14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

ROBUST ADAPTIVE FILTERS USING STUDENT-T DISTRIBUTION

Guang Deng

Department of Electronic Engineering,
La Trobe University, Bundoora, Victoria 3086, Australia.
Email: d.deng@latrobe.edu.au.
Phone: +61 3 9479 2036 Fax: +61 3 9471 0524.

ABSTRACT

An important application of adaptive filters is in system identification. Robustness of the adaptive filters to impulsive noise has been studied. In this paper, we propose an alternative way to developing robust adaptive filters. Our approach is based on formulating the problem as a maximum penalized likelihood (MPL) problem. We use student-t distribution to model the noise and a quadratic penalty function to play a regularization role. The minorization-maximization principle is used to solve the optimization problem. Based on the solution, we propose two LMS-type of algorithms called MPL-LMS and robust MPL-LMS. The robustness of the latter algorithm is demonstrated both theoretically and experimentally.

1. INTRODUCTION

A typical application of an adaptive filter is in identifying an unknown linear system. Formulating the problem in an iterative way, we have at time n the available data, denoted $D_n = \{y_n, \mathbf{x}_n\}$ and the observation model

$$y_n = \mathbf{x}_n \mathbf{w} + e_n \tag{1}$$

where the true impulse response of the unknown system $\mathbf{w}$ is an $(M \times 1)$ vector, $\mathbf{x}_n$ is a $(1 \times M)$ input signal vector, y_n is the system output and e_n is the independent and identically distributed noise with a known distribution function. We also have the estimate of the system impulse response from the (n-1)th iteration, denoted by $\mathbf{w}_{n-1}$. A classical solution to this problem is the LMS algorithm [1]

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \lambda \hat{e} \mathbf{x}_n^T. \tag{2}$$

where λ is called the step size and $\hat{e} = y_n - \mathbf{x}_n \mathbf{w}_{n-1}$. The normalized LMS (NLMS) algorithm can be regarded as a special case of the LMS when the step size is $\lambda = \eta/(\mathbf{x}_n \mathbf{x}_n^T)$, where η is the step size of the NLMS algorithm.

A potential problem for the LMS algorithm is that it is not robust to outliers such as impulsive noise. Let us assume that $\mathbf{w}_{n-1}$ is quite close to $\mathbf{w}$, and e_n is the impulsive noise. In this case $\hat{e}$ is dominated by the noise and the second term of (2) is large. As a result $\mathbf{w}_n$ will be forced to move away from $\mathbf{w}$. Various type of robust adaptive filters have been studied to tackle this problem. These include robust recursive-least-squares filters [2,3] and robust LMS filters [4–6] which have the following form

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \Phi(\hat{e}) \hat{e} \mathbf{x}_n^T \tag{3}$$

where $\Phi(\hat{e})$ is a nonlinear function of $\hat{e}$. Comparing (2) and (3), we can see that a robust LMS algorithm has a variable step size that depends on $\hat{e}$.

In this paper, we develop robust adaptive filters from an alternative perspective. We formulate the problem as a maximum penalized likelihood (MPL) problem. The robustness is achieved by using the student-t distribution for the likelihood and the quadratic penalizing function is used for stabilizing the algorithm. This is presented in section 2. In section 3, we first develop an iterative algorithm based on the minorization-maximization (MM) principle [7] to solve the optimization problem. We then simplify the algorithm and proposed an MPL-LMS algorithm and a robust MPL-LMS algorithm. In section 4, we present simulation results.

2. PROBLEM FORMULATION

Given the observation model (1) and the assumptions, we propose to develop an adaptive filter to identify the unknown system based on the principle of maximum penalized likelihood. More specifically, at time n, the objective function to be maximized is given by

$$J(\mathbf{w}) = \mathcal{L}(\mathbf{w}) - \frac{1}{\alpha} \mathcal{P}(\mathbf{w}) \tag{4}$$

where $\alpha > 0$ is the tuning parameter. The log-likelihood function is $\mathcal{L}(\mathbf{w}) = \log p(y_n|\mathbf{w}, \mathbf{x}_n) = \log p(e_n|\mathcal{H})$ where $\mathcal{H}$ represents model assumptions. The penalty function is $\mathcal{P}(\mathbf{w}) = F(||\mathbf{w} - \mathbf{w}_{n-1}||)$ where the function$F(||\cdot||)$ represents a suitable measure of the distance between the two vectors.

In this paper, we assume that the noise e_n follows a zero mean i.i.d student-t distribution, denoted by $p(e_n|\mathcal{H}) = t_\nu(0, \sigma^2)$ where ν is the degrees of freedom of the distribution and σ^2 is the scaling parameter. This assumption is motivated by the requirement for robustness to outliers. To cope with outliers, we can use heavy-tailed distribution functions such as the student-t distribution. An interesting property of the student-t distribution is that when $\nu \to \infty$, it becomes a Gaussian distribution.

On the other hand, we choose the l_2-norm for the penalty function

$$\mathcal{P}(\mathbf{w}) = \frac{1}{2} (\mathbf{w} - \mathbf{w}_{n-1})^T (\mathbf{w} - \mathbf{w}_{n-1}) \tag{5}$$

which favors a solution that is close to the previous estimate $\mathbf{w}_{n-1}$. Although a generalized l_p-norm may be used as the penalty function, the l_2-norm is chosen to simplify the algorithmic development. We can see that the penalty function plays a regularization role in the optimization process. Without it, the solution of maximizing the log-likelihood will be highly dependent on the data D_n and it may change wildly from one iteration to another. As such, the tuning parameter α controls the relative importance between the two requirements: fitting the data reasonably well and not moving far away from the previous estimate.

3. THE ADAPTIVE FILTERING ALGORITHM

The adaptive filtering algorithm is developed by solving the optimization problem

$$\mathbf{w}_n = \max_{\mathbf{w}} J(\mathbf{w}) \tag{6}$$

Since there is no closed-form solution, we develop an iterative algorithm to solve this problem. We use the idea of MM algorithm [7] and the convexity of the log-likelihood after suitable mapping of variables.

3.1 The MM algorithm

We briefly review the basic idea of the MM algorithm. The function $G_0(x; x^{(k)})$ with a known parameter $x^{(k)}$ is said to minorize $G(x)$ at the point $x^{(k)}$ provided

$$\begin{aligned} G_0(x; x^{(k)}) &\leq G(x) \quad \text{for all } x \\ G_0(x^{(k)}; x^{(k)}) &= G(x^{(k)}) \end{aligned} \tag{7}$$

Let $x^{(k+1)} = \max_x G_0(x; x^{(k)})$. From the definition, we have

$$G_0(x^{(k+1)}; x^{(k)}) \geq G_0(x^{(k)}; x^{(k)}) = G(x^{(k)})$$

and

$$G(x^{(k+1)}) \geq G_0(x^{(k+1)}; x^{(k)}).$$

Therefore, maximizing $G_0(x; x^{(k)})$ results in a non-decreasing sequence $G(x^{(k+1)}) \geq G(x^{(k)})$. This algorithm is called a minorization-maximization (MM) algorithm [7]. Therefore, instead of directly maximizing $G(x)$,we can iteratively maximize its minorizing function $G_0(x; x^{(k)})$.

3.2 The convexity of the log-likelihood function

The student-t distribution of a zero mean random variable x with a fixed degree of freedom ν, can be represented as [8]

$$t_\nu(x|0, \sigma^2) = \int_0^\infty p(x|0, \sigma^2/u)p(u)du \tag{8}$$

where $p(x|0, \sigma^2/u)$ is a Gaussian distribution with variance σ^2/u and $p(u) = \Gamma(u|\frac{\nu}{2}, \frac{\nu}{2})$ is a gamma distribution. Therefore, we have

$$\begin{aligned} \mathcal{L}(\mathbf{w}) &= \log t_\nu(e_n|0, \sigma^2) \\ &= \log \int_0^\infty \sqrt{u}p(u) \exp[-\frac{u}{2\sigma^2}e_n^2]du + c \end{aligned} \tag{9}$$

where c is a constant and we assume both ν and s are known parameters of the distribution. By changing variable $z = e_n^2/(2\sigma^2)$, we define a function

$$f(z) = \log \int_0^\infty \sqrt{u}p(u) \exp[-uz]du \tag{10}$$

Since $f(z)$ is the logarithm of the Laplace transform of $\sqrt{u}p(u)$, $f(z)$ is convex [9], .

3.3 The iterative algorithm

Since $f(z)$ is convex, we have for any constant $z^{(k)} \geq 0$

$$f(z) \geq f(z^{(k)}) + f^{'}(z^{(k)})(z - z^{(k)}) \tag{11}$$

where

$$f^{'}(z^{(k)}) = \frac{df}{dz}|_{z=z^{(k)}} = -\frac{\nu+1}{\nu+2z^{(k)}}$$

Ignoring constants, we can write the objective function $J(\mathbf{w})$ as

$$\begin{aligned} J(\mathbf{w}) &= f(z) - \frac{1}{\alpha}\mathcal{P}(\mathbf{w}) \\ &\geq f(z^{(k)}) + f^{'}(z^{(k)})(z - z^{(k)}) - \frac{1}{\alpha}\mathcal{P}(\mathbf{w}) \end{aligned}$$

Denote the solution to the problem stated in (6) at the kth iteration by $\mathbf{w}_n^{(k)}$. Here the subscript n indicates the time and the superscript (k) indicates the iteration. Let $z^{(k)} = (y_n - \mathbf{x}_n\mathbf{w}_n^{(k)})^2/(2\sigma^2)$. We can now define a new objective function

$$J_0(\mathbf{w}; \mathbf{w}_n^{(k)}) = f(z^{(k)}) + f^{'}(z^{(k)})(z - z^{(k)}) - \frac{1}{\alpha}\mathcal{P}(\mathbf{w}). \tag{12}$$

We can see that $J(\mathbf{w})$ is minorized by $J_0(\mathbf{w}; \mathbf{w}_n^{(k)})$ in that $J(\mathbf{w}) \geq J_0(\mathbf{w}; \mathbf{w}_n^{(k)})$ for any $\mathbf{w}_n^{(k)}$ and $J(\mathbf{w}_n^{(k)}) = J_0(\mathbf{w}_n^{(n)}; \mathbf{w}_n^{(k)})$. To simplify notation, let $\beta^{(k)} = -f^{'}(z^{(k)})$. Then using the MM algorithm, the solution at the $(k+1)$th iteration is given by

$$\begin{aligned} \mathbf{w}_n^{(k+1)} &= \max_{\mathbf{w}} J_0(\mathbf{w}; \mathbf{w}_n^{(k)}) \\ &= \min_{\mathbf{w}}[\frac{\beta^{(k)}}{2\sigma^2}(y_n - \mathbf{x}_n\mathbf{w})^2 + \frac{1}{\alpha}\mathcal{P}(\mathbf{w})] \\ &= \mathbf{w}_{n-1} + \frac{\mu^{(k)}\hat{e}\mathbf{x}_n^T}{1 + \mu^{(k)}\mathbf{x}_n\mathbf{x}_n^T} \end{aligned} \tag{13}$$

where $\hat{e} = \mathbf{y}_n - \mathbf{x}_n\mathbf{w}_{n-1}$ and $\mu^{(k)} = \alpha\beta^{(k)}/\sigma^2$.

In summary, the MM algorithm that iteratively solves the maximum penalized likelihood problem stated in (6) involves the following steps.

Input: y_n, $\mathbf{x}_n$ and $\mathbf{w}_{n-1}$
Output: $\mathbf{w}_n$
Step-1: $k = 0$; $\mathbf{w}_n^{(k)} = \mathbf{w}_{n-1}$.
Step-2: Calculate $\mathbf{w}_n^{(k+1)}$ using (13).
Step-3: If convergence condition is satisfied or maximum number of iterations is reached, then stop the iteration and output $\mathbf{w}_n = \mathbf{w}_n^{(k+1)}$; else, $k = k + 1$; goto Step 2.

3.4 Proposed algorithms

Since the MM algorithm guarantees non-decreasing penalized likelihood in each iteration $J(\mathbf{w}_n^{(k+1)}) \geq J(\mathbf{w}_n^{(k)})$, a possible simplification of the above iterative algorithm is to use only one iteration. As such we can write

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\mu \hat{e} \mathbf{x}_n^T}{1 + \mu \mathbf{x}_n \mathbf{x}_n^T} \tag{14}$$

where

$$\mu = \alpha\beta/\sigma^2$$

and

$$\beta = \frac{\nu+1}{\nu + 2z^{(0)}} = \frac{\nu+1}{\nu + \hat{e}^2/\sigma^2}$$

There are a number of motivations for such simplification. Using only one iteration will produce a non-decreasing penalized likelihood in light of the new data D_n, i.e., $J(\mathbf{w}_n) \geq J(\mathbf{w}_{n-1})$. As such, $\mathbf{w}_n$ is no worse than $\mathbf{w}_{n-1}$ in terms of the objective function, although $\mathbf{w}_n$ is not necessarily an optimal solution which requires significantly more computations. The optimal solution is thus sacrificed for reduced computation time. This is similar to the basic idea of a generalized EM algorithm [8] where instead of pursuing the maximum likelihood, only an increase of the likelihood is sought for simpler implementation. Another appealing motivation is that using only one iteration leads to an LMS type of adaptive filter. This permits us to study it using the established theories on LMS algorithms.

3.4.1 The MPL-LMS algorithm

For example, let us consider a special case where $\nu \to \infty$. In this case, the student-t distribution becomes a Gaussian distribution. The log-likelihood function is $\mathcal{L}(\mathbf{w}) = -\frac{1}{2\sigma^2}(y_n - \mathbf{x}_n\mathbf{w})^2$ and the maximum penalized likelihood problem becomes a minimum penalized least squares problem where the objective function is

$$J(\mathbf{w}) = \frac{1}{2\sigma^2}(y_n - \mathbf{x}_n\mathbf{w})^2 + \frac{1}{2\alpha}(\mathbf{w} - \mathbf{w}_{n-1})^T(\mathbf{w} - \mathbf{w}_{n-1}) \tag{15}$$

In this case, (14) is the exact solution to this problem with $\lim_{\nu\to\infty} \beta = 1$ and $\mu = \alpha/\sigma^2$. Furthermore, the objective function (15) can be interpreted as the log-posterior of $\mathbf{w}$ where the first term is the log-likelihood and the second term is the log-prior. The parameter σ^2 is now the variance of the noise, and α is the variance of the prior which is also a Gaussian distribution. Therefore, μ is the ratio of the prior variance to the noise variance. To relate the MPL-LMS algorithm to the normalized LMS algorithm, we consider an extreme case where $\alpha \to \infty$, i.e., the variance of the prior is very large and the prior is approximately a uniform distribution with its mean at $\mathbf{w}_{n-1}$. In this case, $\mu \to \infty$ and (14) becomes

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\hat{e}\mathbf{x}_n^T}{\mathbf{x}_n\mathbf{x}_n^T}$$

which is the NLMS algorithm with $\eta = 1$. In addition, the MPL-LMS algorithm can be regarded as special cases of the LMS algorithm in that for the MPL-LMS $\lambda = \mu/(1 + \mu\mathbf{x}_n\mathbf{x}_n^T)$.

3.4.2 The robust MPL-LMS algorithm

We now discuss the general algorithm given by (14), which is rewritten as follows

$$\begin{aligned} \mathbf{w}_n &= \mathbf{w}_{n-1} + \frac{\hat{e}\mathbf{x}_n^T}{1/\mu + \mathbf{x}_n\mathbf{x}_n^T} \\ &= \mathbf{w}_{n-1} + \frac{\hat{e}\mathbf{x}_n^T}{\frac{\nu\sigma^2 + \hat{e}^2}{\alpha(\nu+1)} + \mathbf{x}_n\mathbf{x}_n^T} \end{aligned} \tag{16}$$

This equation is of the same form as that of the MPL-LMS algorithm. The difference is that in (16) μ is a function of $\hat{e}$, which makes it robust to outliers and is demonstrated as follows. We also note that (16) is of the same form as (3) in that $\Phi(\hat{e}) = 1/(\frac{\nu\sigma^2 + \hat{e}^2}{\alpha(\nu+1)} + \mathbf{x}_n\mathbf{x}_n^T)$. We call this algorithm the robust MPL-LMS algorithm.

The robustness of the proposed algorithm can be demonstrated by recognising that the second term of (16), denoted $\mathbf{u}$, is the correction term due to the new data D_n. It is a nonlinear function of $\hat{e}$ in the form

$$\mathbf{u} = \frac{\hat{e}}{\hat{e}^2 + b}\mathbf{a} \tag{17}$$

where

$$\mathbf{a} = \alpha(\nu+1)\mathbf{x}_n^T$$

and

$$b = \alpha(\nu+1)\mathbf{x}_n\mathbf{x}_n^T + \nu\sigma^2.$$

When all parameters ν, σ^2 and α are fixed, $\hat{e}$ is the error due to fitting the previous estimate $\mathbf{w}_{n-1}$ to the current data. If $\mathbf{w}_{n-1}$ is not very far away from the true impulse response vector $\mathbf{w}$, then $\hat{e}$ is mainly due to the additive noise. If $|\hat{e}|$ is very large, then it is likely there is an impulsive noise. As such, if the correction term is a linear function of $\hat{e}$, which is the case for both the LMS and NLMS algorithms, then the algorithm is not robust in that it makes unnecessarily large correction due to noise.

We now show that the robustness of the proposed algorithm comes from the nonlinear relationship expressed in (17). Without losing generality, in figure 1, we show the value of the mth element of $\mathbf{u}$, denoted $u(m)$, as a function of $\hat{e}$ (when $0 \leq \hat{e} < 10$) for some particular combinations of $a = \alpha(\nu+1)x_n(m)$ and b. Here we use $x_n(m)$ to represent the mth element of the vector $\mathbf{x}_n^T$. We can see from figure 1 that for small value of $\hat{e}$ the correction increases with it. After a threshold, the correction decreases as $\hat{e}$ continues to increase. The threshold, denoted by $\hat{e}_t$, can be calculated by using (17) and is given by $\hat{e}_t = \sqrt{b}$. Therefore, when $|\hat{e}| \leq \hat{e}_t$, the correction term of the proposed robust MPL-LMS algorithm behaves in a similar way as the NLMS and LMS algorithm. However, when $|\hat{e}| > \hat{e}_t$, the proposed algorithm shrinks the correction term to avoid mis-adjustment. The threshold depends on the all parameters of the algorithm and the input signal energy $\mathbf{x}_n\mathbf{x}_n^T$.

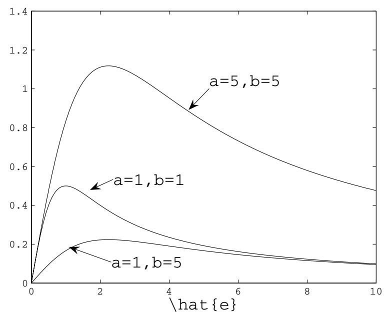

Figure 1: The value of the correction term as a function of $\hat{e}$ (when $0 \leq \hat{e} < 10$) for some combinations of a and b.

4. SIMULATION RESULTS

To study the robustness of the algorithm, we perform the following simulation experiments. The impulse response of the system to be identified is generated as a random (12×1) vector $\mathbf{w}$. In the nth iteration, a random input signal vector $\mathbf{x}_n$ is generated and y_n is calculated using (1), where the noise e_n is generated from a mixture of two zero mean Gaussian distributions with variance s_1 and s_2 ($s_2 = ms_1$, $m > 1$), respectively. This is simulated in Matlab by: en = s1 * `randn` + s2 * `randn` * (`abs(randn)` > T). We perform simulations with $s_1 = 0.1$, $s_2 = 1$ and $T = 3$. We also fix $\sigma^2 = \alpha = 0.01$ and study cases for $\nu = 10^{10}, 10, 3, 1$. The performance of algorithm is measured by the distance between the two vectors $D_n = ||\mathbf{w} - \mathbf{w}_n||^2$. Results are shown in figure 2. We can see that when $\nu = 10^{10}$ the algorithm is essentially the MPL-LMS algorithm which is not robust in that D_n tracks the impulsive noise. When $\nu = 1, 3$, the results is very robust to the impulse noise. This verifies the theoretical analysis in the previous section. When ν is increased, the learning speed is also increased and so is the sensitivity of the algorithm to the impulsive noise.

5. SUMMARY

In this paper, we have proposed a new robust adaptive filters called robust MPL-LMS algorithm. A special case of the algorithm is the MPL-LMS algorithm which is similar to the NLMS algorithm. The development is based on the student-t distribution and penalized maximum likelihood. We have demonstrated the robustness of the algorithm. A key element of the algorithm is the setting of the parameters such that the algorithm achieves the best trade-off between the learning speed and the sensitivity to impulsive noise. This should be studied using practical models of the additive noise.

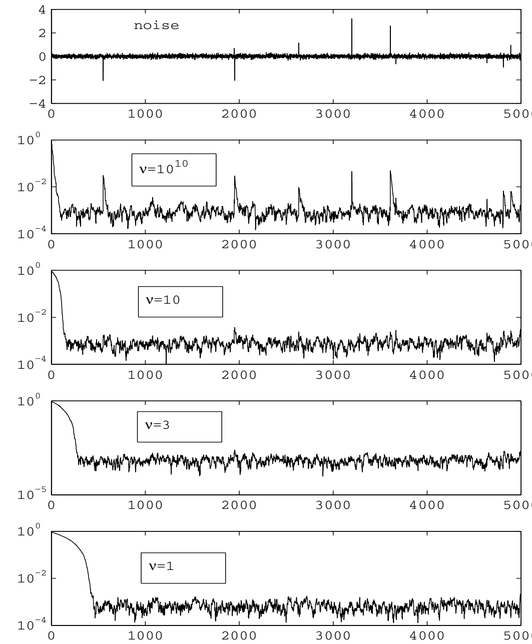

Figure 2: Simulation results using different settings of ν. In each plot, the horizontal axis is the iteration index n, while the vertical axis is $D_n = ||\mathbf{w} - \mathbf{w}_n||^2$

REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, New Jersey, USA: Prentice Hall Inc., 1996.

[2] P. Petrus, "Robust Huber adaptive filter," *IEEE Trans. Signal Processing*, vol. 47, no. 4, pp. 1129–1133, Apr. 1999.

[3] S. C. Chan and Y. X. Zou, "A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergence performance analysis," *IEEE Trans. Signal Processing*, vol. 52, no. 4, pp. 975–991, Apr. 2004.

[4] S. C. Bang and S. Ann, "A robust algorithm for adaptive FIR filtering and its performance analysis with additive contaminated-Gaussian noise," *IEEE Trans. Circuits Syst. I*, vol. 43, no. 5, pp. 361–369, May 1996.

[5] Y. X. Zou, S. C. Chan, and T. S. Ng, "Least mean M-estimate algorithms for robust adaptive filtering in impulse noise," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 12, pp. 1564–1569, Dec. 2000.

[6] G. Deng and W.-Y. Ng, "A model-based approach for the development of LMS algorithms," in *Proc. IEEE ISCAS 2005, CDROM.*

[7] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *American Statistician*, vol. 58, pp. 30–37, 2004.

[8] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, ser. Wiley series in Probability and Statistics. John Wiley & Sons, Inc., 1997.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge, U.K.: Cambridge University Press, 2004.