ON THE LEARNING BEHAVIOR OF THE DR-LS ALGORITHM

Markus Rupp

Bell-Labs, Lucent Technologies, Wireless Research Lab.
Zadelstede 1-10, 3431 JZ Nieuwegein, THE NETHERLANDS
Fax: +31 30609 7498, Tel: +31 30609 7493
e-mail:rupp@lucent.com

ABSTRACT

Adaptive algorithms with data-reuse, like the UNDR-LMS algorithm, have recently received more attention due to their simple structure and their capability to improve the estimates by repeating the same operation. Since the basic operation is that of an LMS algorithm, it is a common belief that these algorithms converge towards the Wiener solution. In this paper, a data-reuse algorithm is analyzed that converges to the LS solution instead. The algorithm is similar to Kaczmarz's row projection method, allows, however, un-normalized regression vectors and a wide range of (time-variant) step-sizes. Similar to the LMS algorithm when compared to the Wiener solution, this algorithm also results in a misadjustment proportional to its step-size. Various step-size control strategies are investigated to improve this misadjustment.

1. INTRODUCTION

When it comes to high-speed implementations of adaptive filters, for example in adaptive equalizer structures, adaptive gradient-type algorithms have always been favored against LS-type algorithms. Gradient-type algorithms have low complexity, their operations are well suited for implementation and they do not suffer of precision and robustness problems when implemented in fixed-point architectures. The price of this preference is their slow convergence and poor estimation often paid and traded against their more desirable properties. Recently, there has been some attention of data-reuse algorithms, for example the Un-Normalized Data-Reuse LMS (UNDR-LMS) algorithm and various derivatives thereof (see for example [1]-[5]).

The gradient-type data-reuse algorithm presented here will be called UNDR-LS algorithm since it is the corresponding Least-Square gradient algorithm to the UNDR-LMS algorithm. It will be shown that UNDR-LS converges indeed to an LS solution and not to the Wiener solution as it is the case for the other data-reusing algorithms.

If the standard LMS algorithm is applied on one data pair (input vector and desired) for an infinite number of times, the same solution as for an LS estimator on the same data pair is obtained. This property has been shown by Nitzberg [6] and from a different point of view already by Goodwin and Sin [7]. An open question, however, is what solution is obtained when the updates are performed over a set of data-pairs and then repeated a number of times, i.e., the operation of the UNDR-LS algorithm. For normalized regression vectors, the Kaczmarz's row projection method (see [3] and references therein) is known to iterate to the LS solution. This pro-

jection method, however, requires divisions and cannot allow for a range of step-sizes to control the additive noise. In contrast to this projection method, we will apply a much simpler, LMS-like operation by utilizing un-normalized vectors with a (time-varying) step-size. Since the projections cannot be applied any more, this solution causes a misadjustment that is proportional to the step-size.

2. DERIVATION

Assume the desired sequence d_k and the training symbols u_k, arranged in a row vector $\mathbf{u}_k^T \triangleq [u_k, u_{k-1}, ..., u_{k-M+1}]$. The problem is to minimize the sequence $d_k - \mathbf{u}_k^T\mathbf{w}$ in an LS sense. Let's observe the sequences over a fixed period, say N time instants. Then the corresponding signals can be arranged into a vector $\mathbf{d}_N^T \triangleq [d_{N-1}, ..., d_0]$, and an $M \times N$ matrix $\mathbf{U}_N \triangleq [\mathbf{u}_{N-1}, \mathbf{u}_{N-2}, ..., \mathbf{u}_0]$. The LS solution solves $\min_{\mathbf{w}} \|\mathbf{d}_N - \mathbf{U}_N^T\mathbf{w}\|^2$, and, assuming rank$(\mathbf{U}_N = M)$, can be written as

$$\mathbf{w}_{LS} = \mathbf{U}_N[\mathbf{U}_N^T\mathbf{U}_N]^{-1}\mathbf{d}_N \,. \tag{1}$$

The UNDR-LS algorithm on the other hand applies the same set of gradient updates several times. Starting with an initial vector $\mathbf{w}_1$, the procedure is

$$e_k = (d_k - \mathbf{u}_k^T\mathbf{w}_k); \quad k = 0..N-1 \tag{2}$$
$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu\mathbf{u}_k e_k; \quad k = 0..N-1, \tag{3}$$

Compared to the DR-LMS with infinite time horizon, the DR-LS algorithm is limited to N points. The so obtained estimate $\mathbf{w}_N$ in (3) is then used as initial value again, and the procedure (2)-(3) is applied again for say L times. Iterative substitution of the N updates from $k = 0..N-1$ results in one update equation from time instant zero to $N-1$,

$$\begin{aligned}
\mathbf{w}_{k+1} &= \mathbf{w}_k + \mu\mathbf{u}_k(d_k - \mathbf{u}_k^T\mathbf{w}_k) \\
&= [\mathbf{I} - \mu\mathbf{u}_k\mathbf{u}_k^T]\mathbf{w}_k + \mu\mathbf{u}_k d_k \\
&= [\mathbf{I} - \mu\mathbf{u}_k\mathbf{u}_k^T][\mathbf{I} - \mathbf{u}_{k-1}\mathbf{u}_{k-1}^T]\mathbf{w}_{k-1} \\
&\quad + \mu\mathbf{u}_k d_k + [\mathbf{I} - \mu\mathbf{u}_k\mathbf{u}_k^T]\mu\mathbf{u}_{k-1}d_{k-1} \\
&= ...
\end{aligned}$$

Thus, obtaining (with $\mathbf{A}_k = \mathbf{I} - \mu\mathbf{u}_k\mathbf{u}_k^T$)

$$\mathbf{w}_N = \prod_{k=0}^{N-1}\mathbf{A}_k\mathbf{w}_0 + \mu\sum_{k=0}^{N-1}\prod_{l=k+1}^{N-1}\mathbf{A}_k\mathbf{u}_k d_k \tag{4}$$

This update equation can be split up in three parts

$$\mathbf{w}_N = \left[\mathbf{I} - \mu\sum_{k=0}^{N-1}\mathbf{u}_k\mathbf{u}_k^T\right]\mathbf{w}_0 + \mu\sum_{k=0}^{N-1}\mathbf{u}_k d_k + O(\mu^2) \tag{5}$$

the third part being of higher order in μ. In the following, this part will be neglected.

Assume now that the initial value $\mathbf{w}_0$ can be described in terms of the update vectors $\mathbf{u}_k$ and, in case these vectors do not span a space of dimension M, an additional term $\mathbf{z}$ orthogonal to the previous ones. Thus,

$$\mathbf{w}_0 = \mathbf{U}_N\mathbf{a}_0 + \mathbf{z}$$

with the vector $\mathbf{a}_0^T = [a_{N-1}, .., a_0]$, the weights for the update vectors $\mathbf{u}_k$, $k = 0..N-1$. Applying such a vector to (5) leads to

$$\mathbf{U}_N\mathbf{a}_1 = \mathbf{U}_N[\mathbf{I} - \mu\mathbf{U}_N^T\mathbf{U}_N]\mathbf{a}_0 + \mathbf{z} + \mu\mathbf{U}_N\mathbf{d}_N \,. \tag{6}$$

In other words, the linear combination in $\mathbf{U}_N$ remains, although the set of coefficients $\mathbf{a}_k$ changes. The orthogonal part $\mathbf{z}$ remains unchanged due to its orthogonality of $\mathbf{U}_N$. The solution can thus be described in terms of the vector $\mathbf{a}_k$ only, neglecting the orthogonal part $\mathbf{z}$ that is not effected. After one set of updates,

$$\mathbf{a}_{k+1} = [\mathbf{I} - \mu\mathbf{U}_N^T\mathbf{U}_N]\mathbf{a}_k + \mu\mathbf{d}_N \,, \tag{7}$$

is obtained. Applying (7) L times leads to

$$\mathbf{a}_L = [\mathbf{I} - \mu\mathbf{U}_N^T\mathbf{U}_N]^L\mathbf{a}_0 + \mu\sum_{k=0}^{L-1}[\mathbf{I} - \mu\mathbf{U}_N^T\mathbf{U}_N]^k\mathbf{d}_N \,. \tag{8}$$

Thus, assuming that

$$|\mathrm{eig}(\mathbf{I} - \mu \mathbf{U}_N^T \mathbf{U}_N)| < 1, \tag{9}$$

the initial values $\mathbf{a}_0$ die out (and can be set to zero with an initial zero estimate) and the right hand expression of the equation grows to some final value. Running $L \to \infty$, the following is obtained asymptotically

$$\begin{aligned} \mathbf{a}_\infty &= \mu \lim_{L\to\infty} \sum_{k=0}^{L-1} [\mathbf{I} - \mu \mathbf{U}_N^T \mathbf{U}_N]^k \mathbf{d}_N \,. \quad (10) \\ &= [\mathbf{U}_N^T \mathbf{U}_N]^{-1} \mathbf{d}_N \,. \quad (11) \end{aligned}$$

This is nothing else but the LS-solution. The only two differences now are

1. Due to the initial value of $\mathbf{w}_0$, and correspondingly $\mathbf{a}_0$, there can exist an orthogonal part $\mathbf{z}$ that is not updated. This will in particular occur when $N < M$ (underdetermined LS solution) and if for $N \geq M$ (overdetermined LS-solution) the matrix $\mathbf{U}_N$ is not of full rank M. Such a solution also occurs when a standard LS problem is defined under the same conditions.

2. The solution is an LS-approximation where terms of order μ^2 and higher were neglected. These terms might cause some difference (misadjustment) when compared to the exact LS-solution. If either the input process becomes orthogonal, i.e., $\mathbf{U}_N^T \mathbf{U}_N = \mathrm{diag}$, or the noise zero, the misadjustment vanishes. This effect is similar to the misadjustment in LMS when compared to the Wiener solution. It is, like the LMS-misadjustment, proportional to μ^2.

The eigenvalue condition (9) requires a statement on the step-size. An upper bound for the step-size μ so that the eigenvalues remain inside the unit circle is given by

$$0 < \mu < \frac{2}{\max \mathrm{eig}(\mathbf{U}_N^T \mathbf{U}_N)} \leq \frac{2}{\sum_{k=0}^{L-1} \|\mathbf{u}_k\|^2} \,.$$

This condition could lead to very small step-sizes. If the input sequence u_k is designed for training, some orthogonal property next to a constant modulus property can be imposed. For perfect orthogonality, $\mathbf{U}_N^T \mathbf{U}_N = M\mathbf{I}$ and thus

$$0 < \mu < \frac{2}{M} \tag{12}$$

is obtained, a familiar step-size condition for the LMS algorithm.

2.1. Example

A linear system of order $M = 10$ is driven by white Gaussian noise and its output is corrupted by additive noise of 40dB SNR. The UNDR-LMS algorithm is run on 20 samples and is repeated 2000 times with a fixed step-size μ. The results are compared to the Wiener solution and the LS solution obtained from (1). Figure 1 displays the convergence behavior. The relative system mismatch is plotted assuming the LS and the Wiener solution, respectively (continuous line). In a second experiment, the μ was decreased by a factor of 10. The solution (dotted line) is now as far away from the Wiener solution as before while the distance to the LS solution has been improved by a factor of 100, as predicted.

3. OPTIMAL STEP-SIZES

In this section a closer look will be taken on the residual error and the impact of the step-size μ on it. For this reason (4) is starting point again, now with time-variant step-size $\mu(l)$. It is assumed that the step-size remains constant for a block of updates but changes from block to block (indicated by index l).

$$\begin{aligned} \mathbf{w}_N &= \prod_{k=0}^{N-1} [\mathbf{I} - \mu(l) \mathbf{u}_k \mathbf{u}_k^T] \mathbf{w}_0 \quad (13) \\ &\quad + \mu(l) \sum_{k=0}^{N-1} \prod_{m=k+1}^{N-1} [\mathbf{I} - \mu(l) \mathbf{u}_m \mathbf{u}_m^T] \mathbf{u}_k d_k \end{aligned}$$

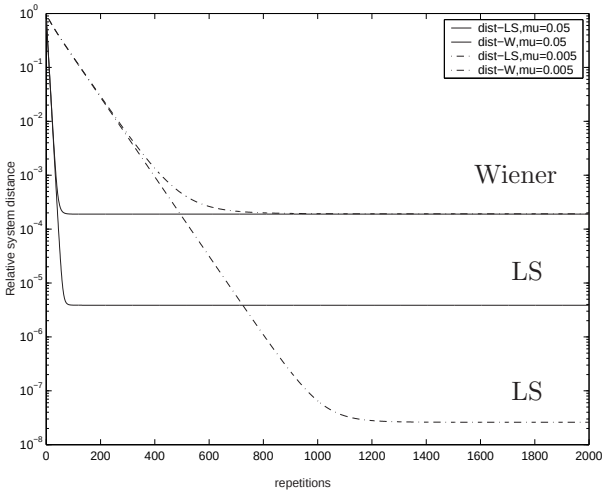

Figure 1: *Comparisons of relative system distances to Wiener and to LS solution.*

which can be approximated to

$$\begin{aligned}\mathbf{w}_N &= [\mathbf{I} - \mu(l)\sum_{k=0}^{N-1}\mathbf{u}_k\mathbf{u}_k^T]\mathbf{w}_0 \\ &\quad +\mu(l)\sum_{k=0}^{N-1}\mathbf{u}_k d_k + \mu^2(l)\sum_{k=0}^{N-1}\mathbf{u}_k x_k\end{aligned} \tag{14}$$

with the not closer specified elements x_k denoting the errors when compared to the LS solution. Using the more compact vector and matrix notation, (15) can be reformulated as

$$\begin{aligned}\mathbf{a}_L &= \prod_{l=1}^{L}[\mathbf{I} - \mu(l)\mathbf{U}_N^T\mathbf{U}_N]\mathbf{a}_0 \\ &\quad + \sum_{k=1}^{L}\mu(k)\prod_{l=k+1}^{L}[\mathbf{I} - \mu(l)\mathbf{U}_N^T\mathbf{U}_N] \\ &\quad \times\{\mathbf{d}_N + \mu(k)\mathbf{x}_N\}\ .\end{aligned} \tag{15}$$

By applying a unitary transformation, $\mathbf{Q}\mathbf{a}_i = \mathbf{b}_i$, the matrix $\mathbf{U}_N^T\mathbf{U}_N$ can be diagonalized into $\mathbf{Q}\mathbf{U}_N^T\mathbf{U}_N\mathbf{Q}^T = \Lambda$ with the M eigenvalues λ_1 to $\lambda_M = \lambda_{\max}$, and the iterative equation (15) can completely be diagonalized into

$$\begin{aligned}\mathbf{b}_{L,i} &= \prod_{l=1}^{L}(1-\mu(l)\lambda_i)\mathbf{b}_{0,i} \\ &\quad + \sum_{k=1}^{L}\mu(k)\lambda_i\prod_{l=k+1}^{L}(1-\mu(l)\lambda_i) \\ &\quad \times\{\mathbf{b}_{LS,i} + \mu(k)x(i)\}\ ,\ \text{for } i = 1..N\ ,\end{aligned} \tag{16}$$

where the LS solution $\mathbf{d}_N = \mathbf{U}_N^T\mathbf{U}_N\mathbf{a}_{LS}$ was used to substitute $\mathbf{d}_N$ and finally $\mathbf{Q}\mathbf{a}_{LS} = \mathbf{b}_{LS}$. The transformed noise terms are simply denoted as $x(i)$. From this form, and the definitions

$$\begin{aligned}S_L &\triangleq \sum_{k=1}^{L}\mu(k)\lambda_i\prod_{l=k+1}^{L}(1-\mu(l)\lambda_i), \\ T_L &\triangleq \sum_{k=1}^{L}\mu^2(k)\lambda_i\prod_{l=k+1}^{L}(1-\mu(l)\lambda_i).\end{aligned}$$

the general conditions for approaching the LS solution asymptotically can be derived:

$$\lim_{L\to\infty}\prod_{l=1}^{L}(1-\mu(l)\lambda_i) = 0, \tag{17}$$

$$\lim_{L\to\infty} S_L = 1, \tag{18}$$

$$\lim_{L\to\infty} T_L = 0. \tag{19}$$

Note that the above definitions for S_L and T_L can be reformulated recursively as

$$\begin{aligned}(S_L - 1) &= (1-\lambda_i\mu(L))(S_{L-1}-1) \\ T_L - \lambda_i\mu(L) &= (1-\lambda_i\mu(L))(T_{L-1}-\lambda_i\mu(L)).\end{aligned}$$

Note that as long as $|1-\lambda_i\mu(L)| < 1$, the terms converge to one and $\lambda_i\mu(L)$, respectively. Thus, in order for T_L to disappear, the step-size needs to decrease to zero.

This leads us to the question which step-size is best for achieving the LS solution. So far, only a constant step-size has been used,

$$\mu_0(l) = \mu\ . \tag{20}$$

In gradient-type approximation theory it is well-known that a decreasing step-size of the form

$$\mu_1(l) = \frac{c_1}{l};\ l = 1, 2... \tag{21}$$

can achieve the Wiener solution without errors (see for example [7]). Another decreasing sequence of great practical importance is given by

$$\mu_2(l) = c_2\mu_2(l-1) \tag{22}$$

since a simple multiplication can derive the following step-size value. Yet, another interesting choice is

$$\mu_3(l) = c_2\mu_3(l-1) + c_3\ . \tag{23}$$

For all sequences it can be shown that given the eigenvalues with the proper choice of the parameters, the product term $\prod(1-\mu(l)\lambda_i)$ tends asymptotically to zero. For $\mu_1(l) = c_1/l$, a ratio of Γ functions is obtained which can be approximated for small values $c_1\lambda_i$, showing that by

$$\frac{\Gamma(l-\lambda_i c_1)}{\Gamma(l)} \approx e^{-c_1\lambda_i}\ .$$

for every $0 < c_1\lambda_i \leq 1$ the sequence converges to zero.

In the following, we will compare the sequences in terms of their approximation qualities. The convergence of condition (19), however, is not so simple to conclude. It should also be noted that the eigenvalues are usually not known a-priori. In an experiment, the eigenvalues were in the range from 0.01 to 100. Table 1 shows the results for the terms S_L and T_L after $L = 100,000$ iterations. The corresponding step-size selections were $\mu = 0.01, c_1 = 0.1$ and $c_2 = 0.99, \mu(0) = 0.1, c_3 = 0.0001$. As the table reveals, the step-size sequences behave quite differently. While a constant step-size as well as the decreasing step-size with lower bound always approach $S_L = 1$, with a certain drawback in precision for T_L, the other two sequences show distinctly different behavior. Although theoretically known for (21) that $S_L \to 1$, it can take a huge amount of iterations for approaching this.

λ	S_L (20) T_L (20)	S_L (21) T_L (21)	S_L (22) T_L (22)	S_L (23) T_L (23)
100	1.0 1.0	1.0 0.01	1.0 0.01	1.0 1.0
10	1.0 0.1	1.0 1e-4	1.0 0.01	1.0 0.1
1	1.0 0.01	0.7337 8.6e-3	1.0 0.01	1.0 0.01
0.1	1.0 0.001	0.022 1e-4	0.633 0.0027	1.0 0.001
0.01	1-4e-5 1e-4	1.1e-3 1e-6	0.095 0.47e-4	1-4e-5 1e-4

Table 1: Approximation results for the four step-size sequences.

3.1. Examples

The conditions (17)-(19) do not reflect how precise the LS approximation is in a particular case. To study their influence on precision, the following identification problem is carried out. A system of order $M = 10$ is identified by taking $N = 20$ vectors and repeating them up to $L = 2000$ times. The step-size sequences as presented in the above example were applied. The relative system mismatch to the LS solution is plotted in Figure 2 for the four step-size sequences in (20) to (23) denoted μ_0 to μ_3. It now shows that the exponential decaying step-sizes are superior. After about 100 iterations they are very close to the LS solution and keep improving slowly afterwards. In particular the sequence (21) does not show very promising behavior. Thus, in practical problems were only a few repetitions can be realized an exponential decaying μ is suited best.

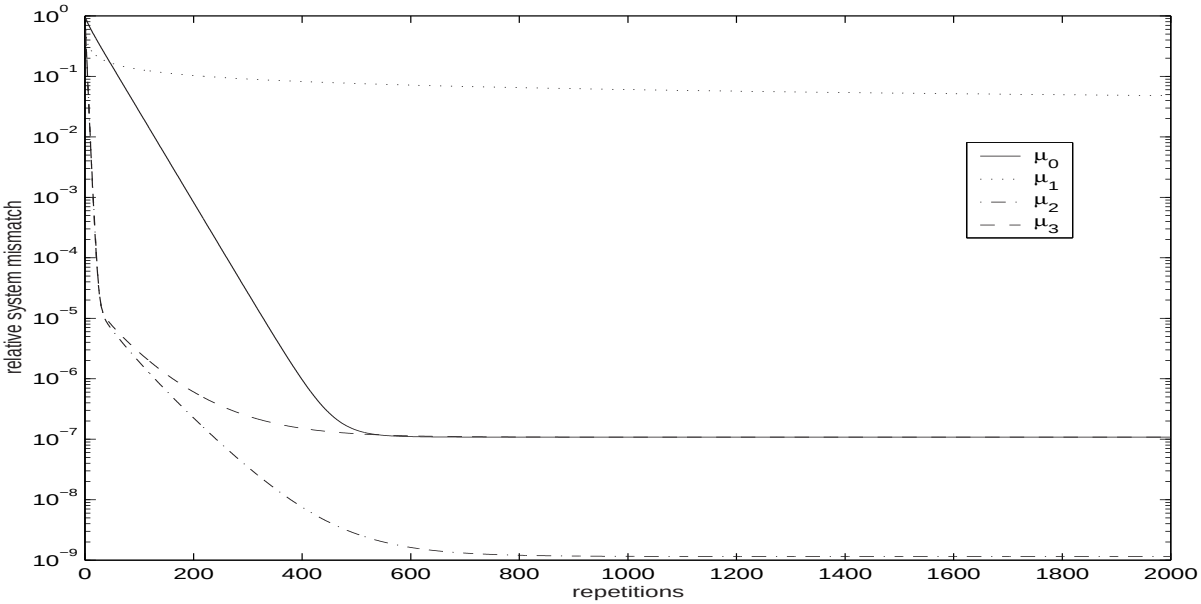

Figure 2: *Comparisons of various step-sizes.*

4. CONCLUSIONS

A new type of data-reusing algorithm has been proposed that exhibits, in contrast to the LMS algorithm, LS behavior. The algorithm is simple to implement and since it follows very similar update rules than an LMS algorithm, it is robust against numerical approximations, i.e., fixed-point implementations. Misadjustment, occurring due to approximations, can be reduced by a carefully selected step-size control.

REFERENCES

[1] B. Schnaufer, W.K.Jenkins, "New data-reusing LMS algorithm for improved convergence," Proc. of Asilomar Conf., pp. 1584-1588, Nov. 1993.

[2] W.K.Jenkins, A.W.Hull, J.C.Strait, B.A.Schnaufer, X.Li, "Advanced Concepts in Adaptive Signal Processing," Kluwer Academic Publishers, 1996.

[3] R.A.Soni, K.A.Gallivan, W.K.Jenkins, "Projection methods for improved performance in FIR adaptive filtering," Proc. of Midwest Symp. on Circ. & Syst., pp. 746-749, Aug. 1997.

[4] J.A.Apolinario Jr.,M.L.R. de Campos, P.S.R. Diniz, "Convergence analysis of the binormalized data-reusing LMS algorithm, " Proc. of European Conf. on Circuit Theory and Design, Budapest, pp. 972-977, 1997.

[5] R.A.Soni, K.A.Gallivan, W.K.Jenkins, "Convergence properties of affine projection and normalized data reusing methods," Proc. of Asilomar Conf., pp. 1166-1170, Nov. 1998.

[6] R. Nitzberg, "Application of the normalized LMS algorithm to MSLC," IEEE Trans. Aerosp. Electron. Syst., vol. AES–21, no. 1., pp. 79–91, Jan. 1985.

[7] G.C. Goodwin, K.S. Sin, *Adaptive Filtering, Prediction and Control,* Prentice–Hall, 1984.