Convex Regularized Recursive Maximum Correntropy Algorithm

Xie Zhang¹, Kaixin Li¹, Zongze Wu¹, Yuli Fu¹, Haiquan Zhao², Badong Chen³

1. School of Electronic and Information Engineering, South China University of Technology, Guangzhou, 510640, China
2. School of Electrical Engineering, Southwest Jiaotong University, Chengdu, China
3. School of Electronic and Information Engineering, Xi’an Jiaotong University, Xi’an, 710049, China

Abstract—In this brief, a robust and sparse recursive adaptive filtering algorithm, called convex regularized recursive maximum correntropy (CR-RMC), is derived by adding a general convex regularization penalty term to the maximum correntropy criterion (MCC). An approximate expression for automatically selecting the regularization parameter is also introduced. Simulation results show that the CR-RMC can significantly outperform the original recursive maximum correntropy (RMC) algorithm especially when the underlying system is very sparse. Compared with the convex regularized recursive least squares (CR-RLS) algorithm, the new algorithm also shows strong robustness against impulsive noise. The CR-RMC also performs much better than other LMS-type sparse adaptive filtering algorithms based on MCC.

Key Words: maximum correntropy criterion (MCC), sparse adaptive filtering, recursive maximum correntropy (RMC), convex regularized recursive maximum correntropy (CR-RMC).

I. INTRODUCTION

Sparse adaptive filtering has found many applications, such as acoustic and network echo cancellation [1] and communication channel identification [2]. There are a lot of adaptive algorithms for sparse system identification (where the impulse response to be identified contains many near-zero coefficients and only a few large ones) which use a priori knowledge of the sparsity. In [3], a sparse least mean square (LMS) algorithm, called the \( l_0 - LMS \), was derived by adding a convex approximation for the \( l_0 \) norm penalty to the squared error cost function. The \( l_1 \) norm and log-sum terms were also used to regularize the LMS algorithm to sparse solutions [4]. In [5], a more general convex function was proposed to regularize the...
LMS algorithm. In [6], a sparse recursive least squares (RLS) algorithm was developed by adding a weighted $l_1$ norm penalty to the least squared cost function. In [7], a general convex function was proposed to substitute the $l_1$ norm, and the resulting RLS-type algorithm is called the convex regularized recursive least squares (CR-RLS) algorithm. This generalization allows utilization of any convex function for regularization. In recent years, the sparsity idea has also been applied in kernel adaptive filtering [8-12].

The algorithms mentioned above usually perform well in the presence of Gaussian noise. However, their performance may deteriorate seriously in non-Gaussian situations, especially when the signals are disturbed by some impulsive noises, which are rather common in real world applications. Recently, the maximum correntropy criterion (MCC) [13-22] has been successfully used to derive various robust adaptive filtering algorithms which can perform very well in impulsive noises. Since correntropy is a local similarity measure, it is insensitive to large outliers. In [14], the correntropy was used as a cost function for a linear adaptive filter. A kernel adaptive filtering algorithm, called the kernel maximum correntropy (KMC) was derived in [15]. An RLS-type recursive algorithm under MCC criterion was derived in [20] and its kernelized version was presented in [21]. More recently, under the MCC criterion, several LMS-type (i.e. stochastic gradient based) robust and sparse adaptive filtering algorithms were developed in [22]. The goal of this work is to present an RLS-type robust and sparse adaptive filtering algorithm, called convex regularized recursive maximum correntropy (CR-RMC), which is derived by using a general convex function to regularize the MCC cost. The new algorithm can achieve excellent performance for sparse system identification and also show strong robustness under impulsive noise.

The rest of the brief is organized as follows. In section II, we develop the CR-RMC algorithm. In section III, we present simulation results to demonstrate the performance of CR-RMC. Finally, we give the conclusion in section IV.

II. CONVEX REGULARIZED RMC ALGORITHM

2.1 Maximum Correntropy Criterion

First, we briefly review the maximum correntropy criterion (MCC) for adaptive system training.
Given two random variables \( X \) and \( Y \), the correntropy is defined by [13,23]

\[
V(X,Y) = E[\kappa(X,Y)] = \int \kappa(x,y) dF_{X,Y}(x,y)
\]

(1)

where \( \kappa(\cdot) \) is a \textit{shift-invariant} Mercer kernel, and \( F_{X,Y}(x,y) \) denotes the joint distribution function of \((X,Y)\).

In this work, the following Gaussian kernel is selected as the kernel function in correntropy:

\[
\kappa_\sigma(x,y) = \exp\left(-\frac{||x-y||^2}{2\sigma^2}\right)
\]

(2)

where \( \sigma > 0 \) stands for the kernel size (or kernel bandwidth). Taking Taylor series expansion around \( X - Y = 0 \), we have

\[
V_0(X,Y) = \sum_{n=0}^{\infty} \frac{(-1)^n}{2^n n!} E[\frac{(X - Y)^{2n}}{\sigma^{2n}}]
\]

(3)

As we can see, correntropy contains higher order moments of the error \( e = X - Y \). Correntropy measures how similar two random variables are within a neighborhood controlled by the kernel width \( \sigma \). This property is very useful to reduce the bad influence of outliers or impulsive noises.

Correntropy can be used as a cost function for robust machine learning [24-28] and adaptive filtering [14-22]. Let \( \{x_n,d_n\}_{n=1}^{N} \) be a finite number of training data, with \( x_n \) being the input vector, and \( d_n \) the desired signal, generated via

\[
d_n = w^T x_n + \eta_n
\]

(4)

where \( w^* = [w_0,w_1,\ldots,w_{M-1}]^T \) denotes an unknown finite-impulse-response (FIR) weight vector to be estimated, \( x_n = [x(n),x(n-1),\ldots,x(n-M+1)]^T \) with \( M \) being the filter order, and \( \eta_n \) accounts for the observation noise at instant \( n \). Under MCC, the objective of adaptive filtering is to find an optimal weight vector \( \hat{w} \) by maximizing the correntropy between the desired signal \( d_n \) and the filter output. The MCC cost function with exponential forgetting factor \( \lambda \) is given by

\[
L_n = \sum_{n=0}^{\infty} \lambda^{n-m} \exp\left(-\frac{e_n^2}{2\sigma^2}\right)
\]

(5)

where \( w_n = [w_{0,n},w_{1,n},\ldots,w_{M-1,n}]^T \) denotes the weight vector of an adaptive filter at instant \( n \), and \( e_n \) is the
estimation error:

\[ e_n = d_n - w_n^T x_n \]  

(6)

2.2 CR-RMC Algorithm

We can regularize the MCC cost function by including a convex function of the weight vector. In this way, some a priori knowledge, such as sparsity, about the unknown system can be utilized [7]. In this work, the regularized MCC cost function is

\[ J_n = \sum_{n=0}^{N} \lambda^{n-m} \exp\left(\frac{(d_n - w_n^T x_n)^2}{2\sigma^2}\right) - \gamma_n f(w_n) \]  

(7)

where \( f(.) \) is a convex function, and \( \gamma_n \) is a positive number that can change with time. The parameter \( \gamma_n \) provides a balance between the MCC cost and the regularizing term. When \( \gamma_n \) is larger, the regularizing term will have more influence on the adaptation. There are subgradients for a convex and non-differentiable function [29]. The collection of subgradients is called subdifferential, which is denoted by \( \partial f(w_n) \). We denote a subgradient vector of \( f \) at \( v \) with \( \nabla^s f(v) \in \partial f(v) \). The optimal weight vector under the cost function \( J_n \) can be found by setting the subgradient \( \nabla^s J_n \) equal to 0. After some straightforward manipulations, we obtain

\[ \sum_{n=0}^{N} \lambda^{n-m} \left\{ \exp\left(\frac{(d_n - \tilde{w}_n^T x_n)^2}{2\sigma^2}\right) \cdot x_{n-k} \cdot (d_n - \sum_{k=0}^{N-1} \tilde{w}_k x_{n-k}) \right\} = \sigma^2 \gamma_n \left\{ \nabla^s f(\tilde{w}_n) \right\}_i \]  

(8)

where \( \left\{ \nabla^s f(\tilde{w}_n) \right\}_i \) denotes the \( i \)th component of \( \nabla^s f(\tilde{w}_n) \). One can write (8) in a matrix form:

\[ \Phi_n \tilde{w}_n = r_n - \sigma^2 \gamma_n \nabla^s f(\tilde{w}_n) \]  

(9)

where \( \Phi_n \in \mathbb{R}^{N \times N} \) is a weighted autocorrelation matrix for the input signal \( x_n \):

\[ \Phi_n = \sum_{m=0}^{N} \lambda^{n-m} \exp\left(-\frac{e_n^2}{2\sigma^2}\right) \cdot x_m \cdot x_m^T \]  

(10)

\[ = \lambda \Phi_{n-1} + \exp\left(-\frac{e_n^2}{2\sigma^2}\right) \cdot x_n \cdot x_n^T \]

and \( r_n \in \mathbb{R}^N \) is a weighted cross-correlation vector between \( x_n \) and \( d_n \).
\( \mathbf{r}_n = \sum_{m=0}^{n} \lambda^{n-m} \exp\left(-\frac{e_n^2}{2\sigma^2}\right) d_m \cdot \mathbf{x}_m \)  
(11)

\( = \lambda \mathbf{r}_{n-1} + \exp\left(-\frac{e_n^2}{2\sigma^2}\right) d_n \cdot \mathbf{x}_n \)

\( \Phi_n \) and \( \mathbf{r}_n \) can be updated by the above two recursive equations. The right hand side of (9) can be defined as a new vector \( \mathbf{\theta}_n \):

\( \mathbf{\theta}_n = \mathbf{r}_n - \sigma^2 \gamma_n \nabla^i f(\mathbf{w}_n) \)  
(12)

The above equation depends on the update equation (11). Assume that \( \gamma_{n-1} \) and \( \nabla f(\mathbf{w}_{n-1}) \) do not change considerably over a single time step [7]. Then the vector \( \mathbf{\theta}_n \) can also be recursively updated as

\( \mathbf{\theta}_n = \lambda \mathbf{\theta}_{n-1} + \exp\left(-\frac{e_n^2}{2\sigma^2}\right) d_n \mathbf{x}_n - \gamma_{n-1} (1-\lambda) \sigma^2 \gamma_n \nabla^i f(\mathbf{w}_{n-1}) \)  
(13)

The equation (9) can be rewritten as

\( \mathbf{\hat{w}}_n = \mathbf{P}_n \mathbf{\theta}_n \)  
(14)

where \( \mathbf{P}_n = \Phi_n^{-1} \). Using the matrix inversion lemma, we come up with the update equation for \( \mathbf{P}_n \):

\( \mathbf{P}_n = \lambda^{-1} \left( \mathbf{P}_{n-1} - \mathbf{k}_n \mathbf{x}_n^T \mathbf{P}_{n-1} \right) \)  
(15)

where the gain vector \( \mathbf{k}_n \) is

\( \mathbf{k}_n = \mathbf{P}_{n-1} \mathbf{x}_n / (\lambda[\exp(-\frac{e_n^2}{2\sigma^2})]^{-1} + \mathbf{x}_n^T \mathbf{P}_{n-1} \mathbf{x}_n) \)  
(16)

Rewriting (14) in a recursive way, one can get the following update equation for \( \mathbf{\hat{w}}_n \):

\( \mathbf{\hat{w}}_n = \mathbf{\hat{w}}_{n-1} + \mathbf{k}_n e_n - \sigma^2 \gamma_{n-1} (1-\lambda) \nabla^i f(\mathbf{w}_{n-1}) \)  
(17)

If the subgradient term vanishes, the above update equation will reduce to the standard recursive maximum correntropy algorithm (RMC) [20, 21]:

\( \mathbf{\hat{w}}_n = \mathbf{\hat{w}}_{n-1} + \mathbf{k}_n e_n \)  
(18)

The update equation (17) is referred to as the convex regularized RMC (CR-RMC) algorithm.

2.3 Selection of the Regularization Parameter

Usually the convex penalty functions can be used to provide some a priori knowledge about the
unknown system. Here we assume that the a priori knowledge is an upper bound on the convex function

\[ f(w^*) \leq \rho \]  

Let \( \tilde{w}_n \) be the solution to the normal equation \( (\Phi_n w = r_n) \) without regularization. The deviations of the weight vectors \( \tilde{w}_n \) and \( \tilde{w}_n \) from the true weights are denoted as \( \tilde{e}_n = \tilde{w} - w^* \) and \( \tilde{\epsilon}_n = \tilde{w}_n - w^* \), respectively. We can get the following equation from (9)

\[ \tilde{e}_n = \tilde{\epsilon}_n - \sigma^2 \gamma_n P_n \nabla f(\tilde{w}_n) \]  

The squared deviation (SD) for \( \tilde{e}_n \) can be calculated as

\[ \hat{D}_n = \hat{\epsilon}_n^T \hat{\epsilon}_n = \|\tilde{\epsilon}_n\|^2 \]
\[ = \hat{D}_n - 2\sigma^2 \gamma_n \|P_n \nabla f(\tilde{w}_n)\|^2 \]  

where \( \hat{D}_n = \|\tilde{\epsilon}_n\|^2 \) is the SD for \( \tilde{\epsilon}_n \). One can obtain the following theorem from the equation (21) [7].

**Theorem 1:** \( \hat{D}_n \leq \hat{D}_n \) if \( \gamma_n \in [0, \max(\gamma_n, 0)] \), where

\[ \hat{\gamma}_n = 2 \frac{\nabla f(\tilde{w}_n)^T P_n \tilde{\epsilon}_n}{\sigma^2 \|P_n \nabla f(\tilde{w}_n)\|^2} \]  

**Proof:** When \( \sigma^2 \gamma_n \|P_n \nabla f(\tilde{w}_n)\|^2 \leq 2\gamma_n \|P_n \nabla f(\tilde{w}_n)\|^2 \), we get \( \hat{D}_n \leq \hat{D}_n \). Therefore we give a constraint on \( \gamma_n \):

\[ 0 \leq \gamma_n \leq 2 \frac{\nabla f(\tilde{w}_n)^T P_n \tilde{\epsilon}_n}{\sigma^2 \|P_n \nabla f(\tilde{w}_n)\|^2} \]  

if \( \nabla f(\tilde{w}_n)^T P_n \tilde{\epsilon}_n \geq 0 \), and \( \gamma_n = 0 \) if \( \nabla f(\tilde{w}_n)^T P_n \tilde{\epsilon}_n < 0 \).

Theorem 1 indicates that the CR-RMC algorithm can achieve a mean squared deviation (MSD) less than or equal to that of the original RMC algorithm. However, we cannot evaluate \( \hat{\gamma}_n \) in fact, because the true weight vector is needed for calculating \( \tilde{\epsilon}_n \). In a similar fashion to [7] and by using the definition of the subgradient of a convex function [29], one can get an approximate expression of \( \hat{\gamma}_n \) provided that the input is white and \( n \) is large enough:

\[ \hat{\gamma}_n \approx \gamma_n' = 2 \frac{\text{tr}(P_n) (f(\tilde{w}_n) - \rho) + \nabla f(\tilde{w}_n)^T P_n \epsilon'}{\sigma^2 \|P_n \nabla f(\tilde{w}_n)\|^2} \]  

\[ \text{(23)} \]
where $\hat{\varepsilon}'_n = \hat{\mathbf{w}}_n - \hat{\mathbf{w}}_n$. Here $\text{tr}(\cdot)$ denotes the trace operator. Equation (23) provides an approximation for calculating $\hat{\gamma}_n$ when the input is white. Thus $\hat{\gamma}_n$ can be automatically updated at every iteration by $\hat{\gamma}_n = \max(\hat{\gamma}'_n, 0)$. Automatically updating $\gamma_n$ requires only $O(N^2)$. This method can save us much time in finding an optimal value of $\gamma$.

III. SIMULATION RESULTS

In this section, we apply the CR-RMC algorithm to identify sparse systems by employing two sparsity inducing convex penalty functions. Because the $l_0$ norm is a non-convex function, we adopt two approximations of $l_0$ norm. One is the $l_1$ norm, for which we have $f(\mathbf{w}) = \|\mathbf{w}\| = \sum_{k=0}^{N-1} |w_k|$ , with subgradient $\nabla^i(\|\mathbf{w}\|) = \text{sgn}(\mathbf{w})$ [4, 6], where $\text{sgn}(\cdot)$ is the component-wise sign function. Another approximation for $l_1$ norm is as follows [6]:

$$\|\mathbf{w}\| \approx f^\beta(\mathbf{w}) = \sum_{k=0}^{N-1} (1 - e^{-\beta|w_k|})$$

(24)

where $\beta$ is an appropriate constant. The subgradient of (24) can be, approximately, calculated as [3]:

$$\nabla^i f^\beta(\mathbf{w})_k = \begin{cases} \beta \text{sgn}(w_k) - \beta^2 w_k, & |w_k| \leq \frac{1}{\beta} \\ 0, & \text{elsewhere} \end{cases}$$

(25)

Equation (25) has been utilized in LMS, resulting in $l_0-LMS$ [3]. In the following, the CR-RMC algorithms with the $l_1$ norm constraint and the penalty function (24) are called respectively as the $l_1-RMC$ and $l_0-RMC$.

The first sparse system to be identified has a total of $M = 64$ taps, of which $S$ tap weights are nonzero. The positions of nonzero weights are chosen randomly and their values are generated from a Gaussian distribution $N(0, (1/S))$. The input signal is $x_n - N(0,1)$. The regularization parameter in CR-RMC algorithms is determined in two different ways. One is chosen as a constant value $\gamma_n = \gamma$, another is set automatically as $\gamma_n = \max(\gamma', 0)$. For constant case, an optimal value of $\gamma$, with which a minimum steady-state MSD is achieved, can be found by trial and error method. For the $l_0-RMC$ we set $\rho = \|\mathbf{w}\|$, with $\rho = \max(\rho', 0)$.
while for the $l_p - RMC$, $\rho = \|w\|$. Other parameters are set at $\lambda = 0.995$, $\delta = 1$. And for $l_p - RMC$, $\beta = 50$. All the simulation results are obtained by averaging over 1000 independent Monte Carlo runs.

First, we show the performance of the CR-RMC algorithms ($l_0 - RMC$ and $l_1 - RMC$) under Gaussian noise. For comparison purpose we also show the performance of the original RMC algorithm. Assume that the unknown system is sparse with $S = 4$, and the disturbance noise is zero-mean with a proper variance such that $SNR = 20 dB$ (here the SNR is calculated from the ratio between the input variance and the noise variance). Fig.1 illustrates the convergence curves in terms of the mean squared deviation (MSD). In Fig.1, $Auto l_0 - RMC$ and $Auto l_1 - RMC$ denote, respectively, $l_0 - RMC$ and $l_1 - RMC$ with automatically determined $\gamma_n$. From Fig.1 we observe: 1) the CR-RMC algorithms perform much better (i.e. achieve much lower MSD) than the original RMC algorithm; 2) the $l_0 - RMC$ can outperform the $l_1 - RMC$ in sparse system identification; 3) the CR-RMC algorithms with automatically determined $\gamma_n$ can achieve almost the same steady-state MSDs as those of the CR-RMC algorithms with optimized constant $\gamma$ (For $l_0 - RMC$, the optimal value is $\gamma = 1.5$, while for $l_1 - RMC$, $\gamma = 0.2$).

Second, we compare the performance of the $Auto l_0 - RMC$, $Auto l_1 - RMC$, and the original RMC for different values of $S$ ($S = 2$, 4, 8 and 64). Table I presents the steady-state MSDs. It can be seen that the performance of RMC does not change with different $S$. When the system is highly sparse (i.e. when $S$ is small), the $Auto l_0 - RMC$ and $Auto l_1 - RMC$ perform much better than the RMC algorithm. All the algorithms achieve almost the same steady-state MSDs when the underlying system is totally non-sparse ($S = 64$). Similar results can be observed from Fig. 2 where the convergence curves of the $Auto l_0 - RMC$ and RMC for different $S$ are plotted.

Third, we compare the performance of the CR-RMC and CR-RLS algorithms for sparse system identification under impulsive noise. We set $S = 4$. The impulsive noise is modeled as a symmetric alpha-stable ($S\alpha S$) distribution [22] with characteristic factor $\alpha=1.7$ and dispersion parameter $\gamma=0.1$. The average convergence curves in terms of the MSD are shown in Fig. 3. For comparison purpose, we also
include in Fig. 3 the results of the unregularized RLS and RMC algorithms. One can observe that $Auto \ l_0 - RMC$, $Auto \ l_1 - RMC$ and RMC have strong ability to suppress the adverse effects of impulsive disturbances, while $Auto \ l_0 - RLS$, $Auto \ l_1 - RLS$ and RLS are sensitive to outliers and their convergence curves contain large fluctuations. Simulation results confirm the fact that the MCC is robust with respect to impulsive noises. In addition, $Auto \ l_0 - RMC$ and $Auto \ l_1 - RMC$ perform much better than the unregularized RMC algorithm.

Finally, we identify another sparse system and demonstrate the performance of the CR-RMC and several LMS-type sparse adaptive filtering algorithms under MCC, including ZAMCC, RZAMCC, and CIMMCC algorithms [22]. Assume that the parameter vector of the unknown channel is time-varying, given by

$$\begin{align*}
    w^* &= \begin{cases} [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] & n \leq 2000 \\
                     [1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0] & 2000 < n \leq 4000 \\
                     [1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1] & 4000 < n
\end{cases}
\end{align*}$$

(26)

As one can see, the unknown system becomes non-sparse as $n$ increasing. We consider again the case in which there is an $\alpha$-S impulsive noise, with $\alpha=1.7$, $\gamma=0.1$. The average convergence curves are shown in Fig. 4. Evidently, no matter how sparse the system is, the CR-RMC algorithms (especially the $l_0 - RMC$) perform very well and outperform significantly the three LMS-type algorithms.
Fig. 2. Convergence curves of $\text{Auto } l^0_{\omega} - \text{RMC}$ and RMC for different values of $S$ ($SNR = 20 dB$)

<table>
<thead>
<tr>
<th></th>
<th>$S=2$</th>
<th>$S=4$</th>
<th>$S=8$</th>
<th>$S=64$</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMC</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0017</td>
</tr>
<tr>
<td>$\text{Auto } l^0_{\omega} - \text{RMC}$</td>
<td>0.00084</td>
<td>0.00093</td>
<td>0.0011</td>
<td>0.0017</td>
</tr>
<tr>
<td>$\text{Auto } l^0_{\omega} - \text{RMC}$</td>
<td>0.00011</td>
<td>0.00017</td>
<td>0.00031</td>
<td>0.0017</td>
</tr>
</tbody>
</table>

Table I. Steady-state MSDs for different values of $S$ ($SNR = 20 dB$)

Fig. 3. Convergence curves of the CR-RMC and CR-RLS algorithms in impulsive noises ($S = 4$)
Fig.4. Convergence curves of CR-RMC and LMS-type sparse adaptive filtering algorithms under MCC

IV. CONCLUSION

A convex regularized recursive adaptive filtering algorithm, called CR-RMC, has been presented in this brief, which is derived by utilizing a general convex function to regularize a correntropy based cost function. Simulation results show that CR-RMC can perform much better than the original RMC algorithm for sparse system identification, and significantly outperform the CR-RLS algorithm in impulsive noise environments. The CR-RMC can also perform much better than the LMS-type sparse adaptive filtering algorithms under MCC.

ACKNOWLEDGEMENTS

This work was supported by 973 Program (no. 2015CB351703) and National Natural Science Foundation of China (no. 61372152, no. 61271210).

REFERENCES


