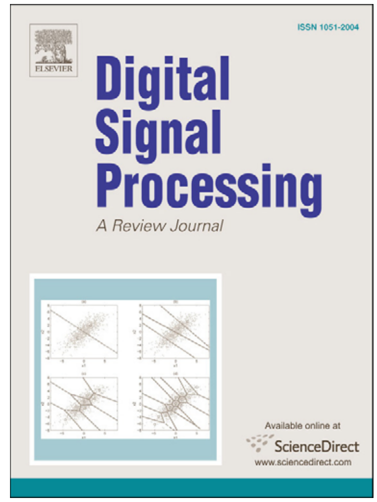

A Class of Stochastic Gradient Algorithms with Exponentiated Error Cost Functions

C. Boukis, D.P. Mandic, A.G. Constantinides

PII: S1051-2004(08)00181-4
DOI: 10.1016/j.dsp.2008.11.006
Reference: YDSPR 872

To appear in: *Digital Signal Processing*

Received date: 30 October 2007
Revised date: 16 June 2008
Accepted date: 23 November 2008

A Class of Stochastic Gradient Algorithms with Exponentiated Error Cost Functions

C. Boukis [a,*], D.P. Mandic [b] and A.G. Constantinides [b]

[a] *Athens Information Technology, Autonomic and Grid Computing Group, Markopoulo Ave, Peania/Athens 19002,Greece*

[b] *Imperial College, Electrical and Electronic Engineering Dept, Communications and Signal Processing Group, Exhibition Rd, London SW7 2BT, UK*

Abstract

A novel class of stochastic gradient descent algorithms is introduced based on the minimisation of convex cost functions with exponential dependence on the adaptation error, instead of the conventional linear combinations of even moments. The derivation is supported by rigourous analysis of the necessary conditions for convergence, the steady state mean square error is calculated and the optimal solutions in the least exponential sense are derived. The normalisation of the associated step size is also considered in order to fully exploit the dynamics of the input signal. Simulation results support the analysis.

Key words: Adaptive Filtering, Cost Functions, Stochastic Gradient Descent, Online Optimisation
PACS: 43.60.Mn, 45.10.Db

1 Introduction

Gradient descent (GD) algorithms aim at updating the coefficients of an adaptive tap-delay filter in a recursive manner, in order to minimise a chosen cost function. This is achieved in the following way

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu(n) f[e(n)] g[\boldsymbol{x}_n]$$

* Corresponding author
Email addresses: `cbou@ait.edu.gr` (C. Boukis), `d.mandic@imperial.ac.uk` (D.P. Mandic), `agc@imperial.ac.uk` (A.G. Constantinides).

Preprint submitted to Signal Processing *16 June 2008*

where $\boldsymbol{w}_n = [w_0(n), w_1(n), \ldots, w_{N-1}(n)]^t$ is the vector of filter coefficients, $e(n)$ the adaptation error defined as the difference between the desired response and the output of the adaptive filter, $\boldsymbol{x}_n = [x(n), x(n-1), \ldots, x(n-N+1)]^t$ the input regressor vector, $\mu(n)$ the time-varying learning rate and $(\cdot)^t$ the vector transpose operator [1] . The (possibly nonlinear) functions $f[\cdot]$ and $g[\cdot]$, and consequently the performance of GD algorithms, depend critically on the choice of the cost function.

The vast majority of GD algorithms use quadratic cost functions, due to their mathematical tractability and convenience of analysis. In this case the functions $f[\cdot]$ and $g[\cdot]$ are linear. The so derived second order statistics (SOS) based algorithms have low computational complexity. Members of this class are the least mean square (LMS) [2] and the normalised least mean square (NLMS) [3].

Using high order even powers of the adaptation error as cost functions (nonlinear $f[\cdot]$ and $g[\cdot]$) results in higher order statistics (HOS) adaptive algorithms [4]. These algorithms have potentially faster convergence than SOS based algorithms, due to their steeper error surfaces, that is they penalise heavier for deviations from the optimal solution. Moreover, unless the probability distribution of the measurement noise is Gaussian, HOS algorithms exhibit reduced misadjustment as compared to SOS algorithms. Typical representatives of this class are the least mean fourth (LMF) [4] and the least mean kurtosis (LMK) [5] algorithm, which have been shown to outperform the LMS, especially in the presence of non-Gaussian additive measurement noise.

Mixed norm GD algorithms, that are robust under several noise conditions, can be derived when using finite sums of even error powers as cost functions. To that end Chambers, Tanrilulu and Constantinides introduced the Least Mean Mixed Norm (LMMN) algorithm [6] where second and fourth order moments were linearly combined in a convex manner. Later on, Chambers and Avlonitis presented the Robust Mixed Norm (RMN) algorithm [7] which is based on a convex mixing of the L1 and L2 norms. A normalised version of RMN was introduced in [8]. A generalisation of the mixed norm approach was introduced by Barros et al termed the weighted even moments (WEM) algorithm [9]. This algorithm is general enough to cater for as many even error powers as necessary, however, the weighting coefficients of the error powers need to be determined empirically.

In this paper, we propose stochastic gradient adaptation based on cost functions that have exponential dependence on the chosen error. Contrary to existing approaches, this class of functions takes into account an infinite number of even moments of the error, resulting in nonlinear functions $f[\cdot]$ and $g[\cdot]$. Exponentiated error cost functions have much steeper surfaces than linear combinations of even moments, thus penalising heavily for deviation from the optimal solution. Simulations in a system identification setting have shown that the proposed least exponentials class of algorithms (LE) outperform least mean square (LMS) algorithms in terms of convergence, together with increased robustness in the presence of impulsive noise.

The proposed LE class algorithms differ from the exponentiated gradient (EG) algorithms [10–12], since in our approach the coefficient adaptation formula is additive while the latter use multiplicative updating formulas. In addition, the cost function within EG algorithm attempts to minimise is the square of the error, while LE algorithms aim at the minimisation of exponentiated error cost functions.

Section 2 introduces the class of exponentiated error functions and Section 3 presents the associated least exponential algorithms. The performance of these algorithms is then analysed in Section 4 within the energy conservation framework [14]. Simulation results are presented in Section 5 and Section 6 concludes the paper.

2 Exponentiated Error Cost Functions

In order for GD algorithms to converge to global minima of error surfaces, they employ convex and unimodal cost functions. The most general choice of such cost functions is based on linear combination of even error powers [6,9], that is

$$J(n) = \sum_{i=1}^{M} \alpha_i e^{2i}(n) \tag{1}$$

where α_i is the weighting factor that represents the contribution of of the $(2i)$-th power of the adaptation error $e(n)$. In the case of tap-delay (transversal) filters, the output error $e(n)$ is given by

$$e(n) = d(n) - \boldsymbol{w}_n^t \boldsymbol{x}_n \tag{2}$$

where $d(n)$ is the desired response. Functions of the form (1) are convex with a single minimum at $e(n) = 0$. Choosing $M = 1$ yields second order statistics (SOS) algorithms (e.g. the least mean square [2]), while for $k > 1$ we have higher order statistics (HOS) algorithms [4,6,9] (e.g. for $M = 2$ the least mean mixed norm algorithm is derived [6]). In most of the cases, the coefficients α_k are chosen empirically.

To reduce the dependence on an empirical choice of the parameters and to provide a closed form solution, we propose cost functions that have exponential dependence on the error. These are convex, unimodal and have steeper error surfaces than those given by (1). Moreover, they take into account all the even moments of the adaptation error. Two such functions are considered in this paper: the exponential of the squared error and the hyperbolic cosine. The cost function based on the exponential of the squared error (Fig. 1) is given by

$$J_{e2}(n) = \frac{1}{2}\exp[e^2(n)] \tag{3}$$

Evaluation of (3) as a Taylor Series Expansion (TSE) around zero gives

$$J_{e2}(n) = \frac{1}{2}\sum_{i=0}^{+\infty}\frac{1}{i!}e^{2i}(n) \tag{4}$$

indicating that the objective function $J_{e2}(n)$ takes into account all the even moments of the adaptation error. Also, as desired, since the weight of the $(2i)-$th error moment is $1/(i!)$ more emphasis is given to lower order moments.

The second cost function considered is the hyperbolic cosine (sum of error exponentials) given by

$$J_{se}(n) = \frac{1}{2}\Big(\exp[e(n)] + \exp[-e(n)]\Big) \tag{5}$$

As illustrated in Fig. 1, $J_{se}(n)$ is less steep than $J_{e2}(n)$ and both are steeper than a quadratic function. This can also be observed from the TSE of $J_{se}(n)$ around $e(n) = 0$

$$J_{se}(n) = \sum_{i=0}^{+\infty}\frac{1}{(2i)!}e^{2i}(n) \tag{6}$$

where the $2i$-th power of the adaptation error is weighted by $1/(2i)!$. So $J_{se}(n)$ emphasises less than $J_{e2}(n)$ on the high-order error moments. Notice also that for small error values, the exponentiated error cost functions become quadratic, that is

$$J_{e2}(n) \approx \frac{1}{2}\left(1 + e^2(n)\right) \tag{7}$$

and

$$J_{se}(n) = 1 + \frac{1}{2}e^2(n) \tag{8}$$

The LE algorithms therefore reduce to LMS for small errors; the term "1" in (8) is irrelevant when taking gradients.

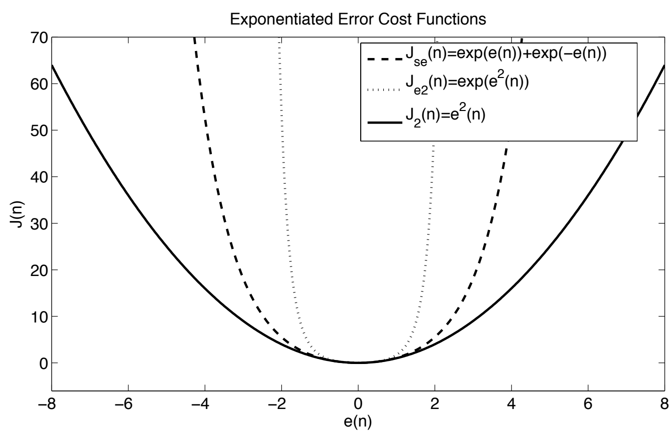

Fig. 1. Comparison of the standard and proposed cost functions.

3 The Class of Least Exponential Algorithms

Based oin the gradient of $J(n)$ from (1) we have a general stochastic gradient descent update formula [1]

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n - \mu(n)\frac{\partial J(n)}{\partial \boldsymbol{w}_n} \tag{9}$$

which yields

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu(n)\boldsymbol{x}_n \sum_{i=1}^{M} 2i\alpha_i e^{2i-1}(n) \tag{10}$$

For $M = 1$, $\alpha_1 = 1/2$ and a constant step size $\mu(n) = \mu$ this simplifies into the least mean square (LMS) algorithm. For $M = 2$ the proposed algorithms become similar to the least mean mixed norm (LMMN) algorithm [6]. Choosing $M > 2$ and heuristically finding the most appropriate values of α_k $(k = 1, 2, \ldots, M)$ results in the weighted even moments (WEM) algorithm [9].

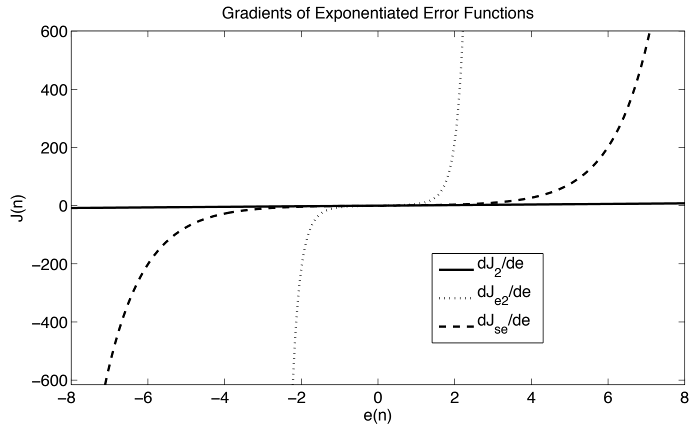

Fig. 2. The gradients of the square, exponential of the square, and sum of exponentials cost functions.

3.1 The Least Exponentiated Square (LE2) Algorithm

Taking the gradient of $J_{e2}(n)$ with respect to the vector of the filter coefficients yields

$$\frac{\partial J_{e2}(n)}{\partial \boldsymbol{w}_n} = -e(n)\boldsymbol{x}(n)\exp[e^2(n)] \tag{11}$$

Substituting (11) into the general SGD update from (9), and assuming a time invariant step size results in the least exponentiated square (LE2) algorithm, which updates its coefficient estimates according to

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu e(n)\boldsymbol{x}_n \exp[e^2(n)] \tag{12}$$

Since the cost function $J_{e2}(n)$ is much steeper than the square of the error (Fig. 1), the value of the gradient $\partial J_{e2}(n)/\partial \boldsymbol{w}_n$ is significantly larger than that of the gradient of the squared error with respect to the coefficients (Fig. 2). Hence, the LE2 converges faster than the least mean square (LMS) algorithm for a given constant learning rate, provided stability conditions. The nonlinearity of this algorithm with respect to the adaptation error is obvious from

$$f_{e2}[e(n)] = e(n)exp[e^2(n)] = \sum_{i=0}^{+\infty} \frac{1}{i!} e^{2i+1}(n) \tag{13}$$

Using (13), the weight update (12) can be re-written as

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu \boldsymbol{x}_n \sum_{i=0}^{+\infty} \frac{1}{i!} e^{2i+1}(n) \tag{14}$$

illustrating that LE2 algorithm comprises the odd moments of the adaptation error.

3.2 The Least Sum of Exponentials (LSE) Algorithm

The Least Sum of Exponentials (LSE) algorithm is derived by substituting $\partial J(n)/\partial \boldsymbol{w}_n$ in the general SGD formula given by (9), with the partial gradient of $J_{se}(n)$ w.r.t. the coefficients vector $\boldsymbol{w}_n$. Its update is given by

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \frac{\mu}{2} \boldsymbol{x})n \left(\exp[e(n)] - \exp[-e(n)]\right) \tag{15}$$

since

$$\frac{\partial J_{se}(n)}{\partial \boldsymbol{w}_n} = -\frac{1}{2} \boldsymbol{x}_n \left(\exp[e(n)] - \exp[-e(n)]\right) \tag{16}$$

The error nonlinearity in the recursion of LSE is therefore

$$f_{se}[e(n)] = \frac{1}{2} \left[\exp[e(n)] - \exp[-e(n)]\right] = \sum_{i=0}^{+\infty} \frac{1}{(2i+1)!} e^{2i+1}(n) \tag{17}$$

Combining (15) and (17) yields the final weight update in the form

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu \boldsymbol{x}_n \sum_{i=0}^{+\infty} \frac{1}{(2i+1)!} e^{2i+1}(n) \tag{18}$$

The only difference between LSE and LE2 is the fact that the $(2i+1)-th$ power is weighted by $1/(2i+1)!$ in the LSE update,whereas in the LE2 the coefficient associated with the same error power is $1/i!$.

4 Convergence Analysis

In this section the performance of the least exponential algorithms is examined in terms of the optimal solution, mean-square stability and steady state behaviour. This is achieved mainly based on the energy conservation framework [14,15], which relies on the observation that [16]

$$\|\tilde{\boldsymbol{w}}_{n+1}\|^2 + \frac{1}{\|\boldsymbol{x}_n\|^2}|e_a(n)|^2 = \|\tilde{\boldsymbol{w}}_n\|^2 + \frac{1}{\|\boldsymbol{x}_n\|^2}|e_p(n)|^2 \tag{19}$$

where $e_a(n)$ is the a priori error given by

$$e_a(n) = [\boldsymbol{w}_o - \boldsymbol{w}_n]^t \boldsymbol{x}_n = \tilde{\boldsymbol{w}}_n \boldsymbol{x}_n \tag{20}$$

and

$$e_p(n) = [\boldsymbol{w}_o - \boldsymbol{w}_{n+1}]^t \boldsymbol{x}_n = \tilde{\boldsymbol{w}}_{n+1} \boldsymbol{x}_n \tag{21}$$

is the a posteriori error.

In this analysis we also make the following standard assumptions [14]

- The desired response is produced by a linear model given by

$$d(n) = \boldsymbol{w}_o^t \boldsymbol{x}_n + v(n) \tag{22}$$

  where $\boldsymbol{w}_o$ is a vector containing the optimal coefficient values, and $v(n)$ is an additive noise component;
- The noise sequence $\{v(n)\}$ is independent, identically distributed and independent of the input sequence $\{\boldsymbol{x}_n\}$;
- The filter is long enough such that the a priori error is Gaussian. This assumption implies that the systematic component of the unknown signal is adequately modelled and hence the modelling error is not biased;
- The random variables $||\boldsymbol{x}_n||^2$ and $f^2[e(n)]$ are asymptotically uncorrelated. This assumption can be mathematically expressed as

$$\lim_{n\to\infty} E\left[||\boldsymbol{x}_n||^2 f^2[e(n)]\right] = E\left[||\boldsymbol{x}_n||^2\right] \lim_{n\to\infty} E\left[f^2[e(n)]\right]$$

4.1 Optimal Solution

In order for the behaviour of the algorithm to be controllable the optimal solution should be unique (unimodal error surfaces). This is also the case with the exponentiated square and the sum of exponentials cost functions.

The optimal solution of the LE2 algorithm can be analysed based on

$$-\frac{1}{2}\Big(E\left\{d(n)\boldsymbol{x}_n\exp[e^2(n)]\right\} - E\left\{\boldsymbol{x}_n\boldsymbol{x}_n^t\exp[e^2(n)]\right\}\boldsymbol{w}_{o,e2}\Big) = 0 \tag{23}$$

where $\boldsymbol{w}_o$ is the optimal solution in the least exponentiated square sense. Substituting the TSE of $\exp[e(n)]$ into (23) results in

$$E\left\{d(n)\boldsymbol{x}_n\left[\sum_{i=0}^{+\infty}\frac{1}{i!}e^{2i}(n)\right]\right\} = E\left\{\boldsymbol{x}_n\boldsymbol{x}_n^t\left[\sum_{i=0}^{+\infty}\frac{1}{i!}e^{2i}(n)\right]\right\}\boldsymbol{w}_{o,e2} \tag{24}$$

Consequently, the optimal solution in the least exponential squares sense is given by

$$\boldsymbol{w}_{o,e2} = \left[\boldsymbol{R} + \sum_{i=1}^{+\infty}\frac{1}{i!}E\left\{\boldsymbol{x}_n\boldsymbol{x}_n^t e^{2i}(n)\right\}\right]^{-1}\cdot\left[\boldsymbol{p} + \sum_{i=1}^{+\infty}\frac{1}{i!}E\left\{d(n)\boldsymbol{x}_n e^{2i}(n)\right\}\right] \tag{25}$$

where $\boldsymbol{R} = E\{\boldsymbol{x}_n\boldsymbol{x}_n^t\}$ is the autocorrelation matrix of the input signal and $\boldsymbol{p} = E\{d(n)\boldsymbol{x}_n\}$ the cross-correlation between the input $\boldsymbol{x}_n$ and the desired signal $d(n)$. Moreover, assuming that $\{\boldsymbol{x}_n\}$ and $\{e(n)\}$ are asymptotically uncorrelated yields

$$\boldsymbol{w}_{o,e2} = \left[\boldsymbol{R}\sum_{i=0}^{+\infty}\frac{1}{i!}E\left\{e^{2i}(n)\right\}\right]^{-1}\cdot\left[\boldsymbol{p}\sum_{i=0}^{+\infty}\frac{1}{i!}E\left\{e^{2i}(n)\right\}\right] \tag{26}$$

Unless the measurements of the desired signal $d(n)$ are extremely noisy, the adaptation error $e(n)$ is small at the steady state and the terms that include high powers of $e(n)$ can be neglected resulting in

$$\boldsymbol{w}_{o,e2} = \boldsymbol{R}^{-1}\boldsymbol{p}$$

hence conforming with the Wiener solution. This is due to the fact that when $e(n)$ is very small, which is the case in the steady state, $J_{e2}(n)$ becomes approximately quadratic. Eqn. (25) can be further analysed by expressing the desired response $d(n)$ and the adaptation error $e(n)$ as functions of the a priori error $e_a(n)$ as dictated by eqn. (22) and by

$$e(n) = e_a(n) + v(n) = \tilde{\boldsymbol{w}}_n^t\boldsymbol{x}_n + v(n) \tag{27}$$

Similarly to (25) the optimal coefficient vector in the least sum of exponentials sense $\boldsymbol{w}_{o,se}$ is derived by minimising the expectation of $J_{se}(n)$ as

$$\boldsymbol{w}_{o,se} = \left[\boldsymbol{R} \sum_{i=0}^{+\infty} \frac{1}{(2i+1)!} E\left\{ e^{2i}(n) \right\} \right]^{-1} \cdot \left[\boldsymbol{p} \sum_{i=0}^{+\infty} \frac{1}{(2i+1)!} E\left\{ e^{2i}(n) \right\} \right] \quad (28)$$

The least sum of exponentials solution from (28) differs from the least exponentiated square solution (eqn (26)) only in the weighting of the terms that contain high order powers of $e(n)$. Assuming that the measurement noise is negligible, the terms containing high order powers of the adaptation error can be ignored, resulting in

$$\boldsymbol{w}_{o,se} = \boldsymbol{R}^{-1} \boldsymbol{p}$$

that is the Wiener solution.

4.2 Step-size bounds for stability

Similar to all gradient descent algorithms, the choice of the step size in least exponential algorithms is crucial. To guarantee stability, the step size should satisfy

$$E\left\{ \|\tilde{\boldsymbol{w}}_{n+1}\|^2 \right\} \leq E\left\{ \|\tilde{\boldsymbol{w}}_n\|^2 \right\} \quad (29)$$

Embarking upon (29), and using (19) in order to preserve stability the bound on the step size can be calculated as

$$\mu \leq \frac{2}{\left[\|\boldsymbol{x}_n\|^4 \right]^{1/2}} \left(\inf_{E\{e_a^2\} \in \Omega''} \frac{E\{e_a^2\}\, h_G\left[E\{e_a^2\}\right]}{\sqrt{h_C\left[E\{e_a^2\}\right]}} \right) \quad (30)$$

where

$$h_G\left[E\left\{ e_a^2 \right\} \right] \triangleq \frac{E\{e_a(i) f[e(i)]\}}{E\{e_a^2\}}, \quad (31)$$

$$h_C\left[E\left\{ e_a^2 \right\} \right] \triangleq E\left\{ f^4[e(i)] \right\} \quad (32)$$

and the set $\Omega"$ is defined as

$$\Omega" = \left\{E\left\{e_a^2\right\} : \lambda \leq E\left\{e_a^2\right\} \leq \frac{1}{4}Tr(\boldsymbol{R})E\left\{\|\tilde{\boldsymbol{w}}_o\|^2\right\}\right\} \tag{33}$$

with λ the Cramer-Rao bound [17]. Using (30) the set of values of the step size that guarantee stability can be computed by finding the values of the functions $h_G[E\{e_a^2\}]$ and $h_C[E\{e_a^2\}]$ for the LE2 and the LSE algorithms. Indeed, substituting the nonlinearity of the LE2 algorithm $f_{e2}[e(n)]$ from (13) into (31) yields

$$h_G[E\{e_a^2(n)\}] = \frac{E\{e_a(n)e(n)\exp[e^2(n)]\}}{E\{e_a^2(n)\}}. \tag{34}$$

Replacing $\exp[e^2(n)]$ with its TSE, expressing $e(n)$ as a function of the a priori error according to (27), and assuming that $e_a(n)$ and $v(n)$ are mutually independent, results in

$$h_G[E\{e_a^2(n)\}] = \sum_{i=0}^{+\infty} \frac{1}{i!} \frac{E\{e_a^{2i+2}(n)\}}{E\{e_a^2(n)\}} \tag{35}$$

In a similar manner the value of the function $h_C[e_a^2(n)]$ can be found for the LE2 algorithm by combining (13) and (32) that is

$$h_C[E\{e_a^2(n)\}] = \sum_{i=0}^{+\infty} 2^{2i}\frac{1}{i!}\left[E\left\{e_a^{2i+4}(n)\right\} + E\left\{v^{2i+4}(n)\right\}\right] \tag{36}$$

Hence, in order to guarantee stability, the step size of the LE2 algorithm should be upper bounded by

$$0 \leq \mu \leq \frac{2}{\left[\|\boldsymbol{x}_n\|^4\right]^{1/2}} \left(\inf_{E\{e_a^2\}\in\Omega"} \frac{\sum_{i=0}^{+\infty} \frac{1}{i!} \frac{E\left\{e_a^{2i+2}(n)\right\}}{E\{e_a^2(n)\}}}{\sqrt{\sum_{i=0}^{+\infty} 2^{2i}\frac{1}{i!}\left[E\left\{e_a^{2i+4}(n)\right\} + E\left\{v^{2i+4}(n)\right\}\right]}} \right) \tag{37}$$

From (37), it is apparent that the step size depends strongly on the even moments of the measurement noise. It appears that the larger the amount of the injected noise the smaller the maximum step size is. Similar conditions for the LSE algorithm can be derived as

$$h_G[E\{e_a^2(n)\}] = \sum_{i=1}^{+\infty} \frac{1}{(2i+1)!} \frac{E\left\{e_a^{2i+2}(n)\right\}}{E\left\{e_a^2(n)\right\}} \tag{38}$$

and

$$h_C[E\{e_a^2(n)\}] = 2\sum_{i=1}^{+\infty}(4^{2i} - 4^{i+1})\frac{1}{2i!}\left[E\left\{e_a^{2i+4}(n)\right\} + E\left\{v^{2i+4}(n)\right\}\right] \tag{39}$$

Therefore the step size of the LSE algorithm should satisfy

$$0 \le \mu \le \frac{2}{\left[\|\boldsymbol{x}_n\|^4\right]^{1/2}} \left(\inf_{E\{e_a^2\}\in\Omega''} \frac{\sum_{i=1}^{+\infty} \frac{1}{(2i+1)!} \frac{E\left\{e_a^{2i+2}(n)\right\}}{E\{e_a^2(n)\}}}{\sqrt{\sum_{i=1}^{+\infty} \left(4^{2i} - 4^{i+1}\right) \frac{1}{(2i)!} \left[E\left\{e_a^{2i}(n)\right\} + E\left\{v^{2i}(n)\right\}\right]}} \right) \tag{40}$$

4.3 Step size normalisation

Using a constant step size in LE gradient descent algorithms is very restrictive; it should be very small in order for the algorithm to converge – especially in the presence of large modelling or measurement error $v(n)$ – according to (37) and (40). This does not allow for the full exploitation of the benefits of the exponential cost function.

To circumvent this problem, recall that minimisation of the *a posteriori* error during every iteration results in time varying normalised step sizes [18], given by (Appendix A)

$$\mu_{e2}(n) = \frac{\mu}{\boldsymbol{x}_n^t \boldsymbol{x}_n \exp[e^2(n)]} \tag{41}$$

and

$$\mu_{se}(n) = \frac{\mu e(n)}{\boldsymbol{x}_n^t \boldsymbol{x}_n \left(\exp[e(n)] - \exp[-e(n)]\right)} \tag{42}$$

These normalised step sizes completely remove the exponential factor in the update of the LE algorithms, resulting in the standard normalised least mean square (NLMS) algorithm. In order to control the steepness of the error surface, we shall introduce a positive multiplicative factor α in the exponential terms (41) and (42), which results in partially normalised step sizes given by

$$\mu_{e2}(n) = \frac{\mu}{\boldsymbol{x}_n^t \boldsymbol{x}_n \exp[\alpha e^2(n)]} \tag{43}$$

and

$$\mu_{se}(n) = \frac{\mu e(n)}{\boldsymbol{x}_n^t \boldsymbol{x}_n \left(\exp[\alpha e(n)] - \exp[-\alpha e(n)]\right)} \tag{44}$$

The closer α to unity, the less pronounced the effect of the exponential term and the greater the similarity with the NLMS algorithm is. For $\alpha < 1$, algorithms that are faster, but less robust than the standard NLMS algorithm are derived. Having values of α greater than unity results in algorithms with slow response but increased robustness to impulsive noise.

4.4 Excess Mean Squared Error

The excess mean square error (EMSE) is defined as the expectation of the square value of the a priori error in the steady state, that is

$$S \triangleq \lim_{n\to\infty} E\left\{|e_a(n)|^2\right\} \tag{45}$$

This quantity measures the ability of the algorithm to model the desired signal; the lower the value of S the more accurate the modelling is. According to the energy preservation framework [13], EMSE is the fixed point of the equation

$$S = Tr(\boldsymbol{R})\frac{h_U[S]}{h_G[S]} \tag{46}$$

where function $h_G[\cdot]$ is given by eqn (31) and $h_U[\cdot]$ is defined as the expectation of the square of the nonlinear error function,

$$h_U[E\{e_a^2(n)\}] = E\left\{f^4[e(i)]\right\} \tag{47}$$

Taking into account the nonlinearity within the LE2 algorithm (13), and using the standard assumptions of the energy conservation framework yields

$$h_U[E\{e_a^2(n)\}] = \sum_{i=0}^{+\infty} \frac{1}{i!} 2^i \left[E\left\{ e_a^{2i+2}(n) \right\} + E\left\{ v^{2i+2}(n) \right\} \right] \tag{48}$$

Similarly, for the LSE algorithm, we have

$$h_U[E\{e_a^2(n)\}] = 2\sum_{i=1}^{+\infty} \frac{1}{(2i)!} 2^{2i} \left[E\left\{ e_a^{2i}(n) \right\} + E\left\{ v^{2i}(n) \right\} \right] \tag{49}$$

Hence, the EMSE of the LE2 algorithm is the positive root of the nonlinear equation

$$S = \frac{\mu}{2} Tr(\boldsymbol{R}) \frac{S + \sigma_U^2 + \sum_{i=1}^{+\infty} \frac{2^i}{i!} \left[E\left\{ e_a^{2i+2} \right\} + E\left\{ v^{2i+2} \right\} \right]}{1 + \sum_{i=1}^{+\infty} \frac{1}{i!} \frac{E\left\{ e_a^{2i+2} \right\}}{S}} \tag{50}$$

Solving (50) for S yields

$$S = \frac{\beta\sigma_U^2 + \sum_{i=1}^{+\infty} \left[\frac{2^i \beta}{i!} E\left\{ v^{2i+2} \right\} + \frac{2^i \beta - 1}{i!} E\left\{ e_a^{2i+2} \right\} \right]}{1 - \beta} \tag{51}$$

where $\beta = \frac{\mu}{2} Tr(\boldsymbol{R})$. The EMSE of the LSE algorithm is the positive root of the nonlinear equation

$$S = \mu Tr(\boldsymbol{R}) \frac{S + \sigma_U^2 + \sum_{i=2}^{+\infty} \frac{2^i}{(2i)!} \left[E\left\{ e_a^{2i} \right\} + E\left\{ v^{2i} \right\} \right]}{1 + \sum_{i=1}^{+\infty} \frac{1}{(2i+1)!} \frac{E\left\{ e_a^{2i+2} \right\}}{S}} \tag{52}$$

As a consequence the steady state error of the LSE algorithm is found to be

$$S = \frac{2\beta\sigma_U^2 + \sum_{i=1}^{+\infty} \left[\frac{2^{i+1}\beta}{(2i)!} E\left\{ v^{2i+2} \right\} + \frac{2^{i+1}(2i+1)\beta - 1}{(2i+1)!} E\left\{ e_a^{2i+2} \right\} \right]}{1 - 2\beta} \tag{53}$$

This completes the analysis of the proposed class of algorithms. Obviously, the study of the steady state behaviour of the proposed LE algorithms through eqn. (51) and (53) is a demanding task that requires a thorough analysis. However, the following straightforward conclusions can be drawn

- The steady state MSE of LE algorithms depends on the even high order moments of the *a priori* error $e_a(n)$
- The EMSE is a function of the even high order moments of the additive noise $v(n)$

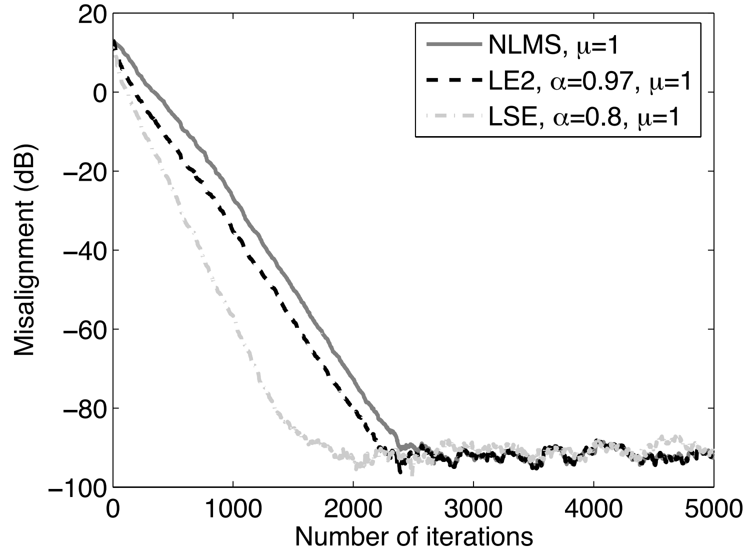

Fig. 3. Misalignment of the normalised LMS, the LE2 and the LSE algorithms for additive white measurement noise in a system identification context.

- The steady state MSE of the LSE is slightly lower than that of the LE2.

5 Simulation Results

The performance of the proposed LSE and LE2 algorithms was evaluated in a system identification setting. Both the unknown channel and the identifying filter were FIR filters of the same order. The input signal was coloured noise, that was produced by passing a white noise signal with Gaussian distribution $\mathcal{N} \sim (0, 1)$ though an autoregressive model with transfer function

$$A(z) = \frac{1}{1 - 1.79z^{-1} + 1.85z^{-2} - 1.27z^{-3} + 0.41z^{-4}} \tag{54}$$

The quantitative performance measure was the misalignment defined as

$$\|\tilde{\boldsymbol{w}}_n\|^2 = (\boldsymbol{w}_o - \boldsymbol{w}_n)^t (\boldsymbol{w}_o - \boldsymbol{w}_n) \tag{55}$$

where $\boldsymbol{w}_n = [w_0(n), w_1(n), \ldots, w_{N-1}(n)]^t$ the values of the digital filter coefficients at time instant n and $\boldsymbol{w}_o = [w_{0o}, w_{1o}, \ldots, w_{(N-1)o}]^t$ the samples of the impulse response of the linear time invariant (LTI) unknown channel. The performance of the NLMS algorithm whose step size is given by

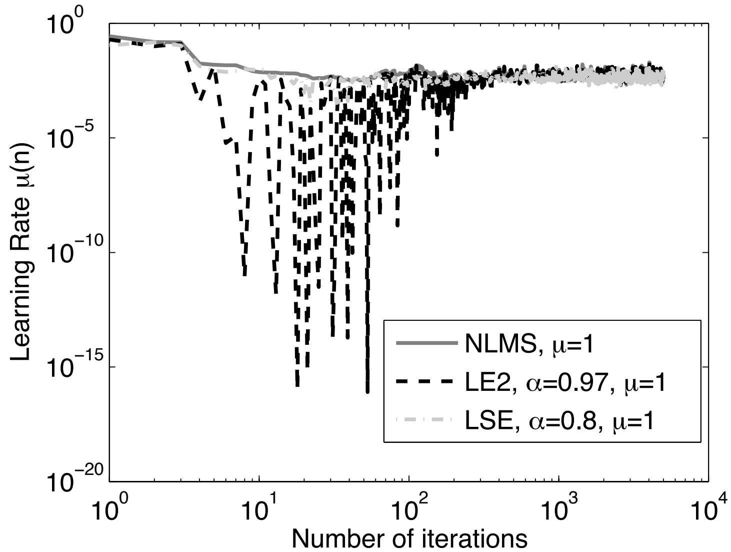

Fig. 4. The time varying step sizes for the normalised LMS and the normalised least exponential squared error algorithms, for the misalignment curves from Fig. 3

$$\mu_{NLMS}(n) = \frac{\mu}{\|\boldsymbol{x}_n\|^2} \tag{56}$$

was used as benchmark. A set of simulations for partially normalised learning rates within the LE2 and the LSE algorithms is also provided.

In Fig. 3 the misalignment curves of the LE2, the LSE and the NLMS algorithms are shown when the system output is contaminated with additive white Gaussian noise $v(n)$ with 80dB SNR. The step size of the LE2 was varying according to (43) with $\mu = 1$ and $\alpha = 0.97$, while that of the LSE was calculated from (44) with $\mu = 1$ and $\alpha = 0.8$. Both LE algorithms outperformed the NLMS, since they reached the same steady state error level within fewer iterations.

Assuming that the error is the steady state is negligible ($\lim_{n\to\infty} e(n) = 0$) results in

$$\exp[\alpha e^2(n)] \approx 1 \tag{57}$$

and

$$(\exp[\alpha e(n)] - \exp[-\alpha e(n)]) \approx \alpha. \tag{58}$$

Substituting (57) and (58) into (43) and (44) respectively yields

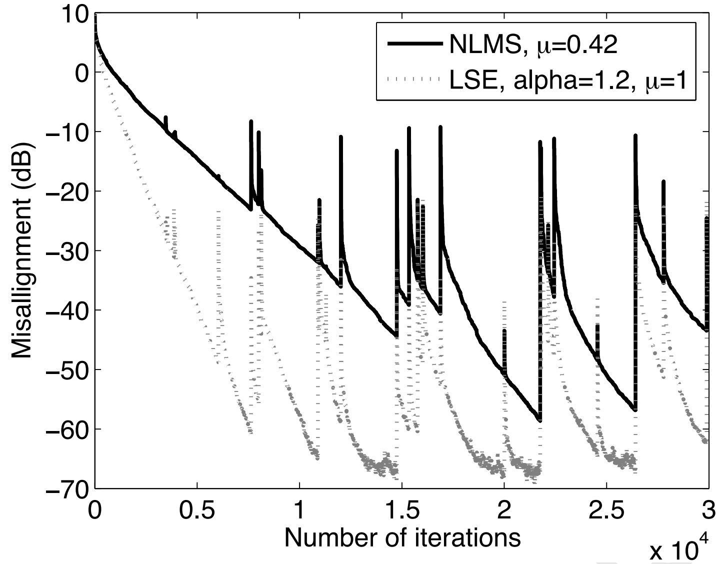

Fig. 5. Misalignment curves of the normalised LMS and the normalised LSE algorithms for impulsive noise disturbances.

$$\mu_{e2}(n) \approx \frac{\mu}{\|\boldsymbol{x}_n\|^2} = \mu_{NLMS}(n) \tag{59}$$

and

$$\mu_{se}(n) \approx \frac{\mu}{\alpha\|\boldsymbol{x}_n\|^2} = \frac{\mu_{NLMS}(n)}{\alpha} \tag{60}$$

This is also observed from the step size trajectories, depicted in Fig. 4, where all the evaluated SGD algorithms have the same step size at the steady state.

The performance of the LSE algorithm when the output of the unknown system is contaminated with impulsive noise along with 80 dB SNR of white Gaussian noise, is presented in Fig. 5. The step size was varied according to (44), with $\alpha = 1.2$ and $\mu = 1$. The misalignment curve for the NLMS algorithm with $\mu = 0.42$, is also provided. Observe that the LSE algorithm is more immune to impulsive noise than the NLMS, since it has faster convergence, lower steady state error, and less pronounced overshoot every time an impulse occurs. The learning rates of the LSE and the NLMS algorithms were chosen so as to have similar values at the steady state. This is presented in Fig. 6, where it is shown that both the LSE and the NLMS algorithms, have similar learning rate values. As desired, when an impulse occurs, the learning rate of the LSE algorithm becomes very small, preventing erroneous updating of the values of the filter coefficients.

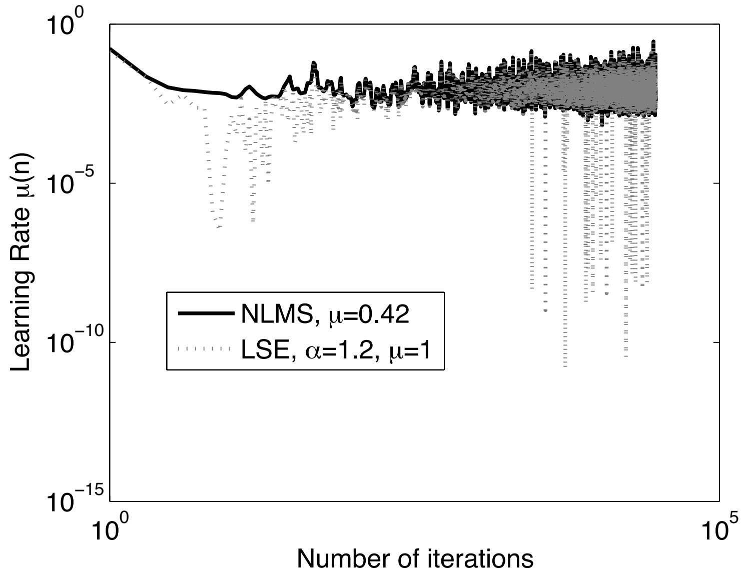

Fig. 6. The corresponding step sizes for the misalignment curves of Fig. 5.

Notice that the parameter α of the partially normalised learning rate is very important, since it greatly affects the performance of the LSE (or the LE2) algorithm for the same μ. For $\alpha \in (0, 1)$ algorithms that converge faster than the conventional NLMS are obtained, but are very sensitive. Having $\alpha > 1$, on the other hand, results in algorithms that are more robust under impulsive noise than the NLMS but have slower convergence.

6 Conclusions

A novel class of least exponential (LE) algorithms has been presented . These have been derived by minimising cost functions that have exponential dependence on the adaptation error. It has been shown that LE algorithms can be considered as a generalisation of the mixed norm stochastic gradient descent algorithms since they take into account an infinite number of error norms. A rigourous mathematical analysis has been provided resulting in closed form expressions for the optimal solutions and an upper bounds for the learning rate. For robustness, normalisation of the step size of the proposed algorithms has been addressed. Simulation results in a system identification setting and under various noise conditions support the analysis.

A Normalisation of the step size

Normalisation of the step size can be achieved through the minimisation of the magnitude of the *a posteriori* error given by

$$e_p(n) = d(n) - \boldsymbol{w}_{n+1}^t \boldsymbol{x}_n \tag{A.1}$$

as was presented in [18] for the case of the LMS algorithm. Applying similar considerations, normalised step sizes for the LE algorithms can be derived. Indeed, substituting (12) in (A.1), yields

$$e_p(n) = d(n) - \Big[\boldsymbol{w}_n + \mu_{e2} e(n) \boldsymbol{x}_n \exp[e^2(n)]\Big]^t \boldsymbol{x}_n \tag{A.2}$$

Using (22), (A.2) can be re-written as

$$\varepsilon_p(n) = \Big[1 - \mu_{e2} \boldsymbol{x}_n^t \boldsymbol{x}_n \exp[e^2(n)]\Big] e(n) \tag{A.3}$$

The magnitude of the a posteriori error is minimised when a time varying step size is employed, that is

$$\mu_{e2}(n) = \frac{1}{\boldsymbol{x}_n^t \boldsymbol{x}_n \exp[e^2(n)]} \tag{A.4}$$

Thus an the optimal learning rate of the LE2 algorithm becomes

$$\mu_{e2}(n) = \frac{\mu_{e2}}{\boldsymbol{x}_n^t \boldsymbol{x}_n \exp[e^2(n)]} \tag{A.5}$$

where $0 \leq \mu_{e2} \leq 2$. Similarly, an appropriate choice for the step size for the LSE algorithm is

$$\mu_{se}(n) = \frac{\mu_{se} e(n)}{\boldsymbol{x}_n^t \boldsymbol{x}_n \left(\exp[e(n)] - \exp[-e(n)]\right)} \tag{A.6}$$

These normalised step sizes completely remove the exponential terms from the recursive equations of the LE2 and the LSE algorithms, given by (12) and (15) respectively, and reduce the derived LE algorithms to the standard NLMS algorithm. Introducing a positive factor α such that

$$\mu_{e2}(n) = \frac{\mu_{e2}}{\boldsymbol{x}_n^t \boldsymbol{x}_n \exp[\alpha e^2(n)]} \tag{A.7}$$

and

$$\mu_{se}(n) = \frac{\mu_{se} e(n)}{\boldsymbol{x}_n^t \boldsymbol{x}_n \left(\exp[\alpha e(n)] - \exp[-\alpha e(n)]\right)} \tag{A.8}$$

the effect of these exponential term can be controlled.

References

[1] A. Benveniste, M. Metivier, P. Priouret, Adaptive Algorithms and Stochastic Approximations, New York:Springer-Verlang, 1990.

[2] B. Widrow, S. Stearns, Adaptive Signal Processing, Prentice Hall, 1985.

[3] N. Bershad, Analysis of the Normalized LMS with Gaussian Inputs, IEEE Trans. Acoust., Speech, Signal Processing 34 (4) (1986) 793–806.

[4] E.Walach, B.Widrow, The least mean fourth (LMF) adaptive algorithm and its family, IEEE Trans. Inform. Theory 30 (2) (2003) 275–283.

[5] O. Tanrikulu, A. Constantinides, Least mean kurtosis: A novel higher-order statistics based adaptive filtering algorithm, Elec. Lett 30 (3) (1994) 189–190.

[6] J. Chambers, O. Tanrikulu, A. Constantinides, Least mean mixed-norm adaptive filtering, Elec. Lett., vol. 30, no 3, pp. 1–190, 1994.

[7] J. Chambers, A.Avlonitis, A Robust Mixed-Norm Adaptive Filter Algorithm, IEEE Signal Processing Lett., vol. 4, no 2, pp. 46–48, 1997.

[8] D. Mandic, E. Papoulis, C. Boukis, A Normalized Mixed-Norm Adapitive Filtering Algorithm Robust Under Impulsive Noise Interference, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. VI, pp. 333–336, 2003.

[9] A. Barros, J. Principe, Y. Takeuchi, C. Sales, N. Ohnishi, An algorithm based in the even moments of the error, in: IEEE XIII Workshop on Neural Networks for Signal Processing, pp. 879–885, 2003.

[10] J. Kivinen, M. Warmuth, Exponentiated gradient versus gradient descent for linear predictors, Inform. Comp., vol. 132, no 1, pp. 1–64, 1997.

[11] R. Martin, W. Sethares, R. Williamson, C. Johnson Jr., Exploiting sparsity in adaptive filters, IEEE Trans. Signal Processing, vol. 50, no 8, pp. 1883–1893, 2002.

[12] J. Benesty, Y. Huang, J. Chen, An exponentiated gradient adaptive algorithm for blind identification of sparse simo systems, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 2, pp. 829–832, 2002.

[13] M. Rupp, A.H. Sayed, A Time-Domain Feedback Analysis of Filtered-Error Adaptive Gradient Algorithms, IEEE Trans. Signal Processing, vol. 44, no 6, pp. 1428–1429, 1996.

[14] T.Y. Al-Naffouri, A.H. Sayed, Adaptive Filters with Error Nonlinearities: Mean-Square Analysis and Optimum Design, EURASIP Journal of Applied Signal Processing, vol. 4, pp. 192–205, 2001.

[15] T.Y. Al-Naffouri, A.H. Sayed, Transient Analysis of Adaptive Filters with Error Nonlinearities, IEEE Trans. Signal Processing, vol. 51, no 3, pp. 653–661, 2003.

[16] N.R. Yousef, A.H. Sayed, A Unified Approach to the Steady-State and Tracking Analyses of Adaptive Filters, IEEE Trans. Signal Processing, vol. 49, no 2, pp. 314–324, 2001.

[17] L. Van Trees, Detection, Estimation and Modulation Theory: Part I, John Wiley and Sons, New York, 1968.

[18] E. Soria-Olivas, J. Calpe-Maravilla, J. Guerrero-Martinez, M. Martinez-Sober, J. Espi-Lopez, An Easy Demonstration of the Optimum Value of the Adaptation Constant in the LMS Algorithm, IEEE Transactions on Education, vol. 41, no 1, pp. 81, 1998.