CONVERGENCE PROPERTIES OF THE BLOCK ORTHOGONAL PROJECTION ALGORITHM

Kazushi IKEDA and Hideaki SAKAI

Graduate School of Informatics
Kyoto University, Kyoto 606-8501 Japan
kazushi@i.kyoto-u.ac.jp.

ABSTRACT

The normalized LMS (N-LMS) algorithm has a disadvantage that the convergence rate is much worse when the input signal is colored. To overcome this, the affine projection algorithm and the block orthogonal projection (BOP) algorithm which are applied the block signal processing technique to the N-LMS algorithm are proposed although the reason why they are tough against the coloredness is not given yet. This paper gives the convergence rate of the BOP algorithm for colored input signals, which shows the superiority of the BOP algorithm. To put it concretely, we derive the expression of the convergence rate, propose an approximation method to calculate it, and confirm the result by computer simulations. We also consider the relation between the block size and the convergence rate formally and geometrically.

1. INTRODUCTION

The normalized LMS (N-LMS) algorithm [6] is one of the most popular adaptive algorithm for transversal filters because of its simplicity and stability. However, its convergence rate gets much worse when the input signal becomes colored. In order to overcome the disadvantage, the affine projection algorithm [3, 7] and the block orthogonal projection (BOP) algorithm [2] are proposed which are straightforwardly applied the block signal processing technique to the N-LMS algorithm. Because they are easily unified and generalized by considering a general period of update [4], we only treat the BOP algorithm in this paper for simplicity.

The BOP algorithm uses a set of input vectors in one update and it has good convergence properties even for colored inputs. Its effect is confirmed by computer simulations, however, it is not supported theoretically yet. Either, such an interesting phenomenon has been reported that the improvement of performance stops when the block size approaches a certain value [5] though the previous works have shown that the algorithm which has a larger block size performs better when the input signal is white [4, 8]. The difficulties in theoretical analyses mainly exist in evaluating the expectation of the inverse matrix used in the algorithm.

In this paper, we give the derivation of the convergence rate of the BOP algorithm when the input signal is colored. Since its direct calculation is impossible, we use an approximation which is supported by the law of large numbers to evaluate the expectation. The results based on the approximation agree with those by the computer simulations and explain the reason of the phenomenon well.

2. DEFINITION OF BOP ALGORITHM

The input signal vector, the tap-weight vector and the output of the adaptive transversal filter at the time t are denoted by $\boldsymbol{x}(t)$, $\boldsymbol{w}(t)$ and $y(t)$, respectively. Then,

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + \frac{\boldsymbol{x}(t)e(t)}{\|\boldsymbol{x}(t)\|^2} \tag{1}$$

shows the N-LMS algorithm that minimizes the square error $e(t)^2 = (y_\mathrm{o}(t) - y(t))^2$ where $y_\mathrm{o}(t) = {\boldsymbol{w}_\mathrm{o}}^T\boldsymbol{x}(t)$ is the desired output and $\boldsymbol{w}_\mathrm{o}$ is the optimal tap-weight vector, respectively.

Instead of the square error, the Block Orthogonal Projection (BOP) algorithm with the block size m minimizes the norm of the error vector

$$\boldsymbol{e}(t) = [e(t), \ldots, e(t-m+1)]^T \in \boldsymbol{R}^m$$

and is written as

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + X^+(t)\boldsymbol{e}(t) \tag{2}$$

where $X^+(t) = X(t)(X(t)^T X(t))^{-1}$ is the transposition of the Moore-Penrose generalized inverse matrix of $X(t) = [\boldsymbol{x}(t), \ldots, \boldsymbol{x}(t-m+1)]$ [2].

From the geometrical point of view, the N-LMS algorithm projects the difference vector $\varepsilon(t) = \boldsymbol{w}(t) - \boldsymbol{w}_\mathrm{o}$

to the hyperplane made from the input signal $\boldsymbol{x}(t)$ and the BOP algorithm does to the orthogonal complement of the m input signal vectors $\boldsymbol{x}(t), \boldsymbol{x}(t-1), \ldots, \boldsymbol{x}(t-m+1)$. Its name results from this property. Since the BOP algorithm uses m inputs in one update, the period of the update is set to m. When set to 1, it is called the affine projection algorithm [3, 7] and any natural number is available. Their geometrical relation is elucidated in [4], however, this paper treats only the BOP algorithm for simplicity.

3. CONVERGENCE RATE OF BOP ALGORITHM

To evaluate the convergence speed, it is enough to know how much the magnitude of the difference vector $\varepsilon(t)$ decreases. So, we consider the convergence rate defined as the expectation of the rate $\|\varepsilon(t+1)\|/\|\varepsilon(t)\|$ in the following. Since the difference vectors $\varepsilon(t+1)$ and $\varepsilon(t)$ satisfy

$$\varepsilon(t+1) = (I_N - X^+(t)X(t)^T)\varepsilon(t),$$

we evaluate the maximum eigenvalue $\lambda_{\max}$ of the expectation of $I_N - X^+(t)X(t)^T$. The convergence rate is written as $\lambda_{\max}^{1/m}$ since the update is done for every m times. Note that the independence of $X(t)$ and $\varepsilon(t)$ is implicitly assumed here which makes the problem a little easier. And we also assume that the input signal is stationary.

Since the i, jth element of $X(t)^T X(t)$ is written as

$$\sum_{k=0}^{N-1} x(t-i-k)x(t-j-k),$$

it can be approximated by its average $\mathrm{Ex}\left[x(t-i)x(t-j)\right]$ using the law of large numbers. So, we adopt the approximation

$$\begin{aligned} & \mathrm{Ex}\left[X(t)\left(X(t)^T X(t)\right)^{-1} X(t)^T\right] \\ \approx\ & \mathrm{Ex}\left[X(t)(\mathrm{Ex}\left[X(t)^T X(t)\right])^{-1} X(t)^T\right] \qquad (3) \end{aligned}$$

to calculate $\lambda_{\max}$. This method is available even when the input is colored because only the second or lower order statistics appear in Eq.(3). A similar approximation is used in [1] where the whiteness of the input vector is assumed though.

Define an $n \times n$ matrix R_n as

$$\begin{aligned} R_n &= \mathrm{Ex}\left[\boldsymbol{x}_n(t)^T \boldsymbol{x}_n(t)\right], \\ \boldsymbol{x}_n(t) &= [x(t), \ldots, x(t-n+1)]. \end{aligned}$$

Then, since $\mathrm{Ex}\left[X(t)^T X(t)\right] = N R_m$,

$$P = \frac{1}{N}\mathrm{Ex}\left[X(t) R_m^{-1} X(t)^T\right] \qquad (4)$$

and the i, jth element P_{ij} is written as

$$P_{ij} = \frac{1}{N}\mathrm{Ex}\left[\boldsymbol{x}_m(t-i)^T R_m^{-1} \boldsymbol{x}_m(t-j)\right] \qquad (5)$$

where $\boldsymbol{x}_m(t) = [x(t), \ldots, x(t-m+1)]$. Using $\mathrm{tr}AB = \mathrm{tr}BA$,

$$\begin{aligned} P_{ij} &= \frac{1}{N}\mathrm{Ex}\left[\boldsymbol{x}_m(t-i)^T R_m^{-1} \boldsymbol{x}_m(t-j)\right] \\ &= \frac{1}{N}\ \mathrm{tr}\left(\mathrm{Ex}\left[\boldsymbol{x}_m(t-i)\boldsymbol{x}_m(t-j)^T\right] R_m^{-1}\right) \\ &= \frac{1}{N}\ \mathrm{tr}\left(\mathrm{Ex}\left[\boldsymbol{x}_m(t-(i-j))\boldsymbol{x}_m(t)^T\right] R_m^{-1}\right) \end{aligned}$$

and then

$$P_{ij} = \frac{1}{N}\ \mathrm{tr}(R_m(i-j)R_m^{-1}) \qquad (6)$$

where $R_m(n) = \mathrm{Ex}\left[\boldsymbol{x}_m(t-n)\boldsymbol{x}_m(t)^T\right]$, which means that P is determined by only the autocorrelation function r_n.

4. BOP ALGORITHM AND N-LMS ALGORITHM

The convergence rate derived above has much complicated form, however, when the input signal is white, it is easily derived as follows: R_m is the null matrix, of course, and the diagonal of $R_m(i)$ is zero if $i \neq 0$. Hence, $P = \frac{m}{N} I_N$ and $\lambda_{\max}$ is $\frac{N-m}{N}$. In this section, we consider the geometrical relation of the BOP algorithm and N-LMS algorithm according to [4], and support the result in the previous section from another side.

For simplicity, we at first consider the case $m = 2$. Let the input vectors be denoted by $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, which are chosen uniformly and independently. Since each input vector makes a hyperplane, we denote the hyperplane itself by $\boldsymbol{x}_1$ or $\boldsymbol{x}_2$. When $\boldsymbol{x}_1$ is given, the N-LMS algorithm projects the difference vector ε to the hyperplane $\boldsymbol{x}_1$. And the BOP algorithm projects ε to the intersection of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. Here, the projection to the intersection coincides the point given by projecting ε to $\boldsymbol{x}_1$ and projecting it to the intersection in the hyperplane $\boldsymbol{x}_1$. It means that we can regard $\boldsymbol{x}_2$ as a hyperplane in the hyperplane $\boldsymbol{x}_1$. Since $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are independent and the hyperplane $\boldsymbol{x}_1$ is $N-1$-dimensional, the convergence rate of the BOP algorithm is the square root of the product of those of the N-dimensional and the $N-1$-dimensional N-LMS algorithms. This consideration is easily extended to any block size and it is found that the BOP algorithm with the block size m is equivalent to the combination from the N-dimensional to the $N-m+1$-dimensional N-LMS algorithms when the input is white.

The convergence rate of the k-dimensional N-LMS algorithm derived by the above theory is $\frac{k-1}{k}$. Since

$$\frac{N-k}{N} = \frac{N-1}{N}\frac{N-2}{N-1}\cdots\frac{N-k}{N-k+1},$$

the theory is also supported from their geometrical relation.

5. COMPUTER SIMULATIONS

In order to confirm the theory numerically, we have done some computer simulations. Fig. 1 shows the relation between the number of iterations and the squared error when the transversal filter for system identification with 20 taps learns using the BOP algorithm with $m = 5$ with the input signal

$$x(t) = \sum_{i=0}^{6} u(t)$$

where $u(t)$ is white Gaussian. In Fig. 1, the *'s and the

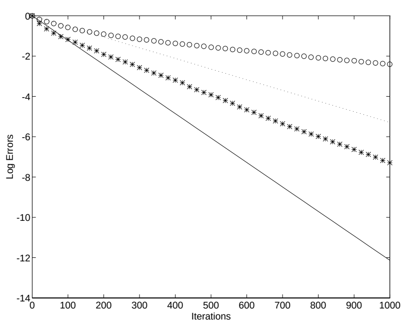

Figure 1: Squared Errors (Colored Inputs)

dot-line respectively show the average of 30 trials and the theoretical value. The slope of the theory agrees with that of the simulations as iterations increase. For next discussion, the experimental result of the N-LMS algorithm is shown by the o's and the solid line means the theoretical result when the input is white.

6. BLOCK SIZE AND CONVERGENCE RATE: FORMAL VIEW

We compare the results of Fig. 1 with the results of white inputs' case (Fig. 2) where the o's and *'s show the experimental results of the N-LMS and BOP algorithms, respectively. The two figures clearly show that the difference of the slopes in white inputs' case is much smaller than the other, which means the block

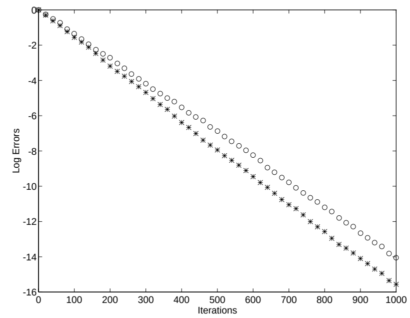

Figure 2: Squared Errors (White Inputs)

implementation is less effective when the input signal is white. We consider about this phenomenon in this section.

From Eq.(6),

$$P_{ii} = \frac{1}{N}\,\mathrm{tr}(I_m) = \frac{m}{N}$$

is easily derived because $R_m(0) = R_m$. This means that the diagonal elements are always m/N therefore P can be rewritten as

$$P = \frac{m}{N}\hat{R}_m$$

where $\hat{R}_m$ has diagonal elements with unity. Denoted the minimum eigenvalue of $\hat{R}_m$ by $\hat{\lambda}_m$, the convergence rate of the BOP algorithm is written as

$$\left(1 - \frac{m}{N}\hat{\lambda}_m\right)^{1/m}.$$

By Taylor expansion with respect to $(\hat{\lambda}_m/N)$, the above is rewritten as

$$1 - \frac{\hat{\lambda}_m}{N} - \frac{m-1}{2}\frac{\hat{\lambda}_m^2}{N^2} + \mathrm{O}\left(\hat{\lambda}_m^3/N^3\right). \qquad (7)$$

When the input is white, $\hat{R}_m$ becomes I_m as shown in Sec.4, therefore $\hat{\lambda}_m$ is invariant to m. Hence, the effect of the increase of m is only the order of $1/N^2$ which appears in the third term. On the other hand, when the input signal is colored, $\hat{R}_m$ itself changes by the increase of m and $\hat{\lambda}_m$ changes, too. Hence the increase of m affects with the order of $1/N$ which appears in the second term. In fact, the results of the computer simulations above have shown that the minimum eigenvalue of $\hat{R}_1$ is 0.0223 and the rate of the minimum and the maximum is 292, and those of $\hat{R}_5$ are 0.237 and 8.65, respectively.

The consideration in this section means that the main reason why the BOP algorithm is effective is not that the number of examples used in one update increases itself but that the spread of the eigenvalues is corrected and that the increase of the block size is needless after the spread is corrected enough. This conclusion does not contradict with the more precise theoretical analyses in [4, 8] because the increase is surely effective though slightly.

The future work should elucidate the relation between the change of $\hat{R}_m$ and the statistical properties of the input signal.

7. BLOCK SIZE AND CONVERGENCE RATE: GEOMETRICAL VIEW

In this section, we consider about the phenomenon from the geometrical point of view. Assume that the input signal $x(t)$ is the output of k-AR model ($k < m$)driven by white Gaussian noise, that is,

$$x(t) = \sum_{i=1}^{k} a_i x(t-i) + u(t). \tag{8}$$

Consider here $N-k$ vectors $\boldsymbol{a}_i, i = 0, \ldots, N-k-1$ that

$$\boldsymbol{a}_i = (0_i, 1, -a_1, -a_2, \ldots, -a_k, 0_{N-k-1-i})^T \in \boldsymbol{R}^N$$

where 0_j is the j-dimensional null vector. From Eq.(8), the inner product of $\boldsymbol{a}_i$ and $\boldsymbol{x}(t) = (x(t), \ldots, x(t-N+1))^T$ is $u(t-i)$. So, if $u(t)$ has a small variance or a_i is rather large, it is neglectable and $\boldsymbol{x}(t)$ is orthogonal to the vectors $\boldsymbol{a}_i, i = 0, \ldots, N-k-1$. Since $\boldsymbol{x}(t)$ is perpendicular to $N-k$ vectors for any t, the space spanned by the vectors $\boldsymbol{x}(t), \boldsymbol{x}(t-1), \ldots, \boldsymbol{x}(t-m+1)$ is only k-dimensional. Because the BOP algorithm projects the difference vector to the complement of the space, the convergence properties do not change even if m increases. That is why the increase of the block size becomes less effective when m is enough large. In practice, even if $u(t)$ is not so small and the spanned space does not degenerate, the vectors $\boldsymbol{x}(t), \ldots, \boldsymbol{x}(t)$ are extremely biased and then the convergence does not becomes faster. To analysis the relation of the bias and the convergence rate is still an open problem.

8. CONCLUSION

The BOP algorithm is said that its convergence speed does not decrease even if the input is colored, though there are no explanations for it. We derived the convergence rate for colored inputs using an approximation supported by the law of large numbers. The results of computer simulations agree with the theoretical values.

We consider the effects of the block implementation in the cases that the input signal is colored from the formal and geometrical viewpoints. The considerations explain the phenomenon well, however, they are still incomplete and quantitative analyses should be given in the future.

9. REFERENCES

[1] Amari, S., Murata, N. and Müller, K.-R., "Mathematical Model of Learning — Generalization and Overtraining," Higher-Order Information Processing in Brain, Asakura Pub., Tokyo, Japan, 1997.

[2] Furukawa, T., Kubota, H. and Tsujii, S., "Orthogonal Projection Algorithm for Block Adaptive Signal Processing and Its Some Properties," Trans. of IEICE, vol. J71-A, pp. 2138–2146, 1988.

[3] Hinamoto, T. and Maekawa, S., "An Extended Learning Identification," Trans. IEEJ., vol. C-95, pp. 227–234, 1975.

[4] Ikeda, K.: Convergence Rate Analysis and Optimization of Block Size of Block Orthogonal Projection Algorithm, J. JSIAM, vol. 7, 1997, in press.

[5] Kaneda, Y., private communications.

[6] Nagumo, J. and Noda, A., "A Learning Method for System Identification," IEEE Trans. AC, vol. 12, pp. 282–287, 1967.

[7] Ozeki, K. and Umeda, T., "An Adaptive Filtering Algorithm Using an Orthogonal Projection to an Affine Subspace and Its Properties," Trans. IEICE, vol. J67-A, pp. 126–132, 1984.

[8] Takamura, M. and Yahagi, T., "On the Convergence Rate of Learning Identification and Orthogonal Projection Algorithms," Trans. IEICE, vol. J73-A, pp. 652–655, 1990.