

反射係数変動に対してフィルタ係数補正を行う Lattice形予測NLMSアルゴリズムのブロック処理

徳井 直樹^{†*} 中山 謙二[†] 平野 晃宏[‡]

[†] 金沢大学 大学院自然科学研究科 [‡] 金沢大学工学部

〒920-8667 金沢市小立野2-40-20

TEL 076-234-4896 FAX 076-234-4900

e-mail:tokui@leo.ec.t.kanazawa-u.ac.jp

(* 石川工業高等専門学校)

Block Implementation of A Joint Lattice Predictor and NLMS Algorithm with Compensation for Reflection Coefficient Fluctuation

Naoki TOKUI^{†*} Kenji NAKAYAMA[†] Akihiro HIRANO[‡]

[†] Graduate School of Natural Science and Technology, Kanazawa University

[‡] Faculty of Engineering, Kanazawa University

2-40-20, Kodatsuno, Kanazawa, 920-8667, Japan

TEL +81-76-234-4896 FAX +81-76-234-4900

e-mail:tokui@leo.ec.t.kanazawa-u.ac.jp

(* Ishikawa College of Technology)

あらまし

Lattice形予測NLMSアルゴリズムは予測器の反射係数変動が適応フィルタの重み係数更新に影響を与えて不安定になる。筆者らはLattice形予測NLMSアルゴリズムで反射係数変動に対してフィルタ係数を補正する安定化法を提案した。しかし、補正係数法はフィルタ長が長くなると補正に必要な計算量が増大する。そこで本稿では予測器のブロック処理により収束特性が安定で計算量を軽減する方法を提案する。ブロック処理は、予測器の次数を制限する方法と、予測器の反射係数の更新のブロック化によって行われる。予測器のブロック化には2重化した予測器を用いる。シミュレーションにより提案したブロック処理が有効であることを確認した。

Abstract

A joint algorithm of orthogonalization and gradient method can improve convergence speed and residual error for colored signals. When a lattice predictor is used for the orthogonalization process, the reflection coefficients change sample by sample following the input signal. This fluctuation make the learning process unstable. This phenomenon was analyzed and the new algorithm was proposed. The filter coefficients are compensated for following the reflection coefficient fluctuation. This algorithm, however, requires $O(M^2)$ computations, where M is adaptive filter length. In this

paper, a block implementation method is proposed. First, the predictor order is limited to that of the AR model of the input signal. Second, the reflection coefficients, used in the compensation process, are updated once a block, that is the reflection coefficients are fixed within a block. Let the AR model order, the adaptive filter length and the block length be L, M and S, respectively. If $L < M < S$ can be held, the computations required in the above compensation method can be well reduced. Simulation results using speech signals show fast convergence and small residual error can be achieved by the proposed block implementation.

1 はじめに

適応フィルタのアルゴリズムとしては、少ない計算量で実現できる点から、Normalized LMS(NLMS)などの勾配法がよく用いられる。勾配法は入力信号の相関行列における固有値広がりの影響を受けて収束特性が劣化する。そのため、入力信号を直交変換で無相関化して適応フィルタに入力し、固有値広がりを低減するTwo-Stage Adaptive Filter[1]がある。Lattice形予測器を用いた直交変換形NLMSアルゴリズムでは、入力信号をARモデルとして推定し無相関化された後向き予測誤差信号が適応フィルタに入力される。しかし、予測誤差フィルタの反射係数は入力信号によって変動するため、その影響を受けて適応フィルタの重み係数更新が不安定になる。

筆者らは予測誤差フィルタの反射係数変動が適応フィ

ルタの重み係数更新に与える影響を解析した [2]. その結果, 適応フィルタの重み係数更新が予測器の反射係数更新に対して常に1サンプル遅れるために, 誤差を発生することが分かった. そこで反射係数から構成される行列を用いて重み係数更新の1サンプル遅れを補正する方法を提案した [2]. この方法では, 反射係数の変動による適応フィルタの重み係数に与える影響を完全に補正できる. しかし, 反射係数から構成される行列の計算と重み係数補正に要する計算量はフィルタ長の自乗値に比例する. 本稿では, 上記の安定化法において, 予測器のブロック処理により計算量を軽減する方法を提案する.

2 Lattice 形予測器を使った適応フィルタの補正重み係数安定化法

Lattice 形予測器における予測誤差の更新は1段あたり,

$$f_m(n) = f_{m-1}(n) + \kappa_m^*(n)b_{m-1}(n-1) \quad (1)$$

$$b_m(n) = b_{m-1}(n-1) + \kappa_m(n)f_{m-1}(n) \quad (2)$$

$$m = 1, 2, \dots, M-1$$

$$f_0(n) = b_0(n) = u(n) \quad (3)$$

である. m 段目の反射係数 κ_m は, 前段の前向き予測誤差 $f_{m-1}(n)$ と後向き予測誤差 $b_{m-1}(n-1)$ から,

$$\kappa_{N,m}(n) = \gamma \kappa_{N,m}(n-1) + b_{m-1}(n-1)f_{m-1}^*(n) \quad (4)$$

$$\kappa_{D,m}(n) = \gamma \kappa_{D,m}(n-1) + (|f_{m-1}(n)|^2 + |b_{m-1}(n-1)|^2) \quad (5)$$

$$\kappa_m(n) = -2 \frac{\kappa_{N,m}(n)}{\kappa_{D,m}(n)} \quad (6)$$

適応フィルタは, この後向き予測誤差

$$\mathbf{b}(n) = [b_0(n), \dots, b_{M-1}(n)]^t \quad (7)$$

を入力として用いる. 重み係数

$$\mathbf{w}(n) = [w_0(n), \dots, w_{M-1}(n)]^t \quad (8)$$

より, フィルタの出力 $y(n)$ は

$$y(n) = \mathbf{w}^H(n)\mathbf{b}(n) \quad (9)$$

となる. 構成図を図1に示す.

後向き予測誤差は入力信号

$$\mathbf{u}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^t \quad (10)$$

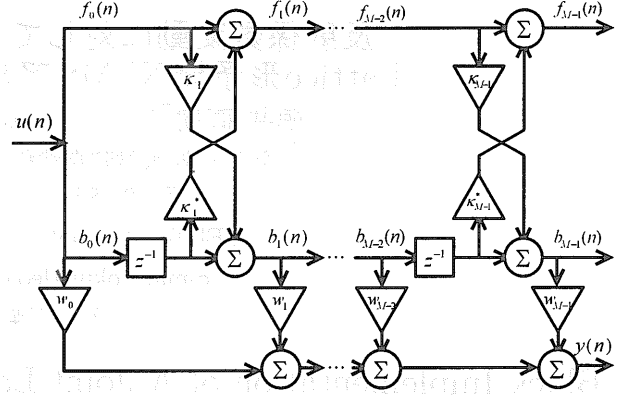


図1: Lattice 形予測器を用いた直交変換形フィルタ構成

と反射係数で構成されるため,

$$\mathbf{b}(n) = \mathbf{K}^H(n)\mathbf{u}(n) \quad (11)$$

となる. ここで $\mathbf{K}(n)$ は,

$$\mathbf{K}(n) = \begin{bmatrix} 1 & \kappa_{12}(n) & \kappa_{13}(n) & \dots & \kappa_{1M}(n) \\ 0 & 1 & \kappa_{23}(n) & \dots & \kappa_{2M}(n) \\ \vdots & \ddots & 1 & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & \kappa_{(M-1)M}(n) \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad (12)$$

$$\kappa_{12}(n) = \kappa_1(n) \quad (13)$$

$$\kappa_{13}(n) = \kappa_2(n) \quad (14)$$

$$\kappa_{23}(n) = \kappa_2(n)\kappa_1^*(n) + \kappa_1(n-1) \quad (15)$$

とあらわされる. この $\mathbf{K}(n)$ は Lattice 形予測誤差フィルタの反射係数から構成される行列で, 対角要素が1である上三角行列になっている.

式(9)と式(12)よりフィルタ出力信号 $y(n)$ は,

$$y(n) = \mathbf{w}^H(n)\mathbf{K}^H(n)\mathbf{u}(n) \quad (16)$$

とあらわされる.

図1で n サンプル時の重み係数 $\mathbf{w}(n)$ は式(16)により次のように更新される.

$$y(n-1) = \mathbf{w}^H(n-1)\mathbf{K}^H(n-1)\mathbf{u}(n-1) \quad (17)$$

$$e(n-1) = d(n-1) - y(n-1) \quad (18)$$

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \frac{\alpha}{\|\mathbf{b}(n)\|^2 + \delta} \mathbf{b}(n-1)e(n-1) \quad (19)$$

n サンプル時の重み係数 $\mathbf{w}(n)$ の計算に用いられる反射係数は $\mathbf{K}(n-1)$ である. しかし, 式(16)の $y(n)$ では重み係数 $\mathbf{w}(n)$ は $\mathbf{K}(n)$ と組み合わせられる. すなわち, $\mathbf{K}(n-1)$ に対して誤差が減少する方向に最適化された $\mathbf{w}(n)$ は次のサンプルの出力 $y(n)$ の計算では $\mathbf{K}(n)$ と組み合わせられ

るため、 $y(n)$ は誤差を減少させる方向に動いている保証はない。

そこで筆者らは、 $\mathbf{K}(n)$ に対する重み係数 $\mathbf{w}(n)$ の 1 サンプル遅れを補正する方法として、行列 $\mathbf{K}(n-1)$ と行列 $\mathbf{K}(n)$ から補正された重み係数 $\hat{\mathbf{w}}(n)$ を用いる方法を提案した [2]。式 (16) の $\mathbf{w}^H(n)\mathbf{K}^H(n)$ は伝達関数に相当するから、 $\mathbf{K}(n)$ に対する伝達関数が $\mathbf{w}^H(n)\mathbf{K}^H(n-1)$ と同じになるように重み係数を補正する。この条件はエルミート変換を用いない形で表すと式 (20) となり、 $\hat{\mathbf{w}}(n)$ が補正された重み係数である。

$$\mathbf{K}(n)\hat{\mathbf{w}}(n) = \mathbf{K}(n-1)\mathbf{w}(n) \quad (20)$$

$$\hat{\mathbf{w}}(n) = \mathbf{K}(n)^{-1}\mathbf{K}(n-1)\mathbf{w}(n) \quad (21)$$

$\hat{\mathbf{w}}(n)$ による適応フィルタ出力信号 $\hat{y}(n)$ は、

$$\hat{y}(n) = \hat{\mathbf{w}}^H(n)\mathbf{K}^H(n)\mathbf{u}(n) \quad (22)$$

よって、これより重み係数の更新が行われる。

$$\hat{\mathbf{e}}(n) = d(n) - \hat{y}(n) \quad (23)$$

$$\mathbf{w}(n+1) = \hat{\mathbf{w}}(n) + \frac{\alpha}{\|\mathbf{b}(n)\|^2 + \delta} \mathbf{b}(n)\hat{\mathbf{e}}(n) \quad (24)$$

この重み係数 $\mathbf{w}(n+1)$ は $\mathbf{K}(n)$ により計算されたものであるため、再び $\mathbf{K}(n+1)$ により補正重み係数 $\hat{\mathbf{w}}(n+1)$ を式 (21) により計算し、次の重み係数更新に利用する。

この補正重み係数法により Lattice 形予測器を用いた直交変換形 NLMS アルゴリズムを安定化させることが可能となった。しかし、補正を行わない方法に比べて、行列 $\mathbf{K}(n)$ と補正重み係数 $\hat{\mathbf{w}}(n)$ の計算にフィルタ長 M の自乗値に比例した計算量を必要とする。

3 反射係数から構成される行列と補正重み係数の計算

式 (12) の反射係数から構成される行列 $\mathbf{K}(n)$ は、

$$\mathbf{K}(n) = [\mathbf{k}_1(n), \mathbf{k}_2(n), \dots, \mathbf{k}_M(n)] \quad (25)$$

とあらわされる。そのため式 (11) より後向き予測誤差は、

$$\mathbf{b}_m(n) = \mathbf{k}_m^H(n)\mathbf{u}(n) \quad (26)$$

である。同様に前向き予測誤差は、

$$\mathbf{f}_m(n) = \mathbf{j}_m^H(n)\mathbf{u}(n) \quad (27)$$

$$\mathbf{J}(n) = [\mathbf{j}_1(n), \mathbf{j}_2(n), \dots, \mathbf{j}_M(n)] \quad (28)$$

とあらわすと、式 (1)、式 (2) と式 (3) より

$$\mathbf{j}_m(n) = \mathbf{j}_{m-1}(n) + \kappa_m^*(n)\hat{\mathbf{k}}_{m-1}(n-1) \quad (29)$$

$$\mathbf{k}_m(n) = \hat{\mathbf{k}}_{m-1}(n-1) + \kappa_m(n)\mathbf{j}_{m-1}(n) \quad (30)$$

$$\hat{\mathbf{k}}_{m-1}(n) = [0, K_{1(m-1)}(n), \dots, K_{(M-1)(m-1)}(n)]^t \quad (31)$$

$$\mathbf{k}_1(n) = \mathbf{j}_1(n) = [1, 0, \dots, 0]^t \quad (32)$$

となり、行列 $\mathbf{K}(n)$ が計算できる。

補正重み係数 $\hat{\mathbf{w}}(n)$ は式 (21) で計算される。しかし、 $\mathbf{K}(n)$ は式 (12) のように対角要素が 1 の上三角行列である。そこで式 (20) より次式のように計算できる。

$$\hat{w}_{M-1}(n) = w_{M-1}(n) \quad (33)$$

$$\hat{w}_i(n) = w_i(n) + \sum_{h=i+1}^{M-1} (w_h(n)K_{(i+1)(h+1)}(n-1) - \hat{w}_h(n)K_{(i+1)(h+1)}(n)) \quad (34)$$

$$i = M-2, \dots, 0$$

4 予測器のブロック処理方法

補正重み係数法では、反射係数により構成される行列 $\mathbf{K}(n)$ と補正重み係数 $\hat{\mathbf{w}}(n)$ の計算を必ず毎サンプル行うため演算量が多い。そこで予測器のブロック処理によって演算量の軽減する方法を提案する。

4.1 予測器の次数制限

フィルタ長 M は未知システムの特長により決定される。Lattice 形予測器は図 1 のようにフィルタ長 M の幅で用いられている。入力信号 $u(n)$ は、Lattice 形予測器によって AR モデルとして推定され、出力される後向き予測誤差 $\mathbf{b}(n)$ が無相関化する。後向き予測誤差 $\mathbf{b}(n)$ は、反射係数により 1 段毎に推定されるため、後段になると白色化される。予測誤差信号が L 段目以降に白色信号化されていると、式 (4) より、

$$\kappa_{N,m}(n) = 0 \quad m = L+1, \dots, M-1 \quad (35)$$

で、式 (6) より反射係数 $\kappa_m(n) = 0$ となる。反射係数が 0 になるため、式 (1) と式 (2) は、

$$\mathbf{f}_m(n) = \mathbf{f}_{m-1}(n) \quad (36)$$

$$\mathbf{b}_m(n) = \mathbf{b}_{m-1}(n-1) \quad (37)$$

Lattice 形予測器は図 2 で示す $L+1$ 段目以降が遅延器の構成となる。

図 2 のように Lattice 形予測器の次数を制限する場合、予測器の次数は入力信号 AR モデルの次数以上に必要がある。つまり、音声信号が AR モデルで 20 次である

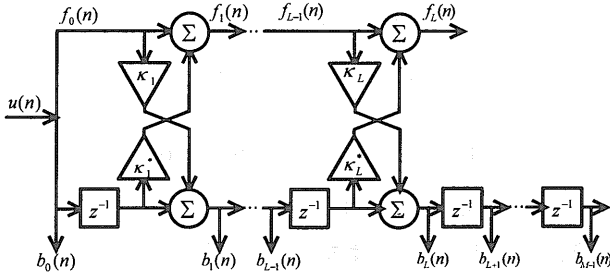


図 2: L 次に制限された M 次 Lattice 形予測器

ならば、予測器は 20 次以上である。予測器の次数を L 次にした場合、反射係数から構成される行列 $\mathbf{K}(n)$ は

$$\mathbf{K}(n) = \begin{bmatrix} 1 & \cdots & K_{1(L+1)}(n) & 0 & \cdots \\ 0 & \ddots & K_{2(L+1)}(n) & K_{1(L+1)}(n-1) & \ddots \\ \vdots & \ddots & \vdots & K_{2(L+1)}(n-1) & \ddots \\ \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & 0 & 0 & 0 & \ddots \end{bmatrix} \quad (38)$$

の帯行列になる。帯行列化された $\mathbf{K}(n)$ の要素は、式 (29) と式 (30) により $L+1$ 列まで計算され、 $L+2$ 列以降は前列の要素を遅延する。帯行列化された $\mathbf{K}(n)$ により補正重み係数 $\hat{w}(n)$ の計算は要素が 0 の項は省略できる。よって式 (34) は

$$\hat{w}_i(n) = w_i(n) + \sum_{h=i+1}^{M-1} (w_h(n)K_{(i+1)(h+1)}(n-1) - \hat{w}_h(n)K_{(i+1)(h+1)}(n)) \quad (39)$$

$$i = M-2, \dots, M-L-1$$

$$\hat{w}_i(n) = w_i(n) + \sum_{h=i+1}^{i+L} (w_h(n)K_{(i+1)(h+1)}(n-1) - \hat{w}_h(n)K_{(i+1)(h+1)}(n)) \quad (40)$$

$$i = M-L-2, \dots, 0$$

と変形できる。予測器を L 次にした計算量を表 1 に示す。

表 1: L 次予測器の計算量の比較

	L 次の予測器	補正係数法
乗算	$5M + 2ML + 9L$	$2M^2 + 14M$
加算	$4M + 2ML + 5L$	$2M^2 + 9M$
遅延	$2M + L^2 + 3L$	$M^2 + 5M$

4.2 反射係数の更新ブロック化

入力信号が音声など非定常な場合、Lattice 形予測器の反射係数 κ は後向き予測誤差 b の直交化のために変動す

る。そのため、行列 \mathbf{K} の要素が変動し、補正重み係数 \hat{w} を計算する必要がある。しかし、定常入力の場合、固定した反射係数 κ が入力信号の AR モデル係数から計算できる。すると、行列 \mathbf{K} の要素は変動しないため補正重み係数 \hat{w} は不要である。

そこで、図 3 に示すように反射係数の更新を毎サンプルではなく、間隔 S 毎に更新するブロック化を提案する。ただし、常に各段の反射係数は式 (4)、式 (5) と式 (6) にしたがって計算され、更新時刻の値が予測器に代入される。3 節で示すように、行列 \mathbf{K} の要素は反射係数 κ の変動に対応する。反射係数 κ が s サンプル固定されると、

$$\mathbf{K}(n) = [\mathbf{k}_1(n-s), \dots, \mathbf{k}_s(n), \dots, \mathbf{k}_M(n)] \quad (41)$$

のように s 列目までは要素が変動しない。このため行列 \mathbf{K} の計算量は減少する。ところが図 3 の斜線で示すフィルタ長 M の間は行列 \mathbf{K} の要素が変動するため、補正重み係数 \hat{w} の計算は必要である。反射係数の更新ブロック長 S がフィルタ長 M より長い場合、補正重み係数を計算しない区間ができる。

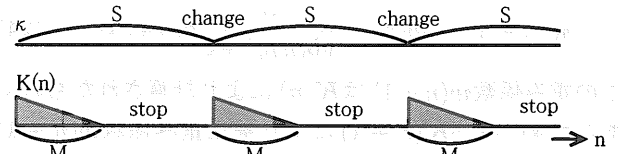


図 3: 反射係数の更新ブロック化

2 節で示すように、反射係数 κ_m は前段の前向き予測誤差 f_{m-1} と後向き予測誤差 b_{m-1} から計算される。反射係数が入力信号の AR モデルを正しく推定しない値になった場合、後向き予測誤差の直交化が悪化し、収束特性も劣化する。この直交化の悪化は後段に伝搬する。反射係数が毎サンプル更新されている場合、直交化の悪化が後段の更新に影響を与える時間は短い。しかし、反射係数の更新がブロック化されている場合、後段に影響する時間が長い。

そこで、図 4 に示すように反射係数の推定とブロック長 S 毎に反射係数を複写されて後向き予測誤差を出力する 2 重化 Lattice 形予測器構成を提案する。2 重化された予測器のため、計算量は増加する。しかし、ブロック化によって 1 サンプルあたりの計算量は減少し、反射係数の推定誤差による収束特性の劣化を低減できる。計算量を表 2 に示す。

4.3 予測器のブロック処理

4.1 節の予測器の次数の制限する方法と、4.2 節の反射係数の更新ブロック化は、それぞれ計算量の軽減に効果がある。そこで、これらを組み合わせた予測器のブロッ

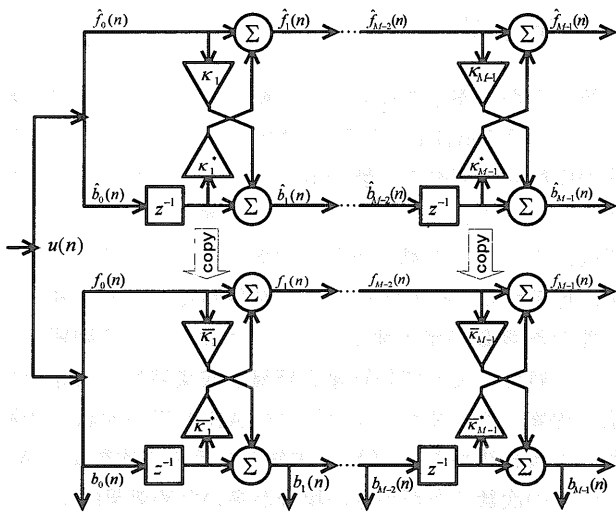


図 4: 2重化Lattice形予測器の更新ブロック化構成

表 2: 反射係数の更新ブロック化による計算量の平均値

	更新ブロック化 $S > M$	補正係数法
乗算	$16M + \left(\frac{5M^2}{3} + \frac{M}{2} - \frac{1}{6}\right) \frac{M}{S}$	$2M^2 + 14M$
加算	$11M + \left(\frac{5M^2}{3} + \frac{M}{2} - \frac{1}{6}\right) \frac{M}{S}$	$2M^2 + 9M$
遅延	$5M + (2M^2 + 2M) \frac{M}{S}$	$M^2 + 5M$

ク処理方法を提案する。この予測器のブロック処理方法によってさらに計算量の軽減が可能である。表3に計算量の平均値を示す。

表 3: 予測器のブロック処理による計算量の平均 $S > M$

乗算	$5M + 11L + \left(2ML - \frac{L^2}{3} + \frac{L}{2} - \frac{1}{6}\right) \frac{L}{S}$
加算	$4M + 7L + \left(2ML - \frac{L^2}{3} + \frac{L}{2} - \frac{1}{6}\right) \frac{L}{S}$
遅延	$3M + 2L + (L^2 + L) \frac{L}{S}$

表4に提案した予測器のブロック処理法と非制限補正法の1サンプルあたりの計算量を示す。ただし、予測器のブロック処理法は平均値である。

5 収束特性の比較

本稿で提案した予測器のブロック処理法の有効性を確認するために、入力に図5に示す日本音響学会編集の研究用連続音声データベースから作成した音声信号を用いてシミュレーションを行う。ブロック処理しない補正重み係数補正法と比較を行う。未知システムは、12次のIIR-LPFである。適応フィルタのフィルタ長を $M = 50$ とした場合、インパルス応答の推定に不足する部分のために発生する誤差は約 -60dB 以下である。適応フィルタのパラメータは、 $\alpha = 1$, $\delta = 0.001$, $\gamma = 0.999$ である。誤差量は未知システムの全出力の自乗平均値で正規化し

表 4: 計算量の比較 $M = 50$

	非制限補正法	$L = 20, S = 200$	
乗算	5700	657.65	11.54%
加算	5450	527.65	9.68%
遅延	2750	232.00	8.44%

ている。

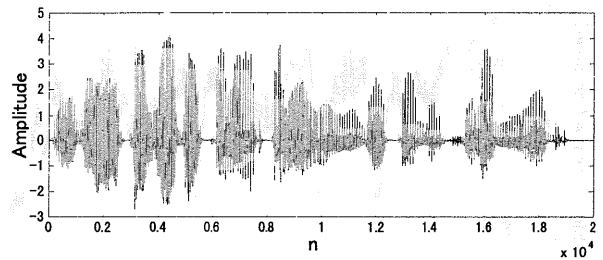


図 5: 音声信号

5.1 予測器の次数を制限する方法

4.1節の予測器の次数を制限する方法によるLattice形予測器を用いた直交変換形NLMSアルゴリズムの特性を図6に示す。図6は予測器の次数 $L = 15$, $L = 20$ と補正係数法 $L = M - 1$ を示す。どちらの次数も不安定ではないが、次数 $L = 15$ は収束に更新回数が約2500回必要である。しかし次数 $L = 20$ では補正係数法とほぼ重なっている。これより、この音声信号のARモデルは $L = 20$ 次以上必要である。

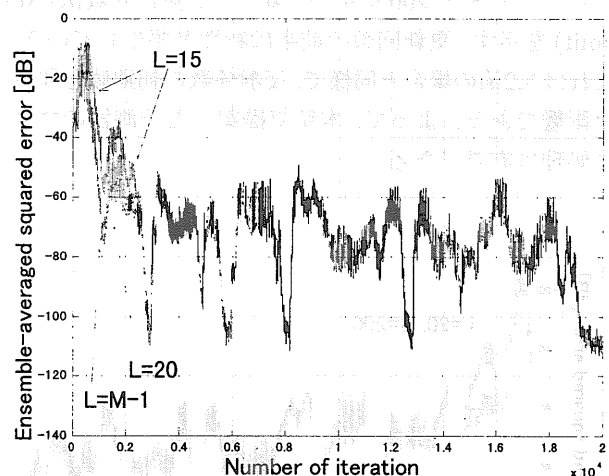


図 6: 予測器の次数を制限した収束特性

5.2 反射係数の更新ブロック化による方法

4.2節の反射係数の更新ブロック化したLattice形予測器を用いた直交変換形NLMSアルゴリズムの特性を図7に示す。図7では、更新ブロック長 $S = 200$ と補正係数

法 $S = 1$ を示す。 $S = 200$ の更新ブロック化では、更新回数が約 2500 回で収束しているが、収束特性は不安定ではない。収束が遅いのは反射係数の初期値 0 を利用した影響が残っているためである。

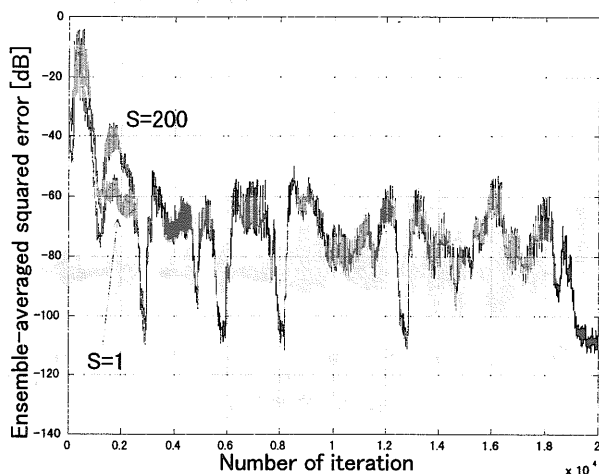


図 7: 反射係数の更新ブロック化による収束特性

5.3 予測器のブロック処理による方法

4.3 節で提案した予測器のブロック処理による Lattice 形予測器を用いた直交変換形 NLMS アルゴリズムの特性を図 8 に示す。図 8 では、予測器の次数 $L = 20$ と更新ブロック長 $S = 200$ を用いた場合と、補正係数法 (Non-limit) を示す。更新回数的前半は特性が悪化しているが、これは 5.2 節の場合と同様で、反射係数の初期値を利用した影響である。よって、本稿で提案した予測器のブロック処理は有効である。

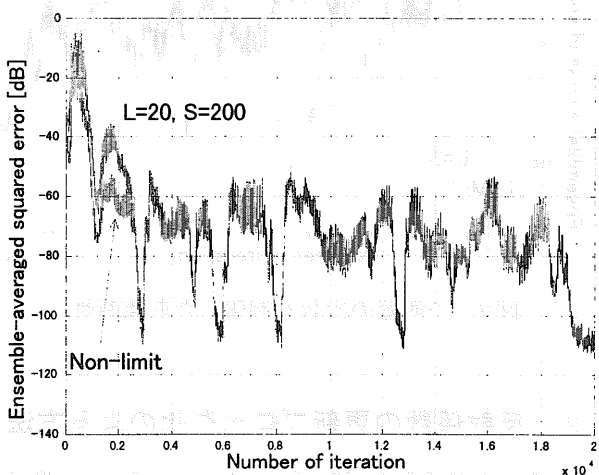


図 8: 予測器のブロック処理による収束特性

6 まとめ

筆者らが提案した反射係数変動に対してフィルタ係数補正を行う Lattice 形予測 NLMS アルゴリズムは、予測器の反射係数変動の影響を適応フィルタの補正重み係数によって完全に補正できる。しかし補正のために、反射係数による行列と補正係数の計算量はフィルタ長の 2 乗値に比例する。そのためフィルタ長が長くなると補正に必要な計算量が増大する。そこで本稿では、予測器のブロック処理によって計算量を軽減し収束特性が安定な方法を提案した。まず、入力信号の AR モデル次数で予測器の次数を決めた。一般に、未知システムの次数より AR モデルの次数の方が小さい場合が多いため効果的である。次に、2 重化予測器による反射係数の更新ブロック化を提案した。計算量はフィルタ長 $M = 50$ の非制限補正法に比べ、予測器の次数 $L = 20$ 、ブロック長 $S = 200$ では約 10% に低減できた。提案したブロック処理を用いて音声入力信号によるシミュレーションを行い、本方法の有効性を確認した。

参考文献

- [1] S. Haykin, "Adaptive Filter Theory," PRENTICE-HALL, 3rd edition, 1996.
- [2] 徳井 直樹 中山 謙二 平野 晃宏, "Lattice 形予測器を用いた直交変換形 LMS アルゴリズムの安定化法", 信学技報, DSP99-81, Sep., 1999.
- [3] J. J. Shynk, "Frequency-domain and multirate adaptive filtering," IEEE SP MAGAZINE, pp.14-37, Jan., 1992.
- [4] V. N. Parikh and A. Z. Baraniecki, "The use of the modified escalator algorithm to improve the performance of transform-domain lms adaptive filters," IEEE Trans. Signal Process., Vol. 46, No. 3, pp.625-635, Mar., 1998.
- [5] Z. Fejzo and H. Lev-Ari, "Adaptive laguerre-lattice filters," IEEE Trans. Signal Processing, Vol. 45, No. 12, pp.3006-3016, Dec., 1997.