The Fifth International Conference
on Neural Information Processing
Kitakyushu, Japan October 21-23,1998

Block-Size Optimization of Block Orthogonal Projection Algorithm for Linear Dichotomies

Kazushi Ikeda†, *Seiji Miyoshi*†† *and Kenji Nakayama*†††
Email:kazushi@kuamp.kyoto-u.ac.jp

†Grad. Sch. of Informatics, Kyoto Univ.
Yoshida-Honmachi, Sakyo, Kyoto 606-8501, Japan
††Kobe City Coll. of Technology
†††Fac. of Engineering, Kanazawa Univ.

ABSTRACT

The block orthogonal projection algorithm which is one for transversal filters can be applied to a linear dichotomy (so called Perceptron) which consists of a transversal filter and a sign function. When the block size which is the number of examples used in one time of renewal is one, the algorithm is equivalent to the normalized LMS algorithm and is proven to stop in a finite number of iterations when the learning coefficient is unity. This report gives the block size which maximizes the convergence rate when the learning coefficient is unity, and confirms it by computer simulations. The results say that larger block size is not necessarily better.

KEYWORDS: Perceptron, Block Orthogonal Projection Algorithm, Convergence Rate

1. Introduction

In the field of linear adaptive filters, transversal filters which output

$$y = x^t w \in R \tag{1}$$

for the input signal vector $x \in R^m$ are most popular. The Least Mean Square (LMS) algorithm,

$$\Delta w = \mu x(d - x^t w), \tag{2}$$

is one of the simplest adaptive algorithms for transversal filters where μ, d and $(d - x^t w)$ are the learning coefficient, the desired output and the output error, respectively. It is a kind of stochastic descent methods and it makes the weight vector converge in probability to the optimal which minimizes the mean square error if $0 < \mu < 2/\lambda_{\max}$ where $\lambda_{\max}$ is the maximum of the eigenvalues of the covariance matrix of the input vector[2]. The discussion above means that we cannot set the learning coefficient to one which guarantees the convergence. The normalized LMS (N-LMS) algorithm[2, 9] is an advanced one in this point It is written as

$$\Delta w = \mu x(d - x^t w)/\|x\|^2 \tag{3}$$

where Δw is independent from the magnitude of the input x and the convergence condition is improved to $0 < \mu < 2$, though the convergence is still slow for colored input signals.

If we neglect the observation errors for simplicity, the N-LMS algorithm orthogonally projects the weight vector onto the hyperplane which intersects the input vector orthogonally and includes the optimal weight vector. From this point of view, the N-LMS algorithm is easily developed to the Block Orthogonal Projection (BOP) algorithm[3, 10, 1] which orthogonally projects the weight vector onto the space which intersects a set of input vectors orthogonally and includes the optimal weight vector. It is therefore written as

$$\Delta w = \mu X^+ e \tag{4}$$

where $X^+ = X(X^t X)^{-1}$ (the transposition of the Moore-Penrose generalized inverse matrix of X), X is an $N \times m$ matrix made from m input vectors

$$X = [x_1, \ldots, x_m], \tag{5}$$

and m dimensional error vector

$$e = [e_1, \ldots, e_m]^t. \tag{6}$$

The BOP algorithm is said to converge fast even when the input signal is colored, which is analyzed theoretically[4, 5].

On the other hand, a linear dichotomy (so called Perceptron)

$$y = \text{sign}\left[x^t w\right] \tag{7}$$

consists of a transversal filter and a sign function and is also used as an element of neural networks. Since the Perceptron Learning, a learning algorithm for linear dichotomies, is written as

$$\Delta w = \frac{1}{2} x \left(\text{sign}\left[x^t w^*\right] - \text{sign}\left[x^t w\right]\right) \tag{8}$$

using the true parameter w^*, it can be regarded as the LM-S algorithm for linear dichotomies. In a linear dichotomy, not only the output sign $\left[x^t w\right]$ but also the value $x^t w$ can be calculated because the input and the weight vector are known. So, the Perceptron Learning can be developed in the same way that the LMS algorithm is to the N-LMS or BOP algorithms.

When w gives the wrong output for the input x, $-aw$ outputs the true sign where a is a positive constant. Then, we can use $-ax^t w$ as the desired output and derive an algorithm

$$\begin{aligned} \Delta w &= \mu x(-a x^t w - x^t w) \\ &= -(1+a)\mu x x^t w \end{aligned} \tag{9}$$

where the renewal is done only when the output is wrong. And we assume $a = 1$ without loss of generality.

Normalizing Δw in Eq. (9) according to the weight as well as the N-LMS algorithm, we derive

$$\Delta w = -2\mu x x^t w/\|x\|^2 \tag{10}$$

which geometrically means that it orthogonally projects w onto the hyperplane $x^t w = 0$ when $\mu = 1/2$ and that it symmetrically moves w with regard to $x^t w = 0$ when $\mu = 1$ (Fig. 1).

It is well-known that the Perceptron Learning stops in a finite times of renewal when the learnable data set is given. The algorithm mentioned above, however, does not necessarily stops and it converges for a given data set if and only if

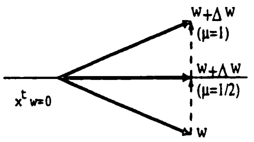


Figure 1: Geometrical Meaning of the N-LMS algorithm

$\mu = 1$[4, 8]. Hence, we consider the case of $\mu = 1$ in the following. The BOP algorithm for linear dichotomies can be defined[6] as well as the N-LMS algorithm as

$$\Delta w = -2\mu X^{+}X^{t}w \tag{11}$$

and its convergence properties are analyzed in [7, 8]. Miyoshi *et al.* have analyzed the convergence rate of the BOP algorithm, which is called Symmetric Learning Algorithm in [7], and have derived that the convergence rate is maximum when the block size (the number of examples used in one time of renewal) is set to a half of the dimension of the weight vector, though they have conjectured that the convergence rate becomes larger as the block size increases when it is small. In this paper, we show the conjecture above by considering how the increase of the block size influences to the convergence properties.

2. Block Orthogonal Projection Algorithm and Assumptions

Let the input and weight vectors x and w of the linear dichotomy be N dimensional vectors. Since the output of the linear dichotomy is independent from the magnitude of w and Δw in Eq. (11) does not change the magnitude, we can assume $\|w\| = \|w^{*}\| = 1$, that is, $w, w^{*} \in S$ where S means $N-1$ dimensional unit hypersphere without loss of generality. In the same reason, we also assume $x \in S_{+}$ where S_{+} is a half of S and the true machine always outputs $+1$ since an example $(x, -1)$ (the true machine with w^{*} outputs -1) is equivalent to $(-x, +1)$ in renewal. In the following, x itself is called an example and assumed to be chosen uniformly from S_{+}. Since the examples for which the current w outputs the true signs are not used in renewal, the examples are chosen from a part of S_{+} as shown in Fig. 2 in practice. The BOP algorithm with the block size k renews the weight vector w according to Eq. (11) using k examples chosen as above. Since the weight vector is symmetrically moved with respect to the complement of the space spanned by k examples, the BOP algorithm with $\mu = 1$ is called "Symmetric Learning Algorithm" in [7].

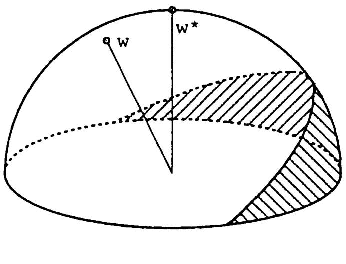


Figure 2: Distribution of Examples

3. Geometrical Meanings of Increase of Block Size

Let the complement R^{N-k} of the space spanned by k examples $x_1, \ldots, x_k$ be denoted by C, and the current weight vector, the true, their midpoint in R^{N}, and the line which includes the three points are defined by w, w^{*}, w_{mid}, and l, respectively. Fig. 3 clearly shows that the distance between w^{*} and w^{+} which is the symmetry of w with respect to C is twice as much as that between w_{mid} and w_{C} which is the orthogonal projection of w onto C, and the distance between w^{*} and w is so as that between w_{mid} and w. Here, we consider the case where another example x_{k+1} is also used in the renewal. Then, since w is moved symmetrically with respect to the complement C' of the space spanned by $k+1$ examples to the point named $w^{+'}$, the distance between w^{*} and $w^{+'}$ is twice of that between w_{mid} and $w_{\mathrm{C}}{}'$ which is the orthogonal projection of w onto C'. Therefore, we consider the distances of w_{C} and $w_{\mathrm{C}}{}'$ from w_{mid} in order to discuss the influence of the added example x_{k+1}.

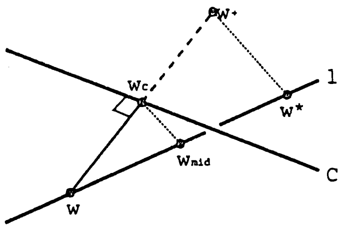


Figure 3: Relation of the weight vectors (in case $C = R^{1}$)

Since C' is a subspace of C and $w_{\mathrm{C}}{}' \in C$, $w_{\mathrm{C}}{}'$ is the orthogonal projection of w_{C} onto C' as shown in Fig. 4. And we divide the vectors to the element in C and its complement in order to compare the distances of w_{C} and $w_{\mathrm{C}}{}'$ from w_{mid}. Since w_{C} and $w_{\mathrm{C}}{}'$ exist in C, the relation of the distances of w_{C} and $w_{\mathrm{C}}{}'$ are equivalent to that from $w_{\mathrm{C}_{\mathrm{mid}}}$ in C, which is the orthogonal projection of w_{mid} onto C (Fig. 5). From its linearity, w_{C}, $w_{\mathrm{C}_{\mathrm{mid}}}$, and w_{C}^{*} which is the projection of w^{*} onto C are on the line l' which is the projection of l onto C, and $w_{\mathrm{C}_{\mathrm{mid}}}$ is the mid point of w_{C} and w_{C}^{*}.

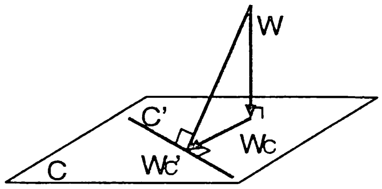


Figure 4: Properties of Orthogonal Projection

By the way, since the example x_{k+1} is chosen from the shadow part of Fig. 2 so that the outputs of w and w^{*} differ, they exist on the different sides of the hyperplane made from x_{k+1}. That means that l on which w and w^{*} exist intersects the hyperplane w_{k+1} at a point between them. If x_{k+1} is perpendicular to $x_1, \ldots, x_k$, it is easily proven that l' and C' intersects at the projection of the intersection point onto

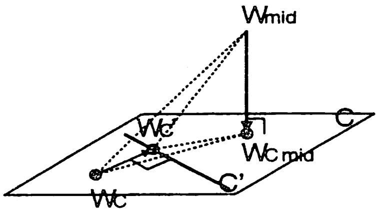


Figure 5: The relation of the Vectors

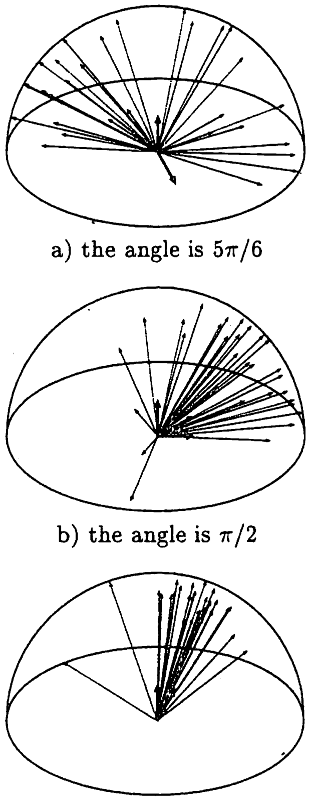
C. Even if not, we can expect that the intersection $w_{C\mathrm{in}}$ of l' and C' exists between w_C and w_C^* because randomly chosen k vectors are almost orthogonal to each other when N is large and k is small. Hence we assume that $w_{C\mathrm{in}}$ is between w_C and w_C^*. Since w_C' is the projection of w_C onto C' and $w_{C\mathrm{in}} \in C'$, the angle w_C-w_C'-$w_{C\mathrm{in}}$ is equal to $\pi/2$ and the angle w_C-w_C'-w_C^* is more than $\pi/2$. That means w_C' exists in the ball which has the segment w_C-w_C^* as a diameter and $w_{C\mathrm{mid}}$ as the center, therefore, w_C' is nearer to $w_{C\mathrm{mid}}$ than w_C. So, when the assumption is satisfied, the addition of x_{k+1} accelerates the learning.

Even when $\mu \neq 1$ but $\mu \in [1/2, 1]$, the same thing can be proven by considering the internally dividing point of w and w^* with $1/2 : \mu - 1/2$ instead of their midpoint w_{mid}.

4. Derivation of the Optimal Block Size

In the previous section, it has been shown that the learning is faster as the block size is larger when k is relatively small. On the other hand, how is the convergence rate when k is large. In this section, we give the answer that the BOP algorithm with the block size k and that with $N - k$ are essentially equivalent and have the same convergence rate under some assumptions.

In the consideration below, the current weight vector w has been fixed. The complement C of the space spanned by k examples $x_1, \ldots, x_k$ distributes uniformly in some sense if each of the k examples is chosen independently uniformly from S_+. In practice, however, examples are chosen a part of S_+ as shown in Fig. 2 according to w and C does not uniformly distribute. Fig. 6 is an example where $N = 3$, $k = 2$, the thick short arrows show w^* and w, and the long arrows show the directions of C's (vectors in this case) on S_+ made from 30 pairs of x_1 and x_2 which are randomly chosen. We can see from the figures that the distribution is biased as the angle of the current and true vectors is small. The bias, however, depends on w and is not simple when C has more dimension. So, we assume uniformity of C in the theoretical analysis. It means that the BOP algorithm with the block size k symmetrically moves w to a point w^1 with respect to C which is an R^{N-k} randomly uniformly chosen. And consider here about the complement $C^\perp$ of C and a point w_1 which is the symmetrical point of w with respect to $C^\perp$. Obviously, w^1 and w_1 are symmetrical with respect to the origin (Fig. 7). Noting the equivalence to choose R^{N-k} and R^k randomly because the whole space is R^N and each is the complement of the other, we can regard $C^\perp$ as the complement of the space spanned by $N - k$ examples, hence, the distribution of w^1 moved by k examples and that of w_1 moved by $N - k$ examples are symmetrical with respect to the origin. When the points gotten by one more time of renewal are denoted by w^2 and w_2, respectively, their distributions perfectly coincide.



a) the angle is $5\pi/6$

b) the angle is $\pi/2$

c) the angle is $\pi/6$

Figure 6: Distribution of the complement C

The consideration above says that the BOP algorithm with the block size k is essentially equivalent to that with the block size $N - k$, and their convergence rates are the same. Though Miyoshi *et al.* gave this result in [7], they considered fixed patterns and assumed that such patterns exist that they can span both a space and its complement and the probabilities they are chosen are the same. Here, we have shown the equivalence more simply by considering the distribution of the moved parameter assuming the uniformity of C .

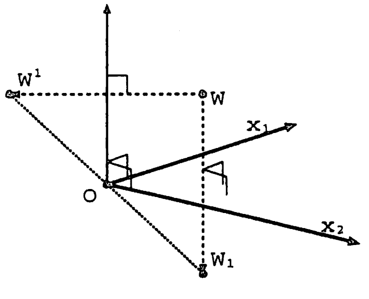


Figure 7: Relation of w^1 and w_1 ($N = 3$, $k = 2$)

Extrapolating the result in the previous section to $k \leq N/2$ and joining it and the result in this section, we can derive that the convergence rate is maximum when $k = N/2$. It is interesting the difference of the results in the linear dichotomies' case and the transversal filters' case when the convergence is fast as the block size is large[4].

5. Computer Simulations

Computer simulations have been done to confirm the theoretical result in the previous section. Though [7] has compared the number of iterations necessary to stop using fixed examples, we evaluate how much the weight vector approaches the true in a certain times of renewal considering that the theory above discusses the movement of the weight vector and that the learning does not stop when the given examples are randomly chosen.

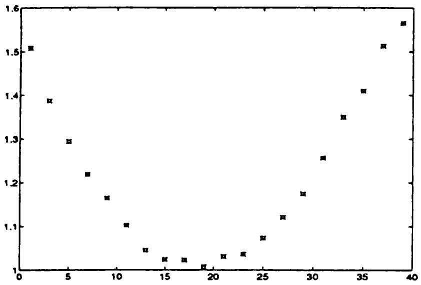


Figure 8: The angle of the weight vector and the true (2 times of renewal)

Fig. 8 shows the relation between the block size and the angle the weight vector w and the true make w^* after 2 times of renewal where the initial vector is randomly chosen so that it is perpendicular to w^*, the dimension of the weight vector is 40, and the angle is the average of 200 trials. It clearly confirms the theoretical result that the convergence rate is maximum when the block size is a half of the vector's dimension.

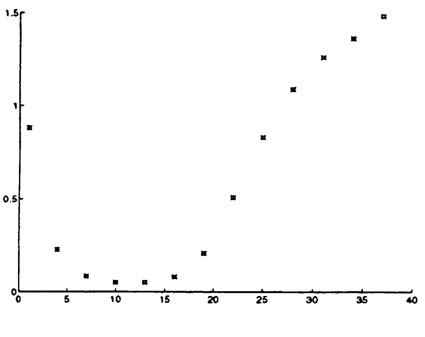


Figure 9: The angle of the weight vector and the true (30 times of renewal)

Fig. 9 shows the result after 30 times of renewal. When the number of renewal increases, the optimal block size becomes smaller than $k = N/2$. The reason is unknown and a future work, though it seems that the bias of the distribution becomes large because the area from which examples are chosen decreases.

6. Conclusion

This paper has discussed the relation of the block size and the convergence rate of the BOP algorithm for linear dichotomies from the geometrical point of view, has derived that the convergence rate is maximum when the block size is a half of the dimension of the weight vector and has confirmed the result by computer simulations.

This result is contrastive to the transversal filters' case where the convergence becomes faster as the block size increases, and it is interesting that the sign function which is a simple nonlinear function gives much influence to the convergence properties.

Finally, the result of computer simulations presents a problem why the optimal number decreases when the number of renewal increases.

References

[1] 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 (1988), 2138–2146.

[2] Haykin, S., *Adaptive Filter Theory*, Prentice-Hall, 2nd edition, 1991.

[3] Hinamoto, T. and Maekawa, S., "An Extended Learning Identification," *Trans. IEEJ. C*, Vol. 95 (1975), 227–234.

[4] Ikeda, K., "Convergence Rate Analysis and Optimization of Block Size of Block Orthogonal Projection Algorithm," *J. JSIAM*, Vol. 7 (1997), 403–413.

[5] Ikeda, K., "Convergence Rate of Block Orthogonal Projection Algorithm — Colored Input Signals' Case," *Trans. of IEICE*, submitted.

[6] Miyoshi, S., Nakayama, K. and Ikeda, K., "Geometric Learning Algorithm for Elementary Perceptron, "Convergence Condition and Noise Performance," *Tech. Rep. of IEICE*, NC 97-5, 1997.

[7] Miyoshi, S., Ikeda, K. and Nakayama, K., "Convergence Properties of Symmetric Learning Algorithm for Pattern Classification," *Trans. of IEICE*, Vol. J81-A (1998), 361–368.

[8] Miyoshi, S., Ikeda, K and Nakayama, K., "Geometric Learning Algorithm for Elementary Perceptron and Its Convergence Condition," *Trans. of IEICE*, Vol. J81-A (1998), in press.

[9] Nagumo, J. and Noda, A., "A Learning Method for System Identification," *IEEE Trans. AC*, Vol. 12 (1967), 282–287.

[10] 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 (1984), 126–132.