A NEW TYPE OF NORMALIZED LMS ALGORITHM BASED ON THE KALMAN FILTER

Paulo A. C. Lopes, Gonçalo Tavares and José B. Gerald

IST and INESC-ID, INESC-ID
Rua Alves Redol n° 9
1000-029 Lisboa
Email: paulo.c.lopes@inesc-id.pt

ABSTRACT

While the LMS algorithm and its normalized version (NLMS), have been thoroughly used and studied, and connections between the Kalman filter and the RLS algorithm have bean established, the connection between the Kalman filter and the LMS algorithm has not received much attention. By linking these two algorithms, a new normalized LMS algorithm can be derived that has some advantages to the classical one. Firstly, its stability is guaranteed since it is a special case of the Kalman filter. Secondly, it suggests a new way to control the step size that results in optimum convergence for a large range of input signal powers, that occur in many applications. Finally, it prevents measurement noise amplification that occurs in the NLMS algorithm for low order filters, like the ones used in OFDM equalization systems.

Index Terms— LMS, Kalman filter, OFDM, step size, convergence

1. INTRODUCTION

The Least Mean Squares (LMS) algorithm for adaptive filters has bean extensively studied and tested in a broad range of applications [1–4]. In [1] and in [5] a relation between the Recursive Least Squares (RLS) and the Kalman filter [6] algorithm is determined, and in [1] the tracking convergence of the LMS, RLS and extended RLS algorithms, based on the Kalman filter, are compared. However, there is no link established between the Kalman filter and the LMS algorithm.

The classical adaptive filtering problem can be stated in the following manner. Given an input signal $u(n)$ and a desired signal $d(n)$ determine the filter, $\mathbf{w}$, that minimizes the error, $e(n)$, between the output of the filter, $y(n)$, and the desired signal, $d(n)$. For the case of Finite Impulse Responde (FIR) filters, an algorithm that solves this problem is the well known LMS. This is given by,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\,\mathbf{u}(n)^*\,e(n). \tag{1}$$

This work was funded by “Fundação para a Ciência e Tecnologia” project POSC/EEA-CPS/59401/2004.

This equation updates the vector of the filter coefficients $\mathbf{w}(n)$. The output of the filter is $y(n) = \mathbf{w}^{\mathrm{T}}(n)\,\mathbf{u}(n)$ with $\mathbf{u}(n) = [u(n)\ldots u(n{-}N{+}1)]$ were N is the filter length, and $e(n) = d(n) - y(n)$.

It is known that the LMS algorithm is only stable if the step size is limited, namely it should be inversely proportional to the power of the reference signal [1]. This lead to the normalized LMS algorithm (NLMS), as represented in (2).

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \alpha\,\frac{\mathbf{u}(n)^*\,e(n)}{\mathbf{u}(n)^{\mathrm{T}}\,\mathbf{u}(n)^*} \tag{2}$$

It is shown in [1] that this algorithm is stable as long as $0 < \alpha < 2$ and of course, $\mathbf{u}(n)^T\,\mathbf{u}(n)^* \neq 0$. In order to prevent this last possibility, in practice, the algorithm is usually modified to,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \alpha\,\frac{\mathbf{u}(n)^*\,e(n)}{\mathbf{u}(n)^{\mathrm{T}}\,\mathbf{u}(n)^* + q}. \tag{3}$$

were q is selected to be small enough when compared with $\mathbf{u}(n)^{\mathrm{T}}\,\mathbf{u}(n)^*$. This is usually chosen in an ad doc fashion. Techniques to select this value based on the proposed algorithm are presented in the paper.

2. THE KALMAN FILTER

The Kalman filter is based on a state space formulation of a continuous or discrete time system. We will limit our discussion to discrete time. The system must be linear but may be time variant. The Kalman filter gives an estimate of the state of the system given a set of outputs. For the case of Gaussian signals, and given the assumed linear model, the state estimate is optimum in the sense that it minimizes the norm of the difference between the estimate and the actual state. The system is described by the equations,

$$\mathbf{x}(n+1) = \mathbf{F}(n)\,\mathbf{x}(n) + \mathbf{n}(n) \tag{4}$$

$$\mathbf{z}(n) = \mathbf{H}(n)^{\mathrm{T}}\,\mathbf{x}(n) + \mathbf{v}(n). \tag{5}$$

The system state vector is $\mathbf{x}(n)$, and the measured signal vector is given by $\mathbf{z}(n)$. The state transition matrix is $\mathbf{F}(n)$,

$\mathbf{n}(n)$ is the state noise, $\mathbf{H}(n)$ is the observation matrix and $\mathbf{v}(n)$ is the measurement noise. The state noise and measurement noise are Gaussian random variables with known autocorrelation functions. The autocorrelation of the state noise is $\mathbf{Q}_{nn}(n)$ and of the measurement noise is $\mathbf{Q}_{vv}(n)$ as in,

$$\mathbf{Q}_{vv}(n) = \mathrm{E}\left[\mathbf{n}(n)\,\mathbf{n}(n)^{\mathrm{T}}\right] \tag{6}$$

$$\mathbf{Q}_{nn}(n) = \mathrm{E}\left[\mathbf{v}(n)\,\mathbf{v}(n)^{\mathrm{T}}\right]. \tag{7}$$

The state estimate is given by $\bar{\mathbf{x}}_{|n}(n)$ in table 1. This table represents the Kalman filter algorithm. The estimate is calculated recursively based on the estimate at the previous time instant, $\bar{\mathbf{x}}_{|n-1}(n-1)$. Along with the state estimate, the algorithm updates the state covariance matrix, $\mathbf{\Sigma}_{x|n}(n)$.

Initialize

$$\bar{\mathbf{x}}_{|0}(0) = \bar{\mathbf{x}}_{0|0} \tag{8}$$

$$\mathbf{\Sigma}_{x|0}(0) = \mathbf{\Sigma}_{x|0} \tag{9}$$

Iterate from $n = 1$ to ...

$$\alpha(n) = \mathbf{z}(n) - \mathbf{H}^{\mathrm{T}}(n)\bar{\mathbf{x}}_{|n-1}(n) \tag{10}$$

Order update

$$\mathbf{K}(n) = \Sigma_{x|n-1}(n)\mathbf{H}(n) \tag{11}$$
$$(\mathbf{H}^{\mathrm{T}}(n)\mathbf{\Sigma}_{x|n-1}(n)\mathbf{H}(n) + \mathbf{Q}_{vv}(n))^{-1}$$

$$\bar{\mathbf{x}}_{|n}(n) = \bar{\mathbf{x}}_{|n-1}(n) + \mathbf{K}(n)\alpha(n) \tag{12}$$

$$\mathbf{\Sigma}_{x|n}(n) = \mathbf{\Sigma}_{x|n-1}(n) - \mathbf{K}(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{\Sigma}_{x|n-1}(n) \tag{13}$$

Time update

$$\bar{\mathbf{x}}_{|n}(n+1) = \mathbf{F}(n)\bar{\mathbf{x}}_{|n}(n) \tag{14}$$

$$\mathbf{\Sigma}_{x|n}(n+1) = \mathbf{F}(n)\mathbf{\Sigma}_{x|n}(n)\mathbf{F}^{\mathrm{T}}(n) + \mathbf{Q}_{nn}(n) \tag{15}$$

Table 1. Kalman Filter.

3. DERIVATION OF THE KALMAN BASED LMS ALGORITHM

The Kalman filter can be used in adaptive filtering by making a number of correspondences. The adaptive filtering problem is reformulated as a state estimation problem, were the state vector corresponds to the filter coefficients vector. Since the state estimate is the state that minimizes the square of the error at each coefficient, it will also minimize the output error of the filter [6]. The optimal filter variation in time is modelled as a Markov model with white noise input, $\mathbf{n}(n)$, and state transition matrix, $\mathbf{F}(n) = \lambda\,\mathbf{I}$ with λ close to one. The measured signal $d(n)$ is related to the state through the reference signal vector $\mathbf{u}(n)$ plus an additional measurement noise $v(n)$. This is summarized in table 2.

Kalman	Kalman LMS
$\mathbf{z}(n)$	$d(n)$
$\mathbf{H}(n)$	$\mathbf{u}(n)$
$\mathbf{x}_{\|n}(n)$	$\mathbf{w}(n)$
$\mathbf{\Sigma}_{x\|n-1}(n)$	$\mathbf{\Sigma}_{w}(n)$
$\mathbf{Q}_{nn}(n)$	$\mathbf{Q}_{nn}(n)$
$\mathbf{Q}_{vv}(n)$	$q_v(n)\,\mathbf{I}$
$\mathbf{F}(n)$	$\lambda\,\mathbf{I}$

Table 2. Correspondences from the Kalman filter variables to adaptive filter variables (NLMS).

The resulting algorithm is then,

$$\alpha(n) = d(n) - \mathbf{u}(n)^{\mathrm{T}}\mathbf{w}(n) \tag{16}$$

$$\mathbf{w}(n+1) = \lambda\,\mathbf{w}(n) + \lambda\,\frac{\mathbf{\Sigma}_w(n)\mathbf{u}(n)\,\alpha(n)}{\mathbf{u}^{\mathrm{T}}(n)\mathbf{\Sigma}_w(n)\mathbf{u}(n) + q_v(n)} \tag{17}$$

$$\mathbf{\Sigma}_w(n+1) = \lambda^2\,\mathbf{\Sigma}_w(n) + \tag{18}$$
$$-\lambda^2\,\frac{\mathbf{\Sigma}_w(n)\mathbf{u}(n)\mathbf{u}(n)^{\mathrm{T}}\mathbf{\Sigma}_w(n)}{\mathbf{u}^{\mathrm{T}}(n)\mathbf{\Sigma}_w(n)\mathbf{u}(n) + q_v(n)} + \mathbf{Q}_{nn}(n)$$

The variance matrix $\Sigma_w(n)$ can be made diagonal by carefully selecting the state noise autocorrelation matrix $\mathbf{Q}_{nn}(n)$ at each iteration. More, this can be done without changing the state noise total power, $\mathrm{tr}\{\mathbf{Q}_{nn}(n)\}$, were $\mathrm{tr}\{\}$ stands for the trace of the matrix. To do this one simply makes $\mathbf{\Sigma}_w(n) = \sigma_w^2(n)\,\mathbf{I}$ and $\mathrm{tr}\{\mathbf{Q}_{nn}(n)\} = N\,q_n(n)$ and apply the trace operator to (18). The resulting algorithm is the Kalman based LMS algorithm (KLMS) and is represented in table 3. Note that $\mathrm{tr}\{\mathbf{u}(n)\,\mathbf{u}(n)^{\mathrm{T}}\} = \mathbf{u}(n)^{\mathrm{T}}\mathbf{u}(n)$. The actual algorithm presented in table 3 has been modified to allow complex signals. Namely, in the calculation of the power and in the coefficients update, $\mathbf{u}(n)^*$, the conjugate of $\mathbf{u}(n)$, is used in its place.

Initialize

$$\mathbf{w}(0) = \mathbf{w}_0 \tag{19}$$

$$\sigma_w^2(0) = \sigma_{w0}^2 \tag{20}$$

Iterate from $n = 0$ to ...

$$P = \mathbf{u}^{\mathrm{T}}(n)\mathbf{u}(n)^* \tag{21}$$

$$\alpha(n) = d(n) - \mathbf{u}^{\mathrm{T}}(n)\mathbf{w}(n) \tag{22}$$

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mathbf{u}(n)^*\,\alpha(n)}{P + q_v(n)/\sigma_w^2(n)} \tag{23}$$

$$\sigma_w^2(n+1) = \sigma_w^2(n)\left(1 - \frac{P/N}{P + q_v(n)/\sigma_w^2(n)}\right) + q_n(n) \tag{24}$$

Table 3. Normalized LMS based on the Kalman filter, KLMS

4. CHOOSING THE STATE NOISE VARIANCE

The model for the state variation is,

$$w_j(n+1) = \lambda\, w_j(n) + n_j(n). \tag{25}$$

Each coefficient corresponds to a low frequency signal, with time constant given by $\tau = T/\ln(\lambda)$ were T is the sampling period. This can be approximated by $\tau = T/(1-\lambda)$ if λ is close to one. So one has, $\lambda \approx (1 - T/\tau)$. The variance of each coefficient is easily calculated as,

$$\sigma_w^2 = \frac{q_n^2}{1-\lambda^2}. \tag{26}$$

This should be equal to the value chosen to initialize the algorithm $\sigma_w^2 = \sigma_{w0}^2$. This results that the state noise can be chosen as,

$$q_n = \sigma_{w0}^2(1-\lambda^2) \approx 2\sigma_{w0}^2 \frac{T}{\tau} \tag{27}$$

were the last approximation is valid for large τ, where τ is the time constant of the underlaying model, as previously discussed.

5. SIMPLIFICATIONS OF THE ALGORITHM

If one is not interested in the initial convergence, then the algorithm proposed in table 3 can be simplified. If the measurement error, q_v, is small when compared to NPq_n then $\sigma_w^2(n)$ converges to $\sigma_{w\infty}^2 \approx N\, q_n$. If q_n was chosen as shown in the last section then $\sigma_{w\infty}^2 = 2N\sigma_{w0}^2 \frac{T}{\tau}$ and finally we can write,

$$q_v(n)/\sigma_w^2(n) = \frac{q_v\, \tau}{2N\sigma_{w0}^2 T}. \tag{28}$$

Another choice can be to establish an upper value for the step size, namely by making,

$$q_v(n)/\sigma_w^2(n) = q_v(n)/\sigma_{w0}^2. \tag{29}$$

This assures that the algorithm converges smoothly. This is not always the case for the NLMS algorithm.

6. MEASUREMENT NOISE AMPLIFICATION

The use of the NLMS algorithm can lead to amplification of the measurement noise in low order filters when the reference signal power takes low values. This can be seen by assuming $d(n) = \mathbf{u}^{\mathrm{T}}(n)\mathbf{w_o}(n) + v(n)$ and rearranging the NLMS algorithm to,

$$\mathbf{w}(n+1) = (\mathbf{I} - \mathbf{\Gamma})\, \mathbf{w}(n) + \mathbf{\Gamma}\, \mathbf{w_o}(n) + \frac{v(n)}{\mathbf{u}(n)^{\mathrm{T}}\mathbf{u}^*(n) + q} \tag{30}$$

were Γ is a matrix given by,

$$\mathbf{\Gamma} = \alpha\, \frac{\mathbf{u}(n)^*\mathbf{u}(n)^{\mathrm{T}}}{\mathbf{u}(n)^{\mathrm{T}}\mathbf{u}^*(n) + q}. \tag{31}$$

Equation (30) maybe diagonalized to represent a bank of low-pass first order IIR filters with added noise, the last term in the equation. For low reference signal power this term will assume high values, resulting in poor performance of the algorithm. The KLMS solves this problem by carefully selection of the value of q.

7. SIMULATION RESULTS

Simulation results are presented for the case of a one coefficient complex filter and a ten real coefficient filter. Comparisons are made with the LMS and NLMS algorithm. The one-coefficient complex filter is typically used in orthogonal frequency division multiplexing (OFDM) [7] channel equalization. In this application equalization is done in the frequency domain resulting in one-coefficient filters. Also, due to the presence of nulls in the channel frequency response and due to the low pass characteristics of many channels, the input signal power varies considerably. The measurement noise, which is a prior to the algorithm, can be considered constant, resulting in a large variation of the sinal-to-noise ratio. This fits nicely to the KLMS formulation while the NLMS is more suitable for a fixed signal to noise ratio, since the α parameter is related to it. Also, the NLMS will perform poorly when the input signal power takes low values, as shown in the simulations.

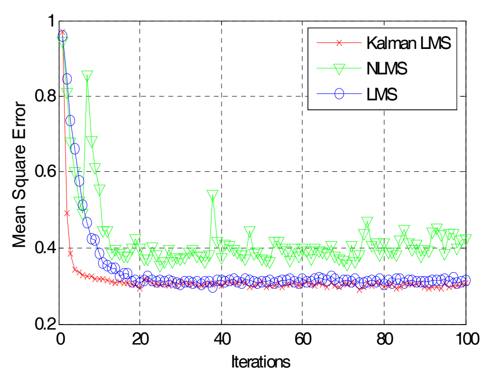

Fig. 1. Mean Square error convergence of the KLMS, NLMS and LMS algorithm. The parameters off all the algorithms were optimized for best performance.

Fig. 1 presents the convergence curves of the mean square error between the output of the adaptive filter and the desired signal for the case of a one coefficient complex filter. The reference signal was uniform distributed with power of one, and the measurement error had a standard deviation or root mean square value (RMS) of 0.3. This results in a signal to noise ratio of $10.4\,\text{dB}$ that is enough to allow fairly low bit error rate in QPSK communication. The measurement noise power of the KLMS has set to, $q_v = (0.3)^2$, the optimal value, and the state noise to zero. The step size of the LMS and NLMS were optimized to achieve a similar residual noise. The curves are the result of the ensemble average of 100 trials.

It can be seen that the KLMS has the best performance. In the case of the NLMS, due to the low filter order, occasional low values of the reference signal power result in very high values of the residual error. The LMS has slower initial convergence.

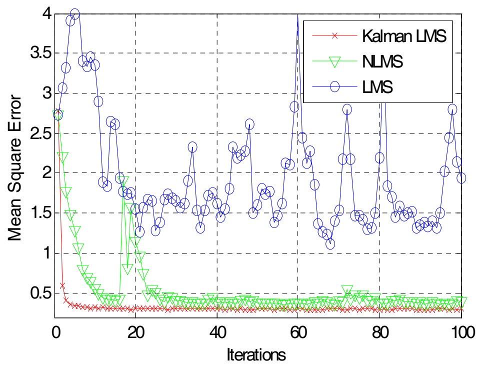

Fig. 2. Mean Square error convergence of the KLMS, NLMS and LMS algorithm, with a 3 times higher reference signal level than in Fig. 1 but with the same algorithms parameters.

In Fig. 2 the reference signal level was amplified three times, while the parameters of all the algorithm were kept constant. It can be seen that the LMS algorithm gets unstable. The NLMS has fewer problems, but it still suffers from measurement noise amplification occasionally. The KLMS still performs accurately. In addition, the KLMS has faster convergence than the NLMS.

Fig. 3 provides a comparison of the convergence of the mean square error of the KLMS, and NLMS. The desired signal was equal to the reference signal filtered by a sinusoidal bandpass filter, with unit gain at the center frequency. The reference signal had unit power and the RMS of the measurement error was 0.3. Some care had to be taken in the initial stages, when the filter buffer was not full. The NLMS buffer was initially filled with ones to prevent the step size to increase to much at the initial stages. In the case of the KLMS the buffer can be left at zero, as long as care is taken in choosing the prior standard deviation of the filter coefficient, σ_{w0}^2. Both algorithms have similar performance.

8. CONCLUSION

A new version of the NLMS algorithm based on the Kalman filter was derived, the KLMS. The new algorithm is stable

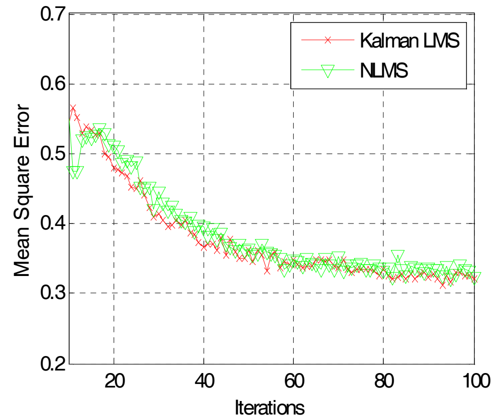

Fig. 3. Mean Square error convergence of the KLMS, NLMS and LMS algorithm, for a 10 real coefficient filter.

since it was derived from the Kalman filter. It allows faster convergence and much higher noise immunity in the cases when the reference signal vector square norm takes low values, namely in the case of low order filters (like in OFDM systems) with uniform or Gaussian distributed references. In the NLMS algorithm, q, prevents division by zero. Comparison with a simplified KLMS result in formulas for q that give the NLMS some of the good properties of the KLMS.

9. REFERENCES

[1] Simon Haykin, *Adaptive Filter Theory*, Prentice-Hall, Inc., 1996.

[2] J. Homer, "Quantifying the convergence speed of lms adaptive fir filter with autoregressive inputs," *Electronics Letters*, vol. 36, no. 6, pp. 585–586, March 2000.

[3] Huijuan Cui Yuantao Gu, Kun Tang and Wen Du, "Modifier formula on mean square convergence of lms algorithm," *Electronics Letters*, vol. 38, no. 19, pp. 1147 – 1148, Sep 2002.

[4] M. Chakraborty and H. Sakai, "Convergence analysis of a complex lms algorithm with tonal reference signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 2, pp. 286 – 292, March 2005.

[5] A.H. Sayed and T. Kailath, "A state-space approach to adaptive rls filtering," *Signal Processing Magazine, IEEE*, vol. 11, no. 3, pp. 18 – 60, July 1994.

[6] Brian D O Anderson, *Optimal Filtering*, Dover Publications, 2005.

[7] J.A.C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come," *Communications Magazine, IEEE*, vol. 28, no. 5, pp. 5–14, May 1990.