THE MULTI-CHANNEL LEAST SQUARES ORDER RECURSIVE LATTICE SMOOTHER

Dong Kyoo Kim, Soo Jin Kim, and PooGyeon Park

Electrical And Electronic Engineering
Pohang Univ. Of Sci. and Tech.
Pohang, KYUNGBUK 790-784 SOUTH KOREA

ABSTRACT

For emerging multiple-input multiple-output communication systems, we investigate an algorithm that compensates inter-symbol interferences and suppresses interferences among channels as well. In general, such an algorithm requires more complicated computations than single channel algorithms, and this causes much numerical errors such as truncation errors during running the algorithm on digital machines. In this paper, we develop a multi-channel least squares order recursive lattice smoother that suppresses the inter-symbol and inter-channel interferences while being numerically stable in the finite precision computation.

1. INTRODUCTION

Multiple-input Multiple-output (MIMO) antenna array systems promise us to achieve high speed data transmission within limited frequency bands. It can be said that the MIMO antenna arrays can be considered as virtual parallel channels in a Rayleigh flat fading environment [1,2], where the the MIMO channel capacity can increase linearly if the channel coefficients are statistically independent and they are known to receivers [3]. Various studies of the MIMO systems have been made for the flat fading channels. In [4], it has been showed that a simple VBLAST(Vertical Bell Laboratories Layered Space-Time) in a quasi-static narrowband indoor radio channel achieved its spectral efficiency of 20-40 bits/sec/Hz at average SNR's ranging from 24 to 34 dB. Other studies exploiting the antenna diversity of the MIMO systems such as an STCM(Space Time Coding Modem) have been also proposed in [5,6]. In these efforts, the flat fading environment have usually been assumed in their algorithms. For wideband wireless communications, frequency selective channels cause multipath inter-channel interferences to degrade the performance of the MIMO systems. To protect the system against the degradation, it is necessary such an algorithm that not only compensates the signal distortion due to each channel but also suppresses the inter-channel interferences. In this paper, we propose a highly numeric-stable algorithm, the multi-channel least squares order recursive lattice smoother (MLSORLS), which equalizes the MIMO channels and suppresses their interferences at the same time. The MLSORLS is numerically much stable than any other adaptive algorithms in the finite precision computation; its lattice structure has inborn numerical stability, furthermore, it is a order recursive smoothing algorithm which has better numerical stability than those of other delayed filtering algorithms [9]. Computational complexity increases rapidly as the number of antennas of the MIMO systems increase. In other words, the numerical property of algorithms being used in the MIMO systems gets more important. In this sense, we believe that the proposed algorithm contributes to such a complex implementation of the MIMO systems. The next section briefly shows the derivation of the MLSORLS. In Section 4, we show some simulation results of the MLSORLS and compare them with other algorithms. Finally, we give our concluding remarks in Section 5.

The authors would like to thank the Ministry of Education of Korea for its financial support toward the Electrical and Computer Engineering Division at POSTECH through its BK21 program, and Postech Research Fund.

2. THE MLSORLS ALGORITHM

The overall system model is depicted in Figure 1. The signal vector $\mathbf{d(k)}$ is transmitted by the l transmitting antennas, and propagates through the MIMO channel. Then, the s receiving antennas receive $\mathbf{x(k)}$ at the end of the MIMO channel. During the transmission, signals are distorted and the white Gaussian noises are added. We shall design the efficient discrete-time MLSORLS system

$$\mathbf{y(k)} = \mathbf{C_M^T(k)x_M(k)}, \tag{1}$$

where

$$\begin{aligned}
\mathbf{y(k)} &= [y_1(k)\, y_2(k)\, \ldots\, y_l(k)]^T, \\
\mathbf{x_M(k)} &= [x_1(k)\, \ldots\, x_s(k)\, \ldots\, \ldots \\
&\quad\; x_1(k-M+1)\, \ldots\, x_s(k-M+1)]^T, \\
C_M(k) &= [\mathbf{c_1^T(k)}\, \mathbf{c_2^T(k)}\, \ldots\, \mathbf{c_M^T(k)}]^{\mathbf{T}},
\end{aligned}$$

where M is the order of the MLSORLS, and $\mathbf{c_i}(\mathbf{k})$ is the i-th order $s \times l$ channel impulse response matrix.
The smoother error vector is

$$\mathbf{e^s_M}(\mathbf{k}) = \mathbf{d}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor) - \mathbf{y}(\mathbf{k}), \tag{2}$$

where $\lfloor \ \rfloor$ is the well-known floor operator. The objective is to minimize the sum of the squared errors,

$$\mathbf{E}(N) = \sum_{k=M-1}^{N} \mathbf{e^{sT}_M}(\mathbf{k}) \mathbf{e^s_M}(\mathbf{k}). \tag{3}$$

The optimal value of the $C_M(N)$ which minimizes (3) can be easily obtained as the following normal equation,

$$R_M(N) C_M(N) = r_M(N), \tag{4}$$

where

$$R_M(N) = \sum_{k=M-1}^{N} \mathbf{x_M}(\mathbf{k}) \mathbf{x^T_M}(\mathbf{k}),$$

$$r_M(N) = \sum_{k=M-1}^{N} \mathbf{x_M}(\mathbf{k}) \mathbf{d^T}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor).$$

Before we go on, we refer to the multi-channel lattice predictor. We abbreviate its derivation in this paper because of paper limitation; it can be seen in [8]. Instead, we summarize the predictor in the Table 1. Symbols used in the predictor are also appeared in Table 1.
To obtain the order-update recursive formulation for the $C_M(N)$, it is required to consider two different order-update recursions, the forward and the backward order-update recursions. We start with the derivation of the backward order-update recursion. The M-th order error vector $\mathbf{e^s_M}(\mathbf{k})$ can be updated to the $M + 1$-th order error vector

$$\mathbf{e^s_{M+1}}(\mathbf{k}) = \mathbf{d}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor) - \mathbf{C^T_{M+1}}(\mathbf{k}) \mathbf{x_{M+1}}(\mathbf{k}).$$

When the correlation matrix $R_{M+1}(N)$ is multiplied by $[C^T_M(N) \ \mathbf{0}]$, this gives the following equation

$$R_{M+1}(N) \begin{bmatrix} C_M(N) \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} r_M(N) \\ H^{bT}_M(N) r_M(n) \end{bmatrix},$$

and it can be written

$$R_{M+1}(N)(C_{M+1}(N) - \begin{bmatrix} C_M(N) \\ \mathbf{0} \end{bmatrix}) = \begin{bmatrix} \mathbf{0} \\ K^b_M(N) \end{bmatrix}, \tag{5}$$

where

$$K^b_M(N) = \sum_{k=M-1}^{N} \mathbf{x}(\mathbf{k} - \mathbf{M}) \mathbf{d^T}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor) - H^{bT}_M(N) r_M(N). \tag{6}$$

From the (5) we can obtain the backward order-update recursion equation of the smoother coefficient matrix as

$$C_{M+1}(N) = \begin{bmatrix} C_M(N) \\ \mathbf{0} \end{bmatrix} + B_M(N) E^{-b}_M(N) K^b_M(N),$$

which can be written with the priori smoother error vector $\alpha^\mathbf{s}_\mathbf{M}(\mathbf{N})$ as

$$\alpha^\mathbf{s}_\mathbf{M+1}(\mathbf{N}) = \alpha^\mathbf{s}_\mathbf{M}(\mathbf{N}) - \mathbf{K^{bT}_M}(\mathbf{N-1}) \mathbf{E^{-bT}_M}(\mathbf{N-1}) \alpha^\mathbf{b}_\mathbf{M}(\mathbf{N}),$$

where

$$\alpha^\mathbf{s}_\mathbf{M}(\mathbf{N}) = \mathbf{d}(\mathbf{N} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor) - \mathbf{C^T_M}(\mathbf{N-1}) \mathbf{x_M}(\mathbf{N}).$$

The M-th order error vector $\mathbf{e^s_M}(\mathbf{k})$ can be also updated to the $(M + 1)$-th error vector

$$\mathbf{e^s_{M+1}}(\mathbf{k-1}) = \mathbf{d}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor - \mathbf{1}) - \mathbf{C^T_{M+1}}(\mathbf{k-1}) \mathbf{x_{M+1}}(\mathbf{k}),$$

where the forward order update equation is required. Similar to the backward order-update recursion, we can write

$$R_{M+1}(N)(C_{M+1}(N-1) - \begin{bmatrix} \mathbf{0} \\ C_M(N-1) \end{bmatrix}) = \begin{bmatrix} K^f_M(N) \\ \mathbf{0} \end{bmatrix}, \tag{7}$$

where

$$K^f_M(N) = \sum_{k=M-1}^{N} \mathbf{x}(\mathbf{k}) \mathbf{d^T}(\mathbf{k} - \lfloor \frac{\mathbf{M}}{\mathbf{2}} \rfloor - \mathbf{1}) - H^{fT}_M(N) r_M(N-1).$$

Then, the forward order-update recursion equation is written as

$$C_{M+1}(N-1) = \begin{bmatrix} \mathbf{0} \\ C_M(N-1) \end{bmatrix} + A_M(N) E^{-f}_M(N) K^f_M(N),$$

and we can rewrite this equation with the priori smoother error vector as

$$\alpha^\mathbf{s}_\mathbf{M+1}(\mathbf{N-1}) = \alpha^\mathbf{s}_\mathbf{M}(\mathbf{N-1}) - K^{fT}_M(N-1) E^{-fT}_M(N-1) \alpha^\mathbf{f}_\mathbf{M}(\mathbf{N}).$$

Till now, we have obtained two order-update recursions for the smoother coefficient $C_M(N)$. To complete the recursions, time-update recursions for $K^b_M(N)$ and $K^f_M(N)$ have to be derived. From (6), we can write

$$
\begin{aligned}
&K_M^b(N) = \\
&\sum_{k=M-1}^{N} \{\mathbf{x(k-M)} - \mathbf{H_M^{bT}(N)x_M(k)}\}\mathbf{d^T(k-\lfloor\frac{M}{2}\rfloor)} \\
&= \sum_{k=M-1}^{N} \{\mathbf{x(k-M)} - \\
&\quad H_M^{bT}(N-1)(\mathbf{x})_{\mathbf{M}}(\mathbf{k})\}\mathbf{d^T(k-\lfloor\frac{M}{2}\rfloor)} - \\
&\quad \alpha_{\mathbf{M}}^{\mathbf{b}}\mathbf{(N)g^T(N)} \sum_{\mathbf{k=M-1}}^{\mathbf{N}} \mathbf{x_M(k)d^T(k-\lfloor\frac{M}{2}\rfloor)} \\
&= K_M^b(N-1) + \{\mathbf{x(N-M)} - \mathbf{H_M^{bT}(N-1)x_M(N)}\}\cdot \\
&\quad \mathbf{d^T(N-\lfloor\frac{M}{2}\rfloor)} - \alpha_{\mathbf{M}}^{\mathbf{b}}\mathbf{(N)g^T(N)r_M(N)} \\
&= K_M^b(N-1) + \alpha_{\mathbf{M}}^{\mathbf{b}}\mathbf{(N)e_M^s(N)}.
\end{aligned}
$$

Similarly, we can write the time-update of $K_M^f(N)$ as

$$K_M^f(N) = K_M^f(N-1) + \alpha_{\mathbf{M}}^{\mathbf{f}}\mathbf{(N)e_M^s(N-1)}.$$

The complete description for the MLSORLS is provided in the Table 1. We shall see the numerical property of the MLSORLS through some simulation results in the next section.

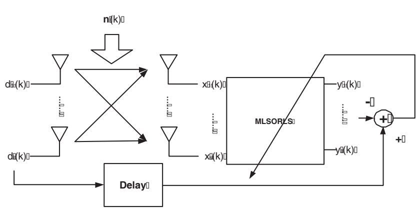

Fig. 1. Figure 1. System model

3. SIMULATION RESULTS

Simulations are performed by using Matlab 6.5. l-independent Bernoulli sequences with values ± 1 are generated and transmitted through the MIMO channel in which each channel impulse response is defined by

$$h_n = \begin{cases} \frac{1}{2}\left[1+cos(\frac{2\pi}{Z}(n-2))\right], & n=1,2,3 \\ 0 & otherwise \end{cases} \quad (8)$$

where Z controls the amount of amplitude distortion, i.e. the eignvalue spread $\chi(\mathbf{R})$. The inter-channel interference is considered; main path gain values are set to 1 and the other path gain values are set to 0.3. Additive white Gaussian noise with variance 0.001 is introduced at the each channel output, and Z is set to 3.5. To investigate effects of the finite-precision arithmetic, we control a threshold value in the inverse operation over the algorihtm. If singular values of a matrix are smaller than the threshold value, their inverted values are considered as zero; similar approach has been performed in [7]. To see these effects we compared the MLSORLS with the delayed MLSORL filter, of which desired vector is generally delayed the half filter order and then fed into the filter. Figure 2 shows the simulation result of the MLSORLS and the delayed MLSORL filter with no threshold value, thus it is possible to compute exact inverse matrix in the algorithms. As supposed, the two algorithms have their nearly identical performances. However, when the threshold is applied to their inverse operations, the MLSORL filter is affected and degraded. As the threshold value increases, its performance gets much deteriorated as shown in Figure 3 and 4. On the other hand, the MLSORLS keeps its performance regardless of the threshold value except for slightly slowed convergent behavior.

4. CONCLUSIONS

We have introduced the multi-channel least squares order recursive lattice smoothing algorithm, which gives advantages for implementing the MIMO communication systems. In the simulation results, we have showed that it has better numerical property than other adaptive algorithms under the poor computational environment.

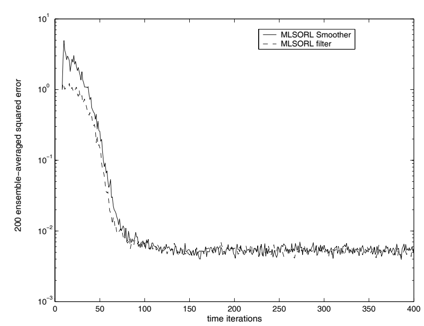

Fig. 2. Figure 2. Learning curves for the MLSORLS and the MLSORL filter with no threshold

5. REFERENCES

[1] G. J. Foschini, "Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multiple-Element Antennas," *Bell Labs Technical Journal*, Autumn 1996, pp.41-59

[2] J. B. Andersen, "Antenna Arrays in Mobile Communications: Gain, Diversity, and Channel Capacity," *IEEE Antennas and Propagation Magazine*, Vol.42, No. 2, April 2000

[3] T. L. Marzetta, "BLAST Training: Estimating Channel Characteristics for High Capacity Space-time Wireless,"

Table 1. The normalized LSORL Smoothing Algorithm

Parameter and variable descriptions:

$\alpha^{\mathbf{f}}_{\mathbf{M}}, \alpha^{\mathbf{b}}_{\mathbf{M}}$: the M-th order priori forward (backward) prediction error vector

Γ^f_M, Γ^b_M: the M-th order forward (backward) reflection coefficient matrix

E^f_M, E^b_M: the M-th order forward (backward) prediction error covariance matrix

MLSORL predictor:

$$\alpha^{\mathbf{f}}_{\mathbf{M}}(\mathbf{N}) = \alpha^{\mathbf{f}}_{\mathbf{M-1}}(\mathbf{N}) + \mathbf{\Gamma^{fT}}(\mathbf{N-1})\alpha^{\mathbf{b}}_{\mathbf{M-1}}(\mathbf{N-1})$$
$$\alpha^{\mathbf{b}}_{\mathbf{M}}(\mathbf{N}) = \alpha^{\mathbf{b}}_{\mathbf{M-1}}(\mathbf{N-1}) + \mathbf{\Gamma^{bT}}(\mathbf{N-1})\alpha^{\mathbf{f}}_{\mathbf{M-1}}(\mathbf{N})$$
$$\Delta_M(N) = W\Delta_M(N-1) + \varphi_{M-1}(N-1)\alpha^{\mathbf{b}}_{\mathbf{M-1}}(\mathbf{N-1})\alpha^{\mathbf{fT}}_{\mathbf{M-1}}(\mathbf{N})$$
$$\Gamma^f_M(N) = -E^{-b}_{M-1}(N-1)\Delta_M(N)$$
$$\Gamma^b_M(N) = -E^{-f}_{M-1}(N-1)\Delta^T_M(N)$$
$$E^f_M(N) = WE^f_M(N-1) + \varphi_M(N-1)\alpha^{\mathbf{f}}_{\mathbf{M}}(\mathbf{N})\alpha^{\mathbf{fT}}_{\mathbf{M}}(\mathbf{N})$$
$$E^b_M(N) = WE^b_M(N-1) + \varphi_M(N)\alpha^{\mathbf{b}}_{\mathbf{M}}(\mathbf{N})\alpha^{\mathbf{bT}}_{\mathbf{M}}(\mathbf{N})$$
$$\varphi_{M+1}(N) = \varphi_M(N) - |\varphi_M(N)|^2\alpha^{\mathbf{bT}}_{\mathbf{M}}(\mathbf{N})\mathbf{E}^{-\mathbf{bT}}_{\mathbf{M}}(\mathbf{N})\alpha^{\mathbf{b}}_{\mathbf{M}}(\mathbf{N})$$

MLSORLS:

$$\alpha^{\mathbf{s}}_{\mathbf{M+1}}(\mathbf{N}) = \alpha^{\mathbf{s}}_{\mathbf{M}}(\mathbf{N}) - \mathbf{K}^{\mathbf{bT}}_{\mathbf{M}}(\mathbf{N-1})\mathbf{E}^{-\mathbf{bT}}_{\mathbf{M}}(\mathbf{N-1})\alpha^{\mathbf{b}}_{\mathbf{M}}(\mathbf{N})$$
$$K^b_M(N) = K^b_M(N-1) + \alpha^{\mathbf{b}}_{\mathbf{M}}(\mathbf{N})\mathbf{e}^{\mathbf{s}}_{\mathbf{M}}(\mathbf{N})$$
$$\alpha^{\mathbf{s}}_{\mathbf{M+1}}(\mathbf{N-1}) = \alpha^{\mathbf{s}}_{\mathbf{M}}(\mathbf{N-1}) - K^{fT}_M(N-1)E^{-fT}_M(N-1)\alpha^{\mathbf{f}}_{\mathbf{M}}(\mathbf{N})$$
$$K^f_M(N) = K^f_M(N-1) + \alpha^{\mathbf{f}}_{\mathbf{M}}(\mathbf{N})\mathbf{e}^{\mathbf{s}}_{\mathbf{M}}(\mathbf{N-1})$$

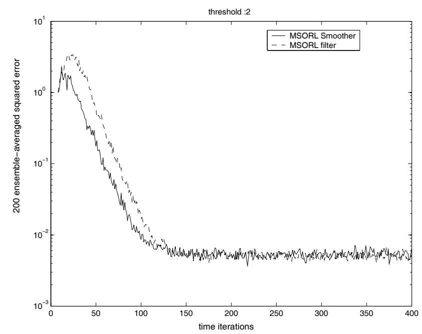

Fig. 3. Learning curves for the MLSORLS and the MLSORL filter with threshold 2

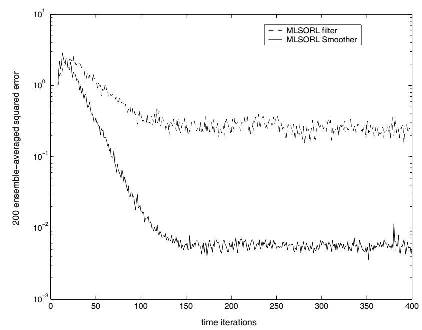

Fig. 4. Learning curves for the MLSORLS and the MLSORL filter with threshold 5

Proceedings 37th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, Sep. 22-24, 1999

[4] P. W. Wolniansky, G. J. Foschini, G. D. Golden and R. A. Valenzuela, "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *IEEE ISSSE-98*, Pisa, Italy, pp. 295-300, Sept. 1998.

[5] V. Tarokh, N. Seshadri and A. R. Calderbank, "Space-time codes for high data rate wireless communications: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744-765, March 1998.

[6] S. M. Alamouti, "A simple transmit diversity techniques for wireless communications," *IEEE J. Selected Areas Commun.*, vol. 16, pp. 1451-1458, Oct. 1998.

[7] J. T. Yuan, and J. A. Stuller, "Least squares order-recursive lattice smoothers", *IEEE Trans. Signal Processing*, Vol. 43, 1995, pp. 1058-1067

[8] N. Kalouptsidis, G. Carayannis, D. Manolakis, and E. Koukoutsis, "Efficient Recursive in Order Least Squares FIR Filtering and Prediction," *IEEE Trans. ASSP*, Vol. ASSP-33, No. 4, Oct. 1985

[9] D. K. Kim, and P. Park, "The Normalized Least Squares Order-Recursive Lattice Smoother," *Signal Processing*, Vol. 82, 2002, pp. 895-905