ROOM IMPULSE RESPONSE ESTIMATION USING SPARSE ONLINE PREDICTION AND ABSOLUTE LOSS

Koby Crammer[†] *and Daniel D. Lee*[‡]

[†]Dept. of Computer and Information Science
[‡]Dept. of Electrical and Systems Engineering
University of Pennsylvania
Philadelphia, PA 19104
crammer@cis.upenn.edu, ddlee@seas.upenn.edu

ABSTRACT

The need to accurately and efficiently estimate room impulse responses arises in many acoustic signal processing applications. In this work, we present a general family of algorithms which contain the conventional normalized least mean squares (NLMS) algorithm as a special case. Specific members of this family yield estimates which are robust both to different noise models and choice of parameters. We demonstrate the merits of our approach to accurately estimate sparse room impulse responses in simulations with speech signals.

1. INTRODUCTION

Many acoustic signal processing systems rely upon the reliable estimation of a linear impulse response. For example, in acoustic echo cancellation, the goal is to estimate the room impulse response in order to effectively compensate for echoes and reverberation. The transfer function of the room is modeled as a linear time-invariant system with a memory size M, with a true room impulse response denoted by $\boldsymbol{h} \in \mathbb{R}^M$. An acoustic source signal is represented as a discrete time signal $\boldsymbol{s}$, whose latest M tap values are given by $\boldsymbol{s}_i = [s_i, s_{i-1}, \ldots, s_{i-M+1}]^T$. The acoustic system then measures the signal $\boldsymbol{x}$, which is given by the convolution of the source signal $\boldsymbol{s}$ with the room impulse response $\boldsymbol{h}$ corrupted by noise $\boldsymbol{n}$: $x_i = \sum_{r=0}^{M-1} h_r s_{i-r} + n_i = \boldsymbol{h} \cdot \boldsymbol{s}_i + n_i$. To effectively predict and cancel any room effects in $\boldsymbol{x}$, the room impulse response $\boldsymbol{h}$ needs to be estimated from these observations.

In the machine learning community, similar problems arise in the context of online prediction problems. In such problems the learner is exposed to an input vector $\boldsymbol{s}_i$ and is required to make a prediction $\hat{x}_i \in \mathbb{R}$. The learner then receives the correct response x_i and suffers a nonnegative loss $\ell(x_i, \hat{x}_i)$, whereby the learner iteratively updates his prediction function. Much prior work in machine learning has focused on linear prediction functions of the form $\hat{x}_i = \boldsymbol{w}_i \cdot \boldsymbol{s}_i$. In this work, we show how techniques from these approaches can be applied to the problem of estimating impulse responses in signal processing. In particular, we demonstrate the utility of incorporating different divergences and loss functions for robustly estimating sparse room impulse responses.

This work was supported by the U.S. NSF and ARO.

2. ALGORITHMS

The popular normalized least mean squares (NLMS) algorithm (see e.g. in [1]) can be viewed as a solution of the following optimization problem,

$$\boldsymbol{w}_{i+1} = \arg\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_i\|^2 + C(x_i - \boldsymbol{w} \cdot \boldsymbol{s}_i)^2, \qquad (1)$$

whose solution is given by:

$$\boldsymbol{w}_{i+1} = \boldsymbol{w}_i + \frac{1}{\frac{1}{C} + \|\boldsymbol{s}_i\|^2}(x_i - \boldsymbol{w}_i \cdot \boldsymbol{s}_i)\boldsymbol{s}_i. \qquad (2)$$

For convenience, we will also refer to the parameter $\gamma = 1/C$ later when describing the NLMS algorithm.

We investigate a broad class of algorithms which contain NLMS as special case. In particular, we consider update rules of the form:

$$\boldsymbol{w}_{i+1} = \arg\min_{\boldsymbol{w}} \mathcal{D}_F\left(\boldsymbol{w} \,\|\boldsymbol{w}_i\right) + C\ell(x_i, \boldsymbol{w} \cdot \boldsymbol{s}_i), \qquad (3)$$

where $\mathcal{D}_F$ is a divergence, ℓ is a loss function and $C > 0$. The new estimate $\boldsymbol{w}_{i+1}$ thus optimizes two opposing terms. The first term tries to keep the new estimate $\boldsymbol{w}_{i+1}$ as close as possible to the current estimate $\boldsymbol{w}_i$. The second term focuses solely on the loss achieved by the estimate on the latest portion of the signal. The constant C encapsulates the tradeoff between these two terms. In the following, we further investigate specific choices for both the divergence $\mathcal{D}_F$ and the loss function ℓ.

The algorithmic framework of Eq. (3) has previously been analyzed for several different loss functions and divergences. For example, [2] presents a general framework for classification and regression, in which the algorithms optimize an epsilon-insensitive loss function with Euclidian distance to measure divergence. In other work, a squared loss function has been used in conjunction with p-norm divergences [3].

2.1. Bregman Divergences

A Bregman divergence is defined via a strictly convex function $F : \mathcal{X} \to \mathbb{R}$ on a closed convex set $\mathcal{X} \subseteq \mathbb{R}^n$. A Bregman function F needs to satisfy a set of specific constraints [4]. We further impose that F is continuously differentiable on all points of $\mathcal{X}_{\rm int}$ (the interior of $\mathcal{X}$) which is assumed to be nonempty. The Bregman divergence that is associated with F, applied to $\boldsymbol{w} \in \mathcal{X}$ and $\boldsymbol{h} \in \mathcal{X}_{\rm int}$ is then defined to be

$$\mathcal{D}_F\left(\boldsymbol{w} \,\|\boldsymbol{h}\right) \stackrel{\rm def}{=} F(\boldsymbol{w}) - F(\boldsymbol{h}) - \nabla F(\boldsymbol{h}) \cdot (\boldsymbol{w} - \boldsymbol{h}). \qquad (4)$$

$\mathcal{D}_F$ measures the difference between two functions evaluated at $\boldsymbol{w}$. The first is the function F itself and the second is the first-order Taylor expansion of F derived at $\boldsymbol{h}$. Note that $\mathcal{D}_F(\cdot\|\cdot)$ is convex in its first argument since F is a convex function. The divergences we employ are defined via a single *scalar* convex function f such that $F(\boldsymbol{w}) = \sum_{l=1}^{n} f([\boldsymbol{w}]_l)$, where $[\boldsymbol{w}]_l$ is the l-th coordinate of $\boldsymbol{w}$. The resulting Bregman divergence between $\boldsymbol{w}$ and $\boldsymbol{h}$ is then given by: $\mathcal{D}_F(\boldsymbol{w}\|\boldsymbol{h}) = \sum_{l=1}^{n} \mathcal{D}_f([\boldsymbol{w}]_l\|[\boldsymbol{h}]_l)$.

In this paper we focus on two commonly used divergences. The first is when $\mathcal{X} \subset \mathbb{R}^n$ and $f(w) = (1/2)\,w^2$ so that $\mathcal{D}_F$ becomes the squared distance between $\boldsymbol{w}$ and $\boldsymbol{h}$,

$$\mathcal{D}_F(\boldsymbol{w}\|\boldsymbol{h}) = \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{h}\|^2.$$

The second divergence we consider is obtained by setting $f(w) = w\log(w) - w$ for nonnegative $\mathcal{X} = \mathbb{R}^n_+$. In this case $\mathcal{D}_F$ is the relative entropy:

$$\mathcal{D}_{\mathrm{RE}}(\boldsymbol{w}\|\boldsymbol{h}) = \sum_{l=1}^{n}\left([\boldsymbol{w}]_l \log\left(\frac{[\boldsymbol{w}]_l}{[\boldsymbol{h}]_l}\right) - [\boldsymbol{w}]_l + [\boldsymbol{h}]_l\right).$$

It has been shown [5] that this divergence yields a sparse regression estimate $\boldsymbol{w}$, where most of the components equal zero. This is due to that fact that the relative entropy is a homogenous divergence, $\mathcal{D}_{\mathrm{RE}}(a\boldsymbol{w}\|a\boldsymbol{h}) = a\mathcal{D}_{\mathrm{RE}}(\boldsymbol{w}\|\boldsymbol{h})$ for $a > 0$, leading to solutions located on the boundaries of the domain $\mathbb{R}^n_+$.

2.2. Loss Functions

We consider the second term in Eq. (3). The squared loss:

$$\ell_2(x_i, \hat{x}_i) = (x_i - \hat{x}_i)^2$$

is ubiquitously found in many domains such as adaptive filtering [6, 7], linear regression and estimation. This is due mainly to the simple functional forms that arise from it. In this paper we consider the absolute loss:

$$\ell_1(x_i, \hat{x}_i) = |x_i - \hat{x}_i|.$$

The absolute loss is more robust to outliers compared to squared loss, since it is relatively less sensitive to large errors $x_i - \hat{x}_i$.

Thus, we except our system to be robust even when the distribution of the noise is *not* Gaussian, and its amplitude level may have extremely high values. The experiments below demonstrate this phenomena.

2.3. Putting It All Together

We now develop a family of algorithms using the absolute loss function $\ell(x_i, \hat{x}_i) = |x_i - \hat{x}_i|$ along with two different Bregman divergences. These algorithms are defined via Eq. (3) to yield the following optimization problem:

$$\boldsymbol{w}_{i+1} = \arg\min_{\boldsymbol{w}} \mathcal{D}_F(\boldsymbol{w}\|\boldsymbol{w}_i) + C|x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i|. \tag{5}$$

Writing the loss function explicitly gives

$$\begin{aligned} \boldsymbol{w}_{i+1} \quad = \quad & \arg\min_{\boldsymbol{w}} \mathcal{D}_F(\boldsymbol{w}\|\boldsymbol{w}_i) + C\xi + C\xi^* \\ \text{s.t.} \quad & x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i \geq -\xi \\ & \xi^* \geq x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i \\ & \xi, \xi^* \geq 0. \end{aligned} \tag{6}$$

Since the objective is strictly convex and the constraints are linear, there is a unique solution to Eq. (6). To characterize the solution $\boldsymbol{w}_{i+1}$, we consider the dual form of Eq. (6):

$$\begin{aligned} \mathcal{L}(\boldsymbol{w};\alpha,\alpha^*,\beta,\beta^*) = \mathcal{D}_F(\boldsymbol{w}\|\boldsymbol{w}_i) + C\xi + C\xi^* - \beta\xi - \beta^*\xi^* \\ +\alpha[-\xi - x_i + \boldsymbol{w}\cdot\boldsymbol{s}_i] + \alpha^*[-\xi^* + x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i] \end{aligned} \tag{7}$$

where $\alpha, \alpha^*, \beta, \beta^* \geq 0$ are the Lagrange multipliers. Taking the derivative of $\mathcal{L}$ with respect $\boldsymbol{w}$ yields $\nabla F(\boldsymbol{w}) = \nabla F(\boldsymbol{w}_i) + (\alpha - \alpha^*)\boldsymbol{s}_i$, so that

$$\boldsymbol{w} = (\nabla F)^{-1}(\nabla F(\boldsymbol{w}_i) + (\alpha - \alpha^*)\boldsymbol{s}_i),$$

where $\nabla F^{-1}(\cdot)$ is the component-wise inverse of $\nabla F(\cdot)$. From convexity, this inverse is well-defined since ∇F is strictly monotone. When the Euclidean divergence is used, $\nabla F(\cdot)$ is the identity function and the update is given by a linear combination of $\boldsymbol{w}_i$ and $\boldsymbol{s}_i$. In contrast, for the relative entropy divergence, the gradient and its inverse are given by component-wise $\log$ and $\exp$ functions, respectively. In this case, the l-th component of $\boldsymbol{w}$ is determined by the multiplicative form:

$$[\boldsymbol{w}]_l = [\boldsymbol{w}_i]_l \exp((\alpha - \alpha^*)[\boldsymbol{s}_i]_l),$$

where $[\boldsymbol{w}]_l$ is the $l$$th$ component of $\boldsymbol{w}$.

Taking derivatives of $\mathcal{L}$ with respect to ξ and ξ^*, and because $\beta, \beta^* \geq 0$, the Lagrange parameters are constrained to be $0 \leq \alpha, \alpha^* \leq C$. The Karush-Kuhn-Tucker (KKT) conditions [8] give conditions for optimality in terms of the primal and dual variables:

$$\begin{aligned} \alpha[-\xi - x_i + \boldsymbol{w}\cdot\boldsymbol{s}_i] &= 0 \\ \alpha^*[-\xi^* + x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i] &= 0 \\ (C-\alpha)\xi &= 0 \\ (C-\alpha^*)\xi^* &= 0 \end{aligned} \tag{8}$$

We first show that the solution can always be chosen such that either $\alpha = 0$ or $\alpha^* = 0$. Assuming that both $\alpha > 0$ and $\alpha^* > 0$. From the first KKT condition we get $x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i = -\xi \leq 0$ and from the second KKT condition we get $x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i = \xi^* \geq 0$. Combining the last two equalities yield $x_i - \boldsymbol{w}\cdot\boldsymbol{s}_i = 0$. Therefore we get $\xi = \xi^* = 0$. By setting $\xi = \xi^* = 0$ in Eq. (7) we get that the Lagrangian of $\mathcal{L}$ is Eq. (7) only a function of the difference $\alpha - \alpha^*$, and not each of them independently. As a consequence, we can always set at least one of the dual parameters to zero, i.e. either $\alpha = 0$ or $\alpha^* = 0$. We thus define $\tau = \alpha + \alpha^*$ and rewrite the update rule:

$$\boldsymbol{w} = (\nabla F)^{-1}(\nabla F(\boldsymbol{w}_i) + \mathrm{sign}(x_i - \hat{x}_i)\tau\boldsymbol{s}_i). \tag{9}$$

A similar argument shows that either $\xi = 0$ or $\xi^* = 0$. If $\xi > 0$, then $\alpha = C > 0$ and $\alpha^* = 0 < C$ so that $\xi^* = 0$. As a consequence we see that the loss is given by $\xi + \xi^* = |x_i - \boldsymbol{w}_{i+1}\cdot\boldsymbol{s}_i|$.

The solution for τ exhibits serveral distinct regimes. Like NLMS, if $x_i = \boldsymbol{w}_i\cdot\boldsymbol{s}_i$ there is no loss and no update will be required: $\boldsymbol{w}_{i+1} = \boldsymbol{w}_i$. In this case $\tau = \alpha = \alpha^* = 0$. When both $0 < \alpha, \alpha^* < C$, then $0 < \tau < C$. In this case $\xi = \xi^* = 0$ and so $|x_i - \boldsymbol{w}_{i+1}\cdot\boldsymbol{s}_i| = 0$. In other words, there is no loss on the current measurement after the update is performed. On the other hand, consider the situation when $\tau = C$, i.e. either $\alpha = C$ or $\alpha^* = C$, and so either $\xi > 0$ or $\xi^* > 0$. Then $|x_i - \boldsymbol{w}_{i+1}\cdot\boldsymbol{s}_i| > 0$, and the loss on the current measurement is non-zero even after $\boldsymbol{w}_{i+1}$ is updated. The resulting update is then analogous to stochastic-gradient algorithms with a fixed learning rate C.

For Euclidean distance, the solution can be expressed analytically as:

$$\tau = \alpha + \alpha^* = \min\left\{\frac{|x_i - \hat{x}_i|}{\|\boldsymbol{s}_i\|^2}, C\right\}, \tag{10}$$

where $\alpha = 0$ if $x_i - \hat{x}_i < 0$, and $\alpha^* = 0$ if $x_i - \hat{x}_i > 0$. For more general divergences, there may not be a closed form solution to the optimal τ. In that case, a binary search algortihm on the interval $[0, C]$ may be used to find an approximate solution up to tolerance level δ in time proportional to $\log_2(C/\delta)$.

2.4. Averaging

One common procedure is to use the latest $\boldsymbol{w}_i$ to estimate the true room impulse response. Other alternatives were described and analyzed in [9]. In this more general formulation, the estimate of $\boldsymbol{h}$ at time i is a function of all the previous vectors $\boldsymbol{w}_1 \ldots \boldsymbol{w}_i$. In particular, one option is to use a filtered average of this sequence rather than just its last element. A theoretical analysis shows that with i.i.d. assumptions over the input sequence, the *average* estimate, $\hat{\boldsymbol{w}}_i = \frac{1}{i}\sum_{j=1}^{i} \boldsymbol{w}_i$, is optimal, in the sense that with high probability the loss suffered over new sequences will be small. This last equation can easily be computed recursively, with a time complexity proportional to the length M. For room impulse response estimation, i.i.d. assumptions do not necessarily hold since $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i-1}$ are statistically dependent. However, the noise is i.i.d. and if the SNR is low, then the data is approximately i.i.d. In the next section we show how averaging may be used to improve the room impulse response estimation.

3. EXPERIMENTS

We demonstrate the utility of the algorithmic approach described above for echo cancellation. In acoustic echo cancellation, adaptive filtering is typically used to estimate the transfer function between a speaker and a microphone present in a room. The adaptive filter estimates the combination of the speaker and microphone characteristics along with the room impulse response. To apply the algorithm to this problem, we assume that the speaker and microphone characteristics are already known, and only the unknown and possibly changing room impulse response needs to be estimated.

In our simulations, an 18 second segment of 291,200 samples of human speech sampled at 16 kHz is used as the source signal, concatenated for long time convergence estimates. The room is simulated by a room impulse response $\boldsymbol{h}$ that is computed from a source image model [10]. The speech source signal is convolved with this room impulse response and corrupted with noise to generate the target signal $\boldsymbol{x}$. The sparse nature of the impulse response motivates the use of divergences in algorithms that can robustly estimate the underlying $\boldsymbol{h}$.

We investigated three different i.i.d. noise models at four different signal-to-noise ratios (SNR): $0, -20, -40, -60$ dB. The first noise model which we will refer to as `Normal` is i.i.d. Gaussian noise with zero mean and different variances corresponding to the SNR ratio. The second noise model is a `Sparse` noise model generated from a binary random variable analogous to a heads/tails coin flip at each discrete time step. If the result is heads, an i.i.d. Gaussian noise is then added to the measurement as in the `Normal` noise model. Otherwise, no noise is added. In other words, the signal is corrupted by additive Gaussian noise for only a fraction of the time steps, and this fraction was varied between 5–50%. The last noise model is generated by corrupting the signal with `Uniform` distributed noise with various mean values.

We evaluated three algorithms: the conventional normalized least mean square (NLMS) algorithm with $\gamma = 1$ and two adaptive filters based on the absolute loss. The first adaptive filter called Eu_1 uses the Euclidean distance and the second called RE_1 uses the Relative-Entropy. All three algorithms were evaluated using the latest $\boldsymbol{w}_i$ estimate (`last`), as well as an averaged estimate $\hat{\boldsymbol{w}}_i$ (`average`) as described in Sec. 2.4.

We use the closed form solution for NLMS and Eu_1 and a binary search with a tolerance level $\delta = 10^{-10}$ for RE_1 . The first two algorithms are initialized with zero coefficients $\boldsymbol{w}_1 = 0$ and the RE_1 algorithm with a uniform vector where all the components equal 10^{-3}. The C parameter used in Eu_1 and RE_1 is set using a previous analysis of similar classification algorithms [11]. This analysis indicates that the value of the parameter should be above and close to $1/R^2$, where R is the radius of the ball that contains all the signal segments of length M, so that $R = \max_i \|\boldsymbol{s}_i\|_p$. We use the first 125 ms of the signal to estimate C. For the Eu_1 algorithm we set $p = 2$ and get $C = 10^{-3}$, and for the RE_1 algorithm set $p = 1$ and get $C = 10^{-2}$.

The results are summarized in Fig. 1, which contains eight plots. The four columns show the results for the SNR levels: -60dB (left), $-40, -20, 0$dB (right). The top row summarizes the results for the `Normal` noise model and the bottom for the `Sparse` model where only 5% of the features are contaminated with noise. The normalized misalignment of the filter estimates:

$$E = \log_{10}\left(\frac{\|\boldsymbol{w} - \boldsymbol{h}\|_2^2}{\|\boldsymbol{h}\|_2^2}\right), \tag{11}$$

is used to evaluate the error of the estimated room impulse response. Each of the eight plots shows the resulting error from the different algorithms: NLMS, Eu_1 and RE_1 where each can be used with only the current estimate (last) and the averaged estimate (averaged).

We first focus on the performance of the algorithms with no averaging. At low noise levels all algorithms recover the room impulse response with a satisfactory misalignment of less then -30dB. The NLMS and Eu_1 achieve even smaller misalignment of about -45dB when the noise level is very small (-60dB). However, the situation is completely different at higher noise levels. In this case, NLMS is not able to recover the room impulse response. Furthermore, the accuracy of its estimates suffers from high variability. On the other hand, algorithms which employ the absolute loss do much more favorably. At noise level of -0dB the Eu_1 recovers the impulse response at a level of -8dB and the RE_1 at a level better than -13dB for the `Normal` noise model, and -19dB for Eu_1 and -25dB for RE_1 for the `Sparse` model.

We see that for very low noise levels, the averaged estimation is worse than the last estimation. The gap ranges from about -3dB for the RE_1 algorithm to about -20dB for NLMS. As the noise level increases this gap reduces. At a noise level of -20dB the averaged estimation is better for NLMS by at least 3dB, and competitive with the two other algorithms. In all cases, the averaged estimate is robust as indicated by the low variance in the estimation error. These results agree with the intuition that averaging should improve results if the input vectors are i.i.d. as in the high noise regime.

We verify the method of choosing the C parameter described above by enumerating over a wide range of C values. Surprisingly our method for choosing C is quite accurate as the best values are indeed close to the values obtained from theoretical arguments. For Eu_1 the estimate favors low noise levels. At high noise levels, lower values of C yield the best performance. The situation is reversed for the RE_1 algorithm. Values of $C = 10^{-2}$ obtained by theory is best for high levels of noise, and larger values of C are better for low noise levels.

We evaluated NLMS by setting the value of $\gamma = 1/C = 10^{+3}$ as obtained by theory. This value improves the performance of the

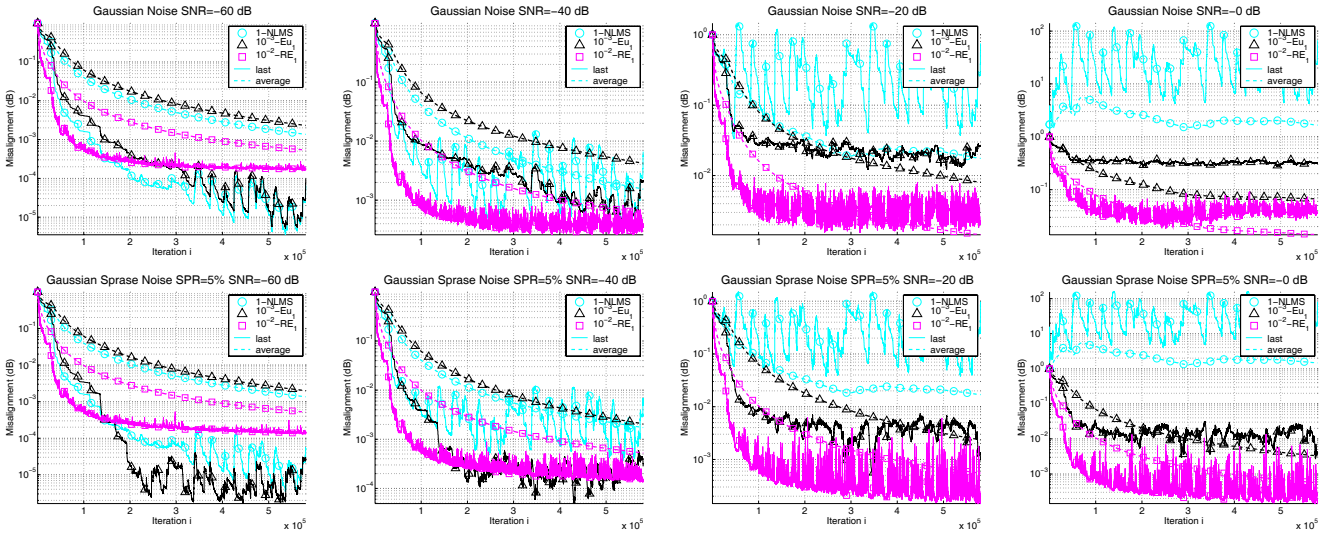

Fig. 1. *Simulations with four noise levels (columns) and two noise models: Normal (top) and Sparse Normal (bottom). Six algorithms are compared: NLMS (circle), Eu_1 (triangle) and RE_1 (square) with instantaneous estimation $\boldsymbol{w}_i$ (solid line) and averaged estimation $\hat{\boldsymbol{w}}_i$ (dashed line). The x-axis represents the sample index and the y-axis the normalized misalignment.*

algorithm for high values of noise, but significantly deteriorates for low noise levels. Examining the best value of γ for each noise level, we observed that it ranges over almost five orders of magnitude 10^0 to 10^{-4}. This range is much smaller and is only about one order of magnitude for Eu_1 and RE_1. These algorithms seem to be more invariant to the noise level, and depend more on the characteristic radius R of the signal. In contrast, NLMS is much more sensitive to fine tuning this parameter according to the noise level.

Finally, in the case of sparse noise, NLMS performed significantly worse than the two other absolute-loss algorithms. The gap in performance is at least 10dB for normalized misalignment.

4. CONCLUSION

We presented an algorithm for adaptive filtering based on ideas borrowed from the machine learning approach to regression. Our system contains three main components : alternative divergence function to measure the difference between two impulse response vectors (relative entropy vs. Euclidian), robust loss function (absolute loss vs. Euclidian) and averaging. The experimental results indicate the usefulness of the approach, especially for high noise levels with outliers. Currently, we are investigating other choices for the Bregman divergence and the loss functions which will perform even better in these acoustic signal processing tasks.

Acknolegments

We thank Yuanqing Lin for his help in preparing the data.

5. REFERENCES

[1] A.H. Sayed, *Fundementals of Adaptive Filtering*, Wiley-Interscience, 2003.

[2] K. Crammer, O. Dekel, S. Shalev-Shwartz, and Y. Singer, "Online passive aggressive algorithms," in *NIPS 16*, 2003.

[3] J. Kivinen, M. K. Warmuth, and B. Hassibi, "The p-norm generalization of the lms algorithm for adaptive filtering," in *Proc. 13th IFAC Symposium on System Identification (SYSID 2003)*, 2003, pp. 1755–1760.

[4] Y. Censor and S.A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*, Oxford University Press, New York, NY, USA, 1997.

[5] J. Kivinen and M. K. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Information and Computation*, vol. 132, no. 1, pp. 1–64, Jan. 1997.

[6] J. Benesty, Y. Huang, and D. R. Morgan, "On a class of exponentiated adaptive algorithms for identification of sparse impulse responses," in *Adaptive Signal Processing: Application to Real-World Problems*, J. Benesty and Y. Huang, Eds. Springer-Verlag, 2003.

[7] J. Benesty and Y. Huang, "The lms, pnlms, and exponentiated gradient algorithms," in *Proc. European Signal Processing Conf.*, 2004.

[8] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[9] N. Cesa-Bianchi, A. Conconi, and C.Gentile, "On the generalization ability of on-line learning algorithms," *IEEE Transactions on Information Theory*, vol. 50, no. 9, 2004.

[10] Y. Lin and D. D. Lee, "Bayesian regularization and nonnegative deconvolution for time delay estimation," in *Advances in Neural Information Processing Systems 17*, 2004.

[11] K. Crammer, *Online Learning for Complex Categorial Problems*, Ph.D. thesis, Hebrew University of Jerusalem, 2005, to appear.