Kernel Adaptive Filtering with Maximum Correntropy Criterion

Songlin Zhao, Badong Chen, and José C. Príncipe

Abstract— **Kernel adaptive filters have drawn increasing attention due to their advantages such as universal nonlinear approximation with universal kernels, linearity and convexity in Reproducing Kernel Hilbert Space (RKHS). Among them, the kernel least mean square (KLMS) algorithm deserves particular attention because of its simplicity and sequential learning approach. Similar to most conventional adaptive filtering algorithms, the KLMS adopts the mean square error (MSE) as the adaptation cost. However, the mere second-order statistics is often not suitable for nonlinear and non-Gaussian situations. Therefore, various non-MSE criteria, which involve higher-order statistics, have received an increasing interest. Recently, the correntropy, as an alternative of MSE, has been successfully used in nonlinear and non-Gaussian signal processing and machine learning domains. This fact motivates us in this paper to develop a new kernel adaptive algorithm, called the kernel maximum correntropy (KMC), which combines the advantages of the KLMS and maximum correntropy criterion (MCC). We also study its convergence and self-regularization properties by using the energy conservation relation. The superior performance of the new algorithm has been demonstrated by simulation experiments in the noisy frequency doubling problem.**

I. Introduction

MANY real-word applications require complex nonlinear models. At present, Neural Networks and Kernel Adaptive Filters are possible solutions. By providing linearity in RKHS and convexity in hypothesis space [1], the kernel adaptive filter is attracting more attention. Through a reproducing kernel, kernel adaptive filter maps data from an input space to a high-dimensional feature space, where appropriate linear methods are applied to the transformed data. Most importantly, the kernel method with universal kernel has universal approximation property, which has been proved by [2], i.e. for any continuous input-output mapping $f : \mathbf{U} \rightarrow \mathbf{R}$, $\forall \varsigma > 0, \exists \{\boldsymbol{u}_i\}_{i \in N} \in \mathbf{U}$ and real number $\{c_i\}_{i \in N}$, such that $\| f - \sum_{i \in N} c_i \kappa(\boldsymbol{u}_i, .) \|_2 < \varsigma$. This universal approximation property guarantees that the kernel method is capable of superior performance in nonlinear tasks. There are many successful examples of this methodology including support vector machines [3], kernel regularization network [4], kernel principal component analysis (kernel PCA) [5], kernel fisher discriminant analysis [6], kernel recursive least squares algorithm (KRLS)[7] and among others. Compared with these algorithms, the Kernel Least Mean Square (KLMS) is different. It provides well-posedness solution with finite data [8] and naturally creates a growing radial-basis function (RBF) network [1]. Moreover, as an online learning algorithm, KLMS is much simpler to achieve, with respect to computational complexity and memory storage, than other batch mode kernel methods.

Similar to most conventional adaptive filtering algorithms, KLMS has utilized the MSE criterion as a cost function. The mere second-order statistics is often not suitable for nonlinear and non-Gaussian situations. Recently, Information theoretic learning (ITL) has been proved more efficient to train adaptive systems. Different from MSE criterion using error energy, ITL utilizes probability density function of the data, estimated by Parzen kernel estimator [9], as the cost function. Owning to taking account into the whole signal distribution, adaptive systems training through information theoretic criteria have better performance in various applications in which the signals are non-Gaussian. Correntropy, developed by Principe et al., is a kind of localized measure to estimate how similar two random variables are: when two random variables are very close, correntropy equals the 2-norm distance, which evolves to 1-norm distance if two random variables get further apart, even falls to zero-norm as they are far apart [10]. Correntropy has already been employed to many applications successfully. Kernel PCA can project the transformed data onto principal directions with correntropy function [11] , and efficiently compute the principal components in the feature space. [12] proposed a power spectral measure for Fourier based surrogate nonlinearity test through correntropy as a discriminant measure. Extending the Minimum Average Correlation Energy (MACE) filter to nonlinear filter via correntropy improves MACE performance, when applied to face recognition [13]. Moreover, similar extension of Granger causality by correntropy can detect causality of a nonlinear dynamical system where the linear Granger causality failed [14]. When it comes to the cost function of adaptive filtering algorithm, MCC is a robust adaptation principle in presence of non-gaussian outliers [15]. Maximazing the similarity between the desire and the prediction output in the senese of correntropy, [16] induced a smooth loss function, C-loss function, to approximate the ideal 0-1 loss in classification problem.

Inspired by KLMS algorithm and MCC criterion, this paper presents a kernel-based MCC learning algorithm, called KMC (Kernel Maximum Correntropy). KMC maps the input data into an RKHS to approximate the input-output mapping f, then utilizes MCC as a cost function to minimize the difference between the desired data and the filter output. This algorithm not only approximates nonlinear system more accurate than linear model, but also is robust in different noisy environment. The computational complexity of our algorithm is similar to KLMS while the robustness is superior. Furthermore, we prove that the KMC is wellposed when finite data is used in the training and therefore does

Songlin Zhao, Badong Chen and José C. Príncipe are with the Department of Electrical and Computer Engineering, The University of Florida, Gainesville, USA (email: slzhao, chenbd, principe@cnel.ufl.edu).

This work was partially supported by NSF grant ECCS 0856441.

not need explicit regularization, which not only simplifies the implementation but also results in the potential to provide better performances because regularization biases the optimala solution as is well known.

The organization of the paper is as follows. Section II is a brief review of correntropy and MCC in linear adaptive filters, and some properties of correntropy are presented to verify the feasibility of correntropy as cost criterion. Afterwards, KMC algorithm is developed in section III, followed by convergence analysis and self-regularization interpretation using energy conservation relation. Finally, simulations for the adaptive frequency-doubler are studied in Section IV, and the conclusions and future lines of work are summarized in Section V.

II. Foundations

The goal of our system is to construct a function $f : \mathbf{U} \rightarrow \mathbf{R}$ based on a known sequence $(\boldsymbol{u}_1, d_1), (\boldsymbol{u}_2, d_2), \ldots, (\boldsymbol{u}_N, d_N) \in Z^N$ where $\boldsymbol{u}_i$ is the system input at sample time i, and d_i is the corresponding desire response. Notice that the desired data may be noisy in practice, that is $d_i = d'_i + \xi_i$, in which d'_i is the real clean data and ξ_i is noise at time i. Actually, what we want to solve is the following empirical risk minimization (ERM) problem:

$$R_{emp}[f \in H, Z_N] = \sum_{i=1}^{N} (d'_i - f(\boldsymbol{u}_i))^2 \tag{1}$$

If the noise distribution has outliers, is non-symmetric, or has nonzero mean, the conventional MSE criterion would result in large variation of weights or output shift.

A. Definition and Properties of Correntropy

As developed in [10] and [17], correntropy is a method to estimate probabilistically the similarity between two arbitrary random variables. The kernel bandwidth controls the "window" in which similarity is assessed.

$$V_\sigma(X, Y) = \mathbf{E}[\kappa_\sigma(X - Y)] \tag{2}$$

in which, $\kappa_\sigma(.)$ is a symmetric positive definite kernel with the kernel width being σ. For simplicity, the Gaussian Kernel is the only one considered in the paper. In practice, we use a set of finite data to approximate the expectation,

$$\hat{V}_{N,\sigma}(X, Y) = \frac{1}{N} \sum_{i=1}^{N} \kappa_\sigma(X - Y) \tag{3}$$

For completeness, we present below some of the most important properties of the correntropy function.

Property 1: Correntropy is positive definite and bounded, that is, $0 < V_\sigma(X, Y) \leq \frac{1}{\sqrt{2\pi}\sigma}$. It reaches its maximum if and only if $X = Y$.

Property 2: Correntropy involves all the even moments of the difference between X and Y:

$$V_\sigma(X, Y) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{n=0}^{\infty} (-1)^n / (2^n n!) \mathbf{E}[(X - Y)^{2n} / (\sigma^{2n})] \tag{4}$$

Compared with MSE $\mathbf{E}[(X - Y)^2]$ which is a quadratic function in the joint input space, correntropy includes second and higher order statistical information. However, when σ increases, the high-order information decays faster, so the second order moment dominates for large σ.

Property 3: Assume i.i.d data sample $(x_i, y_i)_{i \in N}$ follows the joint pdf $f_{X,Y}(x, y)$. Define the error random variable $E = Y - X$, and $\widehat{f}_{E,\sigma}(e)$ as the Parzen estimation of E with the kernel size σ. Such that $\widehat{V}_\sigma(X, Y)$ is the value of $\widehat{f}_{E,\sigma}(e)$ estimated at the point $e = 0$.

The properties above have already been mentioned and proved in [10]. Next, new properties about correntropy are proposed to verify the feasibility of correntropy as cost criterion of adaptive filter.

Property 4: Express correntropy of X, Y with kernel size σ as:

$$h_\sigma(E) = V_\sigma(X, Y) \tag{5}$$

$h_\sigma(E)$ is strictly concave in the range of $E \in [-\sigma, \sigma]$. When correntropy is utilized as a cost criterion in adaptive filters, strictly concavity guarantees the existence and uniqueness for the optimal solution of adaptive filter. Because the concave property is satisfied in the range $E \in [-\sigma, \sigma]$, the initial condition should be chosen carefully, or we can use other criteria to train adaptive filter firstly to make sure current solution is near the global optimal solution.

As a global measure, MSE includes all the samples in the input space to estimate the similarity of two random variables while correntropy is determined by kernel function along $x = y$ line. This property intuitively explains why the correntropy is superior than MSE if the residual of $X - Y$ is non-symmetric or with nonzero mean.

B. Correntropy in Linear Adaptive Filters

When it comes to adaptive filters, the goal is to maximize correntropy between the desired signal d_i and the filter output y_i. Such, criterion is

$$J_n = \frac{1}{N} \sum_{i=n-N+1}^{n} \kappa_\sigma(d_i, y_i) \tag{6}$$

in which, $\kappa_\sigma(.)$ is a positive definite symmetric kernel with the kernel width being σ, and N is the number of samples in Parzen estimate window.

Similar with MSE criterion, we can use an iterative gradient ascent approach to search the optimal solution, that is the next set of filter weights are corrected by taking a value proper to the positive gradient of the cost function in the weight space. Therefore,

$$\boldsymbol{\omega}_{n+1} = \boldsymbol{\omega}_n + \eta \nabla J_n \tag{7}$$

Substituting J_n into Eq.7, we can obtain,

$$\boldsymbol{\omega}_{n+1} = \boldsymbol{\omega}_n + \frac{\eta}{N} \sum_{i=n-N+1}^{n} \frac{\partial \kappa_\sigma(d_i, y_i)}{\partial \boldsymbol{\omega}_{\boldsymbol{n}}} \tag{8}$$

For online mode, the current value ($N = 1$) approximates the stochastic gradient,

$$\begin{aligned}\boldsymbol{\omega}_{n+1} &= \boldsymbol{\omega}_n + \eta \frac{\partial \kappa_\sigma(d_n, y_n)}{\partial \boldsymbol{\omega}_n} \\ &= \boldsymbol{\omega}_n + \eta g(e_n)\boldsymbol{u}_n \\ &= \boldsymbol{\omega}_n + \eta exp(\frac{-e_n^2}{2\sigma^2})e_n\boldsymbol{u}_n\end{aligned} \tag{9}$$

in which $e_n = d_n - \boldsymbol{\omega}_n^T\boldsymbol{u}_n$ is the prediction error, and $g(e_n)$ is a function of e_n in terms of the kernel choice of Correntropy. $g(e_n) = exp(\frac{-e_n^2}{2\sigma^2})e_n$ for the Normalized Gaussian Kernel. In this paper, we assume $N = 1$ is enough to approximate the gradient.

This section showed that MCC shares the computational simplicity of the LMS algorithm. Its computational complexity is $O(N)$, where N is the number of training data. With the smooth dependence of correntropy on kernel bandwith, this criterion is a robust statistical method. In [15], experiments in theoretic and practical applications demonstrate the advantage of MCC in linear adaptive filters thorough comparing with other criteria.

III. Formulation of Kernel Maximum-Correntropy Algorithm

A. KMC Algorithm

If the mapping between d and u is nonlinear, linear adaptive filters cannot obtain good performance. Because of their universal approximation capabilities and convex optimization [1], kernel methods are good choice for this task. In our algorithm, the input data $\boldsymbol{u}_i$ is transformed to a high-dimensional feature space $\mathbb{F}$ as $\boldsymbol{\varphi}(\boldsymbol{u}_i)$ via the kernek-induced mapping. Furthermore, linear adaptive filter is utilized in the feature space. As discussed in Representer Theorem [18], the adaptive filter weight has the representation,

$$\begin{aligned}f &= \sum_{i\in N} c_i < \boldsymbol{\varphi}(\boldsymbol{u}_i), . > \\ &= \sum_{i\in N} c_i \kappa(\boldsymbol{u}_i, .)\end{aligned} \tag{10}$$

where c_i are weighted coefficients obtained from the training data, and κ is a positive define kernel. In general, f is expressed as $\boldsymbol{\Omega}$ in adaptive filters. Then, using the MCC criterion and the stochastic gradient approximation to the new pairwise sample $\{\boldsymbol{\varphi}(\boldsymbol{u}_n), d_n\}$, yields

$$\begin{aligned}\boldsymbol{\Omega}_0 &= 0 \\ \boldsymbol{\Omega}_{n+1} &= \boldsymbol{\Omega}_n + \eta \frac{\partial \kappa_\sigma(d_n, \boldsymbol{\Omega}_n^T\boldsymbol{\varphi}(\boldsymbol{u}_n))}{\partial \boldsymbol{\Omega}_n} \\ &= \boldsymbol{\Omega}_n + \eta[exp(\frac{-e_n^2}{2\sigma^2})e_n\boldsymbol{\varphi}_n] \\ &= \boldsymbol{\Omega}_{n-1} + \eta \sum_{i=n-1}^{n}[exp(\frac{-e_i^2}{2\sigma^2})e_i\boldsymbol{\varphi}_i] \\ &\ldots \\ &= \eta \sum_{i=1}^{n}[exp(\frac{-e_i^2}{2\sigma^2})e_i\boldsymbol{\varphi}_i]\end{aligned} \tag{11}$$

where $\boldsymbol{\varphi}_i$ is a simplified notation for $\boldsymbol{\varphi}(\boldsymbol{u}_i)$. Now, the "kernel trick" is used to obtain the system output, which can be solely expressed in terms of inner products between the new input and previous inputs weighted by prediction errors.

$$\begin{aligned}y_{n+1} &= \boldsymbol{\Omega}_{n+1}^T\boldsymbol{\varphi}_{n+1} \\ &= \eta \sum_{i=1}^{n}[exp(\frac{-e_i^2}{2\sigma^2})e_i\boldsymbol{\varphi}_i^T\boldsymbol{\varphi}_{n+1}] \\ &= \eta \sum_{i=1}^{n}[exp(\frac{-e_i^2}{2\sigma^2})e_i\kappa(\boldsymbol{u}_i, \boldsymbol{u}_{n+1})]\end{aligned} \tag{12}$$

As shown in Eq. (12), the computational complexity of KMC is $O(N)$, where N is the number of training data. In conclusion, the learning algorithm is as follows:

Algorithm 1

```
Initialization
$\eta$: learning rate
$\kappa$: universal kernel
$e_1 = d_1$;
$y_1 = \eta e_1 exp(\frac{-e_1^2}{2\sigma^2})$;
Computation
while $\{\boldsymbol{u}_n, d_n\}$ available do
    $y_n = \eta \sum_{i=1}^{n-1}[exp(\frac{-e_i^2}{2\sigma^2})e_i\kappa(\boldsymbol{u}_i, \boldsymbol{u}_n)]$;
    $e_n = d_n - y_n$;
end while
```

B. Convergence Restriction

Stability is an extremely important aspect in adaptive filtering. In this part, we use the energy conservation relation to analyze well-posedness of our algorithm and derive bounds on the step-size for stability.

An ideal adaptive filter in RKHS attempts to find a weight vector $\boldsymbol{\Omega}^*$,

$$d_n = \boldsymbol{\Omega}^{*T}\boldsymbol{\varphi}_n + v_n \tag{13}$$

where v_n is a measurement noise and modeling errors. The adaptive algorithm yields the prediction error

$$\begin{aligned}e_n &= d_n - \boldsymbol{\Omega}_n^T\boldsymbol{\varphi}_n \\ &= \boldsymbol{\Omega}^{*T}\boldsymbol{\varphi}_n - \boldsymbol{\Omega}_n^T\boldsymbol{\varphi}_n + v_n \\ &= \tilde{\boldsymbol{\Omega}}_n^T\boldsymbol{\varphi}_n + v_n\end{aligned} \tag{14}$$

in which $\tilde{\boldsymbol{\Omega}}_n = \boldsymbol{\Omega}^* - \boldsymbol{\Omega}_n$ is the weight-error vector in RKHS at iteration n. At the next iteration the weight-error vector can be written as,

$$\begin{aligned}\tilde{\boldsymbol{\Omega}}_{n+1} &= \boldsymbol{\Omega}^* - \boldsymbol{\Omega}_{n+1} \\ &= \tilde{\boldsymbol{\Omega}}_n + \boldsymbol{\Omega}_n - \boldsymbol{\Omega}_{n+1} \\ &= \tilde{\boldsymbol{\Omega}}_n - \triangle\boldsymbol{\Omega}_n\end{aligned} \tag{15}$$

In order to study the filter learning process, the prior and posteriori errors are defined as

$$e_n^a = \tilde{\boldsymbol{\Omega}}_n^T\boldsymbol{\varphi}_n \qquad e_n^p = \tilde{\boldsymbol{\Omega}}_{n+1}^T\boldsymbol{\varphi}_n \tag{16}$$

Such that,

$$\begin{aligned} e_n^p &= \tilde{\mathbf{\Omega}}_{n+1}^T \boldsymbol{\varphi}_n \\ &= (\mathbf{\Omega}^* - \mathbf{\Omega}_{n+1})^T \boldsymbol{\varphi}_n \\ &= (\mathbf{\Omega}^* - (\mathbf{\Omega}_n + \triangle\mathbf{\Omega}_n))^T \boldsymbol{\varphi}_n \\ &= (\tilde{\mathbf{\Omega}}_n - \triangle\mathbf{\Omega}_n)^T \boldsymbol{\varphi}_n \\ &= e_n^a - \triangle\mathbf{\Omega}_n^T \boldsymbol{\varphi}_n \\ &= e_n^a - \eta g(e_n)\kappa(\boldsymbol{u}_n, \boldsymbol{u}_n) \end{aligned} \tag{17}$$

For Normalized Gaussian Kernel, $\kappa(\boldsymbol{u}_n, \boldsymbol{u}_n) = 1$. Therefore, Eq. (17) can be simplified to

$$e_n^p = e_n^a - \eta g(e_n) \tag{18}$$

Combining Eq. (17) and Eq. (15), we get,

$$\begin{aligned} \tilde{\mathbf{\Omega}}_{n+1}^T &= \tilde{\mathbf{\Omega}}_n^T + (e_n^p - e_n^a)/\boldsymbol{\varphi}_n^T \\ &= \tilde{\mathbf{\Omega}}_n^T + (e_n^p - e_n^a)\boldsymbol{\varphi}_n/\kappa(\boldsymbol{u}_n, \boldsymbol{u}_n) \\ &= \tilde{\mathbf{\Omega}}_n^T + (e_n^p - e_n^a)\boldsymbol{\varphi}_n \end{aligned} \tag{19}$$

Based on the energy conservation relation, both sides of Eq.(19) should have the same energy,

$$\|\tilde{\mathbf{\Omega}}_{n+1}\|_F^2 = \|\tilde{\mathbf{\Omega}}_n + (e_n^p - e_n^a)\boldsymbol{\varphi}_n\|_F^2 \tag{20}$$

Expanding this equation,

$$\begin{aligned} \tilde{\mathbf{\Omega}}_{n+1}^T \tilde{\mathbf{\Omega}}_{n+1} &= [\tilde{\mathbf{\Omega}}_n + (e_n^p - e_n^a)\boldsymbol{\varphi}_n]^T \times [\tilde{\mathbf{\Omega}}_n + (e_n^p - e_n^a)\boldsymbol{\varphi}_n] \\ &= \tilde{\mathbf{\Omega}}_n^T \tilde{\mathbf{\Omega}}_n + 2\tilde{\mathbf{\Omega}}_n^T (e_n^p - e_n^a)\boldsymbol{\varphi}_n + (e_n^p - e_n^a)^2 \\ &= \tilde{\mathbf{\Omega}}_n^T \tilde{\mathbf{\Omega}}_n - 2\eta e_n^a g(e_n) + \eta^2 g(e_n)^2 \end{aligned} \tag{21}$$

Such that, we obtain,

$$E[\|\tilde{\mathbf{\Omega}}_{n+1}\|_F^2] = E[\|\tilde{\mathbf{\Omega}}_n\|_F^2] - 2\eta E[e_n^a g(e_n)] + \eta^2 E[g(e_n)^2] \tag{22}$$

In order to guarantee a convergence solution, the energy of weight-error vector should decrease gradually. Therefore,

$$\begin{aligned} & E[\|\tilde{\mathbf{\Omega}}_{n+1}\|_F^2] \le E[\|\tilde{\mathbf{\Omega}}_n\|_F^2] \\ & \Leftrightarrow -2\eta E[e_i^a g(e_n)] + \eta^2 E[g(e_n)^2] \le 0 \\ & \Leftrightarrow \eta \le \frac{2E[e_n^a g(e_n)]}{E[g(e_n)^2]} \end{aligned} \tag{23}$$

Therefore, as long as the stepsize of KMC satisfies Eq. (23) the sequence $E[\|\tilde{\mathbf{\Omega}}_i\|_F^2]$ is bounded from below, and the learning process is stable.

C. Self Regularization

Does our method face the ill-posed problem due to small data size or severe noise, like Least Square (LS)? In the LS problem, the Tikhonov regularization [8] is widely used to deal with this issue. If the same method is applied to our method, a regularization term is introduced to the MCC cost function

$$\max_{\mathbf{\Omega}} R_{emp}[\mathbf{\Omega} \in F, Z^N] = \sum_{i=1}^{N} \kappa_\sigma(d_i, \mathbf{\Omega}(\boldsymbol{\varphi}_i)) + \lambda\|\mathbf{\Omega}\|_F^2 \tag{24}$$

this optimization problem is equivalent to the following problem:

$$\begin{aligned} \max_{\mathbf{\Omega}} R_{emp}[\mathbf{\Omega} \in F, Z^N] &= \sum_{i=1}^{N} \kappa_\sigma(d_i, \mathbf{\Omega}(\boldsymbol{\varphi}_i)) \\ & subjectto\|\mathbf{\Omega}\|_F^2 \le C \end{aligned} \tag{25}$$

which was already proven in [8]. Therefore, the following conclusion can be obtained: constraining the norm of the solution has the same result as adding a regularization term in the KMC.

As mentioned in previous section, $\|\mathbf{\Omega}\|_F^2$ monotonically decreases as iterations increase as long as the stepsize satisfies the constrain of Eq. (23). Hence, for any positive value C, we can find a $\mathbf{\Omega}$ with appropriate stepsize and initial condition, such that $\|\mathbf{\Omega}\|_F^2 \le C$. To conclude, KMC learning is a self-regularization method under the appropriate stepsize and inital condition.

IV. Simulation Results

Frequency doubling is an obvious nonlinear problem. In this simulation, the input and desire data for the system both are sine wave with f_0 and $2f_0$ respectively, as shown in Fig. 1. 1500 samples from two sequences are segmented for the training data, and 200 samples as the test data. We use an impulsive Gaussian mixture model to simulate the influence of the non-gaussian noise, whose probability density function is:

$$p_{noise}(i) = 0.9N(0, 0.01) + 0.1N(2, 0.01) \tag{26}$$

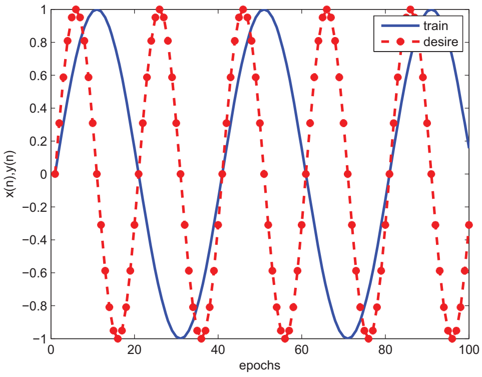


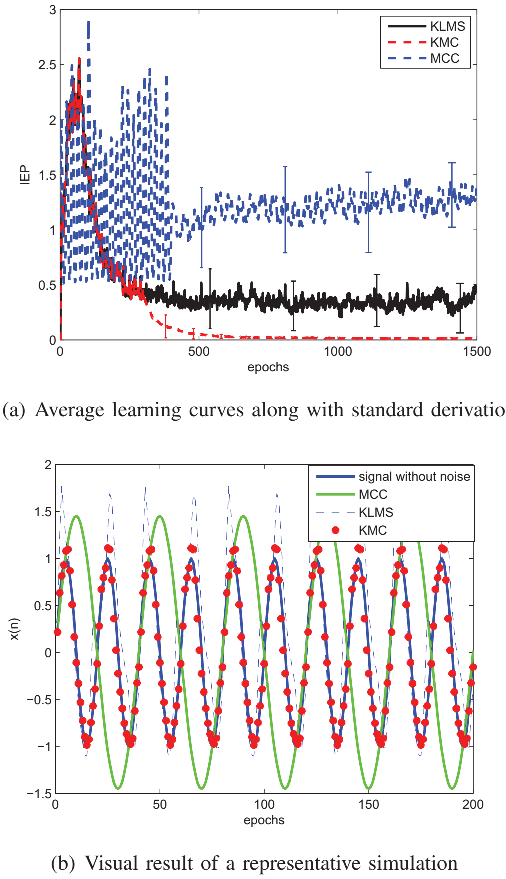
Fig. 1. Simulation data

1) KMC Performance Compared with KLMS and MCC with linear adaptive filter : The input vector is dimension 2 (current and one past sample). Under kernel learning process the learning rate is 0.9, and it is 0.2 for MCC with linear filter. Moreover, the kernel sizes for kernel-induced mapping in KMC and KLMS are 0.5, and the kernel sizes for correntropy criterion in KMC and MCC with linear adaptive filter are set to 0.4 which performs best on the test data. Meanwhile, in order to guarantee KMC and MCC with linear filter reach the global optimal solution, we first train these filters with MSE criterion during the first 300 samples. 100 Monte-Carlo simulations are run for the same data with 100 different starts. All of results are presented with respect to intrinsic error power (IEP) on clean test data, where clean data means desired signal without noise. That is

$$IEP = \mathbf{E}[d - f(\boldsymbol{u})]^2 \tag{27}$$

in which, d and $\boldsymbol{u}$ are the desire and input of clean test data respectively, and $f(\boldsymbol{u})$ is the system output for corresponding test data.

The average learning curves accompanied with standard derivation are shown in Fig. 2a), and the final estimated IEP for MCC with linear filter is 1.2793 ± 0.2475, for KLMS is 0.4765 ± 0.3418 and for KMC is 0.0109 ± 0.0033 (Note that all of these results are summarized in the form of "average ± standard deviation"). All of these results show that the performance of KMC is much better than KLMS and linear filter with MCC. Not only the mean of IEP of KMC is smaller than other two algorithms, but also the output range is narrower. Fig. 2b) is a representative visual result. The MCC with linear filter is incapable of approximating the nonliner transfer function required for the frequency doubling. Even though KLMS follows the frequency doubling, the result is influenced by the noise. However, notice that the kernel width in the correntropy criterion must be selected by the user to attenuate the outliers, which depends on the application.



(a) Average learning curves along with standard derivation

(b) Visual result of a representative simulation

Fig. 2. Performance of different algorithms

2) The effect of the kernel width on MCC in KMC: The kernel width for correntropy is very important, having even more influence than the kernel that defines the RKHS. Actually, correntropy is a measure to estimate how similar two random variables in a local range controlled by the kernel width. Kernel width of MCC affects some important properties of adaptation such as the nature of the performance surface, presence of local optima, rate of convergence and robustness to impulsive noise during adaptation [19].

In this section, the effect of the kernel width on MCC in KMC is demonstrated. We choose eight kernel widths: 0.1, 0.3, 0.4, 0.7, 1.0, 1.5, 4 and the value obtained by Silverman's rule. Similarly, 100 Monte Carlo simulations with different noises are run to study the effect of kernel width, while all the other parameters of filters are the same as the previous simulation. Table I presents the results. From this table, KMC performs at the same level for a large range of kernel sizes, i.e. when the kernel width is in the range of $[0.3, 1.0]$. If a set of kernel sizes σ_{SM} by applying Silverman's rule to the prediction error, σ_{SM} vraies between [0.0602, 0.15], which is in the neighborhood of the best values mentioned above. A large kernel width initially is beneficial to avoid local optimal solution, and the global optimal point will obtain by using a large kernel width. At the same time, large kernel width decreases the "window" effect of correntropy, and adaptive systems with too large kernel width degenerate to those trained with MSE criterion. Therefore an appropriate kernel width is a compromise between global optima and noise outlier cancellation. However, KMC with large kernel width will not perform worse than KLMS.

TABLE I

Effect of Kernel width on MCC in KMC

σ	IEP
0.1	0.3241 ± 0.3279
0.3	0.0119 ± 0.0060
0.4	0.0109 ± 0.0033
0.7	0.0113 ± 0.0038
1.0	0.0209 ± 0.0113
1.5	0.1203 ± 0.1257
4.0	0.2386 ± 0.1966
Silverman's	0.1724 ± 0.1339

V. Conclusions

Owing to universal nonlinear approximation, linearity in RKHS, and convexity in hypothesis space, kernel adaptive filters are widely used in many applications. However, MSE criterion which is popular in most conventional adaptive filters is not appropriate for non-Gaussian and nonlinear cases. As an alternative option, MCC has been proved to be more efficient and robust than MSE criterion in these situations. This paper combines the advantages of these two approaches and brings a new algorithm called Kernel Maximum Correntropy. As shown theoretically and experimentally, the performance of this algorithm is superior to KLMS and conventional linear filters with MCC. Besides, the computational complexity of KMC is similar to that of KLMS, being $O(N)$.

Although the KMC filter obtains good performance, the kernel width in the correntropy criterion must be selected according to the application to attenuate the outliers. We have shown that, when proper kernel width is selected, the KMC provides better performance to attenuate the outliers. How to select the kernel width appropriately and adaptively is the most important problem to be solved in the future. The network size of RBF increases linearly with data number, which is a common bottleneck for both KLMS and KMC when applied to continuous adaption. Therefore, another interesting direction for future work is how to restrict the computation complexity efficiently.

REFERENCES

[1] W. Liu, J. Principe and S. Haykin, *Kernel Adaptive Filtering: A Comprehensive Introduction*, WILEY, 2010.

[2] I. Steinwart, "On the influence of the kernel on the consistency of support vector machines," *Journal of Machine Learning Research*, vol. 2, pp. 67-93, 2001.

[3] V. Vapnik, *The nature of statistical learning theory*, Springer, New York, 1995.

[4] F. Girosi, M. Jones and T. Poggio, "Regularization theroy and neural networks architectures," *Neural Compuatation*, vol. 7, pp. 219-269, 1995.

[5] B. Scholkopf, A. Smola and K. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Compuatation*, vol. 10, no.5, pp. 1299-1319, Jul. 1998.

[6] M. Sebastian, R. Gunnar , W. Jason and S. Bernhard, "Fisher Discriminant Analysis With Kernels," *Neural Networks Signal Process Processing*, pp. 41-48, Aug. 1999.

[7] Y. Engel, S. Mannor and R. Meir, "The Kernel Recursive Least-Squares Algorithm," *IEEE Transactions on signal processing*, vol. 52, no.8, pp. 2275-2285, 2004.

[8] W. Liu, P. Pokharel and J. Principe, "The kernel least mean quare algorithm," *IEEE Transactions on Signal Processing*, vol. 56, iss. 2, pp. 543-554, 2008.

[9] E. Parzen, "On the estimation of a probability density function and the mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065-1076, 1962.

[10] W. Liu, P. Pokharel, J. Principe, "Correntropy: Properties and Applications in Non-Gaussian Signal Processing," *IEEE Transactions on Signal Processing*, vol. 55, iss. 11, pp. 5286-5298, 2007.

[11] J. Xu, P. Pokharel, A. Paiva and J. Principe, "Nonlinear Component Analysis Based on Correntropy," *International Joint Conference on Neural Networks*, pp. 1851-1855, 2006.

[12] A. Gunduz and J. Principe, "Correntropy as a Novel Measure for Nonlinearity Tests," *Signal Processing*, vol. 89, iss. 1, pp. 14-23, Jan. 2009.

[13] K. Jeong and J. Principe, "The Correntropy MACE Filter for Image Recognition," *IEEE International Workshop on Machine Learning for Signal Processing*, pp. 9-14, Sep. 2006.

[14] I. Park and J. Principe, "Correntropy based Granger Causality," *International Conference on Acoustics, Speech and Signal Processing*, pp. 3605-3608, Mar. 2008.

[15] A. Singh, J. Principe, "Using correntropy as a cost function in linear adaptive filters," *Proceedings of International Joint Conference on Neural Network*, pp. 2950-2955, Jun. 2009.

[16] A. Singh, J. Principe, "A loss function for classification based on a robust similarity metric," *Proceedings of International Joint Conference on Neural Network*, pp. 1-6, Jul. 2010.

[17] I. Santamaria, P. Pokarel and J. Principe, "Generalized correlation function: Definition, properties and application to blind equalization," *IEEE Transactions on Signal Processing*, vol. 54, no. 6, pp. 2187-2197, 2006.

[18] C. Micchelli, Y. Xu, H. Zhang and G. Lugosi, "Universal Kernel," *The Journal of Machine Learning Research*, vol. 7, pp. 2651-2667, 2006.

[19] A. Singh, J. Principe, "Information theoretic learning with adaptive kernels," *Signal Processing*, vol. 91, iss. 2, pp. 203-213, Feb. 2011.