Online Learning in L^2 Space with Multiple Gaussian Kernels

Motoya Ohnishi and Masahiro Yukawa
Dept. Electronics and Electrical Engineering, Keio University, Japan

***Abstract*—We present a novel online learning paradigm for nonlinear function estimation based on iterative orthogonal projections in an L^2 space reflecting the stochastic property of input signals. An online algorithm is built upon the fact that any finite dimensional subspace has a reproducing kernel, which is given in terms of the Gram matrix of its basis. The basis used in the present study involves multiple Gaussian kernels. The sequence generated by the algorithm is expected to approach towards the best approximation, in the L^2-norm sense, of the nonlinear function to be estimated. This is in sharp contrast to the conventional kernel adaptive filtering paradigm because the best approximation in the reproducing kernel Hilbert space generally differs from the minimum mean squared error estimator over the subspace (Yukawa and Müller 2016). Numerical examples show the efficacy of the proposed approach.**

I. Introduction

Given a basis containing Gaussian functions with different scale parameters, what space possesses the most preferable geometry for online nonlinear-function estimation? This question naturally arises through our recent studies of multikernel adaptive filtering [1]–[4]. Kernel adaptive filtering [5]–[18] is an adaptive counterpart of the kernel method [19], [20]. It is known that an appropriate design of the metric leads to significant improvements of convergence behaviors for the projection-based adaptive algorithms [21], [22]. For both the monokernel and multikernel approaches, projection-based adaptive algorithms have been studied under the metrics of the Euclidean space and the reproducing kernel Hilbert space (RKHS). It has experimentally observed that the Hilbertian metric enjoys better convergence behaviors than the Euclidean one due to its decorrelation property [3], [16], [17]. It has recently been shown in single-kernel arguments that the eigenvalue spread for the case of the Euclidean metric is reduced to its square root, by the use of the Hilbertian metric, under a certain practical condition [23]. The speed of convergence is actually of particular importance when multiple kernels are employed or when the data under study has a large scale. The question is *what is the best metric in the sense of decorrelation (or whitening)*. This is clearly related to the classical algorithm of the recursive least squares (RLS). An answer is the metric of the real Hilbert space $\mathcal{H} := L^2(\mathbb{R}^L, \mathrm{d}\mu)$ equipped with the inner product

$$\langle f, g\rangle_{\mathcal{H}} := \int_{\mathbb{R}^L} f(\boldsymbol{u})g(\boldsymbol{u})\mathrm{d}\mu(\boldsymbol{u}), \quad f, g \in \mathcal{H}, \tag{1}$$

and its induced norm $\|f\|_{\mathcal{H}} := \sqrt{\langle f, f\rangle_{\mathcal{H}}}$, where $L \in \mathbb{N}^*$ is the dimension of inputs and $\mathrm{d}\mu(\boldsymbol{u}) := p(\boldsymbol{u})\mathrm{d}\boldsymbol{u}$ is the probability measure for the probability density function $p(\boldsymbol{u})$ of the input vector $\boldsymbol{u} \in \mathbb{R}^L$. Throughout, $\mathbb{R}$, $\mathbb{N}$, and $\mathbb{N}^*$ are the sets of real numbers, nonnegative integers, and positive integers, respectively. The space $\mathcal{H}$ unfortunately has no reproducing kernel, although reproducing kernels play an important role in building efficient online nonlinear-estimation algorithms.

In this paper, we present an efficient online algorithm operating iterative orthogonal projections in $\mathcal{H}$ based on the fact that every finite dimensional space has a reproducing kernel. We show how the reproducing kernel can be constructed from a set of basis vectors. The Gram matrix of the basis comes in here, and it is required for evaluating the kernel. Fortunately, in some practical cases, the Gram matrix can be expressed in a closed form (see Section II-B). It is therefore unnecessary to estimate it recursively unlike the kernel RLS (KRLS) algorithm [7]. The proposed online learning paradigm has the following remarkable property: what the algorithm seeks for (i.e., the minimum mean squared error (MMSE) estimator) coincides with the best approximation of the desired nonlinear function in the dictionary subspace. The computational complexity of the proposed algorithm has the same order as that of the Euclidean approach when the selective update strategy [16] is employed. The numerical examples show that the proposed algorithm enjoys an improved decorrelation property, which leads to faster convergence.

II. Online learning in L^2 space

We consider the following nonlinear system model:

$$d_n := z_n + \nu_n = \psi(\boldsymbol{u}_n) + \nu_n, \tag{2}$$

where $\boldsymbol{u}_n$ and d_n are the input and the output, respectively, $\psi \in \mathcal{H}$ is the nonlinear function to be estimated, and ν_n is the noise at time $n \in \mathbb{N}$. Note here that the space $\mathcal{H}$ is known to be a superset of a Gaussian RKHS [24], and hence $\psi \in \mathcal{H}$ is not a stronger assumption than assumed usually in the literature of kernel adaptive filtering. The metric projection of a given vector $f \in \mathcal{H}$ onto a given closed convex set $C \subset \mathcal{H}$ is defined as

$$P_C(f) := \operatorname*{argmin}_{g \in C} \|f - g\|_{\mathcal{H}}. \tag{3}$$

If the set C is affine, P_C is referred to as the orthogonal projection.

A. Algorithm

A multikernel adaptive filter at time n is given by [1], [3]

$$\varphi_n(\boldsymbol{u}) := \sum_{q \in \mathcal{Q}} \sum_{j \in \mathcal{J}_n^{(q)}} h_{j,n}^{(q)} \kappa_q(\boldsymbol{u}_j, \boldsymbol{u}),\ h_{j,n}^{(q)} \in \mathbb{R}, \tag{4}$$

ISBN 978-0-9928626-7-1

where $\mathcal{Q} := \{1, ..., Q\}$ is the set of kernel indices and $\mathcal{J}_n^{(q)}, q \in \mathcal{Q}$, is the set of data indices for the qth kernel. The set $\mathcal{D}_n := \bigcup_{q\in\mathcal{Q}} \{\kappa_q(\boldsymbol{u}_j, \cdot)\}_{j\in\mathcal{J}_n^{(q)}}$ is called a *dictionary*, and the dictionary subspace is defined as

$$\mathcal{M}_n := \mathrm{span}\mathcal{D}_n \subset \mathcal{H}. \tag{5}$$

For the initial estimate $\varphi_0 := \theta$, where θ is the null vector of $\mathcal{H}$, generate the sequence $(\varphi_n)_{n\in\mathbb{N}}$ of nonlinear estimators by using a natural extension of [16] to $\mathcal{H}$:

$$\begin{aligned}\varphi_{n+1} &:= \varphi_n + \lambda\left(P_{\Pi_n}(\varphi_n) - \varphi_n\right)\\ &= \varphi_n + \lambda\frac{d_n - \varphi_n(\boldsymbol{u}_n)}{\|\kappa(\boldsymbol{u}_n,\cdot)\|_{\mathcal{H}}^2}\kappa(\boldsymbol{u}_n,\cdot),\end{aligned} \tag{6}$$

where $\lambda \in (0, 2)$ is the step size, $\Pi_n := \{f \in \mathcal{M}_n \mid f(\boldsymbol{u}_n) = \langle f, \kappa(\boldsymbol{u}_n, \cdot)\rangle_{\mathcal{H}} = d_n\}$ is the zero instantaneous-error hyperplane, and $\kappa(\cdot,\cdot)$ is the reproducing kernel of the Hilbert space $(\mathcal{M}_n, \langle\cdot,\cdot\rangle_{\mathcal{H}})$, given by the following proposition.

Proposition 1. *Let $\mathcal{D} := \{f_1, f_2, ..., f_r\} \subset \mathcal{H}$, $r \in \mathbb{N}^*$, be a linearly independent set, and $\boldsymbol{R}$ is its Gram matrix with its (k,l) entry $R_{k,l} := \langle f_k, f_l\rangle_{\mathcal{H}}$. Define $\boldsymbol{f}(\boldsymbol{x}) := [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_r(\boldsymbol{x})]^{\mathsf{T}}$, $\boldsymbol{x} \in \mathbb{R}^L$, where $(\cdot)^{\mathsf{T}}$ is the transpose of the vector. Then,*

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) := \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{y}),\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^L, \tag{7}$$

is the reproducing kernel of the subspace $\mathcal{M} := \mathrm{span}\mathcal{D}$.

Proof. It is clear that $\kappa(\boldsymbol{x},\cdot) \in \mathcal{M}$ for any $\boldsymbol{x} \in \mathbb{R}^L$. By definition of $\langle\cdot,\cdot\rangle_{\mathcal{H}}$, it can be readily verified that

$$\begin{aligned}&\langle\kappa(\boldsymbol{x},\cdot), \kappa(\boldsymbol{y},\cdot)\rangle_{\mathcal{H}}\\ &= \int_{\mathbb{R}^L} \underbrace{\boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{u})}_{\kappa(\boldsymbol{x},\boldsymbol{u})}\underbrace{\boldsymbol{f}(\boldsymbol{u})^{\mathsf{T}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{y})}_{\kappa(\boldsymbol{y},\boldsymbol{u})}\mathrm{d}\mu(\boldsymbol{u})\\ &= \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{R}^{-1}\underbrace{\int_{\mathbb{R}^L}\boldsymbol{f}(\boldsymbol{u})\boldsymbol{f}(\boldsymbol{u})^{\mathsf{T}}\mathrm{d}\mu(\boldsymbol{u})}_{\boldsymbol{R}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{y})\\ &= \kappa(\boldsymbol{x},\boldsymbol{y}).\end{aligned} \tag{8}$$

For any $\boldsymbol{x} \in \mathbb{R}^L$ and $\phi := \sum_{i=1}^r \alpha_i f_i, \alpha_i \in \mathbb{R}$, the reproducing property holds:

$$\begin{aligned}&\langle\phi, \kappa(\boldsymbol{x},\cdot)\rangle_{\mathcal{H}} = \int_{\mathbb{R}^L}\phi(\boldsymbol{u})\boldsymbol{f}(\boldsymbol{u})^{\mathsf{T}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{x})\mathrm{d}\mu(\boldsymbol{u})\\ &= \sum_{i=1}^r \alpha_i\int_{\mathbb{R}^L} f_i(\boldsymbol{u})\boldsymbol{f}(\boldsymbol{u})^{\mathsf{T}}\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{x})\mathrm{d}\mu(\boldsymbol{u})\\ &= \sum_{i=1}^r \alpha_i\underbrace{\boldsymbol{e}_i^{\mathsf{T}}\boldsymbol{f}(\boldsymbol{x})}_{f_i(\boldsymbol{x})} = \phi(\boldsymbol{x}),\end{aligned} \tag{9}$$

where $\{\boldsymbol{e}_i\}_{i=1}^r$ is the standard basis of $\mathbb{R}^r$. □

To attain linear complexity (see Section II-D3), we employ the selective update strategy. The idea is to select a few, say $s_n \in \mathbb{N}^*$, elements from $\mathcal{D}_n$ that are maximally coherent to $\kappa(\boldsymbol{u}_n,\cdot) \in \mathcal{M}_n$ [16]; i.e., select $\tilde{\mathcal{D}}_n(\subset \mathcal{D}_n)$ with $\left|\tilde{\mathcal{D}}_n\right| = s_n$ such that $f(\boldsymbol{u}_n)/\{\|f\|_{\mathcal{H}}\|\kappa(\boldsymbol{u}_n,\cdot)\|_{\mathcal{H}}\} \geq g(\boldsymbol{u}_n)/\{\|g\|_{\mathcal{H}}\|\kappa(\boldsymbol{u}_n,\cdot)\|_{\mathcal{H}}\}$ for any $f \in \tilde{\mathcal{D}}_n$ and for any $g \in \mathcal{D}_n \setminus \tilde{\mathcal{D}}_n$.

B. Computation of inner product

We present two practical examples for which (approximate) analytical expressions of inner product can be obtained and for which the computational advantage of kernel adaptive filters preserves.

Example 1 (Gaussian kernels and uniform distribution)**.** *Suppose that no information is available on the statistical property of the input vector. In this case, one may use the concept of noninformative prior [25], i.e. let $\mathrm{d}\mu(\boldsymbol{u}) = \mathrm{d}\boldsymbol{u}$ and $\tilde{\mathcal{H}} := L^2(\mathbb{R}^L, \mathrm{d}\boldsymbol{u})$. This approach works efficiently in practice, as will be shown by simulations in Section III. Let $\kappa_q(\boldsymbol{u},\boldsymbol{x}) := \frac{1}{(\sqrt{2\pi}\sigma_q)^L}\exp\left(-\frac{\|\boldsymbol{u}-\boldsymbol{x}\|_{\mathbb{R}^L}^2}{2\sigma_q^2}\right)$, $q \in \mathcal{Q}$, $\boldsymbol{u}, \boldsymbol{x} \in \mathbb{R}^L$, $\sigma_q > 0$. Then, for any $k, l \in \mathcal{Q}$ and for any $\boldsymbol{u}, \boldsymbol{x} \in \mathbb{R}^L$,*

$$\langle\kappa_k(\boldsymbol{u},\cdot), \kappa_l(\boldsymbol{x},\cdot)\rangle_{\tilde{\mathcal{H}}} = \frac{1}{(\sqrt{2\pi}\sigma_{k,l})^L}\exp\left(-\frac{\|\boldsymbol{u}-\boldsymbol{x}\|_{\mathbb{R}^L}^2}{2\sigma_{k,l}^2}\right), \tag{10}$$

where $\|\boldsymbol{x}\|_{\mathbb{R}^L} := \sqrt{\boldsymbol{x}^{\mathsf{T}}\boldsymbol{x}}$, $\boldsymbol{x} \in \mathbb{R}^L$, and $\sigma_{k,l} := \sqrt{\sigma_k^2 + \sigma_l^2}$. We mention that the result in (10) is also obtained in the RKHS of a Gaussian kernel by taking the limit of its scale parameter towards zero (see also [26]).

Example 2 (Gaussian kernels and Gaussian distribution)**.** *Suppose that the input vector obeys the zero-mean Gaussian distribution with standard deviation $\sigma > 0$. In this case, using the Gaussian kernels as in Example 1, the inner product can be computed exactly by*

$$\begin{aligned}&\langle\kappa_k(\boldsymbol{u},\cdot), \kappa_l(\boldsymbol{x},\cdot)\rangle_{\mathcal{H}} = \frac{1}{(2\pi\sigma\sigma_{k,l})^L}\\ &\exp\left(-\frac{\sigma^2\|\boldsymbol{u}-\boldsymbol{x}\|_{\mathbb{R}^L}^2 + \sigma_l^2\|\boldsymbol{u}\|_{\mathbb{R}^L}^2 + \sigma_k^2\|\boldsymbol{x}\|_{\mathbb{R}^L}^2}{2\sigma^2\sigma_{k,l}^2}\right).\end{aligned} \tag{11}$$

C. A remarkable property of L^2-space online learning

The online learning paradigm in the L^2 space $\mathcal{H}$ has a remarkable property coming directly from the following basic fact.

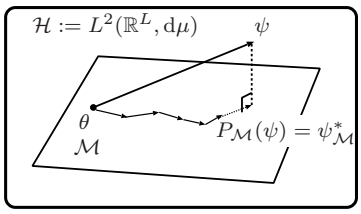
Fact 1. *Assume that $E[\boldsymbol{f}(\boldsymbol{u}_n)\nu_n] = \boldsymbol{0}$. The MMSE estimator $\psi_{\mathcal{M}}^* := \mathrm{argmin}_{f\in\mathcal{M}} E[d_n - f(\boldsymbol{u}_n)]^2$ then coincides with the orthogonal projection $P_{\mathcal{M}}(\psi)$, which is the best approximation of ψ in $\mathcal{M}$ in the L^2-norm sense.*

Figure 1(a) illustrates the proposed learning paradigm. This is in sharp contrast to the conventional picture of the kernel adaptive filtering paradigm depicted in Figure 1(b). Note here that the best approximation of ψ in the RKHS generally differs from the MMSE estimator $\psi_{\mathcal{M}}^*$ [23].

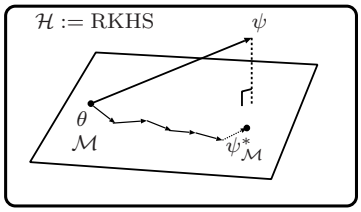
D. Discussion

1) Decorrelation property: The convergence behavior of the proposed algorithm is governed by the autocorrelation

ISBN 978-0-9928626-7-1 © EURASIP 2017



(a) Proposed online learning paradigm



(b) Conventional kernel adaptive filtering paradigm

Fig. 1. An illustrative comparison between the proposed paradigm and the conventional kernel adaptive filtering paradigm.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE ALGORITHMS

NLMS	$3L+2$
KNLMS	$(L+6)r_n+2$
HYPASS	$(L+4)r_n+\frac{L+5}{2}s_n^2-\frac{L-1}{2}s_n+2+v_{\mathrm{inv}}(s_n)$
MXKLMS	$(L+3)r_n+5Q+10$
OMKR	$(L+3Q)r_n+4Q$
MKNLMS	$(L+6)r_n+2$
CHYPASS	$(L+5)r_n+\frac{L+5}{2}s_n^2-\frac{L-1}{2}s_n+2+v_{\mathrm{inv}}(s_n)$
Proposed	$(L+5)r_n+\frac{L+5}{2}s_n^2-\frac{L-1}{2}s_n+2+v_{\mathrm{inv}}(s_n)$

matrix $\widetilde{\boldsymbol{R}} := \hat{\boldsymbol{R}}^{-\frac{1}{2}}\boldsymbol{R}\hat{\boldsymbol{R}}^{-\frac{1}{2}} \approx \boldsymbol{I}$ of the modified kernelized input vector $\tilde{\boldsymbol{k}} := \hat{\boldsymbol{R}}^{-\frac{1}{2}}\boldsymbol{k}$, where $\hat{\boldsymbol{R}}$ is an approximation of $\boldsymbol{R}$ (see [17]). In practice, the exact autocorrelation matrix is unavailable as the exact distribution of the input is unknown. The eigenvalue spread of $\widetilde{\boldsymbol{R}}$ is therefore greater than one usually.

2) Relation to kernel adaptive filter: The major advantages of kernel adaptive filtering include the tractability of inner product (which enables to obtain the update direction easily) and the invariance of metric when the dictionary grows. The proposed paradigm satisfies both of the advantages because a reproducing kernel can always be defined depending on the dictionary. Besides, the scale parameters of basis functions can be chosen arbitrarily and (possibly inappropriate) noninformative distribution of the input vector efficiently works in many cases (see Section III), although the inner product can be expressed in a closed-form only in some limited cases. We emphasize that our approach does not prevent us from using kernels other than Gaussian, for example, when multiple Gaussian kernels and non-Gaussian kernel are employed, the Cartesian product of L^2 space for Gaussian kernels and the RKHS of the other kernel can be exploited instead of the Cartesian products of the RKHSs.

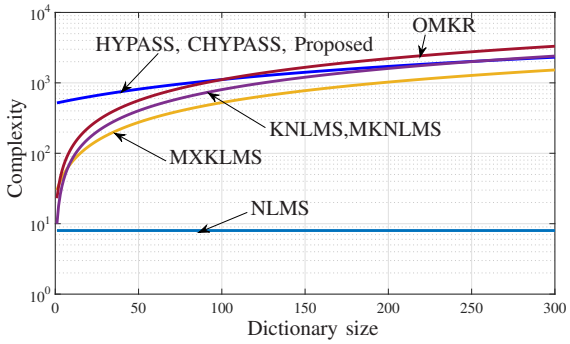


Fig. 2. Computational complexity for $L=2, s_n=7, Q=3$.

3) Computational complexity: We discuss the computational complexity in terms of the number of multiplications required at each iteration. Suppose that the normalized Gaussian kernels as in Example 1 are used. We employ the selective update strategy [16] and s_n is the number of selected elements. Table I summarizes the overall per-iteration complexity of the proposed algorithm (the case of Example 1), normalized least mean squares (NLMS) [27], kernel NLMS (KNLMS) [12], hyperplane projection along affine subspace (HYPASS) [16], deterministic online multiple kernel regression (OMKR) (Hedge) [28], mixture kernel least mean square (MXKLMS) [29], multikernel NLMS (MKNLMS) [1], and Cartesian HYPASS (CHYPASS) [3]. In Table I, the complexity required for the inverse of an $s_n \times s_n$ matrix is denoted as $v_{\mathrm{inv}}(s_n)$. Figure 2 shows the evolution of the computational complexity of the algorithms for $L=2, s_n=7$, $n \in \mathbb{N}$, and $Q=3$, where we let $v_{\mathrm{inv}}(s_n) := s_n^3$.

4) Dictionary design and novelty criteria: Suppose that the Gaussian kernels are employed with $\sigma_1 > \sigma_2 > \cdots > \sigma_Q > 0$. In this case, one possible dictionary design is the following.

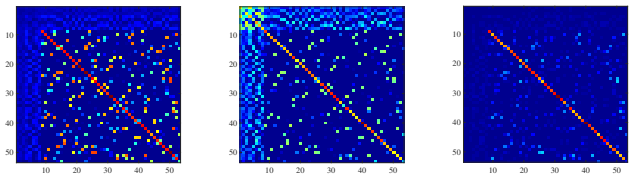
1) $\kappa_1(\boldsymbol{u}_n, \cdot)$ is added into the dictionary when a novelty criterion is satisfied for $\kappa_1(\boldsymbol{u}_n, \cdot)$.
2) $\kappa_i(\boldsymbol{u}_n, \cdot)$, $i \geq 2$, is added into the dictionary if the novelty criterion is satisfied for $\kappa_i(\boldsymbol{u}_n, \cdot)$ but is unsatisfied for all $\kappa_1(\boldsymbol{u}_n, \cdot)$, $\kappa_2(\boldsymbol{u}_n, \cdot)$, $\cdots$, $\kappa_{i-1}(\boldsymbol{u}_n, \cdot)$.

In the experiments, the coherence criterion [12] is employed for all algorithms because of its low computational complexity. We show however that the proposed algorithm takes some benefits from the approximate linear dependency (ALD) criterion, although an experimental study for the ALD criterion is beyond the scope of this paper. Let $\mathcal{M}_+ := \mathrm{span}\mathcal{D}_+$ be the dictionary subspace of $\mathcal{D}_+ := \mathcal{D} \cup \{f_{r+1}\}$. Given a vector $f \in \mathcal{H}$ and a dictionary subspace $\mathcal{M}$, we consider the ALD condition

$$\frac{\|f - P_{\mathcal{M}}(f)\|_{\mathcal{H}}^2}{\|f\|_{\mathcal{H}}^2} \geq \eta \tag{12}$$

where $\eta \in [0,1]$ controls the sparsity level of the dictionary. Then, the following proposition holds.

Proposition 2. *Given $f_{r+1} \in \mathcal{H}$, let $\mathcal{M}_+ := \mathrm{span}\,(\mathcal{D} \cup f_{r+1})$ and $\psi^*_{\mathcal{M}_+} := \sum_{i=1}^{r+1} h_i f_i$, $h_i \in \mathbb{R}$ be the MMSE estimate*



(a) MKNLMS (b) CHYPASS (c) Proposed

Fig. 3. Illustrations of the autocorrelation matrices of the modified kernelized input vectors.

in $\mathcal{M}_+$. *Assume* $E\left[f_i(\boldsymbol{u}_n)\nu_n\right] = 0,\ i \in \{1, ..., r+1\}$, *and* $E\left[\psi(\boldsymbol{u}_n)\nu_n\right] = 0$. *If the ALD condition (12) is satisfied for the vector* $f := f_{r+1}$, *it holds that*

$$\Delta\text{MMSE} := \text{MMSE}(\mathcal{M}) - \text{MMSE}(\mathcal{M}_+) \geq h_{r+1}^2 \left\|f_{r+1}\right\|_{\mathcal{H}}^2 \eta, \tag{13}$$

where $\text{MMSE}(\mathcal{M}) := \min_{f\in\mathcal{M}} E\left[d_n - f(\boldsymbol{u}_n)\right]^2$ *for a closed subspace* $\mathcal{M} \subset \mathcal{H}$.

Proof. By the assumptions and the definition of $\|\cdot\|_{\mathcal{H}}$, we have

$$\begin{aligned}
&\text{MMSE}(\mathcal{M}) \\
&= E\left[d_n - \psi_{\mathcal{M}}^*(\boldsymbol{u}_n)\right]^2 = E\left[\psi(\boldsymbol{u}_n) - \psi_{\mathcal{M}}^*(\boldsymbol{u}_n)\right]^2 + E(\nu_n)^2 \\
&= \left\|\psi - \psi_{\mathcal{M}}^*\right\|_{\mathcal{H}}^2 + E(\nu_n)^2 \qquad (14) \\
&\text{MMSE}(\mathcal{M}_+) \\
&= E\left[d_n - \psi_{\mathcal{M}_+}^*(\boldsymbol{u}_n)\right]^2 = \left\|\psi - \psi_{\mathcal{M}_+}^*\right\|_{\mathcal{H}}^2 + E(\nu_n^2). \qquad (15)
\end{aligned}$$

By Pythagorean theorem, it follows that

$$\begin{aligned}
&\Delta\text{MMSE} \\
&= \left\|\psi - \psi_{\mathcal{M}}^*\right\|_{\mathcal{H}}^2 - \left\|\psi - \psi_{\mathcal{M}_+}^*\right\|_{\mathcal{H}}^2 = \left\|\psi_{\mathcal{M}_+}^* - \psi_{\mathcal{M}}^*\right\|_{\mathcal{H}}^2 \\
&= h_{r+1}^2 \left\|f_{r+1} - P_{\mathcal{M}}(f_{r+1})\right\|_{\mathcal{H}}^2 \geq h_{r+1}^2 \left\|f_{r+1}\right\|_{\mathcal{H}}^2 \eta. \qquad (16)
\end{aligned}$$

□

By Proposition 2, the ALD condition ensures that the amount of MMSE reduction is no smaller than $h_{r+1}^2 \left\|f_{r+1}\right\|_{\mathcal{H}}^2 \eta$.

5) Convergence analysis: Since the proposed paradigm employs the Hilbert space $L^2(\mathbb{R}^L, \mathrm{d}\mu)$ for learning, the convergence analysis in [16, Section III.C] can be applied straightforwardly to the present case.

III. Numerical Examples

We first show the decorrelation property of the proposed algorithm. We then show the efficacy of the proposed algorithm in online prediction of time-series data.

A. Decorrelation property

We compare the eigenvalue spreads of the modified autocorrelation matrices of the proposed algorithm and the existing multikernel adaptive filtering algorithms. Dictionary is constructed by using the coherence criterion with the threshold $\eta = 0.8$ beforehand and is fixed during the experiment. We employ Gaussian kernels with scale parameters $\sigma = 1.0, 0.5, 0.05$, and test 10000 samples drawn from the input space $\mathbb{R}$, i.e. $L = 1$ with the uniform distribution over $[-1, 1]$, and the eigenvalue spreads of the autocorrelation matrices $\widetilde{\boldsymbol{R}}$s are averaged over 300 independent trials. Table II shows the eigenvalue spread of $\widetilde{\boldsymbol{R}}$ for each algorithm. According to Table II, the proposed algorithm has a better decorrelation property. We emphasize here that proposed approach works well despite the use of (possibly inappropriate) noninformative distribution for the input vector. For further clarification, $\widetilde{\boldsymbol{R}}$s for MKNLMS, CHYPASS, and the proposed algorithm are illustrated in Figure 3. In particular, we can observe that the off-diagonal elements of $\widetilde{\boldsymbol{R}}$ are closer to zero compared to the other algorithms.

TABLE II
Eigenvalue spreads of $\widetilde{\boldsymbol{R}}$s.

MKNLMS	CHYPASS	Proposed
1.39×10^{17}	2.04×10^{15}	3.70×10^{13}

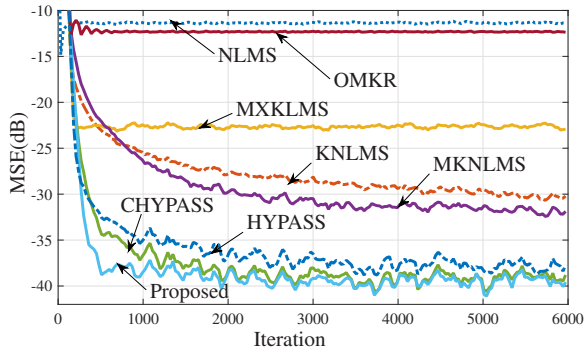
B. Online prediction of time-series data

We consider the time series data generated by the following equation [12]: $d_n := (0.8 - 0.5\exp(-d_{n-1}^2))d_{n-1} - (0.3 + 0.9\exp(-d_{n-1}^2))d_{n-2} + 0.1\sin(d_{n-1}\pi)$ for $0 \leq n \leq 6000$ ($d_{-2} := d_{-1} := 0.1$) The noise is white Gaussian with the signal to noise ratio (SNR) 40 dB. In this experiment, each datum d_n is regarded as a nonlinear function of $\boldsymbol{u}_n := [d_{n-1}, d_{n-2}]^{\mathsf{T}}$, i.e. $L = 2$. In this example, the distribution of the input vector is neither uniform nor Gaussian. Nevertheless, we shall assume noninformative distribution for the input vector to show that the assumption works well even in such a case. We compare the proposed algorithm with NLMS, KNLMS, HYPASS, OMKR, MXKLMS, MKNLMS, and CHYPASS. The proposed algorithm, MKNLMS, and CHYPASS employ the selective update strategy with $s_n := 7$. We employ three Gaussian kernels with $\sigma_1 = 1.5, \sigma_2 = 0.9, \sigma_3 = 0.3$, and for the single-kernel ones, we only employ a Gaussian kernel with scale parameter 0.3661 by following the recommendation in [12]. The maximal dictionary size for OMKR and MXKLMS is set to $M = 100$, and the coherence threshold for the proposed algorithm, KNLMS, HYPASS, MKNLMS, and CHYPASS are selected so that the dictionary sizes at the end of each trial are the same. Figure 4(a) shows the MSE learning curves and Figure 4(b) shows the evolutions of the dictionary size. The proposed algorithm outperforms the compared algorithms. Note that OMKR and MXKLMS compute the coefficients of atoms in the dictionary only once, and this causes a severe degradation of the performance when the maximal dictionary size is limited.

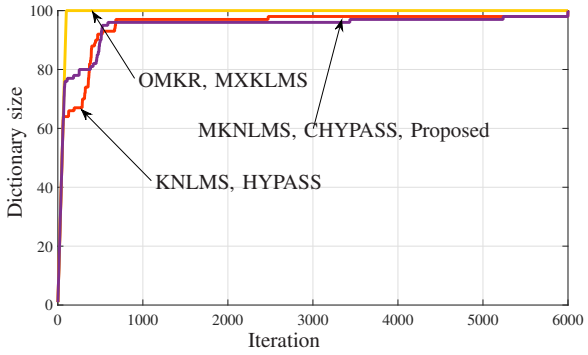
IV. Conclusion

The $L^2(\mathbb{R}^L, \mathrm{d}\mu)$ space possesses the most preferable geometry (in the sense of decorrelation) for online nonlinear-function estimation for a given set of Gaussian functions with different scale parameters. We proposed the online learning algorithm with multiple Gaussian kernels based on the iterative

ISBN 978-0-9928626-7-1 © EURASIP 2017



(a) MSE learning curves



(b) Evolutions of dictionary size

Fig. 4. Results of online prediction.

orthogonal projection operated in $L^2(\mathbb{R}^L, \mathrm{d}\mu)$. Although the L^2 space has no reproducing kernel, its finite dimensional subspace has a reproducing kernel. The update equation of the proposed algorithm therefore resembles a kernel adaptive filtering algorithm. The proposed L^2 space online learning paradigm has a remarkable property that the MMSE estimator (which online algorithms seek for) coincides with the best approximation of the unknown system within the dictionary subspace. The efficacy of the proposed algorithm was shown by simulations.

Acknowledgment

This work was supported by the Support Center for Advanced Telecommunications Technology Research (SCAT) and JSPS Grants-in-Aid (15K06081, 15K13986, 15H02757).

References

[1] M. Yukawa, "Multikernel adaptive filtering," *IEEE Trans. Signal Processing*, vol. 60, no. 9, pp. 4672–4682, 2012.

[2] M. Yukawa and R. Ishii, "Online model selection and learning by multikernel adaptive filtering," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2013, pp. 1–5.

[3] M. Yukawa, "Adaptive learning in Cartesian product of reproducing kernel Hilbert spaces," *IEEE Trans. Signal Processing*, vol. 63, no. 22, pp. 6037–6048, Nov. 2015.

[4] O. Toda and M. Yukawa, "Online model-selection and learning for nonlinear estimation based on multikernel adaptive filtering," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, vol. E100-A, no. 1, pp. 236–250, Jan. 2017.

[5] L. Csato and M. Opper, *Sparse representation for Gaussian process models,* in Advances in Neural Information Processing Systems 13, pp. 444–450, MIT Press, 2001.

[6] J. Kivinen, A. J. Smola, and R. C. Williamson, "Online learning with kernels," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2165–2176, Aug. 2004.

[7] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2275–2285, Aug. 2004.

[8] A. V. Malipatil, Y.-F. Huang, S. Andra, and K. Bennett, "Kernelized set-membership approach to nonlinear adaptive filtering," in *Proc. IEEE ICASSP*, 2005, pp. 149–152.

[9] P. Laskov, C.Gehl, S.Krüger, and K.-R. Müller, "Incremental support vector learning: Analysis, implementation and applications," *J. Mach. Learn. Res.*, vol. 7, pp. 1909–1936, 2006.

[10] W. Liu, P. P. Pokharel, and J. C. Principe, "The kernel least-mean-square algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 543–554, Feb. 2008.

[11] K. Slavakis, S. Theodoridis, and I. Yamada, "Online kernel-based classification using adaptive projection algorithms," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 2781–2796, July 2008.

[12] C. Richard, J. Bermudez, and P. Honeine, "Online prediction of time series data with kernels," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1058–1067, Mar. 2009.

[13] W. Liu, J. Príncipe, and S. Haykin, *Kernel Adaptive Filtering*, Wiley, New Jersey, 2010.

[14] S. Van Vaerenbergh, M. Lázaro-Gredilla, and I. Santamaría, "Kernel recursive least-squares tracker for time-varying regression," *IEEE Trans. Neural Networks and Learning Systems*, vol. 23, no. 8, pp. 1313–1326, Aug. 2012.

[15] B. Chen, S. Zhao, P. Zhu, and J. C. Príncipe, "Quantized kernel least mean square algorithm," *IEEE Trans. Neural Networks and Learning Systems*, vol. 23, no. 1, pp. 22–32, Jan. 2012.

[16] M. Takizawa and M. Yukawa, "Adaptive nonlinear estimation based on parallel projection along affine subspaces in reproducing kernel Hilbert space," *IEEE Trans. Signal Processing*, vol. 63, no. 16, pp. 4257–4269, Aug. 2015.

[17] M. Takizawa and M. Yukawa, "Efficient dictionary-refining kernel adaptive filter with fundamental insights," *IEEE Trans. Signal Processing*, vol. 64, no. 16, pp. 4337–4350, Aug. 2016.

[18] M. Yukawa, "Adaptive learning with reproducing kernels," in *RIMS Kokyuroku 1980 (General Topics on applications of reproducing kernels)*, Jan., pp. 1–15.

[19] K. R. Müller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.

[20] B. Schöelkopf and A. Smola, *Learning with kernels*, MIT Press, Cambridge, 2002.

[21] M. Yukawa, K. Slavakis, and I. Yamada, "Adaptive parallel quadratic-metric projection algorithms," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1665–1680, July 2007.

[22] M. Yukawa and I. Yamada, "A unified view of adaptive variable-metric projection algorithms," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, 2009.

[23] M. Yukawa and K. R. Müller, "Why does a Hilbertian metric work efficiently in online learning with kernels?," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1424–1428, 2016.

[24] A. J. Smola, B. Schölkopf, and K.-R. Müller, "The connection between regularization operators and support vector kernels," *Neural Networks*, vol. 11, no. 4, pp. 637–649, June 1998.

[25] C. M. Bishop, *Pattern recognition and machine learning*, Springer, 2006.

[26] A. Tanaka, H. Imai, M. Kudo, and M. Miyakoshi, "Theoretical analyses on a class of nested RKHS's," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 2072–2075.

[27] J. Nagumo and J. Noda, "A learning method for system identification," *IEEE Trans. Automatic Control*, vol. 12, no. 3, pp. 282–287, June 1967.

[28] D. Sahoo, S. C. Hoi, and B. Li, "Online multiple kernel regression," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2014, pp. 293–302.

[29] R. Pokharel, S. Seth, and J. C. Principe, "Mixture kernel least mean square," in *Proc. International Joint Conference on Neural Networks (IJCNN)*, 2013, pp. 1–7.

ISBN 978-0-9928626-7-1 © EURASIP 2017