_i \tag{15}$$

initialized with $\psi_0 = 0$.

The prediction error is calculated by

$$e_{i+1} = d_{i+1} - \hat{f}_i(u_{i+1}) \tag{16}$$

The prediction variance is calculated using the Kernel least mean squares approach

$$v_{i+1} = \lambda + k(u_{i+1}, u_{i+1}) - max_{\forall c_j \in \mathcal{D}_i} \frac{k^2(u_{i+1}, c_j)}{k(c_j, c_j)} \tag{17}$$

which offers a great advantage due to its simplicity, with no need to compute the Gram matrix inversion in the control dictionary step. It avoids the high computational costs for achieving fast performance.

With the predictions of variance and error, we calculate the Surprise Criterion by

$$S_{i+1} = \frac{1}{2} log\, v_{i+1} + \frac{e_{i+1}^2}{2v_{i+1}} \tag{18}$$

Based on the Surprise Criterion we establish the following rule for dictionary control:

If

- $S_{i+1} > T_1 \Rightarrow$ Abnormal and Discarded
- $T_1 \geq S_{i+1} \geq T_2 \Rightarrow$ Learnable
- $S_{i+1} < T_2 \Rightarrow$ Redundant and Discarded

T_1 the abnormality threshold and T_2 the redundancy threshold, are parameters dependent of the problem. It is possible to make a trade-off between more accuracy or more velocity of performance by adjusting the parameters T_1 and T_2.

If the new data are learnable, they will be inserted in the dictionary $\mathcal{D}_i$ and the expansion coefficient ψ_i will be calculated and updated. If not the system just takes the last estimate and returns to the beginning for waiting new data.

With this step we avoid the waste of resources as any aditional calculation would be done on insignificant data. This helps the algorithm to be more effective. It also improves the execution of the approximation of the gradient direction because this approximation involves calculations with matrices that grows exponentially with the dictionary. With this approach we get a dictionary that uses only the essential quantity of data.

If the data are learnable and inserted in the dictionary we are going to project the current estimate ψ_i onto the closed convex set defined by

$$C_l^{(i)} := \mathcal{V}_l^{(i)} \cap \Pi_l \subset \mathcal{D}_i \tag{19}$$

where $\mathcal{V}$ is an affine subspace of the dictionary defined by

$$\mathcal{V}_l^i := span\,(k(\cdot, u_j))_{j \in \mathcal{J}_l^{(i)}} + \psi_i \subset \mathcal{D}_i \tag{20}$$

and $(k(.\,, u_j))_{j \in \mathcal{J}_l^{(i)}}$ is the set of selected elements.

Then, the projection of ψ onto the convex set is given by

$$P_C \psi = \psi + \beta P_{\tilde{D}} k(\cdot, u) \tag{21}$$

where β is calculated by

$$\beta = \varsigma \frac{max(|\, d - \psi(u)\,|, 0)}{\sum_{j \in \mathcal{J}} \alpha_j k(u, u_j)} \tag{22}$$

with the signum function $\zeta(\cdot)$ and

$$P_{\tilde{D}} k(\cdot, u) = \sum_{j \in \mathcal{J}} \alpha_j k(u, u_j) \tag{23}$$

The projection $P_{\tilde{D}}k(\cdot,u)$ is obtained solving the normal equation:

$$\alpha = G^{-1}y \tag{24}$$

where G is the Gram matrix

$$G = \begin{bmatrix} k(u_1,u_1) & k(u_1,u_2) & \cdots & k(u_1,u_n) \\ k(u_2,u_1) & k(u_2,u_2) & \cdots & k(u_2,u_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(u_n,u_1) & k(u_n,u_2) & \cdots & k(u_n,u_n) \end{bmatrix} \tag{25}$$

and

$$y := \begin{bmatrix} k(u,u_1) & k(u,u_2) & \cdots & k(u,u_n) \end{bmatrix}^T \tag{26}$$

Finally the nonlinear estimate is calculated by

$$\psi_{i+1} = \psi_i + \lambda_i \left(\sum_{l \in \mathcal{L}_i} w_l^i P_{C_l^i} \psi_i - \psi_i \right) \tag{27}$$

with $w_l^i \geq 0$ and ψ_i is updated in equation (15)

4 COMPUTATIONAL EXPERIMENT

In this section we show the performance of the proposed algorithm KSCP and compare it with some methods: Kernel Affine Projection with Coherence Criterion (KAP) (Richard et al., 2009), (Saidé et al., 2012) Kernel Least Mean Square with Coherence-Sparsification criterion and Adaptive L1-norm regularization (KLMS-CSAL1) (Gao et al., 2013) and Extended Recursive Least Squares (EXKRLS) (Liu et al., 2009b). For these algorithms we used the toolbox Kafbox (Van Vaerenbergh and Santamaría, 2013)

We employ the benchmark sinc function estimation which is often used for nonlinear regression aplications. The synthetic data are generated by

$$y_i = sinc(x_i) + \nu_i \tag{28}$$

where

$$sinc(x) = \begin{cases} sin(x)/x & x \neq 0 \\ 1 & x = 0 \end{cases} \tag{29}$$

ν_i is a zero-mean Gaussian noise with variance 0.04.

In the simulation we use 1000 samples for training and 500 samples for testing. The estimation results are showed in Fig. 1 and in Table. 1 where the mean square error (MSE) for the estimation for each method is presented. The MSE is calculated by

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{30}$$

where y_i is the reference or desired value and $\hat{y}_i$ is the estimated value.

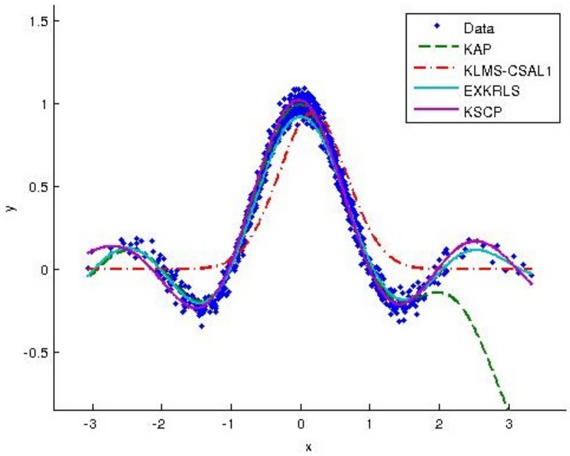

Figure 1: Sinc Function Estimation.

Table 1: Performance of the Algorithms.

Algorithm	Performance	
	MSE (dB)	*Time of Execution (seconds)*
KAP	-10.33	0.31
KLMS-CSAL1	-16.48	0.27
EXKRLS	-29.34	0.88
KSCP	-32.42	0.72

The learning curve of each algorithm is presented in Fig. 2. The parameters setting used by each one of the methods are the following: for KAP the coherence criterion $\mu_0 = 0.95$, the step size $\eta = 0.5$, the regularization term $\lambda = 0.01$, and the kernel parameter $\gamma = 1$;for KLMS-CSAL1 the coherence criterion $\mu_0 = 0.95$, the step size $\eta = 0.1$, the sparsification threshold $\rho = 5x10^{-4}$, and the kernel parameter $\gamma = 0.5$; for EXKRLS the state forgetting factor $\alpha = 0.99$, the data forgetting factor $\beta = 0.99$, the regularization factor $\lambda = 0.01$, the trade-off between modeling variation and measurement disturbance $q = 1x10^{-3}$ and the kernel parameter $\gamma = 1$; for KSCP the abnormality threshold $T_1 = 1$, the redundancy threshold $T_2 = -0.5$, the step size $\eta = 0.5$, the regularization term $\lambda = 0.01$,

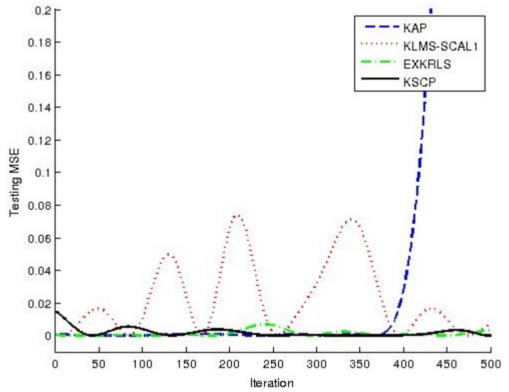

Figure 2: Learning Curve for KAPCC, KLMS-CSAL1 and KSCP.

number of hyperslabs $p = 8$, the weight coefficient $\omega = 0.1250$ and the kernel parameter $\gamma = 1$.

The obtained results lead to the following observations:

- The four algorithms presented in this experiment have a fast performance, taking less than 1 minute in the estimation with 1000 training samples. As is showed in Table I, the KSCP achieves the minimum mean square error, the KAP and the KLMS-SCAL1 achieves more velocity in execution but with a higher mean square error. It is important to highlight that the settings of the T_1 and T_2 parameters are essential for the execution time and the accuracy. The abnormality threshold parameter T_1 can be adjusted for achieving more velocity and the redundancy threshold parameter T_2 must be adequately limited to guarantee high accuracy.
- The learning curve figure showed that in the Kernel Affine Projection algorithm, the mean square error begins to grow exponentially in the iteration number 400; the KLMS-SCAL1 takes several iterations to stabilize and the EXKRLS and KSCP reach zero and remain stable from the first 50 iterations for the considered situation.

5 CONCLUSIONS

This paper presents our proposed algorithm KSCP for adaptive nonlinear estimation. The KSCP improves the estimation performance based on 3 aspects: First, an effective dictionary control is established, guaranteeing an exhaustive selection of the most important data for the estimation, and low computational complexity, basing not on heuristics but on a strong mathematical foundation approach with statistical and probabilistic techniques. Second, we take advantage of the Kernel Least Mean Square and the Surprise Criterion combining them to reduce the complexity of the calculations of variance prediction and error prediction of the incoming data. Third, for the high accuracy goal, we use the Parallel Hyperslab Projection Along Affine Subspace.

With all of these ideas, we achieve a fast convergence, high accuracy, small size of dictionary and fast performance algorithm, which is demonstrated by the computational experiment and the comparison with some important and recognized algorithms like Kernel Affine Projection with Coherence Criterion, Kernel Least Mean Square with Coherence-Sparsification criterion and Adaptive L1-norm regularization and Extended Recursive Least Squares.

REFERENCES

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404.

Engel, Y., Mannor, S., and Meir, R. (2004). The kernel recursive least-squares algorithm. *IEEE Transactions on Signal Processing*, 52(8):2275–2285.

Gao, W., Chen, J., Richard, C., Huang, J., and Flamary, R. (2013). Kernel lms algorithm with forward-backward splitting for dictionary learning. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5735–5739.

Haykin, S., Sayed, A. H., Zeidler, J. R., Yee, P., and Wei, P. C. (1997). Adaptive tracking of linear time-variant systems by extended rls algorithms. *IEEE Transactions on Signal Processing*, 45(5):1118–1128.

Kivinen, J., Smola, A. J., and Williamson, R. C. (2004). Online learning with kernels. *IEEE Transactions on Signal Processing*, 52(8):2165–2176.

Liu, W., Park, I., and Principe, J. C. (2009a). An information theoretic approach of designing sparse kernel adaptive filters. *IEEE Transactions on Neural Networks*, 20(12):1950–1961.

Liu, W., Park, I., Wang, Y., and Principe, J. C. (2009b). Extended kernel recursive least squares algorithm. *IEEE Transactions on Signal Processing*, 57(10):3801–3814.

Liu, W., Pokharel, P. P., and Principe, J. C. (2008). The kernel least-mean-square algorithm. *IEEE Transactions on Signal Processing*, 56(2):543–554.

Liu, W., Principe, J., and Haykin, S. (2010). *Kernel Adaptive Filtering: A comprehensive Introduction*. Wiley, 1 edition.

Ozeki, K. (2016). *Theory of affine projection algorithms for adaptive filtering*. Springer.

Platt, J. (1991). A resource-allocating network for function interpolation. *Neural computation*, 3(2):213–225.

Richard, C., Bermudez, J. C. M., and Honeine, P. (2009). Online prediction of time series data with kernels. *IEEE Transactions on Signal Processing*, 57(3):1058–1067.

Saidé, C., Lengellé, R., Honeine, P., Richard, C., and Achkar, R. (2012). Dictionary adaptation for online prediction of time series data with kernels. In *2012 IEEE Statistical Signal Processing Workshop (SSP)*, pages 604–607.

Scholkopf, B. and Smola, A. (2001). *Learning with Kernels. Cambridge*. MIT Press, 1 edition.

Slavakis, K., Theodoridis, S., and Yamada, I. (2008). Online kernel-based classification using adaptive projection algorithms. *IEEE Transactions on Signal Processing*, 56(7):2781–2796.

Takizawa, M. and Yukawa, M. (2013). An efficient data-reusing kernel adaptive filtering algorithm based on parallel hyperslab projection along affine subspaces. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3557–3561.

Takizawa, M. and Yukawa, M. (2015). Adaptive nonlinear estimation based on parallel projection along affine subspaces in reproducing kernel hilbert space. *IEEE Transactions on Signal Processing*, 63(16):4257–4269.

Van Vaerenbergh, S. and Santamaría, I. (2013). A comparative study of kernel adaptive filtering algorithms. In *2013 IEEE Digital Signal Processing (DSP) Workshop and IEEE Signal Processing Education (SPE)*.