

音響エコーキャンセラのための
低演算量適応信号処理の研究

田中 雅史

システム情報工学研究科

筑波大学

2006年7月

目次

第 1 章 序論	1
1.1 序言	1
1.2 研究の背景	2
1.2.1 エコーの発生	2
1.2.2 エコーの知覚	4
1.2.3 適応フィルタによるエコー消去	5
1.2.4 適応フィルタ以外のエコー抑圧方式	7
1.2.5 適応フィルタ長の選択	8
1.3 本研究の考え方	8
1.3.1 研究のねらい	8
1.3.2 本論文の構成	9
第 2 章 適応フィルタアルゴリズム	11
2.1 記号の定義	11
2.2 LMS 法と NLMS 法	13
2.3 AP 法	16
2.4 RLS 法	19
2.4.1 指数窓 RLS 法	20
2.4.2 方形窓 RLS 法	23
2.5 高速 RLS 法	24
2.5.1 予備	24

2.5.2	線形予測フィルタと最小二乗誤差和の再帰更新	26
2.5.3	カルマンゲインと変換係数の再帰更新	27
2.5.4	方形窓の FTF [1]	31
2.5.5	FTF の数値的安定化	32
2.6	組み合わせて用いる手法	33
2.6.1	ブロック処理に基づくもの	33
2.6.2	音響の知識に基づくもの	36
2.7	計算機シミュレーション	36
2.8	本章のまとめ	39
第 3 章	アフィン射影法の複合 NLMS 法としての解釈と収束特性解析	45
3.1	はじめに	45
3.2	準備	46
3.2.1	NLMS 法	46
3.2.2	AP 法	48
3.2.3	前向き予測とコレスキー分解	49
3.3	AP 法の複合 NLMS 法モデル	51
3.4	複合 NLMS 法モデルに基づく AP 法の収束特性	52
3.5	計算機シミュレーションによる検証	58
3.6	本章のまとめ	62
第 4 章	高速アフィン射影法	69
4.1	はじめに	69
4.2	AP 法	70
4.3	高速 AP 法	70
4.3.1	プレフィルタベクトル $\mathbf{g}(k)$ の再帰計算	71
4.3.2	$\mathbf{X}(k)$ の $\mathbf{g}(k)$ によるプレフィルタリング	73

4.4	本章の結論	77
第 5 章	逐次等価ブロック高速アフィン射影法	79
5.1	はじめに	79
5.2	準備	80
5.2.1	AP 法	80
5.2.2	高速 FIR フィルタリング, Fast FIR Filtering (FFF) 法	81
5.3	提案するブロック AP 法	84
5.3.1	出力の逐次法からの補正	84
5.3.2	補助フィルタベクトルの導入	86
5.3.3	FFF の提案法への適用	87
5.3.4	演算量と他の逐次等価ブロック法との比較	89
5.3.5	従来の AP 法との収束曲線の比較	91
5.4	ブロック長の選択	93
5.5	本章の結論	95
第 6 章	騒音制御のための高速アフィン射影法	99
6.1	はじめに	99
6.2	AP 法	100
6.2.1	考え方	100
6.2.2	AP 法の演算低減法	102
6.3	FILTERED-X 系のための射影アルゴリズム	105
6.3.1	Case 1: 一入力一出力系	105
6.3.2	Case 2: 複数入力, 複数二次音源, 複数制御点	110
6.4	計算機シミュレーション	116
6.4.1	実験条件	116
6.4.2	実験結果	116

6.5 本章のまとめ	118
第 7 章 結論	123
付 録 A NLMS 法の定常誤差 (3.11) の導出	127
謝辞	131
参考文献	133
研究業績目録	138

第1章 序論

1.1 序言

近年，拡声通話と呼ばれる通話形態の利用が広がりつつある．従来の電話では，ハンドセット（送受話器）を手を持って通話するのに対し，拡声通話系ではスピーカとマイクロホンにより通話を行う．拡声通話系には，通話をしながら両手を自由に使える，多数の人が一度に通話に参加できるという，ハンドセットによる通話にはない大きな利点がある．拡声通話系の例としては，テレビ会議システム，音声会議システム，遠隔医療や遠隔講義システムがある．10Mbps以上の映像と音声を通すに十分な容量の固定・無線通信回線の導入とともにこれらサービスも急速に普及してきており，これらの拡声通話品質の向上が急務となっている．

拡声通話系を実現する際に最も問題となるのが，音響エコーやハウリングの原因となる音響帰還である．音響エコーは自分の声が相手側のスピーカから放射され，相手側のマイクロホンに入り，自分の側に戻ってスピーカから放射され，自分の声が発声に遅れて知覚される現象である．エコーの存在下では話し続けるのが困難になり，さらに音響帰還が大きい場合には，発振，すなわち，ハウリングを引き起こし通話不能となる．

音響帰還の制御方法には様々な手法があるが，その中でも適応フィルタを用いてスピーカからマイクロホンへ至るエコーの伝達経路を同定し，そのフィルタと受話信号を畳み込むことでエコーを推定し，マイクロホン受話信号から推定エコーを引き去る方法が，最も精度がよく，また，送話音声への影響が無いことから最も有効な方法であると見なされている．

しかしながら，音響エコーを消去するための適応フィルタを良好に動作させるには幾つ

もの課題がある．音響エコー経路が長く変化しやすいこと，周囲騒音が存在すること，また，適応フィルタへの入力信号が有色で非定常な音声であるために適応フィルタの収束が遅い等がその課題である．

1.2 研究の背景

半導体デバイスとデジタル信号処理技術の進歩に伴い，従来のアナログ技術では装置規模等の問題から実現が困難であった複雑で高精度な処理が，小型の装置で実現できるようになってきている．エコーキャンセラもそれら進歩の恩恵を受け実用化されているが，まだ，前節で触れたように解決されていない課題もある．本節では，研究の背景となる，エコーの発生と知覚，それに続き，適応フィルタ等の既存のエコーキャンセル技術と適応フィルタのフィルタ長の選択について述べる．

1.2.1 エコーの発生

エコーは，やまびこやこだまのように自分の声が時間を置いて話者自身に戻り，知覚される現象である．通信におけるエコーは2線4線変換回路で生じる回線エコーと，スピーカ，マイク間の帰還による音響エコーの二つがある．ここでは，それら二つのエコーの特徴を述べ，比較する．

(a) 回線エコー

図 1.1 は電話回線の構成の概念図を示すものである．局と加入者の間は，送話信号と受話信号が重畳する2線式回路で，局と局の間は多重化とアンプを入れる必要とから送話信号と受話信号を分離する4線式回路により伝送する．この2線式回路と4線式回路を変換する2線4線変換回路は，ハイブリッドトランスからなる平衡回路網であり，平衡インピーダンスが2線側のインピーダンスと平衡していれば，4線の受話ポートの信号は2線側にだけ伝わる．実際には，呼毎に2線側の端末と回線長が変わるので，2線4線変換回路と2線側のインピーダンスの間に不整合が起こり，4線の受話信号が4線の送話信号に漏れ

る．この漏れた受話信号が，局間の伝送や多重化による遅延の後に発話者に戻るのが回線エコーである．回線エコーは電気回路によって生じるエコーであり，インパルス応答は短くまた回線の接続が確立すると不変と考えてよい．エコーとして知覚されるかどうかは，エコーのレベルと発話からエコーが戻ってくるまでの遅延時間により決まる．近年，交換機が全てデジタル化されたことからエコーのレベルは回線がアナログの時代より大きく，また，IP(Internet Protocol) 網や ATM (Asynchronous Transfer Mode) 網のような遅延の長いパケット回線が普及しつつあり，エコーが近くされやすい状況になっている．

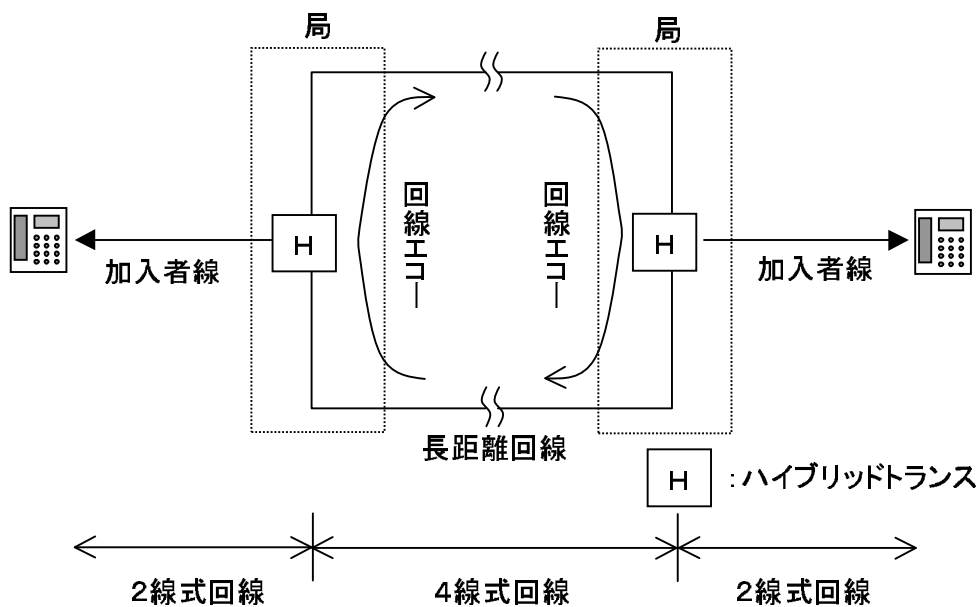


図 1.1: 電話回路構成の概念図．電話回線におけるエコーは，ハイブリッドトランスにおけるインピーダンス不整合により生じる．

(b) 音響エコー

図 1.2 は音響エコーの発生を説明するものである．図のようなマイクロホンとスピーカを使う系は，拡声通話あるいはハンズフリー通話と呼ばれ，TV会議や一部の電話でも採

用されている通話形態である。音響エコーは、スピーカから放射された音がマイクロホンに回り込む音響帰還により生じる。スピーカとマイクロホンの間の回り込みには直接音以外に室内の壁や話者で多重反射するパスがある。音響エコーの特徴は、継続時間が長く回線エコーの場合の10倍以上あることと、話者やスピーカやマイクの移動、あるいはドアの開閉等によって急激に変化することである。

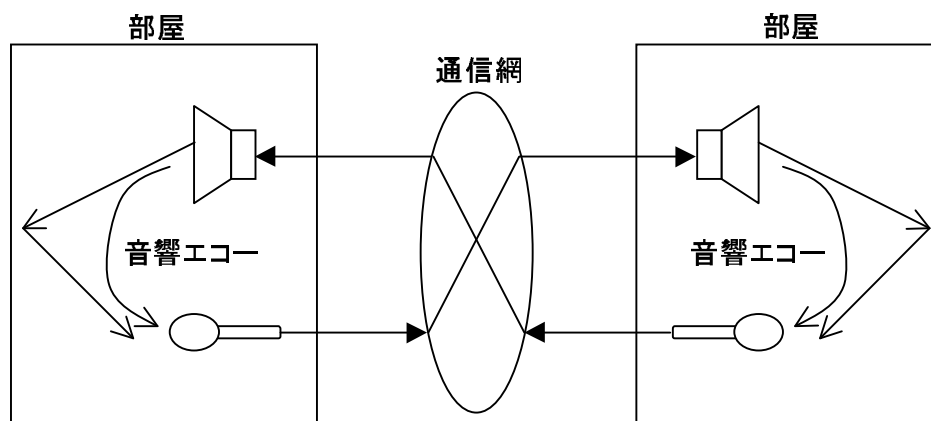


図 1.2: 拡声通話系の構成図。拡声通話系におけるエコーはスピーカからマイクロホンへの音響帰還により生じる。

1.2.2 エコーの知覚

エコーは物理的にあるだけではなく、知覚されて初めて問題となる。エコーが知覚されるかどうかは直接音からの遅延時間が特に重要である。我々は屋内で会話するときには常に直接音以外に残響、つまり、エコーを聞いているが、遅延時間が短いために気にならない。ところが、ひとたび、IP 網のように折り返しで 2, 300 ms かそれ以上になる回線遅延が入ると、エコーのために会話に困難を感じるようになる。図 1.3 は、音響エコーについて遅延時間とエコーが気にならなくなるための所要エコー減衰量の関係を実験的に求めた結果である [2]。同図から遅延時間が数 10 ms と短くてもエコーを除去する必要があり、ま

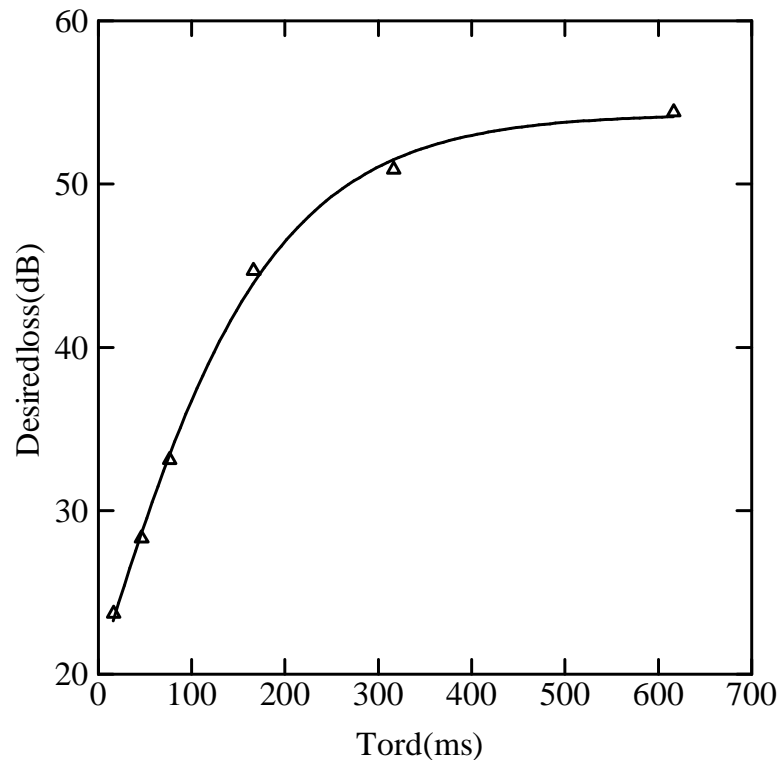


図 1.3: 遅延時間 Tord に対する所望抑圧量．遅延時間が数 10 ms と短くてもエコーを除去する必要があり，また，遅延時間とともに必要なエコー減衰量が増加する．

た，遅延時間とともに必要なエコー減衰量が増加し，300 ms では約 50 dB の抑圧量が必要となることがわかる．

1.2.3 適応フィルタによるエコー消去

図 1.4 は，音響系を例に適応フィルタによるエコー消去の原理を説明するものである．スピーカとマイクロホンの間のエコー伝達経路 $h(k)$ は FIR フィルタ $\hat{h}(k)$ によりモデル化され，エコー伝達経路の変動に追従するために，適応フィルタアルゴリズムにより更新される．適応フィルタ $\hat{h}(k)$ と受話信号 $x(k)$ を畳み込むことで推定エコー $\hat{y}(k)$ が生成される．この推定エコー $\hat{y}(k)$ をマイクロホンで受信した信号 $y(k) + n(k)$ から差し引き，誤差

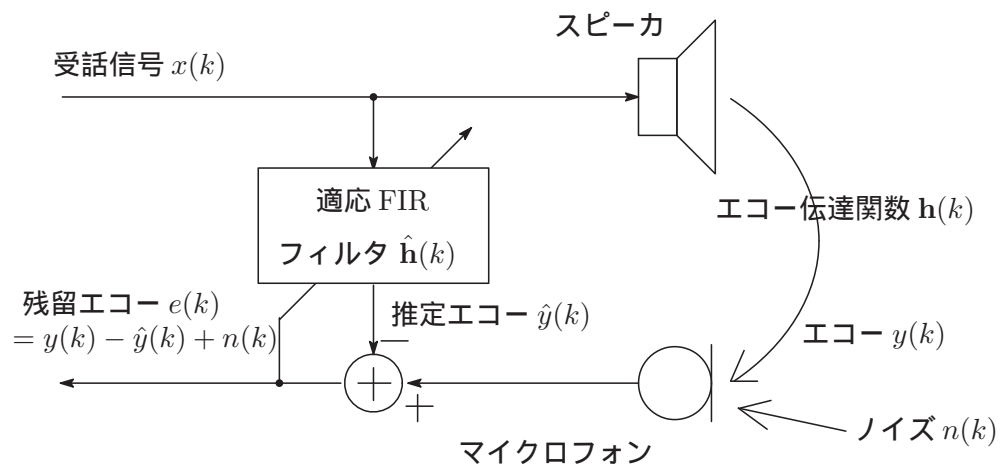


図 1.4: 適応フィルタによるエコー消去の仕組み．音響エコーキャンセラはスピーカとマイクロホンの間のエコー伝達経路 $h(k)$ を適応フィルタ $\hat{h}(k)$ により推定し，適応フィルタ $\hat{h}(k)$ と受信信号 $x(k)$ を畳み込むことで推定エコー $\hat{y}(k)$ が生成し，この推定エコー $\hat{y}(k)$ をマイクロホンで受信した信号 $y(k) + n(k)$ から差し引きことによりエコー $y(k)$ を消去する．

$e(k) = y(k) - \hat{y}(k) + n(k)$ が生成される．ただし， $n(k)$ は周囲雑音である．

適応フィルタアルゴリズムは誤差エコーのパワーが小さくなるように，フィルタ係数 \hat{h} を修正するが，近端話者の音声が入力に加わる場合（同時通話）には，ノイズが増加した場合と同様にフィルタ係数が乱れるので，修正を停止する必要がある．もし，適応フィルタがエコーパスを精度良く近似し，また，同時通話時に修正を停止できれば，同時通話時においてもエコーは消去されるが近端話者の音声は歪まず，理想的なエコー消去が実現できる．

あるエコー消去量を得るのに必要な適応フィルタの長さはエコー伝達経路の長さに比例し，適応フィルタが収束するまでの時間は適応フィルタ長に比例する．従って，長く変化しやすい音響エコー伝達経路を推定するには，収束（追従）の速い適応フィルタアルゴリズムが求められることになる．

適応フィルタアルゴリズムは様々なものが提案されているが，その中で，最も利用されているのが Least Mean Squares(LMS) 法，Normalized LMS 法（NLMS あるいは学習同

定法)である。また、この他に良く知られた手法として Recursive Least Squares(RLS)法がある。この LMS 法と RLS 法を比較すると、LMS 法は、音声などの有色の入力に対して収束が遅いという欠点があるが、演算は適応フィルタのタップ数 L の 2 倍と比較的少ない。一方、RLS 法は、有色入力に対しても白色信号入力と変わらない速い収束速度を有するが、演算は L^2 に比例し、タップ数が数百以上になる音響エコーキャンセラでは、現在のハードウェア技術をもってしても大規模な装置が必要となり、実用化は困難である。

アフィン射影 (AP) 法は、1980 年代に提案された方法 [3, 4] で、NLMS 法の自然な拡張と考えることができる。この手法には、射影次数 p という入力信号の脱相関次数に相当するパラメータがあり、 $p = 1$ で NLMS 法に、 $p = L$ で窓長 $2L + p - 1$ の方形窓 RLS と等価になる。収束速度も NLMS 法と RLS 法の間位置し、有色信号に対して NLMS 法より高速である。演算量については提案されたままの計算法では、 $pL + O(p^3)$ であり、NLMS 法より多いという欠点がある。

1.2.4 適応フィルタ以外のエコー抑圧方式

適応フィルタ以外の代表的な音響エコー抑圧方法として音声スイッチ [5, 6] と呼ばれる方法がある。ここでは、適応フィルタと音声スイッチの得失について述べる。

音声スイッチは、受話信号があるときには送話側の回線に、送話信号があるときには受話側の回線に、どちらもないときには、送受話両方か片方の回線に損失を挿入することでエコーの遮断とハウリングの防止をはかる方式である。なお、受話信号の有無は、受話信号と送話信号のパワーの比較により行われる。この音声スイッチの利点は処理が簡単でアナログ回路でも実現できるということである。他方、欠点は、受話区間、送話区間の検出誤りにより、語頭語尾が欠落し音声品質が低下する、また、同時に発話することができないので会話が不自然になるということである。

適応フィルタの場合は、音声スイッチとは逆に、エコーのみを消去するので、音声品質の劣化がなく、また、同時通話も可能であるという利点があるが、演算の多さが問題となる。実用的なエコーキャンセラ装置では相補的な二つの技術を利用するが、エコーキャン

セラ装置の性能を決めるのは、適応フィルタであり、適応フィルタアルゴリズムの改善が重要となってくる。

1.2.5 適応フィルタ長の選択

1.2.2 項で述べたように、回線遅延の長い IP 網では 50 dB ものエコー抑圧量が必要であるが、通常、適応フィルタでは演算量や記憶容量といったハードウェア上の制限、あるいは、適応フィルタ長に比例して伸びるフィルタの収束までの時間という使用上の制限の上限で可能なだけエコーを消去し、残りを音声スイッチで抑圧する。経験的には、50 dB エコーが減衰するのに 300 ms 程度要する、10 名程度収容可能な会議室を想定した場合、適応フィルタ長は最低でも 1000(8 kHz サンプリングで 125 ms, 消去量にして、約 20 dB) 程度必要である。

本節では、研究の背景として、拡声通話においてエコーが発生する原因、エコーの遅延時間と知覚の関係、適応フィルタや音声スイッチによるエコー消去・抑圧の仕組みと得失、そして、適応フィルタ長の選択について述べた。

1.3 本研究の考え方

1.3.1 研究のねらい

本研究では、従来よく用いられてきた NLMS 法に比較して音声信号に対して速くエコーを消去することが可能なアフィン射影法について、主に以下の三つの点からの検討を加え、このアルゴリズムの有用性を拡大することで、拡声通話系の品質を向上することを目的としている。

- 最初に、アフィン射影法の収束特性の解析を行う。アフィン射影法を複数の学習同定法とみなし、それら複数の学習同定法の収束特性でアフィン射影法の収束特性を記

述する方法によりアフィン射影法の収束特性を理解するところを試みる。

- 次に、アフィン射影法の演算量の低減を行う。従来、アフィン射影法の演算量は良く知られた学習同定法に比べ非常に大きいと考えられていたが、学習同定法と同程度の演算量で計算できることを示す。また、“exact block”処理を導入することで、推定誤差出力の遅延以外の収束特性の劣化無しに、さらに演算量を低減できることを示す。
- 最後に、アフィン射影法を空間的なノイズ抑圧や音場再生に用いられる filtered-x というシステムに応用することを考える。filtered-x システムでは、収束が早く演算量の少ない手法がこれまで無かった。提案する filtered-x システム上のアフィン射影法は従来法から若干の演算量の増加だけで高速な収束を実現する。

1.3.2 本論文の構成

本論文の構成は、次章でアフィン射影法とそれに関連のある適応フィルタアルゴリズムの概略を説明する。つづく、3章では収束特性の解析について論じる。4、5章では、演算量の低減手法について、6章では filtered-x 系への応用について、最後に、7章でまとめる。

第2章 適応フィルタアルゴリズム

本章では、次章以降で述べるアフィン射影 (AP) 法の演算量低減化の予備知識として、AP 法と代表的な適応フィルタアルゴリズムである Least-Mean-Square (LMS) 法、Recursive Least-Squares (RLS) 法とその派生法の導出と、その他の演算量を低減する手法や収束を速くする手法を概観する。加えて、NLMS 法、AP 法と RLS 法について音声信号等に対する計算機シミュレーションを行い収束特性の比較を行う。

2.1 記号の定義

以下では、イタリック体はスカラーを、ボールド体の小文字はベクトルを、ボールド体の大文字は行列を表すこととする。図 2.1 に適応フィルタのブロック図を示す。未知系は FIR フィルタでモデル化できると仮定し、その係数をベクトルで

$$\mathbf{h} = [h_1, h_2, \dots, h_L]^T \quad (2.1)$$

と表わす。ここで、 L は係数の個数を、 T は転置を表わす。未知系は FIR フィルタとして推定され、その離散時刻 k での係数を

$$\hat{\mathbf{h}}(k) = [\hat{h}_1, \hat{h}_2(k), \dots, \hat{h}_L(k)]^T \quad (2.2)$$

とベクトル表示する。図 2.1 の $x(k)$, $y(k)$, $\hat{y}(k)$, $n(k)$ そして $e(k)$ は、それぞれ、未知系への入力、目的信号、推定出力、雑音そして推定誤差を表わす。推定出力 $\hat{y}(k)$ は

$$\hat{y}(k) = \mathbf{x}^T(k) \hat{\mathbf{h}}(k-1) \quad (2.3)$$

と計算される．ここで， $\mathbf{x}(k)$ は

$$\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-L+1)]^T \quad (2.4)$$

である．また，推定誤差は，

$$e(k) = y(k) - \hat{y}(k) \quad (2.5)$$

である．同図で推定フィルタ $\hat{\mathbf{h}}(k)$ は推定出力 $\hat{y}(k)$ が目的信号 $y(k)$ に近づくように，修正ベクトル $\Delta\hat{\mathbf{h}}(k)$ を加え修正する．

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu(k)\Delta\hat{\mathbf{h}}(k) \quad (2.6)$$

ここで，修正の大きさを決める $\mu(k)$ はステップサイズと呼ばれ，収束速度と残留誤差の量を調節する．ステップサイズは $\mu(k)$ は時変として扱うが，実際の使用では一定値とすることがある．

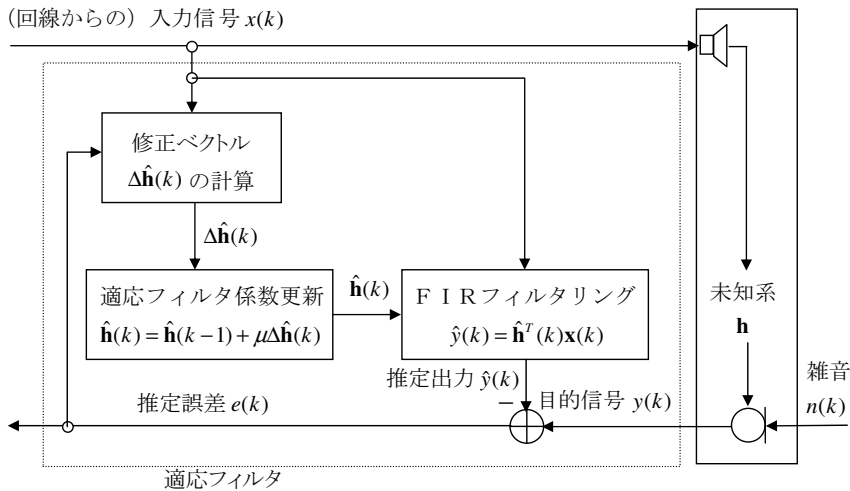


図 2.1: 適応フィルタの一構成．時刻: k ，未知系と適応フィルタへの入力信号: $x(k)$ ，適応フィルタ: $\hat{\mathbf{h}}(k)$ ，修正ベクトル: $\Delta\hat{\mathbf{h}}(k)$ ，ステップサイズ: μ ，推定出力 $\hat{y}(k)$ ，雑音: $n(k)$ ，目的信号: $y(k)$ ，推定誤差: $e(k)$ ，推定誤差のレベルが小さくなるように適応フィルタが修正される．

2.2 LMS 法と NLMS 法

推定誤差は適応フィルタ係数 $\hat{\mathbf{h}}$ に依存するので，推定誤差の二乗平均値を適応フィルタ係数の関数として次式のように $J(\hat{\mathbf{h}})$ と表わす．

$$J(\hat{\mathbf{h}}) = E [e^2(k)] \quad (2.7)$$

ここで， $E[\cdot]$ は期待値を取る演算である．LMS 法は，この二乗誤差の期待値 $J(\hat{\mathbf{h}})$ が最小になるように， $J(\hat{\mathbf{h}})$ の勾配に沿って推定フィルタ係数を修正する．(2.7) に (2.3) と (2.5) を代入し展開すると，

$$J(\hat{\mathbf{h}}) = E [y^2] - \hat{\mathbf{h}}^T \mathbf{q} - \mathbf{q}^T \hat{\mathbf{h}} + \hat{\mathbf{h}}^T \mathbf{R} \hat{\mathbf{h}} \quad (2.8)$$

となる．ただし，

$$\mathbf{R} = E [\mathbf{x}\mathbf{x}^T] \quad (2.9)$$

$$\mathbf{q} = E [y\mathbf{x}] \quad (2.10)$$

(2.8) を推定フィルタ $\hat{\mathbf{h}}$ に関して微分すると，

$$\frac{1}{2} \frac{\partial J(\hat{\mathbf{h}})}{\partial \hat{\mathbf{h}}} = -\mathbf{q} + \mathbf{R} \hat{\mathbf{h}} \quad (2.11)$$

となる．この微係数の符号を反転したものが最急勾配方向である．(2.8) の J は $\hat{\mathbf{h}}$ の二次形式であるから， J の唯一の極小点は最小点でもある．したがって， $\hat{\mathbf{h}}(k)$ 近傍で最急勾配方向に，十分小さく推定フィルタを修正すれば J は減少する．(2.11) では，統計量 \mathbf{q} と \mathbf{R} が必要であるが，LMS 法では，(2.9) と (2.10) の平均操作を外した瞬時値を用いて，推定フィルタを次式により修正する．

$$\begin{aligned} \hat{\mathbf{h}}(k) &= \hat{\mathbf{h}}(k-1) + \mu_{\text{LMS}}(y(k)\mathbf{x}(k) - \mathbf{x}(k)\mathbf{x}^T(k)\hat{\mathbf{h}}(k-1)) \\ &= \hat{\mathbf{h}}(k-1) + \mu_{\text{LMS}}\mathbf{x}(k)e(k) \end{aligned} \quad (2.12)$$

以上，LMS 法をまとめる．ただし，演算量は乗算，加減算と積和演算の回数の和である．

演算	式番号	演算量
1) $\hat{y}(k) = \mathbf{x}^T(k)\hat{\mathbf{h}}(k-1)$	(2.3)	L
2) $e(k) = y(k) - \hat{y}(k)$	(2.5)	1
3) $\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu_{\text{LMS}}\mathbf{x}(k)e(k)$	(2.12)	L

アルゴリズム 2.1: LMS 法

LMS 法により適応フィルタ係数の期待値が収束する，ステップサイズ μ_{LMS} の必要十分条件は，

$$0 < \mu_{\text{LMS}} < 2/\lambda_{\max} \quad (2.13)$$

である．ただし， λ_{\max} は相関行列 \mathbf{R} の最大固有値である．このことは，以下のようにして分かる．推定フィルタと未知系の真の応答の差を

$$\boldsymbol{\epsilon}(k) = \hat{\mathbf{h}}(k) - \mathbf{h} \quad (2.14)$$

と表わす．雑音 n を無視し，

$$y(k) = \mathbf{x}^T(k)\mathbf{h} \quad (2.15)$$

とすると，(2.12) より、

$$\boldsymbol{\epsilon}(k) = (\mathbf{I} - \mu_{\text{LMS}}\mathbf{x}(k)\mathbf{x}^T(k))\boldsymbol{\epsilon}(k-1) \quad (2.16)$$

推定フィルタと真の応答の差の時間更新式が得られる．この (2.16) 両辺の期待値を取ると，

$$E[\boldsymbol{\epsilon}(k)] = E[(\mathbf{I} - \mu_{\text{LMS}}\mathbf{x}(k)\mathbf{x}^T(k))\boldsymbol{\epsilon}(k-1)] \quad (2.17)$$

となる．さらに， $\boldsymbol{\epsilon}(k-1)$ と $\mathbf{x}(k)$ が統計的に独立であると仮定すると (2.16) は，

$$E[\boldsymbol{\epsilon}(k)] = E[\mathbf{I} - \mu_{\text{LMS}}\mathbf{R}]E[\boldsymbol{\epsilon}(k-1)] \quad (2.18)$$

となる．従って， $E[\boldsymbol{\epsilon}(k)]$ が 0 に収束するためには， $\mathbf{I} - \mu_{\text{LMS}}\mathbf{R}$ の全ての固有値の絶対値が 1 未満である必要がある．すなわち， λ_i を \mathbf{R} の固有値とすれば，

$$|1 - \mu_{\text{LMS}}\lambda_i| < 1, \quad i = 1, 2, \dots, L \quad (2.19)$$

\mathbf{R} が相関行列であるから固有値は全て 0 以上である．また，それより，最大固有値 λ_{max} について成立すればそれ以下の固有値についても上式が成立することから (2.13) が導かれる．また，逆に，(2.13) が成立すれば， $E[\epsilon(k)]$ は 0 に収束する．

このように，LMS 法の場合，収束範囲が入力信号の固有値（パワー）に依存するので，非定常な入力信号を扱う場合にステップサイズを固定にすると，入力パワーの大きなどこでしかフィルタの学習が進まないという問題がある．NLMS 法は，この問題が起きないようにステップサイズを

$$\mu_{LMS} = \mu_{NLMS} / (\|\mathbf{x}(k)\|^2 + \delta) \quad (2.20)$$

と置き換えたものとして説明される．ここで， δ は正則化のための小さい正数値である．この場合，(2.12) は，

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu_{NLMS} \mathbf{x}(k) e(k) / (\|\mathbf{x}(k)\|^2 + \delta) \quad (2.21)$$

であり，ステップサイズが

$$0 < \mu_{NLMS} < 2 \quad (2.22)$$

であることが，適応フィルタの係数の誤差の平均値が 0 に収束し，誤差の二乗平均値も有限の値に収束することの必要十分条件である [7]．下表に NLMS 法をまとめる．

	演算	式番号	演算量
1)	$\hat{y}(k) = \mathbf{x}^T(k) \hat{\mathbf{h}}(k-1)$	(2.3)	L
2)	$e(k) = y(k) - \hat{y}(k)$	(2.5)	1
3)	$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu_{NLMS} \mathbf{x}(k) e(k) / (\ \mathbf{x}(k)\ ^2 + \delta)$	(2.21)	L

アルゴリズム 2.2: NLMS 法

LMS と NLMS 法の演算量（乗算，加減算，積和の総数）は，ともに誤差 e の計算に L ，推定フィルタ $\hat{\mathbf{h}}$ の更新に L の合計 $2L$ であり，演算も非常に簡単である．しかしながら，有色，つまり，自己相関のある入力信号に対しては，フィルタの収束が遅いという欠点がある．これは，修正ベクトル $\Delta \hat{\mathbf{h}}(k)$ が (2.12) と (2.21) の右辺にあるように入力信号 $\mathbf{x}(k)$ に

単に比例しており、また、自己相関の高い信号では $\mathbf{x}(k)$ の方向が時刻とともに大きく変化しないことから、適応フィルタ係数の更新回数を増やしても $\mathbf{x}(k)$ 以外の方向への学習が進まないためである。

2.3 AP 法

AP 法 (AP Algorithm, APA と略すこともある) は、学習同定法あるいは NLMS 法を一般化した手法として提案された [3, 4]。彼らの導出では、以下の連立方程式の最小二乗ノルム解という見地から一般化とアルゴリズムの導出を行っている。

$$y(k-i) = [\hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k)]^T \mathbf{x}(k-i), \quad i = 0, 1, \dots, p-1 \quad (2.23)$$

ここで、方程式の数 p は射影次数と呼ばれる ($p = 1$ のとき NLMS に等しい。) この AP 法の導出のアプローチは文献 [8, 9, 10] と次章以降での説明に委ねることとし、ここでは、LMS 法と同様に確率的最急降下法の見地から導出することにする。AP 法における最小とすべき評価量は

$$J_{APA}(\hat{\mathbf{h}}) = E [\mathbf{e}_p^T(k) \mathbf{R}_p^{-1}(k) \mathbf{e}_p(k)] \quad (2.24)$$

であると考えることができる。ただし、

$$\mathbf{e}_p(k) = \mathbf{y}_p(k) - \mathbf{X}_{L,p}^T(k) \hat{\mathbf{h}}(k-1) \quad (2.25)$$

$$\mathbf{y}_p(k) = [y(k), y(k-1), \dots, y(k-p+1)]^T \quad (2.26)$$

$$\mathbf{X}_{L,p}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-p+1)] \quad (2.27)$$

$$\mathbf{R}_p(k) = \mathbf{X}_{L,p}^T(k) \mathbf{X}_{L,p}(k) \quad (2.28)$$

である。(2.25) の $\mathbf{e}_p(k)$ は誤差ベクトルと呼ぶことにする。戻って、(2.24) の右辺の $\mathbf{R}_p^{-1}(k)$ なる重みは、入力信号 x のパワーによる正規化の作用に加え、入力信号 x の自己相関に由来する推定誤差 e の自己相関の除去の役目を果たしている。

(2.24) の右辺に (2.25) を代入し展開すると ,

$$J_{APA}(\hat{\mathbf{h}}) = E \left[\mathbf{y}\mathbf{R}^{-1}\mathbf{y} - \mathbf{y}\mathbf{R}^{-1}\mathbf{x}\hat{\mathbf{h}} - \hat{\mathbf{h}}\mathbf{x}\mathbf{R}^{-1}\mathbf{y} + \hat{\mathbf{h}}^T\mathbf{x}\mathbf{R}^{-1}\mathbf{x}^T\hat{\mathbf{h}} \right] \quad (2.29)$$

となる ((k), ($k-1$) は略した .) これを $\hat{\mathbf{h}}$ で偏微分し , 整理すると ,

$$\frac{1}{2} \frac{\partial J_{APA}(\hat{\mathbf{h}})}{\partial \hat{\mathbf{h}}} = E \left[-\mathbf{X}\mathbf{R}^{-1}(\mathbf{y} - \mathbf{X}^T\hat{\mathbf{h}}) \right] \quad (2.30)$$

となる . NLMS と同様に , 期待値操作を外し , 符号を逆にすると AP 法の修正量が得られる .

$$\Delta\hat{\mathbf{h}}(k) = \mathbf{X}_{L,p}(k)\mathbf{g}(k) \quad (2.31)$$

$$\mathbf{g}(k) = \mathbf{R}_p^{-1}(k)\mathbf{e}_p(k) \quad (2.32)$$

これより , フィルタの更新式は ,

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu(k)\mathbf{X}_{L,p}(k)\mathbf{g}(k) \quad (2.33)$$

となる . (2.31) の $\mathbf{g}_p(k)$ は , $\mathbf{X}(k)$ の行ベクトルをフィルタリングすることで修正ベクトル $\Delta\hat{\mathbf{h}}(k)$ を生成するので , プレフィルタベクトルと呼ぶことにする . (2.32) によりプレフィルタベクトルを求める際 , $\mathbf{R}(k)$ の条件数が大きく特異に近いことがあるので , (2.32) での逆行列を計算では , 次式のように , 正則化のための正定値行列 (通常 , $\delta\mathbf{I}$) を $\mathbf{R}(k)$ に加えたのち逆行列を計算することが行われる .

$$\mathbf{g}(k) = (\mathbf{R}_p(k) + \delta\mathbf{I})^{-1}\mathbf{e}_p(k) \quad (2.34)$$

ところで , $\Delta\hat{\mathbf{h}}(k-1)$ は時刻 $k-1$ で

$$y(k-j) - \mathbf{x}^T(k-j) \left[\hat{\mathbf{h}}(k-2) + \Delta\hat{\mathbf{h}}(k-1) \right] = 0, \quad j = 1, 2, \dots, p \quad (2.35)$$

を満たすので , 誤差ベクトル $\mathbf{e}_p(k)$ の $j+1$ 番目 ($j = 1, 2, \dots, p-1$) の要素は次式で更新

できる .

$$\begin{aligned}
e_{j+1}(k) &= y(k-j) - \mathbf{x}^T(k-j)\hat{\mathbf{h}}(k-1) \\
&= y(k-j) - \mathbf{x}^T(k-j)(\hat{\mathbf{h}}(k-2) + \mu(k-1)\Delta\hat{\mathbf{h}}(k-1)) \\
&= \mu(k-1) \left[y(k-j) - \mathbf{x}^T(k-j)(\hat{\mathbf{h}}(k-2) + \Delta\hat{\mathbf{h}}(k-1)) \right] \\
&\quad + (1 - \mu(k-1))(y(k-j) - \mathbf{x}^T(k-j)\hat{\mathbf{h}}(k-2)) \\
&= (1 - \mu(k-1))e_j(k-1).
\end{aligned} \tag{2.36}$$

この更新式をベクトル表現すると

$$\mathbf{e}_p(k) = [e(k), (1 - \mu(k-1))\mathbf{e}_{p-1}^T(k-1)]^T \tag{2.37}$$

となる . ここで , $\mathbf{e}_{p-1}(k-1)$ は $\mathbf{e}(k-1)$ の最初の $p-1$ 個の要素からなるベクトルである .
すなわち ,

$$\mathbf{e}_{p-1}(k-1) = [e_1(k-1), e_2(k-1), \dots, e_{p-1}(k-1)]^T. \tag{2.38}$$

(2.3) と (2.37) によれば , 誤差ベクトルは p 個の積和演算で更新できる [11] .

以上 , AP 法をまとめると ,

	演算	式番号	演算量
1)	$\hat{y}(k) = \mathbf{x}^T(k)\hat{\mathbf{h}}(k-1)$	(2.3)	L
2)	$\mathbf{e}(k) = [y(k) - \hat{y}(k), 1 - \mu(k-1)\mathbf{e}_{p-1}^T(k-1)]^T$	(2.37)	p
3)	$\mathbf{g}(k) = (\mathbf{R}_p(k) + \delta\mathbf{I})^{-1}\mathbf{e}_p(k)$	(2.34)	$O(p^3)$
4)	$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu(k)\mathbf{X}_{L,p}(k)\mathbf{g}(k)$	(2.33)	pL

アルゴリズム 2.3: AP 法

NLMS と同様にアルゴリズムが安定であるための必要十分条件は , $0 < \mu < 2$ である .

AP 法の演算量は , 推定値の計算に , L , プレフィルタベクトルの計算に $O(p^3)$, プレフィルタリングに , pL となり , フィルタ長 L と射影次数 p の積に比例する . 収束速度は , 射影次数 p を増加させると速くなり , 有色の入力信号に対しては LMS に比べて高速である .

2.4 RLS 法

最小二乗法とその再帰解法である RLS 法の評価関数は、

$$J_{LS}(k) = \sum_{i=1}^k w(k, i) |e(k, i)|^2 = \mathbf{e}_k^T(k, k) \mathbf{W}(k) \mathbf{e}_k(k, k) \quad (2.39)$$

である。ただし、時刻 k はフィルタ長 L 以上であると仮定する。(2.39) の推定誤差ベクトル $\mathbf{e}_k(k)$ は、NLMS や AP 法のとおり異なり、次式に示すように事後誤差、つまり、時刻 k でのフィルタ修正をした後のフィルタを用いた推定誤差である。

$$e(k, i) = y(i) - \mathbf{x}^T(i) \hat{\mathbf{h}}(k) \quad (2.40)$$

$$\mathbf{e}_k(k, k) = \mathbf{y}_k(k) - \mathbf{X}_{L,k}^T \hat{\mathbf{h}}(k) \quad (2.41)$$

$$\mathbf{X}_{L,k}(k) = [\mathbf{x}_L(k), \mathbf{x}_L(k-1), \dots, \mathbf{x}_L(k-k+1)] \quad (2.42)$$

$$\mathbf{x}_L(k) = [x(k), x(k-1), \dots, x(k-L+1)]^T \quad (2.43)$$

(2.39) 中の w は重み係数または忘却係数と呼ばれ、未知系の変化に適応フィルタが追従するために、過去のデータに窓かけして、忘却していくためのものである。よって、その値は正で、通常、最大値は 1 である。忘却係数の与え方の代表的なものとして以下の二つがある。

- 指数窓

最も一般的に用いられる時間とともに指数減衰する窓である。

$$w(k, i) = \lambda^{k-i}, \quad 0 < \lambda < 1, \quad i = 1, 2, \dots, k \quad (2.44)$$

- 方形窓

誤差評価する時間区間長が一定で、その区間内で忘却係数が 1、区間外で 0 のものである。AP 法の高速度算法でこの窓を用いた RLS 法が利用される。

(2.39) の J_{LS} の $\hat{\mathbf{h}}$ に関する微分を 0 と置くと, J_{LS} の最小値を与える解 (LS 解) $\hat{\mathbf{h}}_{LS}$ が得られる.

$$\hat{\mathbf{h}}_{LS} = \mathbf{R}^{-1}(k)\mathbf{q}(k) \quad (2.45)$$

ただし,

$$\mathbf{R}(k) = \mathbf{X}_{L,k}(k)\mathbf{W}(k)\mathbf{X}_{L,k}^T(k) \quad (2.46)$$

$$\mathbf{q}(k) = \mathbf{X}_{L,k}(k)\mathbf{W}(k)\mathbf{y}_k(k) \quad (2.47)$$

ここで, $\mathbf{R}(k)$ は自己相関行列, $\mathbf{q}(k)$ は相互相関ベクトルである. (2.45) は逆行列の計算があるので, $O(L^3)$ の演算が必要となる. 音響系のように未知系が絶えず変化する場合には $\hat{\mathbf{h}}_{LS}$ を絶えず計算する必要があるので, この演算量の多さは問題である. RLS 法は, 以下に示すように, 自己相関行列 $\mathbf{R}(k)$ と相互相関ベクトル $\mathbf{q}(k)$ が逐次的に更新できることと, 逆行列の補題を利用して演算量を $O(L^2)$ としたアルゴリズムである.

2.4.1 指数窓 RLS 法

指数窓を用いた RLS 法の導出を行う. まず, (2.46) と (2.47) の定義からわかるように自己相関行列 $\mathbf{R}(k)$ と相互相関 $\mathbf{q}(k)$ ベクトルは, それぞれ

$$\mathbf{R}(k) = \lambda\mathbf{R}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) \quad (2.48)$$

$$\mathbf{q}(k) = \lambda\mathbf{q}(k-1) + \mathbf{x}(k)y(k) \quad (2.49)$$

と更新することができる. (2.48) に以下に述べる逆行列の補題を適用することで相関行列の逆行列 \mathbf{R}^{-1} の時間更新式を導出し, 次に, その時間更新式と相互相関の時間更新式 (2.49) からフィルタ係数の時間更新式を導出する.

逆行列の補題によると, \mathbf{C} を M 行 N 列の行列, \mathbf{D} を N 行 N 列の正定値行列として, L 行 L 列の正定値行列 \mathbf{A} と \mathbf{B} に

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T \quad (2.50)$$

なる関係があるとするとき、 \mathbf{A} の逆行列が

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{BC}(\mathbf{D} + \mathbf{C}^T\mathbf{BC})^{-1}\mathbf{C}^T\mathbf{B} \quad (2.51)$$

と表わすことができる。(2.50)において、 $\mathbf{A} = \mathbf{R}(k)$, $\mathbf{B}^{-1} = \lambda\mathbf{R}(k-1)$, $\mathbf{C} = \mathbf{x}(k)$, $\mathbf{D} = 1$ と置き換えると、(2.48)に逆行列の補題が適用できる。 $\mathbf{R}(k)$ が正定値であると仮定すると、逆行列の補題より、 $\mathbf{R}(k)$ の逆行列の時間更新式が、

$$\mathbf{R}^{-1}(k) = \lambda^{-1}\mathbf{R}^{-1}(k-1) - \frac{\lambda^{-2}\mathbf{R}^{-1}(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{R}^{-1}(k-1)}{1 + \lambda^{-1}\mathbf{x}^T(k)\mathbf{R}^{-1}(k-1)\mathbf{x}(k)} \quad (2.52)$$

と得られる。初期条件として、 $\mathbf{R}(0)$ が正則であることが必要であるから、

$$\mathbf{R}^{-1}(0) = \delta^{-1}\mathbf{I} \quad (2.53)$$

とする。さらに、事前カルマンゲインベクトル $\tilde{\mathbf{k}}$ 、変換係数 γ 、事後カルマンゲインベクトル \mathbf{k} を、それぞれ、

$$\tilde{\mathbf{k}}(k) = \lambda^{-1}\mathbf{R}^{-1}(k-1)\mathbf{x}(k) \quad (2.54)$$

$$\begin{aligned} \gamma(k) &= 1/\{1 + \lambda^{-1}\mathbf{x}^T(k)\mathbf{R}^{-1}(k)\mathbf{x}(k)\} \\ &= 1/\{1 + \mathbf{x}^T(k)\tilde{\mathbf{k}}(k)\} \end{aligned} \quad (2.55)$$

$$\mathbf{k}(k) = \gamma(k)\tilde{\mathbf{k}}(k) \quad (2.56)$$

と定義すると、(2.52) は、

$$\mathbf{R}^{-1}(k) = \lambda^{-1}\mathbf{R}^{-1}(k-1) - \gamma(k)\tilde{\mathbf{k}}(k)\tilde{\mathbf{k}}^T(k) \quad (2.57)$$

と書き表される。

つぎに，フィルタ係数の更新を行う．(2.45) に (2.49) と (2.57) を代入すると，

$$\begin{aligned}
\hat{\mathbf{h}}_{LS}(k) &= \left[\lambda^{-1} \mathbf{R}^{-1}(k-1) - \mathbf{k}(k) \tilde{\mathbf{k}}^T(k) \right] [\lambda \mathbf{q}(k-1) + \mathbf{x}(k)y(k)] \\
&= \hat{\mathbf{h}}_{LS}(k-1) - \lambda \mathbf{k}(k) \tilde{\mathbf{k}}^T(k) \mathbf{q}(k-1) + \\
&\quad \lambda^{-1} \mathbf{R}^{-1}(k-1) \mathbf{x}(k)y(k) - \mathbf{k}(k) \tilde{\mathbf{k}}^T(k) \mathbf{x}(k)y(k) \\
&= \hat{\mathbf{h}}_{LS}(k-1) - \mathbf{k}(k) \mathbf{x}^T(k) \hat{\mathbf{h}}_{LS}(k-1) + \\
&\quad y(k) \tilde{\mathbf{k}}(k) - \{\gamma^{-1}(k) - 1\} y(k) \mathbf{k}(k) \\
&= \hat{\mathbf{h}}_{LS}(k-1) - \hat{y}(k) \mathbf{k}(k) + y(k) \gamma^{-1}(k) \mathbf{k}(k) - \{\gamma^{-1}(k) - 1\} y(k) \mathbf{k}(k) \\
&= \hat{\mathbf{h}}_{LS}(k-1) + e(k) \mathbf{k}(k) \\
&= \hat{\mathbf{h}}_{LS}(k-1) + \gamma(k) e(k) \tilde{\mathbf{k}}(k)
\end{aligned} \tag{2.58}$$

となる．ただし， $e(k)$ は事前誤差であり，次式で定義される．

$$e(k) = y(k) - \mathbf{x}^T(k) \hat{\mathbf{h}}_{LS}(k-1) \tag{2.59}$$

以上，RLS 法の手順を下表にまとめる．

	演算	式番号	演算量
初期化	$\mathbf{R}^{-1}(0) = \delta^{-1} \mathbf{I}$	(2.53)	
1)	$\tilde{\mathbf{k}}(k) = \lambda^{-1} \mathbf{R}^{-1}(k-1) \mathbf{x}(k)$	(2.54)	L^2
2)	$\gamma(k) = 1 / \{1 + \mathbf{x}^T(k) \tilde{\mathbf{k}}(k)\}$	(2.55)	L
3)	$\mathbf{R}^{-1}(k) = \lambda^{-1} \mathbf{R}^{-1}(k-1) - \gamma(k) \tilde{\mathbf{k}}(k) \tilde{\mathbf{k}}^T(k)$	(2.57)	$L^2/2$
4)	$e(k) = y(k) - \mathbf{x}^T(k) \hat{\mathbf{h}}_{LS}(k-1)$	(2.59)	L
5)	$\hat{\mathbf{h}}_{LS}(k) = \hat{\mathbf{h}}_{LS}(k-1) + \gamma(k) e(k) \tilde{\mathbf{k}}(k)$	(2.58)	L

アルゴリズム 2.4: RLS 法

RLS 法の演算量は，事前カルマンゲインベクトルの計算に L^2 ，相関行列の逆行列の更新に $L^2/2$ ，その他，誤差の計算，フィルタ更新等に， $4L$ であり，ほぼ，フィルタ長 L の 2 乗に比例し，LMS 法や AP 法より多い．収束速度は，入力信号の自己相関には依存せず，

常に白色信号が入力された場合と同じである .

2.4.2 方形窓 RLS 法

方形窓の窓長を M とすれば , (2.46) で定義される相関行列 $\mathbf{R}(k)$ と (2.47) で定義される相互相関 $\mathbf{q}(k)$ の時間更新式は ,

step 1

$$\mathbf{R}'(k) = \mathbf{R}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) \quad (2.60)$$

$$\mathbf{q}'(k) = \mathbf{q}(k-1) + \mathbf{x}(k)y(k) \quad (2.61)$$

step 2

$$\mathbf{R}(k) = \mathbf{R}'(k) - \mathbf{x}(k-M)\mathbf{x}^T(k-M) \quad (2.62)$$

$$\mathbf{q}(k) = \mathbf{q}'(k) - \mathbf{x}(k-M)y(k-M) \quad (2.63)$$

のように二段階の更新に表わすことができる . このそれぞれについて , 忘却係数 $\lambda = 1$ の指数窓の RLS 法を適用することで , 方形窓の RLS 法が導出できる . step 1 では指数窓の場合と同様にして $\hat{\mathbf{h}}'_{LS}(k)$ を求める . step 2 では , 新たなカルマンゲインと変換係数

$$\tilde{\mathbf{k}}'(k) = \mathbf{R}'(k)\mathbf{x}(k-M) \quad (2.64)$$

$$\gamma'(k) = 1/(1 - \mathbf{x}^T(k-M)\tilde{\mathbf{k}}'(k)) \quad (2.65)$$

を計算し , これらを用いて相関行列 \mathbf{R} の逆行列と適応フィルタ $\hat{\mathbf{h}}_{LS}(k)$ の更新を以下のように行う .

$$\mathbf{R}^{-1}(k) = \mathbf{R}'^{-1}(k) + \gamma'(k)\tilde{\mathbf{k}}'(k)\tilde{\mathbf{k}}'^T(k) \quad (2.66)$$

$$e'(k, k-M) = y(k-M) - \mathbf{x}^T(k-M)\hat{\mathbf{h}}'_{LS}(k) \quad (2.67)$$

$$\hat{\mathbf{h}}_{LS} = \hat{\mathbf{h}}'_{LS}(k) - \gamma'(k)e'(k, k-M)\tilde{\mathbf{k}}'(k) \quad (2.68)$$

2.5 高速 RLS 法

RLS の演算量は $O(L^2)$ (L は適応フィルタ長) であるが、さらに、 $O(L)$ に減じることができる。これが (演算量が少ないという意味で) 高速 RLS 法である。フィルタ形状、時間窓に応じた幾つかのアルゴリズムが知られているが、ここでは、トランスバーサルフィルタ型のアルゴリズムである Fast Transversal filters (FTF) 法について述べる。時間窓は、指数窓として説明するが、方形窓についても、第 4 章で必要となるので、簡単に触れる。

FTF 法は、これから見ていくように、入力信号が同一の、以下の四つのフィルタの更新に帰着する。

- 前向き予測フィルタ
- 後向き予測フィルタ
- カルマンゲイン
- 未知系フィルタ

FTF 法は、これらのフィルタの事前予測誤差の計算、フィルタ係数と最小二乗誤差和 (後述) の再帰更新から構成される。次節では、FTF 法導出の基礎となる共分散逆行列の性質等を示す。それを基に、前向き、後向き予測フィルタとカルマンゲインの三つのフィルタの再帰更新と最小二乗誤差和の再帰更新を導出する。

2.5.1 予備

前節で示した RLS 法の手順をみると、もし、カルマンベクトル $\tilde{\mathbf{k}}$ と変換係数 γ が時間的に再帰的に更新できれば、共分散逆行列 \mathbf{R}^{-1} の時間更新式が不要であることが分かる。高速 RLS 法では、共分散逆行列の時間と次数の更新に関する公式と、入力ベクトルのシフト時不変性を利用することで、カルマンゲインと変換係数を $O(L)$ で更新する。

まず，共分散逆行列 \mathbf{R}^{-1} の性質について述べる．次数に関する更新を行うので，次式のように正方行列の次元を下付文字で明記した表記に改める．

$$\mathbf{R}_L(k) = \mathbf{R}(k) \quad (2.69)$$

ここで，下付の L が行列の次元（次数）である．さて，この共分散逆行列には，以下の性質があることが知られている [12]．

$$\mathbf{R}_L^{-1}(k) = \begin{bmatrix} 0 & \mathbf{0}_{L-1}^T \\ \mathbf{0}_{L-1} & \mathbf{R}_{L-1}^{-1}(k-1) \end{bmatrix} + \frac{\mathbf{a}(k)\mathbf{a}^T(k)}{F(k)} \quad (2.70)$$

$$\mathbf{R}_L^{-1}(k) = \begin{bmatrix} \mathbf{R}_{L-1}^{-1}(k) & \mathbf{0}_{L-1} \\ \mathbf{0}_{L-1}^T & 0 \end{bmatrix} + \frac{\mathbf{b}(k)\mathbf{b}^T(k)}{B(k)} \quad (2.71)$$

ここで， $\mathbf{a}(k)$ ， $\mathbf{b}(k)$ は，それぞれ，線形前向き予測フィルタ，線形後ろ向き予測フィルタであり，

$$\mathbf{a}(k) = [1, a_1(k), a_2(k), \dots, a_{L-1}(k)]^T \quad (2.72)$$

$$\mathbf{b}(k) = [b_{L-1}(k), \dots, b_2(k), b_1(k), 1]^T \quad (2.73)$$

と定義される．また， $F(k)$ と $B(k)$ は，それぞれ，最小事後前向き推定誤差二乗和と最小事後後ろ向き推定誤差二乗和を表わす（前向き / 後ろ向き最小二乗誤差和と略す．） $\mathbf{a}(k)$ ， $\mathbf{b}(k)$ ， $F(k)$ と $B(k)$ は次式の augmented 正規方程式を満たす．

$$\mathbf{R}_L(k)\mathbf{a}(k) = [F(k), 0, \dots, 0]^T \quad (2.74)$$

$$\mathbf{R}_L(k)\mathbf{b}(k) = [0, \dots, 0, B(k)]^T \quad (2.75)$$

さて，(2.56) で定義した事後カルマンゲインは，同式右辺の $\gamma(k)$ の分母を払い，整理することで，

$$\mathbf{k}_L(k) = \mathbf{k}(k) = \mathbf{R}_L^{-1}(k)\mathbf{x}(k) \quad (2.76)$$

なる関係があることがわかる．また， $\gamma(k)$ も，

$$\begin{aligned}\gamma_L(k) = \gamma(k) &= 1 - 1 + \gamma_L(k) \\ &= 1 - \mathbf{x}_L^T(k) \mathbf{R}_L^{-1}(k) \mathbf{x}_L(k) \\ &= 1 - \mathbf{x}_L^T(k) \mathbf{k}_L(k)\end{aligned}\quad (2.77)$$

と書き直すことができる．上式は， γ が，目的信号を 1 としたカルマンゲインフィルタの事後誤差であるとみなせることを意味している．

2.5.2 線形予測フィルタと最小二乗誤差和の再帰更新

線形予測フィルタの更新は，RLS 法を参考にして次式ようにかける．

$$\mathbf{a}(k) = \mathbf{a}(k-1) - \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L-1}(k-1) \end{bmatrix} \gamma_{L-1}(k-1) e_a(k) \quad (2.78)$$

$$\mathbf{b}(k) = \mathbf{b}(k-1) - \begin{bmatrix} \tilde{\mathbf{k}}_{L-1}(k) \\ 0 \end{bmatrix} \gamma_{L-1}(k) e_b(k) \quad (2.79)$$

ただし， $e_a(k)$ ， $e_b(k)$ は，それぞれ，前向き，後ろ向きの事前予測誤差であり，次式で求められる．

$$e_a(k) = \mathbf{x}^T(k) \mathbf{a}(k-1) \quad (2.80)$$

$$e_b(k) = \mathbf{x}^T(k) \mathbf{b}(k-1) \quad (2.81)$$

これら事前推定誤差 $e_a(k)$ ， $e_b(k)$ と事後推定誤差 $e_a(k, k)$ ， $e_b(k, k)$ の関係は，(2.78) と (2.79) の左から $\mathbf{x}^T(k)$ を乗じ，(2.77) に注意すると，次式のように得られる．

$$e_a(k, k) = \gamma_{L-1}(k-1) e_a(k) \quad (2.82)$$

$$e_b(k, k) = \gamma_{L-1}(k) e_b(k-1) \quad (2.83)$$

次に，最小二乗誤差和の更新について述べる．(2.74) 左辺に (2.48) と (2.78) を代入すると，

$$\begin{aligned}\mathbf{R}_L(k)\mathbf{a}(k) &= (\lambda\mathbf{R}_L(k-1) + \mathbf{x}(k)\mathbf{x}^T(k)) \left(\mathbf{a}(k-1) - \begin{bmatrix} 0 \\ \mathbf{k}_{L-1}(k-1) \end{bmatrix} e_a(k) \right) \\ &= \lambda\mathbf{R}_L(k-1)\mathbf{a}(k-1) + \mathbf{x}(k)e_a(k) - \mathbf{R}_L(k) \begin{bmatrix} 0 \\ \mathbf{k}_{L-1}(k-1) \end{bmatrix} e_a(k)\end{aligned}\quad (2.84)$$

このベクトルの先頭の要素に着目すると，

$$\begin{aligned}F(k) &= \lambda F(k-1) + x(k)e_a(k) - \hat{x}(k)e_a(k) \\ &= \lambda F(k-1) + e_a(k, k)e_a(k)\end{aligned}\quad (2.85)$$

と，前向き最小二乗誤差和 $F(k)$ の時間更新式が導出される．ただし，

$$\hat{x}(k) = -[a_1(k), a_2(k), \dots, a_{L-1}(k)]\mathbf{x}_{L-1}(k-1)\quad (2.86)$$

同様に，後ろ向き最小二乗誤差和 $B(k)$ の再帰更新式は，

$$B(k) = \lambda B(k-1) + e_b(k, k)e_b(k)\quad (2.87)$$

となる．

2.5.3 カルマンゲインと変換係数の再帰更新

カルマンゲインの再帰更新式は，(2.70) と (2.71) それぞれの両辺に， $\mathbf{x}(k)$ を右から乗じ，(2.76) のカルマンゲインの定義を代入することで，以下のように得られる．

$$\mathbf{k}_L(k) = \begin{bmatrix} 0 \\ \mathbf{k}_{L-1}(k-1) \end{bmatrix} + \frac{e_a(k, k)}{F(k)}\mathbf{a}(k)\quad (2.88)$$

$$\mathbf{k}_L(k) = \begin{bmatrix} \mathbf{k}_{L-1}(k) \\ 0 \end{bmatrix} + \frac{e_b(k, k)}{B(k)}\mathbf{b}(k)\quad (2.89)$$

ただし, $e_a(k, k)$, $e_b(k, k)$ は, それぞれ, 前向き, 後ろ向きの事後予測誤差である. さらに, $\mathbf{k}_L(k)$ と $\mathbf{k}_{L-1}(k)$ が $\mathbf{a}(k)$ と $\mathbf{b}(k)$ ではなく, $\mathbf{a}(k-1)$ と $\mathbf{b}(k-1)$ から求まるように, (2.78) と (2.79) を (2.88) と (2.89) 代入すると,

$$\mathbf{k}_L(k) = \lambda \frac{F(k-1)}{F(k)} \begin{bmatrix} 0 \\ \mathbf{k}_{L-1}(k-1) \end{bmatrix} + \frac{e_a(k, k)}{F(k)} \mathbf{a}(k-1) \quad (2.90)$$

$$\mathbf{k}_L(k) = \lambda \frac{B(k-1)}{B(k)} \begin{bmatrix} \mathbf{k}_{L-1}(k) \\ 0 \end{bmatrix} + \frac{e_b(k, k)}{B(k)} \mathbf{b}(k-1) \quad (2.91)$$

となる. ところで, (2.70) の両辺から $\mathbf{x}(k)$ を乗じると,

$$\gamma_L(k) = \gamma_{L-1}(k-1) e_a^2(k, k) / F(k) \quad (2.92)$$

となる. さらに, この式から, $e_a^2(k, k)$ を消すために, (2.82) を (2.85) を代入すると,

$$\gamma_L(k) = \lambda \frac{F(k-1)}{F(k)} \gamma_{L-1}(k-1) \quad (2.93)$$

となる. 同様に,

$$\gamma_L(k) = \lambda \frac{B(k-1)}{B(k)} \gamma_{L-1}(k) \quad (2.94)$$

という関係がある. これら (2.93) と (2.94), さらに, (2.54), (2.82) と (2.83) を (2.90) と (2.91) に代入すると, $\tilde{\mathbf{k}}$ の更新式が得られる.

$$\tilde{\mathbf{k}}_L(k) = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L-1}(k-1) \end{bmatrix} + \lambda^{-1} \frac{e_a(k)}{F(k-1)} \mathbf{a}(k-1) \quad (2.95)$$

$$\begin{bmatrix} \tilde{\mathbf{k}}_{L-1}(k) \\ 0 \end{bmatrix} = \tilde{\mathbf{k}}_L(k) - \lambda^{-1} \frac{e_b(k)}{B(k-1)} \mathbf{b}(k-1) \quad (2.96)$$

変換係数の更新はまず, 前述した (2.93) と (2.85) により次数と時刻の更新を行う.

$$\begin{aligned} \gamma_L^{-1}(k) &= \frac{F(k)}{F(k-1)} \lambda^{-1} \gamma_{L-1}^{-1}(k-1) \\ &= \frac{\lambda F(k-1) + e_a(k, k) e_a(k)}{F(k-1)} \lambda^{-1} \gamma_{L-1}^{-1}(k-1) \\ &= \gamma_{L-1}^{-1}(k-1) + e_a(k) \frac{e_a(k)}{F(k-1)} \\ &= \gamma_{L-1}^{-1}(k-1) + e_a(k) \tilde{k}_{L,1}(k) \end{aligned} \quad (2.97)$$

ここで, $\tilde{k}_{L,1}(k)$ は $\tilde{\mathbf{k}}(k)$ の先頭の要素を表わす. 次数の低減更新は, (2.94) によりできそうだが, (2.83) と (2.87) より分かるように, $B(k)$ は $\gamma_{L-1}(k)$ を計算した後でなくては求まらない. そこで, 次の関係, つまり, (2.96) より, $\tilde{\mathbf{k}}_L$ の L 番目の要素 $\tilde{k}_{L,L}$ が,

$$\tilde{k}_{L,L}(k) = \lambda^{-1} \frac{e_b(k)}{B(k-1)} \quad (2.98)$$

または, $e_b(k)$ について整理して,

$$e_b(k) = \lambda \tilde{k}_{L,L}(k) B(k-1) \quad (2.99)$$

とかける, ことを利用する. これを (2.87) に代入し, (2.83) により事後誤差 $e_b(k, k)$ を事前誤差 $e_b(k)$ に置き換えると,

$$B(k) = \lambda \left[1 - e_b(k) \gamma_{L-1}(k) \tilde{k}_{L,L}(k) \right] B(k-1) \quad (2.100)$$

となる. これを (2.94) に代入して $B(k)$ を消去し, $\gamma_{L-1}(k)$ について整理すると, γ の次数低減更新式が得られる.

$$\gamma_{L-1}^{-1}(k) = \left[1 - e_b(k) \gamma_L(k) \tilde{k}_{L,L}(k) \right] \gamma_L^{-1}(k) \quad (2.101)$$

以上, FTF 法を下表にまとめる.

演算	式番号	演算量
1) $e_a(k) = \mathbf{x}^T(k)\mathbf{a}(k-1)$	(2.80)	L
2) $e_a(k, k) = \gamma_{L-1}(k-1)e_a(k)$	(2.82)	
3) $\tilde{\mathbf{k}}_L(k) = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L-1}(k-1) \end{bmatrix} + \lambda^{-1} \frac{e_a(k)}{F(k-1)} \mathbf{a}(k-1)$	(2.95)	L
4) $F(k) = \lambda F(k-1) + e_a(k, k)e_a(k)$	(2.85)	
5) $\mathbf{a}(k) = \mathbf{a}(k-1) - \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}_{L-1}(k-1) \end{bmatrix} \gamma_{L-1}(k-1)e_a(k)$	(2.78)	L
6) $e_b(k) = \lambda \tilde{k}_{L,L}(k)B(k-1)$	(2.99)	
7) $\gamma_L^{-1}(k) = \gamma_{L-1}^{-1}(k-1) + e_a(k)\tilde{k}_{L,1}(k)$	(2.97)	
8) $\gamma_{L-1}^{-1}(k) = \left[1 - e_b(k)\gamma_L(k)\tilde{k}_{L,L}(k)\right] \gamma_L^{-1}(k)$	(2.101)	
9) $e_b(k, k) = \gamma_{L-1}(k)e_b(k)$	(2.83)	
10) $\begin{bmatrix} \tilde{\mathbf{k}}_{L-1}(k) \\ 0 \end{bmatrix} = \tilde{\mathbf{k}}_L(k) - \lambda^{-1} \frac{e_b(k)}{B(k-1)} \mathbf{b}(k-1)$	(2.96)	L
11) $B(k) = \lambda B(k-1) + e_b(k, k)e_b(k)$	(2.87)	
12) $\mathbf{b}(k) = \mathbf{b}(k-1) - \begin{bmatrix} \tilde{\mathbf{k}}_{L-1}(k) \\ 0 \end{bmatrix} \gamma_{L-1}(k)e_b(k)$	(2.79)	L
13) $e(k) = y(k) - \mathbf{x}^T(k)\hat{\mathbf{h}}_{LS}(k-1)$	(2.59)	L
14) $\hat{\mathbf{h}}_{LS}(k) = \hat{\mathbf{h}}_{LS}(k-1) + \gamma(k)e(k)\tilde{\mathbf{k}}(k)$	(2.58)	L

アルゴリズム 2.5: FTF 法

ここに示した FTF の演算量 (積, 積和の回数) は約 $7L$ である. ステップ 1)–12) が RLS 法でのカルマンゲインの更新に相当する. 最後の二つのステップ 13) と 14) は指数窓 RLS 法の誤差フィルタリングと更新の式と同一である.

2.5.4 方形窓の FTF [1]

方形窓の場合は, $\lambda = 1$ か, もしくは, 1 に近い値とり, 二回, 前節の FTF を実行する. このとき, 二回の FTF 法で \mathbf{a} , \mathbf{b} , F と B は共通に用い, $\tilde{\mathbf{k}}$, γ は個別に用意する. 一回目の FTF 法は指数窓の FTF 法に同じだが, 途中経過であるため, $\mathbf{a}(k-1)$, $\mathbf{b}(k-1)$, $F(k-1)$, $B(k-1)$ と $\hat{\mathbf{h}}_{LS}(k-1)$ はそれぞれ $\mathbf{a}'(k)$, $\mathbf{b}'(k)$, $F'(k)$, $B'(k)$ と $\hat{\mathbf{h}}'_{LS}(k)$ に更新されたとする. 二回目の FTF 法の手順を以下に示すが, 前項の FTF の手順とは何箇所かで符号が反転することに注意が必要である.

- 1) $e'_a(k, k-M) = \mathbf{x}^T(k-M)\mathbf{a}'(k)$
- 2) $e_a(k, k-M) = \gamma'_{L-1}(k-1)e'_a(k, k-M)$
- 3) $\tilde{\mathbf{k}}'_L(k) = \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}'_{L-1}(k-1) \end{bmatrix} + \frac{e'_a(k, k-M)}{F'(k)}\mathbf{a}'(k)$
- 4) $F(k) = F'(k) - e_a(k, k-M)e'_a(k, k-M)$
- 5) $\mathbf{a}(k) = \mathbf{a}'(k) + \begin{bmatrix} 0 \\ \tilde{\mathbf{k}}'_{L-1}(k-1) \end{bmatrix} \gamma'_{L-1}(k-1)e'_a(k, k-M)$
- 6) $e'_b(k, k-M) = \tilde{\mathbf{k}}'_{L,L}(k)B'(k)$
- 7) $\gamma'^{-1}_L(k) = \gamma'^{-1}_{L-1}(k-1) - e'_a(k, k-M)\tilde{\mathbf{k}}'_{L,1}(k)$
- 8) $\gamma'^{-1}_{L-1}(k) = \left[1 + e'_b(k, k-M)\gamma'_L(k)\tilde{\mathbf{k}}'_{L,L}(k)\right] \gamma'^{-1}_L(k)$
- 9) $e_b(k, k-M) = \gamma'_{L-1}(k)e'_b(k, k-M)$
- 10) $\begin{bmatrix} \tilde{\mathbf{k}}'_{L-1}(k) \\ 0 \end{bmatrix} = \tilde{\mathbf{k}}'_L(k) - \frac{e'_b(k, k-M)}{B'(k)}\mathbf{b}'(k)$
- 11) $B(k) = B'(k) - e_b(k, k-M)e'_b(k, k-M)$
- 12) $\mathbf{b}(k) = \mathbf{b}'(k) + \begin{bmatrix} \tilde{\mathbf{k}}'_{L-1}(k) \\ 0 \end{bmatrix} \gamma'_{L-1}(k)e'_b(k, k-M)$
- 13) $e'(k, k-M) = y(k-M) - \mathbf{x}^T(k-M)\hat{\mathbf{h}}'_{LS}(k)$
- 14) $\hat{\mathbf{h}}_{LS}(k) = \hat{\mathbf{h}}'_{LS}(k) - \gamma'(k)e'(k, k-M)\tilde{\mathbf{k}}'(k)$

2.5.5 FTF の数値的安定化

2.5.3 項で導出した FTF 法は数値的に不安定、つまり、有限語長での演算による量子化誤差が指数的に累積する。本項では、FTF 法が発散するのを避ける幾つかのアプローチを概説する。

不安定の検出と回避

このアプローチは、FTF 法自体を数値的に安定にするのではなく、不安定になる前兆を検出して、FTF 法を再初期化するものである。最初期化の最中は、LMS 法などのより安定なアルゴリズムで代用することもある [13]。具体的な前兆としては、

- 変換係数 γ_L と γ_{L-1} の比 [14]

アルゴリズム 2.5 中の、 $1 - e_b(k)\gamma_L(k)\tilde{k}_{L,L}(k)$ に対応する。変換係数は 0 以上 1 以下であるから、その比は非負であるはずだが、量子化誤差が累積しアルゴリズムが不安定になる直前に負の値をとる。

- 初期化からの経過時間 [13]

この方法では FTF 法が不安定になるまでに再初期化を繰り返す。

不安定化機構の抑圧

このアプローチは、FTF 法の手順を変更して、量子化誤差を累積させる正帰還を絶つものである。

- 量子化誤差のフィードバック [15]

変換係数の比の符号が、アルゴリズムが不安定になる前に符号が正から負に変化することから、変換係数の比の計算に関係のある後ろ向き予測残差 e_b 、変換係数 γ_L と正規化カルマンゲインの最後の要素 $\tilde{k}_{L,L}(k)$ がアルゴリズムの安定性に関係していると推測される。この方法は、後ろ向き予測残差を (2.99) によりスカラー変換で求める方法と、(2.81) によりフィルタリングで求める方法の二通りの方法で計算し、その差を後ろ向き予測残差を用いる他の変数の計算に導入するものである。

- 相関行列の正則化 [16]

量子化誤差の大きさが、相関行列の条件数、つまり、最大固有値と最小固有値の比で近似値としてはパワースペクトルの最大値と最小値の比、に比例することから、小さい固有値の底上げをして量子化誤差の発生を抑圧する方法である。固有値の底上げのために、フィルタ長毎に仮想的にインパルスを FTF に入力する。

2.6 組み合わせて用いる手法

これまで代表的な適応フィルタ手法である、(N)LMS 法、AP 法、そして (高速) RLS 法を説明した。ここでは、それらと組み合わせることで収束速度を改善したり、演算量を低減する手法を紹介する。

2.6.1 ブロック処理に基づくもの

- 単純なブロック処理

ブロック処理は、主に演算量の低減のために用いられる。ブロック長毎に適応フィルタを更新することで、推定出力 y の計算が固定フィルタとデータ列との畳み込みになるので、これを FFT を使う等の高速畳み込みで計算できる。同様に、適応フィルタの更新も高速畳み込みにより計算できるので演算量が低減することが可能である。ブロック処理の欠点は、データの蓄積と推定出力を計算することによる遅延が生じることと収束速度が若干遅くなることである。

代表的なもとして、ブロック LMS 法 [17] を示す。

$$1) \quad \mathbf{e}(k) = \mathbf{y}(k) - \mathbf{X}_{L,N}^T(k) \hat{\mathbf{h}}(k - N)$$

$$2) \quad \hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k - N) + (\mu/N) \mathbf{X}_{L,N} \mathbf{e}(k)$$

N : ブロック長

アルゴリズム 2.6: ブロック LMS 法

AP 法をブロック処理にしたものには、ブロック直交射影アルゴリズム [18] や第 5.2.1 項で述べる方法がある。BOP 法とは数学的に等価だが異なる定式化したものとして PRA 法 [10] がある。

- Exact ブロック処理

ブロック処理の収束速度の劣化を回避し、遅延以外はオリジナルのアルゴリズムと全く同じ収束特性を持つブロック処理 LMS 法が、Benesty と Duhamel によって提案されている [19]。この考えを AP 法に適用し、さらに、推定出力の計算のブロック長をと適応フィルタの更新のブロック長より短くして出力遅延を短縮することを可能にしたアルゴリズムを第 5 章で説明する。

- サブバンド処理

サブバンド処理は信号を幾つかの周波数帯域に分割し、それぞれの帯域で処理を行う手法である [20, 21]。処理の流れを図 2.2 で説明する。目的信号 d と入力信号 x は $Wj^{\frac{2\pi}{N}ni}$ を乗じて中心周波数を移動した後、ポリフェーズフィルタにより帯域制限され、次に M 分の 1 に間引かれる。各バンドでの残留エコー E_i は M 倍に内挿され、低域通過フィルタ f' を通過し、中心周波数を元に戻して加算されフルバンド信号となる。サブバンド処理を行うことの利点としては、まず、演算量の低減がある。間引きによって M 倍にサンプル間隔が伸び、さらに、フィルタの長さも $1/M$ になるので、各周波数帯域の演算量は全帯域の $1/M^2$ となる。さらに帯域の数が $N/2 + 1$ あり、演算が実数から複素数になり約 4 倍になることから、全帯域一括で処理するのに

比べ、演算量は約 $(2N + 4)/M^2$ になる．例えば $N = 32, M = 16$ の場合、0.27 となる．この演算量低減の利点を生かして、高速 RLS 法を DSP 上で実時間動作させる試みも行われている [22]．もうひとつの利点は周波数帯域毎の性質の違いを反映できることである．帯域毎にみると入力信号のレベルのばらつきが減り、白色信号に近づくので、適応フィルタの収束は速くなる．また、音響エコーキャンセラの場合、高い周波数ほど室内のインパルス応答が短いという性質を反映できる．

サブバンド処理の欠点は、ブロック処理と同じく処理遅延である．アンチエイリアシングフィルタ f 、アンチイメージングフィルタ f' により $N = 32$ の場合で 200 サンプルほど遅延する．分割数 N が大きいほど、フィルタ f と f' の阻止域のレベルを低くする必要があるので、フィルタ f と f' の長さとともに処理遅延が増加する．

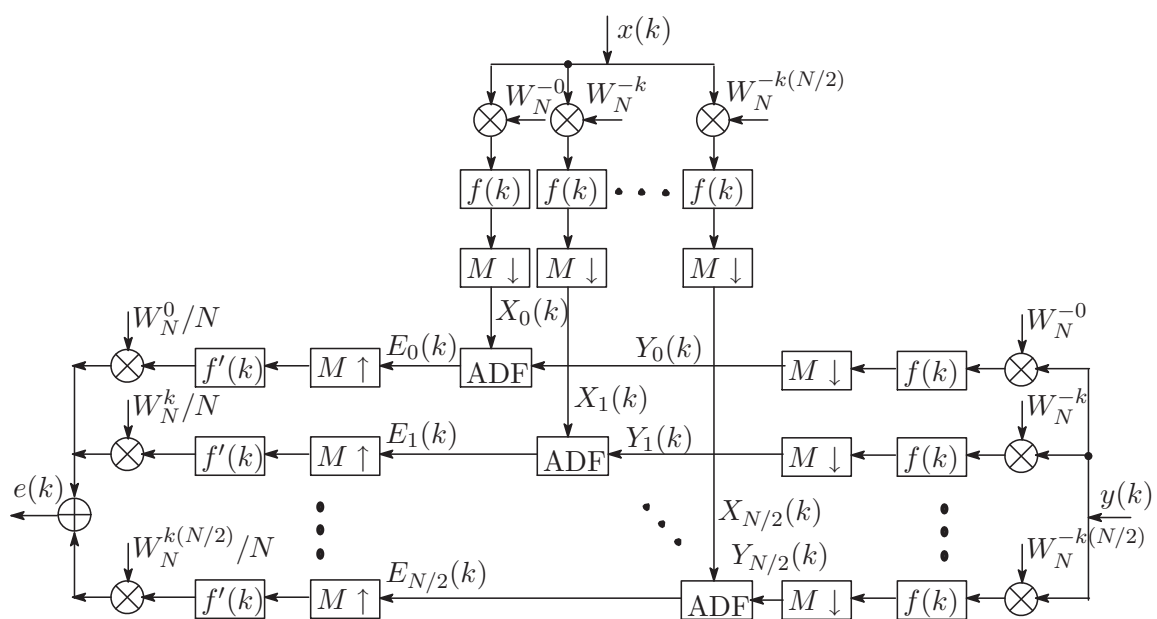


図 2.2: ポリフェーズフィルタバンクによるサブバンドエコーキャンセラの一構成. $W_N = e^{j2\pi/N}$, $f(k)$:アンチエイリアシングフィルタ, $f'(k)$:アンチイメージングフィルタ, $M \downarrow$:間引き, $M \uparrow$:内挿

2.6.2 音響の知識に基づくもの

- 室内インパルス応答の変動特性を利用したもの

牧野らは室内のインパルス応答の変動特性がインパルス応答と同じく指数減衰していることを適応フィルタの更新に反映する ES アルゴリズムを提案している [23, 24] .

- 室内インパルス応答の共通極零を利用したもの

羽田らは室内の複数のインパルス応答に共通する極と零を抜き出し, それを適応フィルタ入力の前に配置することで, 800Hz 以下では, 同じ消去量を得るための適応フィルタ長が削減でき, また, 収束時間も短縮することを示した [25] .

2.7 計算機シミュレーション

本節では, NLMS 法, AP 法, RLS 法の収束特性を計算機シミュレーションにより示すとともに三者の比較を行う. 用いる信号は白色ガウス信号, 4 次の IIR フィルタに白色ガウス信号を通過させた有色信号と音声 (日本語, 女性) である. 白色信号を用いた実験では, 上記三つの手法の最速の収束速度が等しくなることと, NLMS 法と RLS 法の等価性について, 有色信号では, 三つの方法の収束特性の違いについて, そして音声信号では, 音響エコーキャンセラを想定した条件での AP 法の他の手法との得失について述べる.

(1) 白色信号

実験条件は, フィルタ長は $L = 128$, 目的信号に -40 dB の白色ガウス信号を雑音として付加している. 正則化のための定数は, NLMS 法, AP 法では, $\delta_{\text{NLMS}} = 0.01L\sigma_x^2$ (σ_x^2 は入力信号の分散), RLS 法では, $\delta_{\text{RLS}} = 10\sigma_x^2$ である. 実験では, $2L$ で適応フィルタの学習を開始し, 以降, $32L$ まで, 目的信号は入力信号に等しい. 時刻 $32L$ で, 目的信号の符号を反転することで, 推定する系の変化を模擬する. このように途中で系の応答を変化させる理由は, 0 状態から学習を開始すると収束が速くなり, アルゴリズムの追従性能を比較するには不適當だからである. 図 2.3(a)-(d) に実験結果を示す. 縦軸は正規化した係数

誤差で，

$$\xi(k) = \frac{\|h(k) - h_o\|^2}{\|h_o\|^2} \quad (2.102)$$

で定義される．ここで， h_o は真のインパルス応答で，この図の場合， $\{-1, 0, 0, \dots\}$ である．同図 (a) は NLMS 法でステップサイズ μ を変化させたときの係数誤差の収束曲線である．ステップサイズが小さくなるに従い，定常状態に達するまでの時間が長くなるが，定常状態の誤差量は小さくなるというトレードオフの関係がある．最速の収束速度は，ステップサイズ $\mu = 1$ のときであり，このときの収束過渡での収束速度は，約 $6/L$ dB/iteration である [26]．同図 (b) は射影次数 $p = 2$ の AP 法でステップサイズ μ を変化させたときの係数誤差の収束曲線である． $\mu = 1, 0.5$ は，図 (a) NLMS のときの最速の収束速度と同じ収束速度で収束している．同図 (c) は AP 法で，ステップサイズを $\mu = 0.3$ と固定にし，射影次数を変化させたときの係数誤差の収束曲線である．射影次数を上げていくと収束速度は速くなるが，NLMS 法の最速収束速度で飽和する．同図 (d) は RLS 法で忘却係数 λ を変化させたときの係数誤差の収束曲線である．収束速度は忘却定数が $\lambda = 1 - 1/L$ のときに最速で忘却係数を 1 に近づけると遅くなる．図 (a) と図 (d) とを比較すると $L(1 - \lambda)$ が NLMS 法のステップサイズ μ に相当することがわかる [27]．

このように，白色信号が入力するときには，NLMS 法，AP 法，RLS 法のいずれでも，最速の収束速度は同じである．また，NLMS 法と RLS 法の収束特性は，ステップサイズ μ と $L/(1 - \lambda)$ とが等しければほぼ同じ収束特性を示す．

(2) 有色信号

入力信号が白色ガウス信号を 4 次の IIR フィルタ (係数:1, -1., 0.620097, -0.429807, 0.627265) を通した有色信号であること以外の条件は白色信号での実験に同じである．図 2.4(a), (b) には (1) 白色信号と同じく，それぞれ，NLMS 法と 2 次の AP 法でステップサイズを変えたときの係数誤差の収束曲線を示す．収束曲線は，入力信号が白色信号の場合と異なり，収束の初期においては傾きが急，つまり，収束が速く，時間が経つにつれて収束が遅くなる．収束の初期では，収束速度はステップサイズに依らないが，係数誤差が減少するに従って，ステップサイズによる収束速度の差が生じる．図 2.4(c) は AP 法で射影次

数を変化させたときの係数誤差の収束曲線を示している．次数を上げることで収束速度が改善される．4次以上のAP法は入力信号を白色化できるので，図2.3(c)にある $p-4+1$ 次の白色信号入力時の収束曲線と同様の収束速度を有する．図2.4(d)はRLS法で忘却係数を変化させたときの係数誤差の収束曲線である．これらも白色信号入力するとき，図2.3(d)と同一の収束速度を有することがわかる．

このように，入力信号が有色信号の場合には，NLMS法では，収束が遅くなることがある，他方，RLS法は白色信号のときと同じ収束速度を保つ．AP法は，次数を上げていくことで，白色信号入力時の収束速度にできることが分かる．

(3) 音声信号

実験条件が若干前二つの実験と異なる．まず、適応フィルタ長は $L = 1024$ とした．これは，音声信号は非定常なため，分析する区間の長さにより分析結果が異なるためである．音響エコーキャンセラへの応用を考えるので，1.2.5項で述べた例に沿って適応フィルタ長を $L = 1024$ とこれまでの128より長くとした．NLMS法とAP法におけるステップサイズは，定常時の係数誤差がほぼ等しくなるように決定した．また，実験時間の都合で，ノイズレベルは -40 dB から -30 dB とし，RLS法に替え高速RLS(SFTF)法を用いた．正則化定数は48次のAP法では $\delta = 0.1L\sigma_x^2$ それ以外のNLMS法とAP法で， $\delta = 0.05L\sigma_x^2$ ，RLS法では， $\delta_{\text{RLS}} = 10\sigma_x^2$ とした．図2.5は，NLMS法，AP法とSFTF法の音声に対する係数誤差の収束曲線である．NLMS法と次数2のAP法との間で大きな収束速度の向上が認められる．さらに，射影次数を48次までにすると，さらに収束速度が向上する．RLS法は系が変化した直後の収束は遅いが，収束が始まるとその速度は最も速い．RLS法での定常誤差が大きい理由は，音声信号が帯域制限されているために通過帯域以外での推定誤差が多いためである．係数誤差では，入力信号のない，あるいはレベルの小さい帯域の誤差の影響があるので，ノイズを除いたエコー消去量の大きさを評価するERLE(Echo Return Loss Enhancement)と呼ばれる値を

$$\text{ERLE} = E[(y - n)^2] / E[(e - n)^2] \quad (2.103)$$

と定義し，ERLE の収束特性を図 2.6 に示す．同図では，SFTF 法だけが定常状態での評価が悪いということではなく，残留エコーは他の手法と同レベルであることが分かる

入力信号が音声信号でフィルタ長が 1000 程度の場合には，(2) の有色信号の実験結果と同様に，NLMS 法はその他の手法の収束速度に比べて極めて遅い．実験 (2) と異なるのは，AP 法で 8 次程度では収束速度の限界に達せず，さらに大きな射影次数，ここでは 48，にしても収束速度の改善が見られることである．

2.8 本章のまとめ

本章では，主に代表的な適応フィルタアルゴリズムである NLMS 法，AP 法と RLS 法を，評価関数の最小化として導出することで概説した．特に，RLS 法の低演算算法である高速 RLS 法については，後の高速 AP 法との関連があるので詳しく説明した．また，計算機シミュレーションにより，白色雑音信号入力に対しては，NLMS 法，AP 法と RLS 法のどの適応フィルタアルゴリズムでも同様の収束が得られること，有色雑音信号と音声入力に対しては NLMS 法と AP 法では収束が遅くなるが AP 法では射影次数を上げることで収束速度が RLS 法並みに改善することをみた．

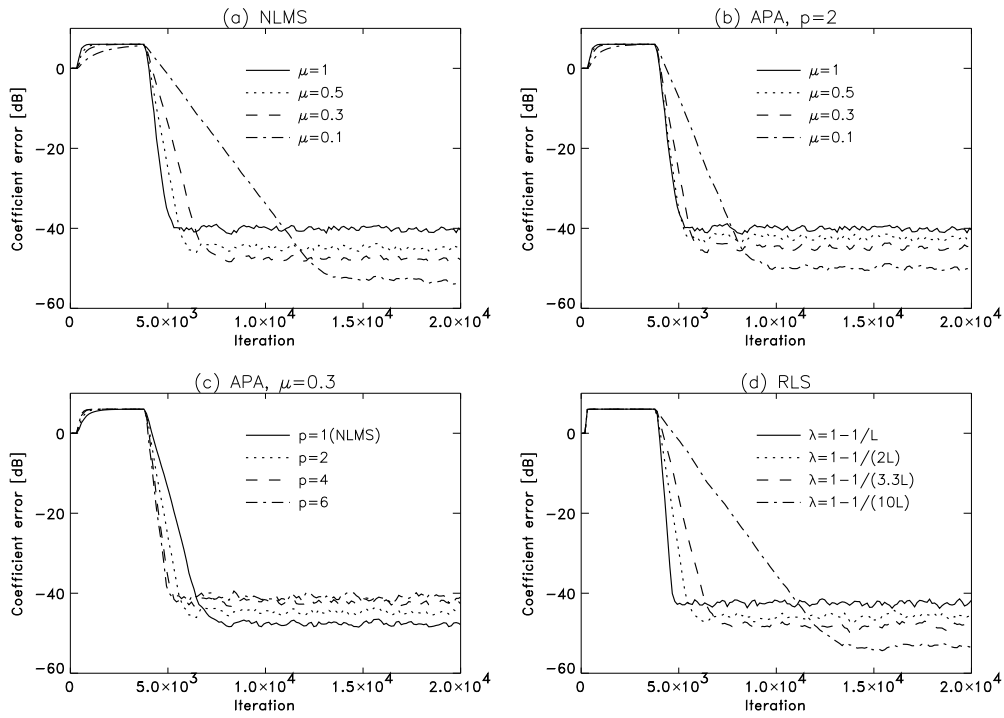


図 2.3: 白色ガウス信号に対する収束特性 . (a) NLMS 法, ステップサイズ $\mu = 1, 0.5, 0.3, 0.1$, (b)AP 法, 射影次数 $p = 2$, ステップサイズ $\mu = 1, 0.5, 0.3, 0.1$, (c) AP 法, 射影次数 $p = 1, 2, 4, 8$, ステップサイズ $\mu = 0.3$, (d) RLS 法, 忘却係数 $\lambda = 1 - 1/L, 1 - 1/(2L), 1 - 1/(3.3L), 1 - 1/(10L)$, (L :フィルタ長), 最速の収束速度は NLMS 法, AP 法, RLS 法のいずれでも同じあり, 約 $6/L$ dB/iteration . NLMS 法のステップサイズ μ と RLS 法の $L/(1 - \lambda)$ が等しければ, 両者はほぼ同じ収束特性を示す . フィルタ長 $L = 128$, 正則化定数 $\delta_{\text{NLMS}} = \delta_{\text{APA}} = 0.01L\sigma_x^2$, $\delta_{\text{RLS}} = 10\sigma_x^2$

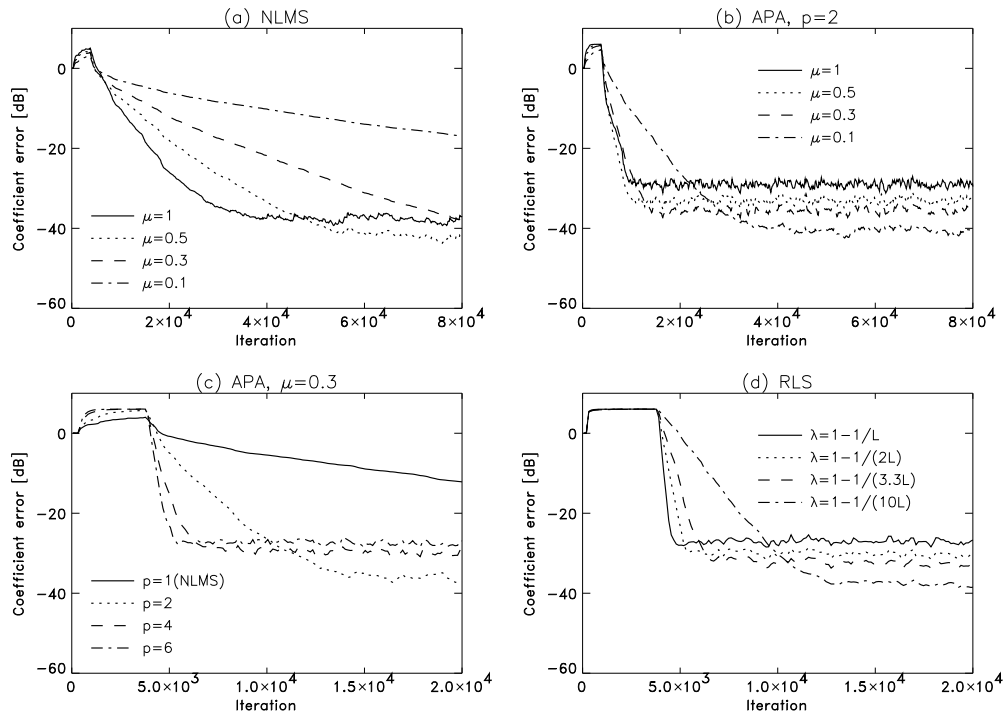


図 2.4: 有色ガウス信号に対する収束特性 . (a) NLMS 法, ステップサイズ $\mu = 1, 0.5, 0.3, 0.1$, (b)AP 法, 射影次数 $p = 2$, ステップサイズ $\mu = 1, 0.5, 0.3, 0.1$, (c) AP 法, 射影次数 $p = 1, 2, 4, 8$, ステップサイズ $\mu = 0.3$, (d) RLS 法, 忘却係数 $\lambda = 1 - 1/L, 1 - 1/(2L), 1 - 1/(3.3L), 1 - 1/(10L)$, (L :フィルタ長) NLMS 法は収束が遅くなる . AP 法は射影次数を上げると収束速度が改善され , 十分大きければ白色信号入力の際の収束速度が得られる . RLS 法は白色入力信号の際と同じ収束速度である . フィルタ長 $L = 128$, 正則化定数 $\delta_{\text{NLMS}} = \delta_{\text{APA}} = 0.01L\sigma_x^2$, $\delta_{\text{RLS}} = 10\sigma_x^2$

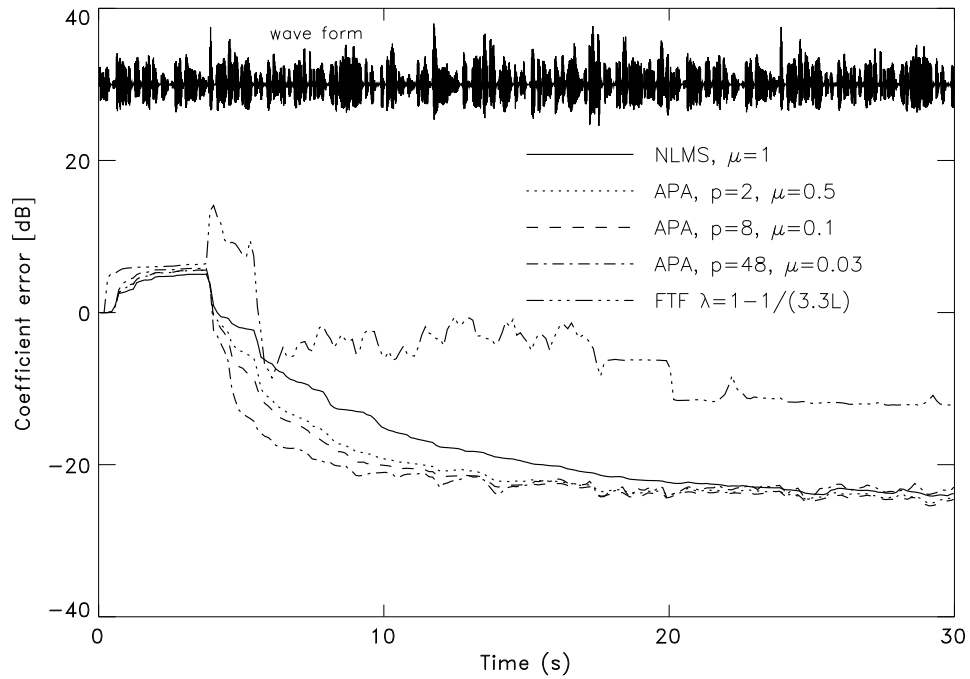


図 2.5: 音声信号に対する収束特性 . NLMS 法と次数 2 の AP 法との間で大きな収束速度の向上が認められる . さらに , 48 次までにすると , さらに収束速度が向上する . RLS 法は系が変化した直後の収束は遅いが , 収束が始まるとその速度は最も速い . SFTF 法での定常誤差が大きい理由は , 音声信号が帯域制限されているために通過帯域以外での推定誤差が多いためである . フィルタ長 $L = 1024$, 正則化定数 $\delta_{\text{NLMS}} = 0.05L\sigma_x^2$, $p = 2, 8$, に対しては $\delta_{\text{APA}} = 0.05L\sigma_x^2$. $p = 48$ に対しては $\delta_{\text{APA}} = 0.1L\sigma_x^2$, $\delta_{\text{RLS}} = 10\sigma_x^2$

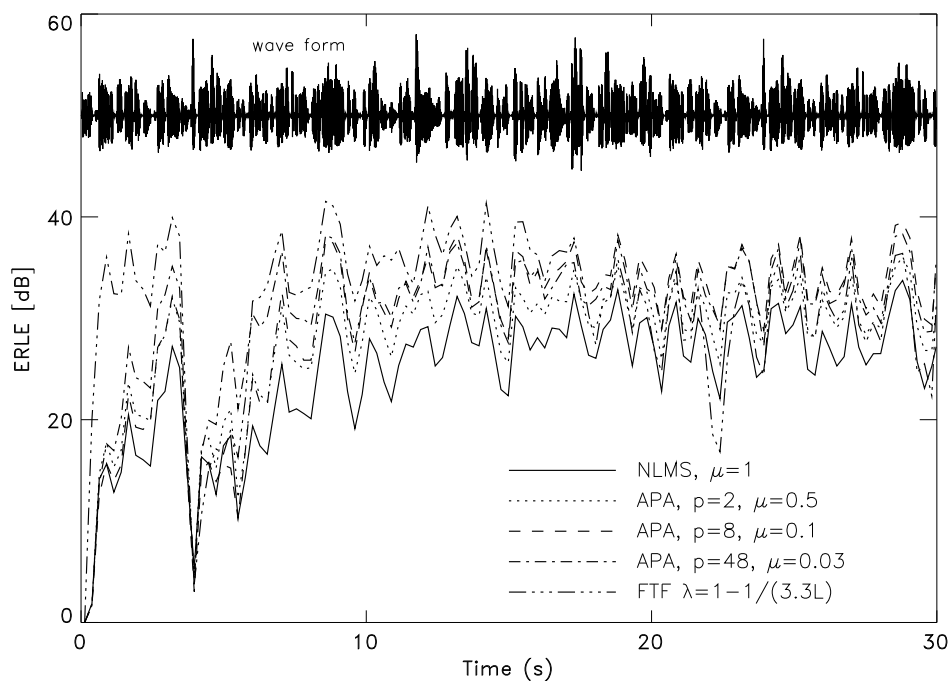


図 2.6: 音声信号に対する ERLE の収束特性 . 図 2.5 のように , FTF 法の定常状態での評価量が悪いということはなく , 他の手法と同レベルであることが分かる . フィルタ長 $L = 1024$, 正則化定数 $\delta_{\text{NLMS}} = 0.05L\sigma_x^2$

第3章 アフィン射影法の複合NLMS法としての解釈と収束特性解析

3.1 はじめに

第2章でみたように，AP法が射影次数を上げるとともに収束速度が向上し，定常状態での係数誤差が増加するという性質がある．本章ではAP法がこのような収束特性を有することの定性的，定量的でかつ容易な理解を目的として収束特性の解析を行う．AP法の収束特性の解析がこれまで幾つか行われてきた [26, 40, 41, 42, 7]．しかしながら，これらの多くは，係数誤差の収束を特異値分解によりフィルタ長個のモードに分解する手法を取っているため，射影次数よりはるかに多いフィルタ長個のモードを導入しておいた点で冗長であり，さらに，導出過程が長く様々な近似を導入することも理解の妨げになっている．さらに，有色信号に対する過渡状態，定常状態の誤差レベルの近似度はそれほど高くない．そこで，この章では，収束特性がよく知られているNLMS法を基本要素としてAP法を表現し，収束特性解析を行うこととした．収束特性の評価量としては，エコー伝達経路を模擬するデジタルフィルタ係数値の真値との誤差（係数誤差）に着目して，その二乗和の収束の過程（過渡状態係数二乗誤差）および収束後の値（定常状態係数二乗誤差）を取り上げる．なお，以下では，過渡状態係数二乗誤差は過渡誤差，定常状態係数二乗誤差は定常誤差と略すこととする．

次節以降，2節ではNLMS法とAP法，加えて，解析の道具であるコレスキー分解を概説する．3節ではAP法を複数のNLMS法の集まりと解釈する複合NLMS法モデルを示す．次に，4節では，2，3節を受けてAP法の収束特性をNLMS法の収束特性を用いて表現する．節で計算機シミュレーションによりこの表現の妥当性を検証し，最後にまと

める .

3.2 準備

この節では，記号の定義，収束特性の指標とする量の定義と，次節以降で必要となる NLMS 法，AP 法とコレスキー分解の説明を行う．

3.2.1 NLMS 法

図 3.1 に NLMS 法を用いた適応フィルタ系の構成を示す．同図で， k は離散時刻を表し， x, d, y, e, n は，それぞれ，入力信号，目的信号，推定信号，推定誤差，雑音を表す．また， $\hat{\mathbf{h}}, \mathbf{h}_o$ は，それぞれ，適応フィルタ係数，未知系の最適ウィナーフィルタ係数を表わす．太文字はその変数がベクトルあるいは行列であることを表す．NLMS 法は，

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu \left\{ \mathbf{x}(k) / \|\mathbf{x}(k)\|^2 \right\} e(k) \quad (3.1)$$

$$\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-L+1)]^T \quad (3.2)$$

により適応フィルタ係数 $\hat{\mathbf{h}}$ を更新する．ここで， L は適応フィルタ長， μ はステップサイズを表わす．また，目的信号 $d(k)$ ，推定信号 $y(k)$ ，推定誤差 $e(k)$ は，それぞれ，

$$d(k) = \mathbf{x}^T(k) \mathbf{h}_o + n(k) \quad (3.3)$$

$$y(k) = \mathbf{x}^T(k) \hat{\mathbf{h}}(k-1) \quad (3.4)$$

$$e(k) = d(k) - y(k) \quad (3.5)$$

である．これら (3.3),(3.4),(3.5) を (3.1) に代入すると，フィルタ係数誤差 $\boldsymbol{\epsilon}(k) = \hat{\mathbf{h}}(k) - \mathbf{h}_o$ に関する再帰式

$$\boldsymbol{\epsilon}(k) = [\mathbf{I} - \mu \mathbf{P}_x(k)] \boldsymbol{\epsilon}(k-1) + \mu \tilde{\mathbf{x}}(k) n(k) \quad (3.6)$$

が得られる．ただし， \mathbf{I} は単位行列， $\mathbf{P}_x(k)$ はベクトル $\mathbf{x}(k)$ に対する射影行列であり，

$$\mathbf{P}_x(k) = \tilde{\mathbf{x}}(k)\mathbf{x}^T(k) \quad (3.7)$$

$$\tilde{\mathbf{x}}(k) = \mathbf{x}(k) / \|\mathbf{x}(k)\|^2 . \quad (3.8)$$

過渡誤差

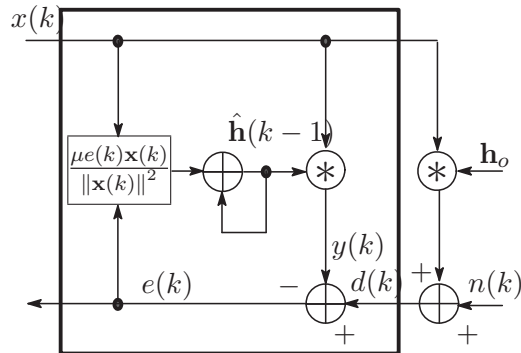


図 3.1: NLMS 法を用いた適応系

入力信号 x に対する，時刻 k での NLMS 法による係数二乗誤差の期待値を $J_{\text{NLMS},x}(k)$ と表す．すなわち，

$$J_{\text{NLMS},x}(k) \equiv E \left[\|\epsilon(k)\|^2 \right] . \quad (3.9)$$

(3.6) の右辺において，過渡状態においては雑音より係数誤差の方が大きく $\|\text{第一項}\|^2 \gg \|\text{第二項}\|^2$ と見なせる．よって (3.9) は，(3.6) の第一項の二乗ノルムを ϵ に関して再帰的に展開して，次式のように近似できる．

$$\begin{aligned} J_{\text{NLMS},x}(k) &\approx E \left[\left\| [\mathbf{I} - \mu \mathbf{P}_x(k)] \epsilon(k-1) \right\|^2 \right] \\ &= E \left[\left\| \prod_{m=1}^k [\mathbf{I} - \mu \mathbf{P}_x(m)] \epsilon(0) \right\|^2 \right] \end{aligned} \quad (3.10)$$

定常誤差

適応フィルタが収束している定常状態での係数二乗誤差の期待値は $J_{\text{NLMS},x}(\infty)$ は，

$$J_{\text{NLMS},x}(\infty) \approx \frac{\mu}{2 - \mu} \frac{\sigma_n^2}{\sigma_x^2} \quad (3.11)$$

により良く近似される [43] . ただし , σ_n^2 と σ_x^2 はそれぞれ , 雑音 n と入力信号 x の分散である . この式の導出は付録 A を参照 .

3.2.2 AP 法

NLMS 法では一時刻分の推定誤差 $e(k)$ と入力信号ベクトル $\mathbf{x}(k)$ を用いて適応フィルタを更新するが , AP 法では , 射影次数 $p(\leq L)$ 時刻分の推定誤差と入力信号ベクトルを用いて適応フィルタを更新する . AP 法の適応フィルタ係数の更新は , NLMS 法の更新 (3.1) において , ベクトル \mathbf{x} とスカラー e をそれぞれ行列 \mathbf{X} とベクトル \mathbf{e} に置き換えることにより ,

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu \mathbf{X}(k) [\mathbf{X}^T(k) \mathbf{X}(k)]^{-1} \mathbf{e}(k) \quad (3.12)$$

と得られる . ここで , \mathbf{X} は L 行 (フィルタ長) p 列 (射影次数) の入力信号行列

$$\mathbf{X}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-p+1)] \quad (3.13)$$

である . また \mathbf{e} は要素数 p の列ベクトルであり ,

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k) \hat{\mathbf{h}}(k-1) \quad (3.14)$$

$$\mathbf{d}(k) = [d(k), d(k-1), \dots, d(k-p+1)]^T \quad (3.15)$$

$$= \mathbf{X}^T(k) \mathbf{h}_o + \mathbf{n}(k),$$

$$\mathbf{n}(k) = [n(k), n(k-1), \dots, n(k-p+1)]^T \quad (3.16)$$

により定義される .

なお、入力信号の共分散行列 $\mathbf{X}^T(k) \mathbf{X}(k)$ の固有値が 0 か 0 に近いことがあり , その場合には逆行列の固有値が大きな値となり (12) で誤差 $e(k)$ 中のノイズを拡大し , 適応フィルタの係数が乱れる . これを避けるために , 実際的には次式に示すように共分散行列 $\mathbf{X}^T(k) \mathbf{X}(k)$ に対角に微少量 δ を持つ対角行列 $\delta \cdot \mathbf{I}$ を加えることで固有値の最小値を保証した更新式を用いる .

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu \mathbf{X}(k) [\mathbf{X}^T(k) \mathbf{X}(k) + \delta \cdot \mathbf{I}]^{-1} \mathbf{e}(k) \quad (3.17)$$

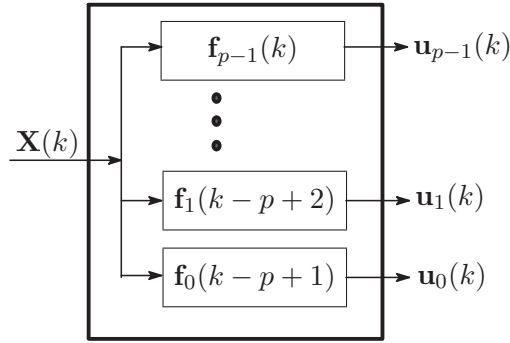


図 3.2: 前向き予測フィルタバンク

3.2.3 前向き予測とコレスキー分解

ここでは，前向きの予測フィルタとコレスキー分解について述べる [44]．図 3.2 のように，入力信号 $x(k)$ の $0, 1, \dots, p-1$ 次の前向き予測フィルタからなるフィルタバンクを考える．ただし，同図で $\mathbf{f}_0(k-p+1), \mathbf{f}_1(k-p+2), \dots, \mathbf{f}_{p-1}(k)$ は予測係数列ベクトルである． i 次の前向き予測係数ベクトル $\mathbf{f}_i(k-p+1+i)$ は，以下の正規方程式の解である．

$$\begin{aligned} \mathbf{X}_i^T(k-p+1+i)\mathbf{X}_i(k-p+1+i) \\ \mathbf{f}_i(k-p+1+i) = [U_i(k-p+1+i), 0, \dots, 0]^T \end{aligned} \quad (3.18)$$

ただし，

$$\begin{aligned} \mathbf{X}_i(k-p+1+i) = \\ [\mathbf{x}(k-p+1+i), \mathbf{x}(k-p+i), \dots, \mathbf{x}(k-p+1)] \end{aligned} \quad (3.19)$$

また， $U_i(k-p+1+i)$ は $\mathbf{f}_i(k-p+1+i)$ による予測残差の二乗和である．予測係数ベクトル $\mathbf{f}_i(k-p+1+i)$ の j 番目の係数を $f_{i,j}(k-p+1+i)$ と表わすと， p 行 p 列のフィル

タバンク行列 $\mathbf{F}(k)$ は ,

$$\begin{aligned} \mathbf{F}(k) &= [\mathbf{f}_{p-1}(k), \dots, \mathbf{f}_1(k-p+2), \mathbf{f}_0(k-p+1)] \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ f_{p-1,1}(k) & \ddots & 0 & 0 \\ \vdots & \dots & \vdots & \vdots \\ f_{p-1,p-2}(k) & \dots & 1 & 0 \\ f_{p-1,p-1}(k) & \dots & f_{1,1}(k-p+2) & 1 \end{bmatrix} \end{aligned} \quad (3.20)$$

とかける . このフィルタバンク行列は入力信号行列 $\mathbf{X}(k)$ の列ベクトル $\mathbf{x}(k-i)$ に対して i 次の前向き予測を行い予測残差 $\mathbf{u}_i(k)$ を生成する . 予測残差 $\mathbf{u}_i(k)$ を列ベクトルにもつ行列 $\mathbf{U}(k)$ はフィルタバンク行列 $\mathbf{F}(k)$ を用いて ,

$$\mathbf{U}(k) = [\mathbf{u}_{p-1}(k), \mathbf{u}_{p-2}(k), \dots, \mathbf{u}_0(k)] = \mathbf{X}(k)\mathbf{F}(k) \quad (3.21)$$

とかくことができる . さらに , 予測残差ベクトル $\mathbf{u}_i(k)$ は ,

$$\mathbf{u}_i^T(k)\mathbf{u}_j(k) = \begin{cases} \|\mathbf{u}_i(k)\|^2 (= U_i(k-p+1+i)) & (i=j), \\ 0 & (i \neq j) \end{cases} \quad (3.22)$$

のように互いに直交することから ,

$$\begin{aligned} \mathbf{U}^T(k)\mathbf{U}(k) &= \mathbf{D}(k) = \\ &\text{diag}(\|\mathbf{u}_{p-1}(k)\|^2, \|\mathbf{u}_{p-2}(k)\|^2, \dots, \|\mathbf{u}_0(k)\|^2) \end{aligned} \quad (3.23)$$

という関係が成り立つ . ここで , $\mathbf{D}(k)$ は予測残差のパワーを対角要素にもつ対角行列である . さらに , (3.21) と (3.23) より , 入力信号の共分散行列は , コレスキー分解として知られるように ,

$$[\mathbf{X}^T(k)\mathbf{X}(k)]^{-1} = \mathbf{F}(k)\mathbf{D}^{-1}(k)\mathbf{F}^T(k) \quad (3.24)$$

と分解される .

3.3 AP法の複合NLMS法モデル

この節では、コレスキー分解を用いてAP法の係数更新式(3.12)を複数のNLMS法の更新式が複合した形に変形する。(3.12)に(3.21)と(3.24)を代入し、(3.23)の関係を用いると、

$$\begin{aligned}
 \hat{\mathbf{h}}(k) &= \hat{\mathbf{h}}(k-1) + \mu \mathbf{U}(k) \mathbf{D}^{-1}(k) \mathbf{F}^T(k) \mathbf{e}(k) \\
 &= \hat{\mathbf{h}}(k-1) + \mu \left[\frac{\mathbf{u}_{p-1}(k)}{\|\mathbf{u}_{p-1}(k)\|^2}, \frac{\mathbf{u}_{p-2}(k)}{\|\mathbf{u}_{p-2}(k)\|^2}, \dots, \frac{\mathbf{u}_0(k)}{\|\mathbf{u}_0(k)\|^2} \right] \begin{bmatrix} \mathbf{f}_{p-1}^T(k) \\ \mathbf{f}_{p-2}^T(k-1) \\ \vdots \\ \mathbf{f}_0^T(k-p+1) \end{bmatrix} \mathbf{e}(k) \quad (3.25) \\
 &= \hat{\mathbf{h}}(k-1) + \mu \sum_{i=0}^{p-1} \frac{\mathbf{u}_i(k)}{\|\mathbf{u}_i(k)\|^2} e_{f,i}(k)
 \end{aligned}$$

と変形できる。ただし、(3.14)、(3.15)と(3.21)より、

$$\begin{aligned}
 e_{f,i}(k) &= \mathbf{f}_i^T(k-p+1+i) \mathbf{e}(k) \\
 &= \mathbf{f}_i^T(k-p+1+i) \left\{ \mathbf{X}^T(k) [\mathbf{h}_o - \hat{\mathbf{h}}(k-1)] + \mathbf{n}(k) \right\} \quad (3.26) \\
 &= \mathbf{u}_i^T(k) [\mathbf{h}_o - \hat{\mathbf{h}}(k-1)] + n_{f,i}(k)
 \end{aligned}$$

$$n_{f,i}(k) = \mathbf{f}_i^T(k-p+1+i) \mathbf{n}(k) \quad (3.27)$$

$$\mathbf{n}(k) = [n(k), n(k-1), \dots, n(k-p+1)]^T \quad (3.28)$$

である。(3.25)から、適応フィルタ係数 $\hat{\mathbf{h}}$ の修正量は、入力信号 $x(k)$ の i 次の予測残差信号 $\mathbf{u}_i(k), (i=0, 1, \dots, p-1)$ を入力とした p 個のNLMS法による修正量の和になることがわかる。

推定誤差 $\mathbf{e}(k)$ については、予測残差に対する推定誤差を要素に持つベクトル $\mathbf{e}_f(k)$ を

$$\mathbf{e}_f(k) = \mathbf{F}^T(k) \mathbf{e}(k) = [e_{f,p-1}(k), \dots, e_{f,1}(k), e_{f,0}(k)]^T \quad (3.29)$$

と定義すると

$$\mathbf{e}(k) = (\mathbf{F}^T)^{-1}(k) \mathbf{e}_f(k) \quad (3.30)$$

と $e_f(k)$ から合成される． F^{-1} の i 列を f_i^{-1} と表記すると (3.30) は図 3.3 に示す構成になる．同図右から入力する予測残差の推定誤差 $e_{f,i}(k)$ を集めてベクトル $e_f(k)$ とし，それを各フィルタ f_i^{-1} に入力する．フィルタの出力は予測残差 $e(k)$ となる．

以上，(3.25) から (3.30) が AP 法の複合 NLMS 法モデルである．図 3.4 はこのモデルに基づく AP 法の構造を示すものであり，AP 法が， p 個の NLMS 適応系と，入力信号系列 $\{x(k)\}$ とノイズ系列 $\{n(k)\}$ をフィルタリングする同一係数の予測フィルタバンク，そして，予測残差に対する推定誤差から入力 $X(k)$ に対する推定誤差を合成するフィルタバンクから構成されることを示している．

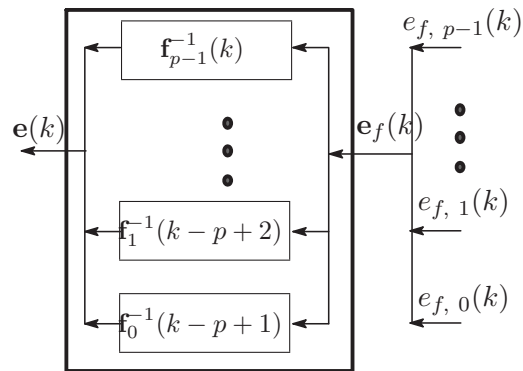


図 3.3: 前向き予測合成フィルタバンク

3.4 複合 NLMS 法モデルに基づく AP 法の収束特性

この節では，AP 法のフィルタ係数の二乗誤差の過渡および定常特性を，前章で示した複合 NLMS 法モデルに沿って近似する式を導出する．

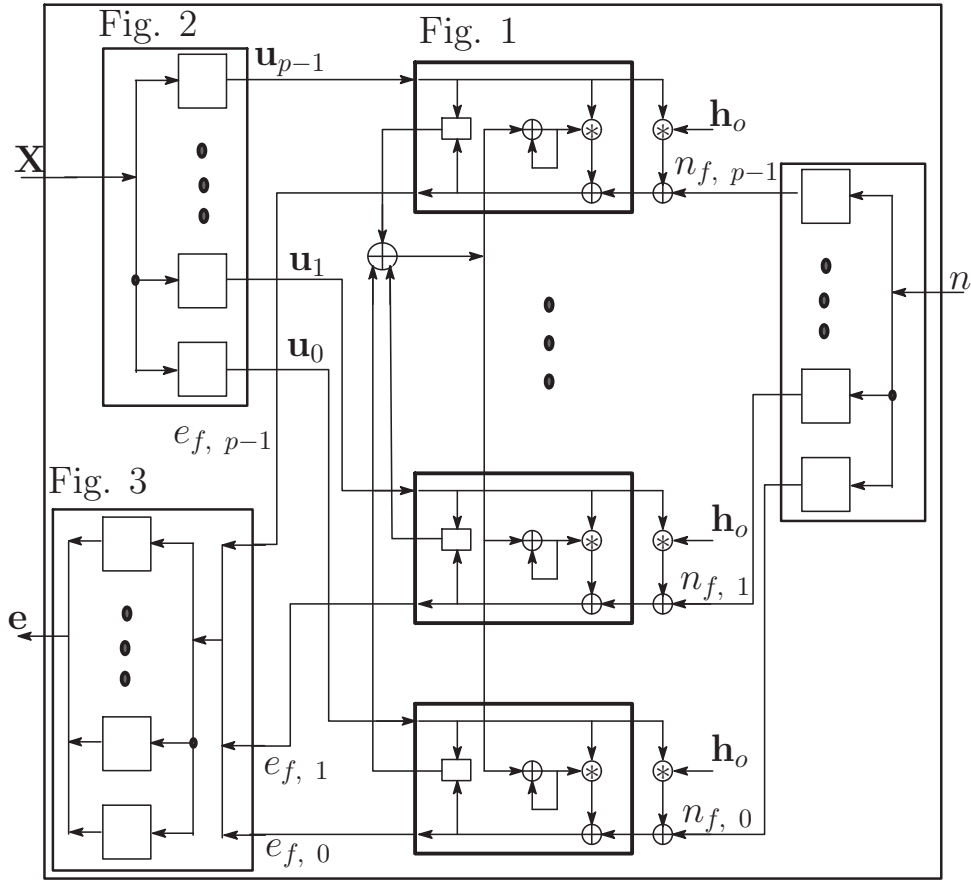


図 3.4: AP 法の複合 NLMS 法モデル．左右中央に縦に並ぶ太線の枠内が予測残差信号 u_0, u_1, \dots, u_{p-1} を入力とした p 個の NLMS 適応系．

AP 法における係数誤差 $\epsilon(k)$ の再帰更新式は，(3.25) と (3.26) を用いて，

$$\begin{aligned}
\epsilon(k) &= \hat{\mathbf{h}}(k) - \mathbf{h}_o \\
&= \hat{\mathbf{h}}(k-1) - \mathbf{h}_o + \mu \sum_{i=0}^{p-1} \frac{\mathbf{u}_i(k)}{\|\mathbf{u}_i(k)\|^2} e_{f,i}(k) \\
&= \hat{\mathbf{h}}(k-1) - \mathbf{h}_o + \mu \sum_{i=0}^{p-1} \frac{\mathbf{u}_i(k)}{\|\mathbf{u}_i(k)\|^2} \left\{ \mathbf{u}_i^T(k) [\mathbf{h}_o - \hat{\mathbf{h}}(k-1)] + n_{f,i}(k) \right\} \quad (3.31) \\
&= \left[\mathbf{I} - \mu \sum_{i=0}^{p-1} \frac{\mathbf{u}_i(k) \mathbf{u}_i^T(k)}{\|\mathbf{u}_i(k)\|^2} \right] [\hat{\mathbf{h}}(k-1) - \mathbf{h}_o] + \mu \sum_{i=0}^{p-1} \frac{\mathbf{u}_i(k)}{\|\mathbf{u}_i(k)\|^2} n_{f,i}(k) \\
&= \left[\mathbf{I} - \mu \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \right] \epsilon(k-1) + \sum_{i=0}^{p-1} \mu \tilde{\mathbf{u}}_i(k) n_{f,i}(k)
\end{aligned}$$

となる．ただし，

$$\mathbf{P}_{u,i}(k) = \tilde{\mathbf{u}}_i(k) \mathbf{u}_i^T(k) \quad (3.32)$$

$$\tilde{\mathbf{u}}_i(k) = \mathbf{u}_i(k) / \|\mathbf{u}_i(k)\|^2 \quad (3.33)$$

である．さらに，(3.22) で示した $\mathbf{u}_i(k)$ の直交性より

$$\mathbf{P}_{u,i}(k) \mathbf{P}_{u,j}(k) = \begin{cases} \mathbf{P}_{u,i}(k) & (i = j), \\ \mathbf{0} & (i \neq j) \end{cases} \quad (3.34)$$

なる関係があるので，(3.31) は，

$$\epsilon(k) = \prod_{i=0}^{p-1} [\mathbf{I} - \mu \mathbf{P}_{u,i}(k)] \epsilon(k-1) + \sum_{i=0}^{p-1} \mu \tilde{\mathbf{u}}_i(k) n_{f,i}(k) \quad (3.35)$$

と書き直すことができる．

過渡誤差

入力信号 $x(k)$ ，射影次数 p に対する係数二乗誤差を $J_{\text{AP},x,p}(k)$ と表わす．過渡状態では NLMS 法と同様に，(3.35) の右辺第二項が無視でき，同右辺第一項の $\epsilon(k-1)$ に関して再帰的に展開することで，

$$\begin{aligned} J_{\text{AP},x,p}(k) &\equiv E \left[\|\epsilon(k)\|^2 \right] \\ &\approx E \left[\left\| \prod_{m=1}^k \prod_{i=0}^{p-1} [\mathbf{I} - \mu \mathbf{P}_{u,i}(m)] \epsilon(0) \right\|^2 \right] \end{aligned} \quad (3.36)$$

で近似することができる．ここで，(3.36) の右辺を展開した式において， μ が十分 1 より小さく， $\mu^2 \|\mathbf{P}_{u,i}(m_i) \mathbf{P}_{u,j}(m_j) \epsilon(0)\|$ などの μ の二乗以上のノルムとその射影行列 $\mathbf{P}_{u,i}(m_i)$ と $\mathbf{P}_{u,j}(m_j)$ の順序を入れ替えたノルム $\mu^2 \|\mathbf{P}_{u,j}(m_j) \mathbf{P}_{u,i}(m_i) \epsilon(0)\|$ の差の総和が μ に比例するノルム $\mu \|\mathbf{P}_{u,i} \epsilon(0)\|$ の総和より十分小さいと仮定できるなら， i と m に関する二つのべき乗を交換することができ，

$$J_{\text{AP},x,p}(k) \approx E \left[\left\| \prod_{i=0}^{p-1} \prod_{m=1}^k [\mathbf{I} - \mu \mathbf{P}_{u,i}(m)] \epsilon(0) \right\|^2 \right] \quad (3.37)$$

と近似できる．

(3.37) と (3.10) で示された NLMS 法の過渡誤差と比較すると, $\prod_{m=1}^k [\mathbf{I} - \mu \mathbf{P}_{u,i}(m)] \epsilon(0)$ は, i 次の予測フィルタからの残差系列 $\{u_i(0), u_i(1), \dots, u_i(k)\}$ (以下 $\{u_i\}$ と略) を入力信号とし, NLMS 法によりフィルタを更新した場合の係数二乗誤差とみなすことができる. よって, (3.37) は, 時刻 k での (過渡) 係数二乗誤差が, まず, 図 3.4 での NLMS 適応系のどれかでフィルタを k 回だけ更新し, 次に, 別の NLMS 適応系で k 回だけフィルタを更新する, ということを全ての NLMS 適応系で行ったときのフィルタ係数二乗誤差として得られるということの意味している.

さらに, この解釈を元に, AP 法の係数二乗誤差 $J_{AP,x,p}(k)$ を予測残差系列 $\{u_i\}$, ($i = 0, 1, \dots, p-1$) に対する NLMS 法の係数二乗誤差 J_{NLMS,u_i} のみで表現することを考える.

(3.37) は途中まで収束している係数を初期値として, NLMS 法を適用することを意味するが, 一般には, NLMS 法の収束の過渡特性はフィルタ係数の初期係数誤差に依存する. このため, AP 法の時刻 k での係数二乗誤差 $J_{AP,x,p}(k)$ の $J_{AP,x,p}(0)$ に対する比は予測残差 $\{u_i\}$, ($i = 0, 1, \dots, p-1$) に対する時刻 k での係数二乗誤差 $J_{NLMS,u_i}(k)$ の $J_{NLMS,u_i}(0)$ に対する比の積にはならない. そこで, 係数誤差の初期値の影響を以下のように扱えと仮定する. つまり, 係数誤差が $J_{AP,x,i}(k)$ から開始する予測残差系列 $\{u_i\}$ を入力とした NLMS 法の係数二乗誤差 $J'_{NLMS,u_i}(k)$ の収束曲線は, 係数誤差の初期値を乱数で与えたときの $J_{NLMS,u_i}(k)$ の収束曲線を $J_{NLMS,u_i}(\kappa) = J_{AP,x,i}(k)$ となるように $J_{NLMS,u_i}(k)$ を時間軸方向に κ だけ平行移動したものになると仮定する.

例えば, 二次の AP 法の時刻 k での係数二乗誤差 $J_{AP,x,2}(k)$ は, 仮定より予測残差系列 $\{u_{2-1}\}$ を入力した係数誤差 $J_{NLMS,u_{2-1}}(k)$ が一次の AP 法の時刻 k での係数誤差 $J_{AP,x,2-1}(k)$ となる時刻 $J_{NLMS,u_{2-1}}^{-1}(J_{AP,x,2-1}(k))$ を開始点として, 一次の予測残差を入力とする NLMS 法の収束曲線を k 時刻進んだ $J_{NLMS,u_1}(J_{NLMS,u_{2-1}}^{-1}(J_{AP,x,2-1}(k)) + k)$ になる. 同様にして, 入力信号が x の p 次の AP 法の係数二乗誤差 (3.37) は,

$$J_{AP,x,p}(k) \approx J_{NLMS,u_{p-1}}(J_{NLMS,u_{p-1}}^{-1}(J_{AP,x,p-1}(k)) + k) \quad (3.38)$$

と近似され, (3.38) を再帰的に適用すると $J_{AP,x,p}$ が残差系列 $\{u_i\}$, ($i = 0, 1, \dots, p-1$) に

対する NLMS 法の係数二乗誤差 J_{NLMS,u_i} のみで表現される .

定常誤差

NLMS 法の場合と同様に (3.31) の両辺の二乗の期待値をとる .

$$\begin{aligned}
E \left[\|\epsilon(k)\|^2 \right] &= \\
& E \left[\left\| \left[\mathbf{I} - \mu \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \right] \epsilon(k-1) \right\|^2 \right] + \\
& E \left[\epsilon^T(k-1) \left[\mathbf{I} - \mu \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \right]^T \left[\sum_{i=0}^{p-1} \mu \tilde{\mathbf{u}}_i(k) n_{f,i}(k) \right] \right] + \\
& E \left[\left[\sum_{i=0}^{p-1} \mu \tilde{\mathbf{u}}_i(k) n_{f,i}(k) \right]^T \left[\mathbf{I} - \mu \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \right] \epsilon(k-1) \right] + \\
& E \left[\left\| \sum_{i=0}^{p-1} \mu \tilde{\mathbf{u}}_i(k) n_{f,i}(k) \right\|^2 \right]
\end{aligned} \tag{3.39}$$

さらに , (3.22) , (3.32) で示した $\mathbf{u}_i(k)$ の直交性より

$$\begin{aligned}
E \left[\|\epsilon(k)\|^2 \right] &= \\
& E \left[\epsilon^T(k-1) \left[\mathbf{I} - \mu(2-\mu) \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \right] \epsilon(k-1) \right] + \\
& E \left[2\mu(1-\mu) \sum_{i=0}^{p-1} \epsilon^T(k-1) \tilde{\mathbf{u}}_i(k) n_{f,i}(k) \right] + \\
& E \left[\sum_{i=0}^{p-1} \mu^2 n_{f,i}^2(k) / \|\mathbf{u}_i(k)\|^2 \right]
\end{aligned} \tag{3.40}$$

上式の右辺第二項にある期待値は , $\epsilon^T(k-1) \tilde{\mathbf{u}}_i(k)$ と $n_{f,i}(k)$ の独立性と $n_{f,i}(k)$ の期待値が 0 であることが仮定できると , 0 になる . また , 右辺の第三項は分子と分母が独立事象となので個別に期待値をとることができる . 以上を考慮すると , (3.40) は

$$\begin{aligned}
E \left[\|\epsilon(k)\|^2 \right] &= E \left[\|\epsilon(k-1)\|^2 \right] - \\
& \mu(2-\mu) E \left[\left\| \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \epsilon(k-1) \right\|^2 \right] + \\
& \sum_{i=0}^{p-1} \mu^2 E \left[n_{f,i}^2(k) \right] / E \left[\|\mathbf{u}_i(k)\|^2 \right]
\end{aligned} \tag{3.41}$$

となる．この式に，定常状態条件

$$E \left[\|\boldsymbol{\epsilon}(k)\|^2 \right] = E \left[\|\boldsymbol{\epsilon}(k-1)\|^2 \right] \quad (3.42)$$

を代入し，極限 $k \rightarrow \infty$ で表すと，(3.41) は，

$$\lim_{k \rightarrow \infty} E \left[\left\| \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \boldsymbol{\epsilon}(k-1) \right\|^2 \right] = \frac{\mu}{2-\mu} \lim_{k \rightarrow \infty} \sum_{i=0}^{p-1} \frac{E \left[n_{f,i}^2(k) \right]}{E \left[\|\mathbf{u}_i(k)\|^2 \right]} \quad (3.43)$$

となる．左辺の期待値の中は，(3.34) より，

$$\left\| \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \boldsymbol{\epsilon}(k-1) \right\|^2 = \sum_{i=0}^{p-1} \|\mathbf{P}_{u,i}(k) \boldsymbol{\epsilon}(k-1)\|^2 \quad (3.44)$$

とかける．予測残差系列 $\{u_i(k)\}$ を入力とした NLMS 法による係数誤差を $\epsilon_{u,i}(k)$ と表すと，付録の式 (A.6) の導出と同様に，

$$\lim_{k \rightarrow \infty} E \left[\|\mathbf{P}_{u,i}(k) \epsilon_{u,i}(k-1)\|^2 \right] L = \lim_{k \rightarrow \infty} E \left[\|\epsilon_{u,i}(k-1)\|^2 \right] \quad (3.45)$$

となる．

左辺は，NLMS 法における付録の式 (A.5) から (A.6) の導出と同様に，予測残差 $u_i(k)$ と係数誤差ベクトル $\boldsymbol{\epsilon}(k-1)$ の各要素が独立であり，また，係数誤差ベクトル $\boldsymbol{\epsilon}(k-1)$ の各要素も互いに独立で期待値が 0 であると仮定すると，予測残差系列 $\{u_i(k)\}$ を入力とした NLMS 法による係数誤差 $\epsilon_{u,i}(k)$ の定常状態二乗誤差が求まる．

$$\lim_{k \rightarrow \infty} E \left[\left\| \sum_{i=0}^{p-1} \mathbf{P}_{u,i}(k) \boldsymbol{\epsilon}(k-1) \right\|^2 \right] = \lim_{k \rightarrow \infty} E \left[\sum_{i=0}^{p-1} \|\epsilon_{u,i}(k-1)\|^2 \right] / L \quad (3.46)$$

ステップサイズ μ の絶対値が十分小さい場合には，毎時刻のフィルタ係数の更新が p 個ある予測残差ベクトル $\mathbf{u}_i(k)$ の方向へ等量行われていると仮定すると，

$$J_{\text{AP},x,p}(\infty) = \lim_{k \rightarrow \infty} E \left[\|\boldsymbol{\epsilon}(k-1)\|^2 \right] = \lim_{k \rightarrow \infty} E \left[\sum_{i=0}^{p-1} \|\epsilon_{u,i}(k-1)\|^2 \right] \quad (3.47)$$

となる．(3.47) に (3.46) の関係と (3.43) を代入することで，AP 法の定常誤差 $J_{\text{AP},x,p}(\infty)$ は，

$$J_{\text{AP},x,p}(\infty) = \frac{\mu}{2-\mu} \lim_{k \rightarrow \infty} \sum_{i=0}^{p-1} \frac{E \left[n_{f,i}^2(k) \right]}{E \left[\|\mathbf{u}_i(k)\|^2 \right]} / L \quad (3.48)$$

と書き直すことができる．ここで，もし，雑音 n が白色雑音であれば，

$$E [n_{f,i}^2(k)] = E [\|\mathbf{f}_i(k)\|^2] \sigma_n^2 \quad (3.49)$$

である．NLMS 法の定常誤差を表す (3.11) において， σ_x と σ_n をそれぞれ $E [\|\mathbf{u}_i(k)\|^2] / L$ と $E [n_{f,i}^2(k)]$ で置き換えたものを (3.48) に代入すると，

$$J_{AP,x,p}(\infty) = \sum_{i=0}^{p-1} J_{NLMS,u_i}(\infty) \quad (3.50)$$

とかける．この式は，AP 法の定常誤差が，予測残差を入力とした NLMS 法による二乗誤差の総和に等しいことを意味している．

3.5 計算機シミュレーションによる検証

複合 NLMS 法モデルに基づいた AP 法の過渡状態と定常状態の係数二乗誤差の表現の妥当性・有効性を検証するために，以下の条件でシステム同定のシミュレーションを行った．

- 最適フィルタ \mathbf{h}_o と適応フィルタ $\hat{\mathbf{h}}$ の要素数がともに $L = 128$.
- ステップサイズ $\mu = 1/16$.
- 入力信号としては，モデル試験用として四つのガウス雑音を，また拡声通信における実際の信号として男性の音声を用いた .

ガウス雑音は自己相関の強さと相関の次数の二つの観点から，自己相関の弱く周波数振幅特性が平坦な信号として白色ガウス雑音と相関次数の高い 8 次の信号，自己相関が強く周波数振幅特性に急峻なピークが現れる信号として相関次数が 2 次と 4 次の信号を用いた．白色信号以外の有色ガウス雑音は白色ガウス雑音を IIR フィルタに通すことで得た．IIR フィルタの係数は，2 次では，1, -0.98, 0.9, 4 次では，1, -1, 0.620097, -0.429807, 0.627265, 8 次では，1, -0.8, 0.7, -0.6, 0.5, -0.4, 0.3, -0.2, 0.1 である．これらの信号の周波数振幅スペクトルを図 3.5(a)-(d) に示す．

一方，音声信号は，8kHz でサンプリングした男性音声を用いた．その波形を図 9(a) に示す．

- 提案モデルにおいて，予測残差 $u_i(k)$ を求めるための予測係数 f_i の値はガウス雑音の場合は，入力信号の共分散行列 $\mathbf{X}^T(k)\mathbf{X}(k)$ の全区間に渡る平均値から求めた．音声の場合には，音声は定常信号ではないため 1 サンプル毎に，予測係数 $f_i(k)$ を求めその予測係数と入力信号系列 $\{x\}$ と雑音系列 $\{n\}$ から入力信号の予測残差 $u_i(k)$ と雑音の予測残差 $n_{f,i}(k)$ を求め，(3.48) 右辺の集合平均は適応フィルタ長 L サンプルの時間平均とした．
- AP 法で処理する際の (3.17) における δ は，過渡誤差のシミュレーションでは， $\delta = 0$ ，定常誤差のシミュレーションでは，信号がガウス雑音の場合は $\delta = 0.01\sigma_x^2 L$ ，音声の場合は $\delta = 0.03\sigma_x^2 L$ とした．

まず，ガウス雑音に対する結果について述べる．

過渡誤差

図 3.6(a)-(d) に四つの入力信号に対する係数二乗誤差の収束曲線を示す．射影次数は，白色信号，2 次の有色信号に対しては $p = 2$ ，4 次と 8 次の有色信号に対しては $p = 3$ である．実線は AP 法によるもので，破線は (3.37) に示した AP 法を NLMS 法を繰り返し適用するという近似式によるものである．実線と破線の差は数 dB 未満であり，AP 法を NLMS 法の繰り返しと近似することが妥当であることがわかる．

図 3.7(a)-(d) には四つの入力信号に対して，AP 法による係数二乗誤差の収束曲線を実線で示し， p 個の予測残差信号を入力としたときの NLMS 法の収束曲線を点線で示す（(a) では二つの点線が重なっている），また，(3.38) による AP 法の近似収束曲線を破線で示す．射影次数は前図 3.6 と同じである．(3.38) は予測残差に対する NLMS 法の収束曲線から AP 法による収束曲線の近似するというもので，この近似による破線で示した近似曲線は AP 法による実線と，(c) の 4 次の有色信号の結果以外では，前図 3.6 の (3.37) による近似と同様により近い近似を与えている．

図 3.7 からは，ステップサイズが同じならば NLMS 法より AP 法の方が収束速度が速いという AP 法の一般的特徴も理解される．図 3.7 における最も上の点線（最も収束の遅い点線）は 0 次の予測残差 $u_0(k)$ に対する収束曲線を表しており， $u_0(k) = x(k)$ なのでこれは $x(k)$ に対する NLMS 法の結果である．一方，予測信号ではスペクトルが平坦化されており NLMS 法であっても速い収束が得られる．AP 法の場合には，元の信号に対する NLMS 法に加えて，予測残差に対するより収束の速い NLMS 法によっても収束するため，NLMS 法より収束速度が速くなる．最終的な収束曲線（破線）は個々の予測残差信号の収束曲線（点線）をさらに下回った（速い速度で収束した）ものとなっている．

定常誤差

提案するモデルと従来法による定常誤差の近似結果の比較を行った．図 3.8(a)-(d) は定常誤差と射影次数の関係を示した図で，横軸は射影次数，縦軸は定常誤差の大きさを表している．(a)-(d) はそれぞれ白色雑音入力，2 次，4 次，8 次の有色信号入力に対する結果を示している．射影次数は， $p = 1, 2, 3, \dots, 10, 16, 32, 64, 128$ である．また，目的音 d に目的音より 30dB 小さい白色ガウス雑音を付加した．実線は各次数において AP 法を動作させた場合の定常誤差の値を結んだもので，次数の増加に伴って誤差が増加していることが分かる．また，「+」印は文献 [41] にある固有値分析に基づく定常誤差の近似式，

$$\mathcal{M}_p \approx \frac{\mu L \sigma_n^2}{2\lambda_{\min,p}} \quad (3.51)$$

による値を示す．ただし， $\lambda_{\min,p}$ は p 次の共分散行列 $E[\mathbf{X}^T(k)\mathbf{X}(k)]$ の最小固有値である．この近似値では次数 p が大きくなるにつれて不正確になっていくことが分かる．一方，「x」印は提案モデルに基づいた (3.48) による近似結果を示す．近似結果と AP 法による結果との差は $p = 128$ 以外では最大で数 dB 程度であり，良い近似を与えることがわかる．

$p = 128$ での AP 法による結果が提案した近似法と既存の近似法より大きくなる原因は，使用する入力信号の共分散行列の違いにある．近似法では $E[\mathbf{X}^T(k)\mathbf{X}(k)]$ と平均値を用いるため，共分散行列から求められる予測残差のパワーや固有値も平均的に値になるが，AP 法では瞬時の共分散行列を用いるため予測残差のパワーや固有値の分散が大きいため小さ

な固有値が現れることが多い。定常誤差は予測残差のパワーや固有値の逆数に比例すると考えられるので、 $p = 128 (= L)$ のように小さい予測残差パワーや固有値が現れる状況では AP 法の方が近似法に比べて定常誤差が大きくなる。

AP 法の一般的な特徴として射影次数あるいは入力信号の自己相関が大きいほど (図 3.5 で、ピークが急峻なほど) 定常誤差も大きくなるということが知られている。この AP 法の特徴が生じる理由を複合 NLMS 法モデルからみると、次のような解釈ができる。

まず、(3.48) において、射影次数が増加するに伴って加算される項が増加し誤差 $J_{AP,x,p}(\infty)$ が単調に増加することから理解できる。また、射影次数 (予測次数) が増加すると、予測残差 $\mathbf{u}_i(k)$ は白色化されほぼ一定の大きさとなる。よって、(3.48) 右辺の総和の中身 $E[n_{f,i}^2(k)] / (E[\|\mathbf{u}_i(k)\|^2] / L)$ は i によらず一定値となるため p に比例して $J_{AP,x,p}(\infty)$ が増加することが説明できる。このことは固有値を利用する従来の予測式 (3.51) では説明できない。

一方、入力信号の自己相関が大きい場合には、予測残差信号 $\mathbf{u}_i(k)$ は小さなものになる。すなわち、(3.48) 右辺の分母項が小さい値となるため、 $J_{AP,x,p}(\infty)$ が大きな値となる。このことより相関の大きい信号 (c) は信号 (a) や (d) より定常誤差が大きくなっていることが説明できる。

次に音声に対する結果を図 9 に示す。同図 (b) は係数二乗誤差の収束曲線を表す。射影次数は $p = 3$ である。実線は AP 法によるもので、破線は (3.37) に示した AP 法を NLMS 法の繰り返し適用するという近似式によるものである。実線と破線の差は数 dB 程度であり、音声に対しても AP 法を NLMS 法の繰り返しと近似することが妥当なことがわかる。図 9(c) は定常誤差と射影次数の関係を示した図であり、実線は AP 法の各次数における結果を結んだものであり、「×」印は提案した (3.48)、「+」印は固有値分析に基づく誤差の近似式による値を示す。なお、提案した (3.48) の計算は、先に実験条件のところで述べたようにサンプルごとに求めた入力信号 x と雑音 n の予測残差から求めたが、固有値分析に基づく方法においても、サンプル毎に入力信号の共分散行列 $\mathbf{X}^T(k)\mathbf{X}(k)$ の最小固有値から (3.51) を求め時間平均した値を用いている。実線と「×」印の差は最大で数 dB 程度であ

ることから提案モデルに基づいた近似結果がよい近似を与えることがわかる。また、「+」印は射影次数を上げると実線から離れ、不正確になっていくことがわかる。

以上、過渡状態と定常状態での係数二乗誤差の近似式と実際の値を比較した結果、近似式と実際の収束特性がよく一致し、複合 NLMS 法モデルが AP 法の収束特性を定量的に説明できることが分かった。また、射影次数や入力信号による収束特性の違いを複合 NLMS 法モデルは定性的に説明できることも分かった。

3.6 本章のまとめ

本章では、AP 法の収束特性の容易な理解を目的として、AP 法を入力信号の予測残差を入力とする複数の NLMS 法の集まりと解釈する複合 NLMS 法モデルを示した。次に、このモデルから、AP 法の過渡状態と定常状態の係数二乗誤差を NLMS 法のそれで表現する近似式を導出した。すなわち、過渡誤差は、入力信号の 0 次から射影次数 -1 次までの予測残差を入力とする NLMS 法を繰り返し適用することで近似されることが示され、AP 法が収束が NLMS に比べて速い理由とその程度が理解可能となった。また、定常誤差は各予測残差を入力とする NLMS 法で生じる定常誤差の和として近似した。白色信号と強い自己相関のある信号を含む四つのガウス雑音および音声入力信号に対する計算機シミュレーションの結果、提案法による定常状態二乗誤差の近似は従来の固有値を用いた解析と比べて良好な近似を行うことができ、射影次数や入力信号による収束特性の違いを定性的に説明できることが示された。

AP 法の収束特性をより簡単で、研究の進んでいる NLMS 法に基づいて解釈することで、これまでよりも単純明快な理解が可能となったといえる。

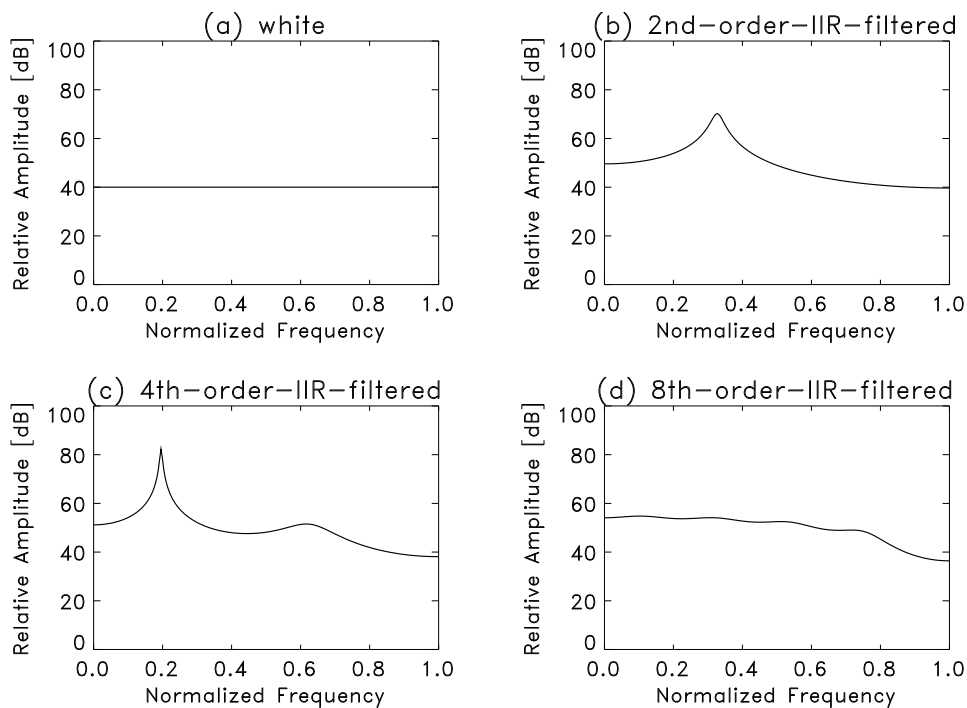


図 3.5: 試験信号のパワースペクトル . 縦軸は相対パワー [dB] , 横軸は規格化周波数 . (a) 白色
 ガウス雑音 , (b) 2 次の IIR フィルタ (係数:1, -0.98, 0.9) に白色ガウス雑音を通過させた信号 , (c) 4 次
 の IIR フィルタ (係数:1, -1., 0.620097, -0.429807, 0.627265) に白色ガウス雑音
 を通過させた信号 , 8 次 の IIR フィルタ (係数:1, -0.8, 0.7, -0.6, 0.5, -0.4, 0.3, -0.2, 0.1)
 に白色ガウス雑音を通過させた信号 .

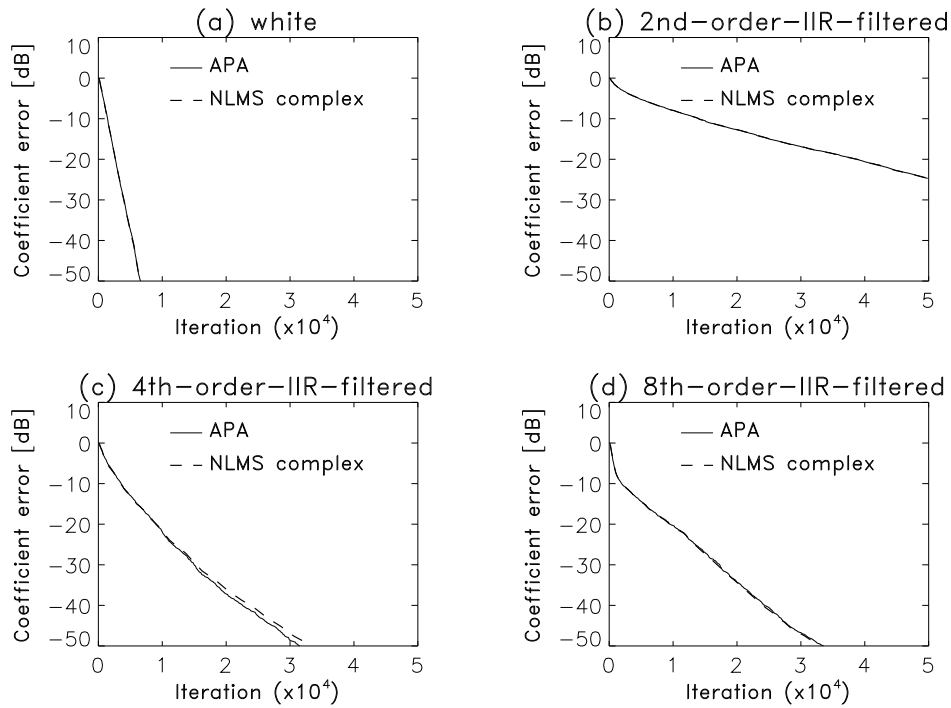


図 3.6: 過渡自乗誤差特性の (3.37) による近似 . (a) 白色雑音入力 , (b) 2 次の有色信号入力 , (c) 4 次の有色信号入力 , (d) 8 次の有色信号入力 . 実線は AP 法による収束曲線 , 破線は (3.37) に示した AP 法を NLMS 法の繰り返し適用として近似した収束曲線である . 実線と破線の差は数 dB 未満であり , AP 法を NLMS 法の繰り返しと解釈することが妥当であることがわかる . 射影次数は , (a) 白色信号入力 , (b) 2 次の有色信号に対しては $p = 2$, (c) 4 次と (d) 8 次の有色信号に対しては $p = 3$ である .

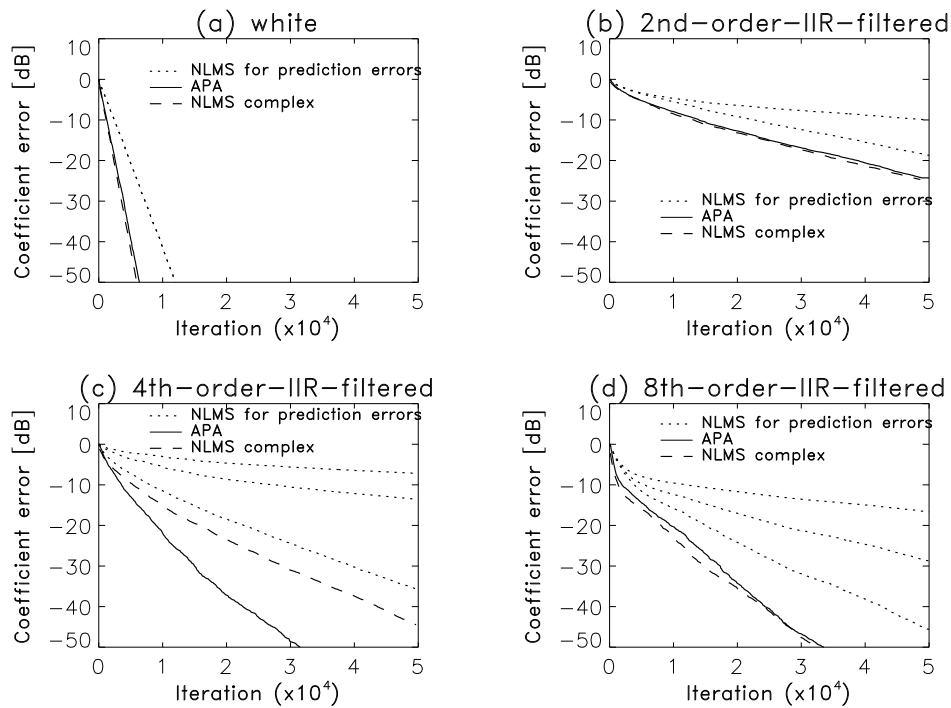


図 3.7: 過渡自乗誤差特性の (3.38) による近似 . (a) 白色雑音入力 , (b) 2 次の有色信号入力 , (c) 4 次の有色信号入力 , (d) 8 次の有色信号入力 . 実線は AP 法による収束曲線 , 点線は入力信号の予測残差を入力としたときの NLMS 法の収束曲線 , そして , 破線は (3.38) による AP 法の近似収束曲線である . (3.38) は予測残差に対する NLMS 法の収束曲線から AP 法による収束曲線の近似するというもので , この近似による破線の近似曲線は AP 法による実線と , (c) の 4 次の有色信号の結果以外では , 前図 3.6 の (3.37) による近似と同様により近似を与えている . 射影次数は , (a) 白色信号入力 , (b) 2 次の有色信号に対しては $p = 2$, (c) 4 次と (d) 8 次の有色信号に対しては $p = 3$ である .

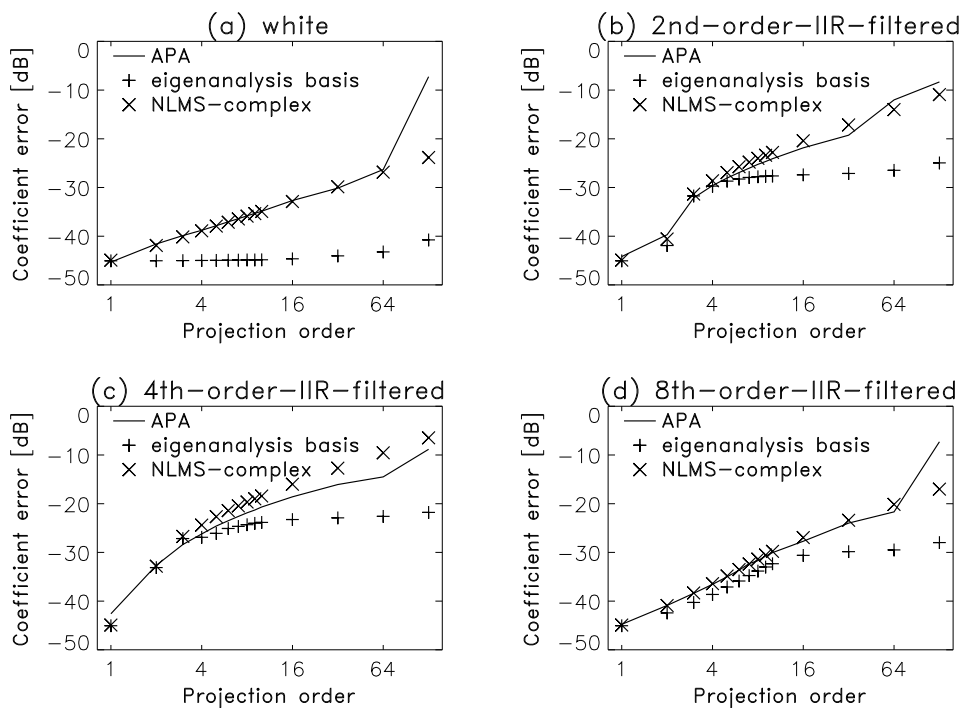


図 3.8: 射影次数と定常自乗誤差の関係の (3.48) による近似 . (a) 白色雑音入力 , (b) 2 次の有色信号入力 , (c) 4 次の有色信号入力 , (d) 8 次の有色信号入力 . 射影次数は , $p = 1, 2, 3, \dots, 10, 16, 32, 64, 128$. 実線は AP 法を動作させて得られた結果 (真値) . 「+」は文献 [41] にある固有値分析に基づく近似結果 , 「x」は提案手法による近似結果 . これらの結果は AP 法の一般的な特徴である射影次数あるいは入力信号の自己相関が大きければ大きいほど定常誤差も大きくなるという点をよく表わしている . そして , 推定値と AP 法による結果の差はほとんどの場合で数 dB 程度であり , 複合 NLMS モデルに基づく (3.48) が良い近似を与えることがわかる .

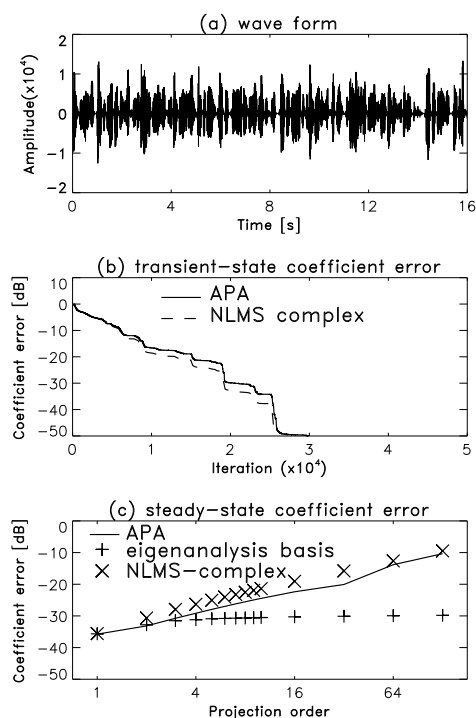


図 3.9: 音声信号とそれに対する近似結果 . (a) 音声波形 , (b) 過渡二乗誤差特性の (3.37) による近似の検証 , (c) 射影次数と定常状態係数二乗誤差の関係の近似結果 , 記号の意味は , 図 3.6 と図 3.8 を参照 . (b) では射影次数は $p = 3$ である . 実線は AP 法によるもので , 破線は (3.37) に示した AP 法を NLMS 法の繰り返し適用するという近似式によるものである . 実線と破線の差は数 dB 程度であり , 音声に対しても AP 法を NLMS 法の繰り返しと近似することが妥当なことがわかる . (c) で実線は AP 法の各次数における結果を結んだものであり , 「 x 」 印は提案した (3.48) , 「 + 」 印は固有値分析に基づく誤差の近似式による値を示す . 実線と 「 x 」 印の差は最大で数 dB 程度であることから提案モデルに基づいた近似結果がよい近似を与えることがわかる . また 「 + 」 印は射影次数を上げると実線から離れ , 不正確になっていくことがわかる .

第4章 高速アフィン射影法

4.1 はじめに

適応フィルタリングはエコーキャンセラや、騒音制御など、多くの分野で用いられている重要な基盤技術である [28]。様々な適応フィルタの手法のなかでも、LMS (Least Mean Squares) 法と NLMS (Normalized LMS) 法 [29] は、簡単で、演算量が少ないことから広く用いられてきた。LMS 法と NLMS 法は演算量は少ないが、音声のような有色、つまり、自己相関の強い信号に対しては収束が遅い。RLS (Recursive Least Squares) 法では有色信号でのこの問題は起きないが、演算量はその（演算量の意味で）高速版でも四倍と多い [14, 15]。

近年、アフィン射影法（略して、AP 法あるいは射影法） [4] が注目を集めている。この適応フィルタリング手法は、NLMS 法と RLS 法の間の特徴をもっている。つまり、演算量は RLS 法より少なく、音声信号のように低次の AR モデル化ができる有色信号に対しては、NLMS 法に比べ、格段に収束が速い。AP 法は、しかしながら、依然として、NLMS 法よりは演算量が多いことが問題として残っていた。

AP 法の演算量低減の努力は幾つも行われてきており、それらは、2つの方法に分類することができる。どちらの方法も同じ式に基づいて修正量を計算するが、フィルタを更新する間隔が異なっている。一つ目の方法は、ブロック演算を適用するもので、修正はブロック長毎に行われる [18, 30, 31, 11]。これらのブロック処理手法では、演算量は低減するものの収束速度が劣化することが報告されている [18]。もう一方の方法は、元の逐次更新の AP 法の無駄な演算を取り除く方法で、元の AP 法の特徴を保持している。本章では、後者の方法に従い、AP 法の演算量を劇的に低減した高速 AP 法と呼ぶ適応フィルタリング

手法を提案する．演算量の低減は，二つの部分からなり，そのうち，一つは，丸山によって最も簡単な場合について取り上げられている [32] ．

本章第 2 節では，まず，AP 法を説明し，次に第 3 節で提案する演算量低減法を導出する．

4.2 AP 法

第 2 章 で示した AP 法のアルゴリズムを再掲する．

$$\hat{y}(k) = \mathbf{x}^T(k) \hat{\mathbf{h}}(k-1) \quad (2.3)$$

$$\mathbf{e}(k) = [y(k) - \hat{y}(k), (1 - \mu(k-1))\mathbf{e}_{p-1}^T(k-1)]^T \quad (2.37)$$

$$\mathbf{g}(k) = \mathbf{R}_p^{-1}(k)\mathbf{e}_p(k) \quad (2.32)$$

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu\mathbf{X}_{L,p}\mathbf{g}(k) \quad (2.33)$$

ただし，

$$\mathbf{X}_{L,p}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-p+1)] \quad (4.1)$$

$$\mathbf{R}_p(k) = \mathbf{X}_{L,p}^T(k)\mathbf{X}_{L,p}(k) \quad (4.2)$$

AP 法の演算量を見積もると，まず，(2.3) により推定出力を求めるのに L ，誤差ベクトルを更新するのに p ，(2.32) によりプレフィルタベクトルを求めるのに $O(p^3)$ ，そして，(2.33) によるプレフィルタリングとフィルタの修正に pL の演算が必要であるから，合計して約 $(p+1)L + O(p^3)$ の演算が必要になる．この演算量は NLMS 法の場合の $2L$ に比べて多く，AP 法が実際的ではないと考えられていた．

4.3 高速 AP 法

この節では高速な AP 法を導出する．この手法によれば，演算量は $2L + 20p$ に削減される．演算量低減は二つの部分からなり，一つは，プレフィルタベクトル $\mathbf{g}(k)$ の再帰計算であり，もう一つは $\mathbf{g}(k)$ による $\mathbf{X}(k)$ のプレフィルタリングに関するものである．

4.3.1 プレフィルタベクトル $\mathbf{g}(k)$ の再帰計算

$\mathbf{g}(k)$ の計算には $O(p^3)$ の積和演算が必要となる。 p が小さい場合には、この量はフィルタ長 L より小さく無視できる。しかし、大きな p を使う場合には、フィルタ長と同程度かそれ以上になるので無視できない演算量となる。本項では、プレフィルタベクトル $\mathbf{g}(k)$ が $15p$ の演算量で再帰的に求められることを示す。

(4.1) と (4.2) より、 $\mathbf{R}_p(k)$ は $p \times p$ の行列で次式のように書き直すことができる。

$$\mathbf{R}_p(k) = \sum_{i=1}^L \mathbf{x}_p(k+1-i) \mathbf{x}_p^T(k+1-i) \quad (4.3)$$

ただし、

$$\mathbf{x}_p(k) = [x(k), x(k-1), \dots, x(k-p+1)]^T \quad (4.4)$$

同様に、 $(p-1)$ 次の共分散行列 $\mathbf{R}_{p-1}(k)$ は次式で定義できる。

$$\mathbf{R}_{p-1}(k) = \sum_{i=1}^L \mathbf{x}_{p-1}(k+1-i) \mathbf{x}_{p-1}^T(k+1-i) \quad (4.5)$$

ただし、

$$\mathbf{x}_{p-1}(k) = [x(k), x(k-1), \dots, x(k-p+2)]^T \quad (4.6)$$

これら 2 つの共分散行列の逆行列には (3.24) で示したコレスキー分解を行列の次数に関して再帰的に表現した次の二つの関係が成立する [12]。

$$\mathbf{R}_p^{-1}(k) = \begin{bmatrix} 0 & \mathbf{0}_{p-1}^T \\ \mathbf{0}_{p-1} & \mathbf{R}_{p-1}^{-1}(k-1) \end{bmatrix} + \frac{\mathbf{a}(k) \mathbf{a}^T(k)}{F(k)}, \quad (4.7)$$

$$\mathbf{R}_p^{-1}(k) = \begin{bmatrix} \mathbf{R}_{p-1}^{-1}(k) & \mathbf{0}_{p-1} \\ \mathbf{0}_{p-1}^T & 0 \end{bmatrix} + \frac{\mathbf{b}(k) \mathbf{b}^T(k)}{B(k)} \quad (4.8)$$

ここで、 $\mathbf{a}(k)$ 、 $\mathbf{b}(k)$ 、 $F(k)$ と $B(k)$ は、それぞれ、線形前向き予測フィルタ、線形後ろ向き予測フィルタ、最小事後前向き推定誤差自乗和、そして、最小事後後ろ向き推定誤差自乗和を表わし、次式の augmented 正規方程式により関係付けられる。

$$\mathbf{R}_p(k) \mathbf{a}(k) = [F(k), 0, \dots, 0]^T \quad (4.9)$$

$$\mathbf{R}_p(k)\mathbf{b}(k) = [0, \dots, 0, B(k)]^T \quad (4.10)$$

この逆行列の性質 (4.7) と残差ベクトルの更新式 (2.37) をプレフィルタの定義式 (2.32) に代入すると、次式が得られる。

$$\mathbf{g}(k) = (1 - \mu(k-1)) \begin{bmatrix} 0 \\ \mathbf{f}(k-1) \end{bmatrix} + \frac{\mathbf{a}^T(k)\mathbf{e}(k)}{F(k)}\mathbf{a}(k) \quad (4.11)$$

ただし、

$$\mathbf{f}(k-1) = \mathbf{R}_{p-1}^{-1}(k-1)\mathbf{e}_{p-1}(k-1) \quad (4.12)$$

同様に、(2.37)、(2.32) と (4.8) から次式を得る。

$$\begin{bmatrix} \mathbf{f}(k) \\ 0 \end{bmatrix} = \mathbf{g}(k) - \frac{\mathbf{b}^T(k)\mathbf{e}(k)}{B(k)}\mathbf{b}(k) \quad (4.13)$$

(4.11) と (4.13) の右辺第 2 項で、分母が小さくなると、有限精度での演算では誤差が増加するので、ある小さな正数 δ を分母 $F(k)$ と $B(k)$ に足すことが望ましい。すなわち、

$$\mathbf{g}(k) = (1 - \mu(k-1)) \begin{bmatrix} 0 \\ \mathbf{f}(k-1) \end{bmatrix} + \frac{\mathbf{a}^T(k)\mathbf{e}(k)}{F(k) + \delta}\mathbf{a}(k) \quad (4.14)$$

$$\begin{bmatrix} \mathbf{f}(k) \\ 0 \end{bmatrix} = \mathbf{g}(k) - \frac{\mathbf{b}^T(k)\mathbf{e}(k)}{B(k) + \delta}\mathbf{b}(k) \quad (4.15)$$

(4.14) と (4.15) により $\mathbf{g}(k)$ が再帰的に射影次数 $O(p)$ の演算で更新されることができる。つまり、(4.14) より $\mathbf{g}(k)$ は $\mathbf{f}(k-1)$ から導出され、次の時刻で必要となる $\mathbf{f}(k)$ は (4.15) により $\mathbf{g}(k)$ から導出される。また、(4.14) と (4.15) での積和演算回数は $5p$ であり、 $\mathbf{a}(k)$ 、 $\mathbf{b}(k)$ 、 $F(k)$ と $B(k)$ は、2.5.4 節で述べた方形窓の Fast Transversal Filter (FTF) 法を用いて約 $10p$ の積和演算で求められる [1] (FTF 法は数値的に不安定であることが知られている。一つの回避策は数値誤差のフィードバックを導入することであり、指数窓の FTF 法の場合について述べられている [15] の方法は方形窓版の FTF にも適用できる。) このように、提案法は約 $15p$ の演算で $\mathbf{g}(k)$ を更新でき、従来の $O(p^3)$ に比べ、演算量を大きく低減する。

4.3.2 $\mathbf{X}(k)$ の $\mathbf{g}(k)$ によるプレフィルタリング

プレフィルタリングの演算量低減のために，変形フィルタベクトル $\mathbf{z}(k)$ を導入する．この変形フィルタベクトルを利用する手法は最初に丸山によって $p = 2$ の場合について示されたが [32]，本項では，丸山のこの手法を一般的な場合に拡張する．

(2.6) より， $\hat{\mathbf{h}}(k)$ は次式のようにかける．

$$\begin{aligned}\hat{\mathbf{h}}(k) &= \mu(k)\Delta\hat{\mathbf{h}}(k) + \mu(k-1)\Delta\hat{\mathbf{h}}(k-1) + \cdots + \hat{\mathbf{h}}(0) \\ &= \sum_{i=1}^k \mu(k+1-i)\Delta\hat{\mathbf{h}}(k+1-i) + \hat{\mathbf{h}}(0)\end{aligned}\quad (4.16)$$

(4.1) を

$$\Delta\hat{\mathbf{h}}(k) = \mathbf{X}_{L,p}(k)\mathbf{g}(k)\quad (4.17)$$

に代入すると， $\Delta\hat{\mathbf{h}}(k)$ は，

$$\Delta\hat{\mathbf{h}}(k) = \sum_{j=1}^p g_j(k)\mathbf{x}(k-j+1)\quad (4.18)$$

とかける．ただし， $g_j(k)$ ， $(j = 1, 2, \dots, p)$ はベクトル $\mathbf{g}(k)$ の j 番目の要素である．(4.16)

と (4.17) より，次式のように， $\hat{\mathbf{h}}(k)$ が過去の入力信号ベクトルの線形和としてかける．

$$\begin{aligned}\hat{\mathbf{h}}(k) &= \sum_{i=1}^k \mu(k+1-i) \sum_{j=1}^p g_j(k+1-i)\mathbf{x}(k-i-j+2) + \hat{\mathbf{h}}(0) \\ &= \mu(k)(g_1(k)\mathbf{x}(k) + g_2(k)\mathbf{x}(k-1) + \cdots + g_p(k)\mathbf{x}(k-p+1)) + \\ &\quad \mu(k-1)(g_1(k-1)\mathbf{x}(k-1) + g_2(k-1)\mathbf{x}(k-2) + \\ &\quad \cdots + g_p(k-1)\mathbf{x}(k-p)) + \\ &\quad \vdots \\ &\quad \mu(1)(g_1(1)\mathbf{x}(1) + g_2(1)\mathbf{x}(0) + \cdots + g_p(1)\mathbf{x}(2-p)) + \hat{\mathbf{h}}(0) \\ &= \sum_{i=1}^{k+p-1} s_i(k)\mathbf{x}(k+1-i) + \hat{\mathbf{h}}(0)\end{aligned}\quad (4.19)$$

ここで,

$$\begin{aligned}
s_1(k) &= \mu(k)g_1(k) \\
s_2(k) &= \mu(k)g_2(k) + \mu(k-1)g_1(k-1) \\
s_3(k) &= \mu(k)g_3(k) + \mu(k-1)g_2(k-1) + \mu(k-2)g_1(k-2) \\
&\vdots \\
s_p(k) &= \mu(k)g_p(k) + \mu(k-1)g_{p-1}(k-1) + \cdots + \\
&\quad \mu(k-p+1)g_1(k-p+1) \\
s_{p+1}(k) &= \mu(k-1)g_p(k-1) + \mu(k-2)g_{p-1}(k-2) + \cdots + \\
&\quad \mu(k-p)g_1(k-p) \\
s_{p+2}(k) &= \mu(k-2)g_p(k-2) + \mu(k-3)g_{p-1}(k-3) + \cdots + \\
&\quad \mu(k-p-1)g_1(k-p-1) \\
&\vdots \\
s_{k+p-2}(k) &= \mu(2)g_p(2) + \mu(1)g_{p-1}(1) \\
s_{k+p-1}(k) &= \mu(1)g_p(1)
\end{aligned} \tag{4.20}$$

$s_i(k)$ を平滑化プレフィルタ係数と呼ぶことにする. (4.20) からわかるように, $s_i(k)$ は次式のように更新される.

$$\begin{aligned}
\text{for } i = 1 : & \quad s_1 = \mu(k)g_1(k) \\
\text{for } i = 2, \dots, p : & \quad s_i(k) = s_{i-1}(k-1) + \mu(k)g_i(k) \\
\text{for } i = p+1, \dots, k+p-1 : & \quad s_i(k) = s_{i-1}(k-1)
\end{aligned} \tag{4.21}$$

実際に更新のための計算が必要なのは $s_i(k)$, $i = 1, \dots, p$ である. それらをベクトル形式でかくと,

$$\mathbf{s}(k) = \begin{bmatrix} 0 \\ \mathbf{s}_{p-1}(k-1) \end{bmatrix} + \mu(k)\mathbf{g}(k) \tag{4.22}$$

となる. ただし,

$$\mathbf{s}(k) = [s_1(k), s_2(k), \dots, s_p(k)]^T \tag{4.23}$$

$$\mathbf{s}_{p-1}(k-1) = [s_1(k-1), s_2(k-1), \dots, s_{p-1}(k-1)]^T \tag{4.24}$$

ここで, (4.19) にあるような $i = 1$ から $i = k + p - 1$ の和に代えて, $i = p$ から $k + p - 1$ までの和をとり, 次式で変形フィルタベクトルを定義する.

$$\mathbf{z}(k) = \sum_{i=p}^{k+p-1} s_i(k)\mathbf{x}(k+1-i) + \hat{\mathbf{h}}(0) \quad (4.25)$$

係数 $s_i(k)$, ($i = p, \dots, k + p - 1$) が時刻 $k + 1$ 以降更新されないので, この線形結合は時刻 $k + 1$ 以降で, 定数となる. (2.32) と (4.20) より, フィルタが収束し推定誤差が小さくなるにしたがって, $g_i(k)$ ひいては $s_i(k)$ ($i = 1, \dots, p$) が零に近づくので, 変形フィルタベクトルはフィルタベクトル $\hat{\mathbf{h}}(k)$ に漸近すると期待される.

ベクトル $\mathbf{z}(k)$ は以下に示すように, 再帰的に L 回の積和演算で更新できる.

$$\begin{aligned} \mathbf{z}(k) &= \sum_{i=p+1}^{k+p-1} s_i(k)\mathbf{x}(k+1-i) + s_p(k)\mathbf{x}(k+1-p) + \hat{\mathbf{h}}(0) \\ &= \mathbf{z}(k-1) + s_p(k)\mathbf{x}(k+1-p) \end{aligned} \quad (4.26)$$

(4.25) から, (4.19) は, 次式のように書き直すことができる.

$$\hat{\mathbf{h}}(k) = \mathbf{z}(k) + \sum_{i=1}^{p-1} s_i(k)\mathbf{x}(k+1-i) \quad (4.27)$$

従って, $\hat{\mathbf{h}}(k)$ を $\mathbf{z}(k)$ を介して計算すると, pL 回の演算が必要となる. この量は, (4.17) による従来の AP 法と同じであり, 変形フィルタベクトル $\mathbf{z}(k)$ を導入したことが効率的ではないように見える. しかし, 適応フィルタを使用する多くの場合では, $\hat{\mathbf{h}}(k)$ が毎時刻得られなくとも, 未知系の推定出力 $\hat{y}(k)$ が得られるだけで十分である.

以下に, 未知系の推定出力 $\hat{y}(k)$ が $\hat{\mathbf{h}}(k)$ ではなく, $\mathbf{z}(k)$ を用いて計算できることを示す.

(4.27) で, k を $k - 1$ として一時刻戻すことで, 次式を得る.

$$\begin{aligned} \hat{\mathbf{h}}(k-1) &= \mathbf{z}(k-1) + \sum_{i=1}^{p-1} s_i(k-1)\mathbf{x}(k-i) \\ &= \mathbf{z}(k-1) + [\mathbf{x}(k-1), \mathbf{x}(k-2), \dots, \mathbf{x}(k-p+1)]\mathbf{s}_{p-1}(k-1) \end{aligned} \quad (4.28)$$

(4.28) を用いて, $y(k)$ を計算すると, (2.3) は次式のようにかける.

$$\hat{y}(k) = \mathbf{x}^T(k)\mathbf{z}(k-1) + \mathbf{r}_{p-1}^T(k)\mathbf{s}_{p-1}(k-1) \quad (4.29)$$

ここで, $\mathbf{r}_{p-1}(k)$ は $(p-1)$ 次のベクトルで, その要素は $x(k)$ の共分散である.

$$\mathbf{r}_{p-1}(k) = [\mathbf{x}^T(k)\mathbf{x}(k-1), \mathbf{x}^T(k)\mathbf{x}(k-2), \dots, \mathbf{x}^T(k)\mathbf{x}(k-p+1)]^T \quad (4.30)$$

(4.29) により, $\hat{y}(k)$ は変形係数ベクトル $\mathbf{z}(k-1)$, 共分散ベクトル $\mathbf{r}_{p-1}(k)$ と平滑化プレフィルタベクトル $\mathbf{s}_{p-1}(k)$ を用いて, $L+p-1$ 回の積和演算で計算できる.

共分散ベクトル $\mathbf{r}_{p-1}(k)$ の i 番目 ($i = 1, \dots, p-1$) の要素が,

$$\begin{aligned} \mathbf{x}^T(k)\mathbf{x}(k-i) &= \sum_{n=0}^{L-1} x(k-n)x(k-i-n) \\ &= \sum_{n=0}^{L-1} x(k-1-n)x(k-1-i-n) + \\ &\quad x(k)x(k-i) - x(k-L)x(k-i-L) \\ &= \mathbf{x}^T(k-1)\mathbf{x}(k-1-i) + x(k)x(k-i) - \\ &\quad x(k-L)x(k-i-L) \end{aligned} \quad (4.31)$$

とかけるので, $\mathbf{r}_{p-1}(k)$ は次式で更新できる.

$$\mathbf{r}_{p-1}(k) = \mathbf{r}_{p-1}(k-1) + x(k)\mathbf{x}_{p-1}(k-1) - x(k-L)\mathbf{x}_{p-1}(k-L-1) \quad (4.32)$$

このように, 共分散ベクトル $\mathbf{r}_{p-1}(k)$ は再帰的に $2(p-1)$ 回の積和演算で更新される. この項では, 変形フィルタベクトル $\mathbf{z}(k)$ を導入した. pL 回の演算が必要であった (2.33) のプレフィルタリングと適応フィルタの更新が L 回の演算しか必要のない (4.26) による変形フィルタベクトルの更新で置き換えられた. 推定出力の計算は変形フィルタベクトル $\mathbf{z}(k-1)$, 共分散ベクトル $\mathbf{r}_{p-1}(k)$ と平滑化プレフィルタベクトル $\mathbf{s}_{p-1}(k)$ を用いて, (4.29) により計算される.

提案法の手順をアルゴリズム 4.1 に, 大まかな演算量とともに示す. 総演算量は約 $2L+20p$ であり, 従来法の $(p+1)L + O(p^3)$ に比べはるかに少ない. 図 4.1 は, $L = 1024$ のときの提案法の演算量 $(2L+20p)$, 従来法の演算量 $(p+1)L + p^3/6 + p^2$ と安定化 FTF 法 (SFTF 法) の演算量 $(8L)$ を比較している. 提案法は, p が大きくなるにつれ従来法との演算量の差が拡大し, 提案法が高効率であることがわかる. また, 安定化 FTF 法に比べても, 射影

次数が 50 次という高次においても少ない．例えば，第 2 章での音声に対する計算機シミュレーションの条件である $L = 1024$ と $p = 48$ では，従来法の演算量は 70912 であるが提案した高速 AP 法では 3008 と約 1/20 に低減される．

	演算	式番号	演算量
初期化	$\mathbf{r}_{p-1}(0) = \mathbf{X}_{L,p}^T(0)\mathbf{x}(0), \mathbf{s}(0) = \mathbf{0}, \mathbf{e}(0) = \mathbf{y}(0)$		
1)	$\mathbf{r}_{p-1}(k) = \mathbf{r}_{p-1}(k-1) + x(k)\mathbf{x}_{p-1}(k-1) - x(k-L)\mathbf{x}_{p-1}(k-L-1)$	(4.32)	$2p$
2)	$\hat{y}(k) = \mathbf{x}^T(k)\mathbf{z}(k-1) + \mathbf{r}_{p-1}^T(k)\mathbf{s}_{p-1}(k-1)$	(4.29)	$L + p$
3)	$\mathbf{e}(k) = [y(k) - \hat{y}(k), (1 - \mu(k-1))\mathbf{e}_{p-1}^T(k-1)]^T$	(2.37)	p
4)	方形窓 FTF 法等で $\mathbf{a}(k), F(k), \mathbf{b}(k), B(k)$ を計算		$10p$
5)	$\mathbf{g}(k) = (1 - \mu(k-1)) \begin{bmatrix} 0 \\ \mathbf{f}(k-1) \end{bmatrix} + \frac{\mathbf{a}^T(k)\mathbf{e}(k)}{F(k) + \delta} \mathbf{a}(k)$	(4.14)	$3p$
6)	$\begin{bmatrix} \mathbf{f}(k) \\ 0 \end{bmatrix} = \mathbf{g}(k) - \frac{\mathbf{b}^T(k)\mathbf{e}(k)}{B(k) + \delta} \mathbf{b}(k)$	(4.15)	$2p$
7)	$\mathbf{s}(k) = \begin{bmatrix} 0 \\ \mathbf{s}_{p-1}(k-1) \end{bmatrix} + \mu(k)\mathbf{g}(k)$	(4.22)	p
9)	$\mathbf{z}(k) = s_p(k)\mathbf{x}(k+1-p) + \mathbf{z}(k-1)$	(4.26)	L

アルゴリズム 4.1: 高速 AP 法

4.4 本章の結論

提案した手法により，AP 法の演算量は $(p+1)L + O(p^3)$ から $2L + 20p$ (L はフィルタ長， p は射影次数) に低減した．低減化は二つの部分からなり，一つは従来法で $O(p^3)$ 要していたプレフィルタの導出を再帰計算により約 $15p$ で行うというものであり，他方は変形フィルタベクトルを導入することにより，従来， pL の演算を要したフィルタの更新とプレフィルタリングを $L + 3p$ の変形フィルタベクトルの更新に置換えるというものである．

第 2 章の計算機シミュレーションでみたように，高次の AP 法は FTF (高速 RLS) 法と同程度の収束速度をもつことから，提案した高速 AP 法により，音声のような自己相関の

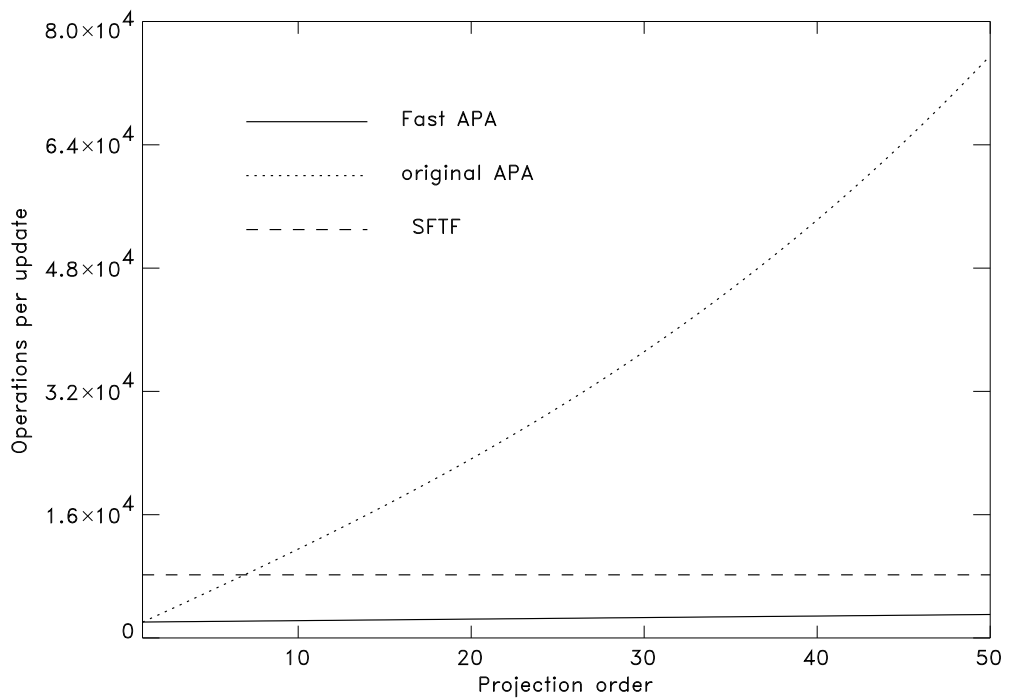


図 4.1: AP 法と高速 AP 法の演算量の比較．フィルタ長 $L = 1024$ のときの提案法の演算量 $(2L + 20p)$ ，従来法の演算量 $(p + 1)L + p^3/6 + p^2$ と安定化 FTF 法の演算量 $(8L)$ として比較．提案法は，射影次数 p が大きくなるにつれ従来法との演算量の差が拡大し，提案法が高効率であることがわかる．また，安定化 FTF 法 (SFTF 法) に比べても，射影次数が 50 次という高次においても少ない．

強い信号に対しても，FTF 法以下の演算で同等の収束速度を得ることが可能となったといえる．

第5章 逐次等価ブロック高速アフィン射影法

5.1 はじめに

ブロック処理は演算量低減の有効な方法であり，これまでにLMS法，アフィン射影(AP)法 [4] に適用されてきた．しかしながら，これらの手法では，フィルタの更新間隔が伸びてしまうために毎時刻フィルタを更新する元の手法に比べると収束速度が低下するという問題点があった．この問題に対し，近年，“exact”ブロック処理がLMS法や高速RLS法等に導入されている [19, 33, 34, 35]．このblock exact ブロック処理手法は，演算量を高速FIRフィルタリング手法により低減し，かつ，出力については元の逐次法の出力からの補正項を計算することで，元の逐次法と同じ収束速度を持たせるものである．逐次法と等価なブロック処理という意味で，本論では逐次等価ブロック処理と呼ぶことにする．

本章では，この逐次等価ブロック処理形式の高速AP法について述べる．他の逐次等価ブロック法には見られなかった，適応フィルタの出力の計算とフィルタの更新に異なるブロック長を選択することを許すことで，アルゴリズムの利便性を向上させている．以下，第2節では提案法導出の準備として，AP法，そのブロック処理版と高速FIRフィルタリング手法を概説し，第3節において提案する逐次等価ブロック処理版の高速AP法を導出し，さらに，既存の手法と演算量の比較を行う．最後に第4節でフィルタ長とブロック処理に伴う出力の遅延時間の関係について示す．

5.2 準備

この節では，高速 AP 法と高速 FIR フィルタリング (FFF) について概説する．他の章と同様に，非ボールド体はスカラーをボールド体の小文字はベクトルを，ボールド体の大文字は行列を表す．

5.2.1 AP 法

第 2 章で示したように，AP 法は次式により実行される．

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{X}^T(k)\hat{\mathbf{h}}(k-1) \quad (5.1)$$

$$\mathbf{g}(k) = \mathbf{R}^{-1}(k)\mathbf{e}(k) \quad (5.2)$$

$$\Delta\hat{\mathbf{h}}(k) = \mathbf{X}(k)\mathbf{g}(k) \quad (5.3)$$

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu(k)\Delta\hat{\mathbf{h}}(k) \quad (5.4)$$

ブロック処理化には，ブロック内で平均化する変数の違いから幾つかの方法が考えられるが，ここでは，修正ベクトルを平均化するブロック版の AP 法を示す．同法は毎 N サンプル時刻に以下の手順を実行する（ $N = 1$ は逐次の AP 法になる）

$$\mathbf{e}_N(k) = \mathbf{y}_N(k) - \mathbf{X}_{L,N}^T(k)\hat{\mathbf{h}}(k-N) \quad (5.5)$$

$$\mathbf{g}(k-ip) = \mathbf{R}^{-1}(k-ip)\mathbf{e}(k-ip), i = 0, 1, \dots, \lfloor (N-1)/p \rfloor \quad (5.6)$$

$$\begin{aligned} \hat{\mathbf{h}}(k) &= \hat{\mathbf{h}}(k-N) + \sum_{i=0}^{\lfloor (N-1)/p \rfloor} \mu(k-ip)\mathbf{X}(k-ip)\mathbf{g}(k-ip) \\ &= \hat{\mathbf{h}}(k-N) + \mathbf{X}_{L,p[\lfloor (N-1)/p \rfloor+1]}(k)\mathbf{s}(k) \end{aligned} \quad (5.7)$$

ここで，

$$\mathbf{y}_N(k) = [y(k), y(k-1), \dots, y(k-N+1)]^T \quad (5.8)$$

$$\mathbf{X}_{L,N}(k) = \begin{bmatrix} x(k) & x(k-1) & \cdots & x(k-N+1) \\ x(k-1) & & & \vdots \\ \vdots & & & \vdots \\ x(k-L+1) & \cdots & \cdots & x(k-N-L+2) \end{bmatrix} \quad (5.9)$$

$$\mathbf{s}(k) = [\mu(k)\mathbf{g}^T(k), \mu(k-1)\mathbf{g}^T(k-1), \cdots, \mu(k-p\lfloor(N-1)/p\rfloor)\mathbf{g}^T(k-p\lfloor(N-1)/p\rfloor)]^T \quad (5.10)$$

であり， $\lfloor x \rfloor$ は floor 関数で x を超えない最大の整数を意味する．この手順中には二つの定係数によるフィルタリングがある．つまり，(5.5) の $\mathbf{X}_{L,N}^T(k)\hat{\mathbf{h}}(k-N)$ と (5.7) の二行目右辺第二項の $\mathbf{X}_{L,p\lfloor(N-1)/p\rfloor+1}(k)\mathbf{s}(k)$ である．これらを高速 FIR フィルタリングにより計算することで演算量の低減が可能ではあるが，逐次法に比べて収束速度が低下するという問題がある．

5.2.2 高速 FIR フィルタリング，Fast FIR Filtering (FFF) 法

この節では高速 FIR 法（または，線形畳み込み法とも呼ばれる）を実現する二つの方法を概説し，その演算量も見積もる．二つの方法の一つは短区間の線形畳みを再帰的に行うものであり [36, 37]，もう一つの方法は巡回畳み込みを利用する方法である．この節では正方 Hankel 行列（行の順序を逆にすれば，Toeplitz 行列に対しても同様の議論ができる）とベクトルの積で表わされる行列ベクトル積（matrix-vector product，MVP）基本として，これを長方形の行列とベクトルの積に応用する．

線形畳み込みに基づく FFF

四つの小正方行列を要素にもつ Hankel 行列にベクトルを乗じる MVP は，通常の直接的な演算では，四つの小行列に関する MVP と二つのベクトル和（vector-vector additions，

VVA) に分解できる．すなわち，

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{u} + \mathbf{B}\mathbf{v} \\ \mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{v} \end{bmatrix} \quad (5.11)$$

一方，短区間畳み込みの手法によると，上式 (5.11) は次のようにもかける．

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{I} \\ \mathbf{I} & \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{C} - \mathbf{B} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{A} - \mathbf{B} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{u} + \mathbf{v} \\ \mathbf{v} \end{bmatrix} \quad (5.12)$$

この場合は，三つの小行列の MVP と三つの VVA，そして二つの行列の差が必要であるので，直接的な方法に比べると必要な MVP の数が一つ少なくて済む．さらに，この三つの MVP は，それぞれ上式 (5.12) を再帰的に適用することで計算される．

正方行列に関する MVP の演算量を見積もる．ここでは，積，和と積和演算の数の総和を演算量として定義する．これは，それらの演算がほとんどの DSP において，1 マシンサイクルで実行されかつ演算時間の主要な部分を占めるからである．行列の次元は $N = 2^n$ と仮定し， $MVP_{\text{FFF}}(N)$ と $MVP_{\text{direct}}(N)$ により，それぞれ，FFF と直接法により次元 N の MVP を実行する際の演算量を，また， $VVA(N)$ により N 次の VVA の演算量を表わす．(5.12) より， $MVP_{\text{FFF}}(N)$ の再帰式は次のとおりかける．

$$MVP_{\text{FFF}}(N) = 3MVP_{\text{FFF}}(N/2) + qVVA(N/2), q = 2 \text{ or } 3 \quad (5.13)$$

ここで，行列の差の演算量は MVP と VVA に比べて小さいので無視している．また， $q = 2$ は VVA $\mathbf{u} + \mathbf{v}$ を行う必要のない場合に対応している．上式 (5.13) から，

$$MVP_{\text{FFF}}(N) = 3^{n-n_0}MVP_{\text{direct}}(N_0) + q \sum_{i=1}^{n-n_0} 3^{i-1}VVA(N/2^i) \quad (5.14)$$

ここで， $N_0 = 2^{n_0}$ は再帰的に計算する小行列の最小の次元であり， $MVP(N_0)$ は直接的に計算する．次に， $VVA(N) = 2VVA(N/2)$ の関係を考慮すると，上式 (5.14) は次式となる

$$MVP_{\text{FFF}}(N) = 3^{n-n_0}MVP_{\text{direct}}(N_0) + q(3^{n-n_0} - 2^{n-n_0})VVA(N_0) \quad (5.15)$$

最後に， $MVP_{\text{direct}}(N_0) = N_0^2$ ， $VVA(N_0) = N_0$ を (5.15) に代入すると MVP_{FFF} が次式で見積もられる．

$$MVP_{\text{FFF}}(N) = 3^n(4/3)^{n_0} + q[3^n(2/3)^{n_0} - 2^n] \quad (5.16)$$

次に，時系列データからなる長方形行列に関する MVP の演算量を見積もる． N サンプル毎に N 行 $L = MN$ (M は自然数) 列の行列に要素数 L の縦ベクトルを右から乗じる，あるいは L 行 N 列の行列に要素数 N の縦ベクトルを右から乗じる場合，行列の差の部分は新たな N 個のデータに対してのみ N 個の差分を $\log_2(N/N_0)$ 回行うので合計 $N \log_2(N/N_0) = 2^n(n - n_0)$ 回の差分となる．また， L 行 N 列の行列に要素数 N の縦ベクトルを右から乗じるという定係数によるフィルタリングを行う場合には，ベクトルの差分 $\mathbf{u} + \mathbf{v}$ は最初に一度だけ求めておけばよい．以上より， r 行 c 列の行列に関する FFF による MVP の演算量を $MVP_{\text{FFF}}(r, c)$ と表すと次式となる．

$$MVP_{\text{FFF}}(r, c) = \begin{cases} (r/c)MVP_{\text{FFF}}(c) + (3^{\log_2 c}(2/3)^{n_0} - c) + c(\log_2 c - n_0), & q = 2, r > c \\ (c/r)MVP_{\text{FFF}}(r) + r(\log_2 r - n_0), & q = 3, r \leq c \end{cases} \quad (5.17)$$

巡回畳み込みに基づく FFF

MVP は Fast Fourier Transform(FFT) のような巡回畳み込みによっても計算できる． \mathbf{A} を次式で定義される N 行 N 列の Hankel 行列とする．

$$\mathbf{A} = \begin{bmatrix} a_1 & a_2 & \cdots & a_N \\ a_2 & a_3 & \cdots & a_{N+1} \\ \vdots & \vdots & \ddots & \vdots \\ a_N & a_{N+1} & \cdots & a_{2N-1} \end{bmatrix} \quad (5.18)$$

また， \mathbf{u} を要素が N 個のベクトルとする．すると， $MVP \mathbf{A} \mathbf{u}$ は FFT を用いて以下の手順で計算できる：

$$\tilde{\mathbf{a}} = \text{FFT}([0, a_1, a_2, \cdots, a_{2N-1}]^T) \quad (5.19)$$

$$\tilde{\mathbf{u}} = \text{FFT}([\mathbf{0}_N^T, u_N, u_{N-1}, \dots, u_1]) \quad (5.20)$$

$$\mathbf{A}\mathbf{u} = \text{the first } N \text{ elements of } \text{IFFT}(\tilde{\mathbf{a}} \otimes \tilde{\mathbf{u}}) \quad (5.21)$$

ここで、(I)FFT、 \otimes はそれぞれ (逆) FFT、要素ごとの積を表わす。適応フィルタでは (5.19) の FFT はデータが最初に入力されたときに行われ、(5.20) の FFT、または (5.21) の IFFT はブロック毎に一回計算される。 $2N$ 点の (I)FFT と要素積の演算量が、それぞれ、 $16N \log_2(2N)$ と 4 であると見積もれば、 MVP_{FFF} は次式で見積もられる。

$$\text{MVP}_{\text{FFF}}(N) = 32N \log_2(2N) + 4N \quad (5.22)$$

また、時系列データからなる r 行 c 列の行列に関する FFF による MVP の演算量 $\text{MVP}_{\text{FFF}}(r, c)$ は、新しいデータに対してのみ (5.18) が必要なことから次式となる。

$$\text{MVP}_{\text{FFF}}(r, c) = \begin{cases} (r/c)(16c \log_2(2c) + 4c) + 16r \log_2(2c), & r > c \\ (c/r)(16r \log_2(2r) + 4r) + 16r \log_2(2r), & r \leq c \end{cases} \quad (5.23)$$

5.3 提案するブロック AP 法

この節では、FFF により演算量を低減しながら、逐次法と同じ収束速度をもつ逐次等価ブロック高速 AP 法について述べる。

5.3.1 出力の逐次法からの補正

もし、係数ベクトル $\hat{\mathbf{h}}(k)$ が毎時刻更新されたとすると、 $\hat{\mathbf{h}}(k+i)$, ($i \geq 0$) は次式のようにかける。

$$\begin{aligned} \hat{\mathbf{h}}(k+i) &= \hat{\mathbf{h}}(k-1) + \sum_{j=1}^{i+1} \mu(k+i-j+1) \mathbf{X}(k+i-j+1) \mathbf{g}(k+i-j+1) \\ &= \hat{\mathbf{h}}(k-1) + \sum_{j=1}^{i+p} s_j(k+i) \mathbf{x}(k+i-j+1) \\ &\quad - \sum_{j=1}^{p-1} s_j(k-1) \mathbf{x}(k-j) \end{aligned} \quad (5.24)$$

ここで, $s_j(k)$ は時刻 k での $\mathbf{x}(k-j+1)$ へのプレフィルタ係数をステップサイズで重み付けした和であり,

$$s_j(k) = \sum_{l=1}^{\min(j,p)} \mu(k-j+l)g_l(k-j+l) \quad (5.25)$$

と定義される. また, ベクトル

$$\mathbf{s}_{i+p}(k+i) = [s_1(k+i), s_2(k+i), \dots, s_{i+p}(k+i)]^T, \quad (i = 0, 1, \dots, N-1) \quad (5.26)$$

は, 次式により更新される.

$$\mathbf{s}_{i+p}(k+i) = [0, \mathbf{s}_{i+p-1}^T(k+i-1)]^T + \mu(k+i)[\mathbf{g}^T(k+i), 0, \dots, 0]^T \quad (5.27)$$

(5.24) より, フィルタ出力 $\hat{y}(k+i)$ ($i \geq 0$) は次式のようにかける.

$$\begin{aligned} \hat{y}(k+i) &= \mathbf{x}^T(k+i)\hat{\mathbf{h}}(k+i-1) \\ &= \mathbf{x}^T(k+i)\hat{\mathbf{h}}(k-1) + \sum_{j=1}^{i+p-1} s_j(k+i-1)\mathbf{x}^T(k+i)\mathbf{x}(k+i-j) \\ &\quad - \sum_{j=1}^{p-1} s_j(k-1)\mathbf{x}^T(k+i)\mathbf{x}(k-j) \\ &= \mathbf{x}^T(k+i)\hat{\mathbf{h}}(k-1) + \sum_{j=1}^{i+p-1} s_j(k+i-1)r_j(k+i) \\ &\quad - \sum_{j=1}^{p-1} s_j(k)r_{i+j}(k+i) \end{aligned} \quad (5.28)$$

ここで, 次式で定義される自己相関

$$r_j(k+i) = \mathbf{x}^T(k+i)\mathbf{x}(k+i-j) \quad (5.29)$$

は, 再帰的に以下のように求められる [19].

$$r_j(k+i) = r_j(k+i-1) + x(k+i)x(k+i-j) - x(k-L+i)x(k-L+i-j) \quad (j = 1, 2, \dots, i+p-2) \quad (5.30)$$

$$r_{i+p-1}(k+i) = r_{i+p-1}(k-1) + \rho_{i+p-1}(k+i) - \rho_{i+p-1}(k-L+i) \quad (5.31)$$

ただし ,

$$\rho_{i+p-1}(k+i) = \sum_{l=0}^i x(k+l)x(k+l-i-p+1) \quad (5.32)$$

(5.28) の右辺第一項は 5.2.1 項のブロック AP のフィルタ出力に相当し同第二項と三項が逐次法の出力からの補正を行う .

5.3.2 補助フィルタベクトルの導入

(5.28) の右辺第三項はフィルタベクトルの代わりに

$$\mathbf{z}(k-1) = \hat{\mathbf{h}}(k-1) - \sum_{j=1}^{p-1} s_j(k-1)\mathbf{x}(k-j) \quad (5.33)$$

を導入することで計算を省ける [32, 38] . (5.28) と (5.33) より , フィルタ出力 $\hat{y}(k+i)$ は

$$\hat{y}(k+i) = \mathbf{x}^T(k+i)\mathbf{z}(k-1) + \mathbf{s}_{i+p-1}^T(k+i-1)\mathbf{r}_{i+p-1}(k+i) \quad (5.34)$$

とかける . ここで ,

$$\mathbf{r}_j(k) = [r_1(k), r_2(k), \dots, r_j(k)]^T \quad (5.35)$$

ブロック長が N なら, (5.34)($i = 0, 1, \dots, N-1$) は行列表現で

$$\begin{aligned} \hat{\mathbf{y}}_N(k+N-1) &= \mathbf{X}_{L,N}^T(k+N-1)\mathbf{z}(k-1) \\ &+ \begin{bmatrix} \mathbf{s}_{p+N-2}^T(k+N-2)\mathbf{r}_{p+N-2}(k+N-1) \\ \vdots \\ \mathbf{s}_p^T(k)\mathbf{r}_p(k+1) \\ \mathbf{s}_{p-1}^T(k-1)\mathbf{r}_{p-1}(k) \end{bmatrix} \end{aligned} \quad (5.36)$$

とかける .

ベクトル $\mathbf{z}(k-1)$ の更新式は以下のようにして導出される . まず , (5.33) において , k を $k+N$ で置き換え次式を得る .

$$\mathbf{z}(k+N-1) = \hat{\mathbf{h}}(k+N-1) - \sum_{j=1}^{p-1} s_j(k+N-1)\mathbf{x}(k+N-j) \quad (5.37)$$

次に, (5.33) と (5.37) を (5.24) ($i = N$) に代入して, 求める $\mathbf{z}(k-1)$ の更新式が得られる.

$$\mathbf{z}(k+N-1) = \mathbf{z}(k-1) + \sum_{j=p}^{p+N-1} s_j(k+N-1)\mathbf{x}(k+N-j) \quad (5.38)$$

行列表現だと,

$$\mathbf{z}(k+N-1) = \mathbf{z}(k-1) + \mathbf{X}_{L,N}(k+N-p)\mathbf{s}_{p+N-1}|_{p+N-1}^p(k+N-1) \quad (5.39)$$

ここで, $\mathbf{s}|_j^i$ は \mathbf{s} の i 番めから j 番めの要素からなる部分ベクトルを表わす.

5.3.3 FFF の提案法への適用

提案法中の以下の二つのフィルタリングが FFF により計算できる.

- $\mathbf{X}_{L,N_1}^T(k+N_1-1)\mathbf{z}(k-1)$ ((5.36) 右辺第一項)
- $\mathbf{X}_{L,N_2}(k+N_2-p)\mathbf{s}_{p+N_2-1}|_{p+N_2-1}^p(k+N_2-1)$ ((5.39) 右辺第二項)

ここで, N_1 はフィルタ出力のためのブロック長であり, N_2 はフィルタ更新のためのブロック長である. ここでは, 簡単のため, 二つのブロック長はともに, フィルタ長 L の因数, つまり, $L = M_i N_i$, ($i = 1, 2$) (M_i は正整数) であり, また, どちらか片方のブロック長はもう片方の倍数であると仮定する. 二つの独立なブロック長を導入する利点は, エコーキャンセラへの応用において, N_1 だけを小さくしてフィルタ出力遅延を小さくすることで, 聴覚的なエコーを減少させることができることである.

一つ目のフィルタリングは N_1 サンプルごとに行われ, 次式のように M_1 個の次元 N_1 の MVP からなる (実装については [19, 37] を参照)

$$\mathbf{X}_{L,N_1}^T(k+N_1-1)\mathbf{z}(k-1) = \sum_{i=1}^{M_1} \mathbf{X}_{N_1,N_1}^T(k-(i-2)N_1-1)\mathbf{z}_{N_1}^{(i)}(k-1) \quad (5.40)$$

ここで, \mathbf{X}_{N_1, N_1} は N_1 行 N_1 列の行列を $\mathbf{z}_{N_1}^{(i)}$ は N_1 要素のベクトルであり, それぞれ, 次式で定義される.

$$\mathbf{X}_{N_1, N_1}(k) = \begin{bmatrix} x(k) & \cdots & x(k - N_1 + 1) \\ \vdots & \ddots & \vdots \\ x(k - N_1 + 1) & \cdots & x(k - 2N_1 + 2) \end{bmatrix} \quad (5.41)$$

$$\mathbf{z}_{N_1}^{(i)}(k - 1) = [z_{(i-1)N_1+1}(k - 1), z_{(i-1)N_1+2}(k - 1), \cdots, z_{(i-1)N_1+N_1}(k - 1)]^T \quad (5.42)$$

上式 (5.40) の演算量は $\text{MVP}_{\text{FFF}}(N_1, L)$ と見積もられる.

同様に, $N_2 (= L/M_2)$ サンプル毎に行われる二つ目のフィルタリングは

$$\mathbf{z}(k + N_2 - 1) = \begin{bmatrix} \mathbf{z}_{N_2}^{(1)}(k - 1) \\ \mathbf{z}_{N_2}^{(2)}(k) \\ \vdots \\ \mathbf{z}_{N_2}^{(M_2)}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{X}_{N_2, N_2}(k + N_2 - p) \\ \mathbf{X}_{N_2, N_2}(k - p) \\ \vdots \\ \mathbf{X}_{N_2, N_2}(k - L + 2N_2 - p) \end{bmatrix} \mathbf{s}_{p+N_2-1|p+N_2-1}^p(k + N_2 - 1) \quad (5.43)$$

と書き直すことができる. ここで, 右辺第二項は M_2 個の $\text{MVP}(N_2)$ からなる.

演算	式番号	演算量
初期化		
$k_{\text{current}} = N_2 - \nu, k = k_{\text{current}} - \nu,$		
ν :delay to avoid data shortage		
$k' = k - N_1 + 1,$		
$\rho_{i+p-1}(k' - jN_2 + i) =$		
$\sum_{l=0}^i x(k' - jN_2 + l)x(k' - jN_2 - i - p + 1 + l),$		
$i = 0, 1, \dots, N_2 - 1, j = 1, 2, \dots, L/N_2$		
$\mathbf{e}(k' - 1) = [y(k' - 1), y(k' - 2), \dots, y(k' - p)]^T$		
$\mathbf{s} = \mathbf{0}, \mu(k' - 1) = 0$		
<hr/>		
For $i = 0, 1, \dots, N_2 - 1$ do 1) to 7)		
1) If $i \bmod N_1 = 0$ then $\hat{\mathbf{y}}'(k) = \mathbf{X}_{L, N_1}^T(k)\mathbf{z}(k' - 1).$	(5.40)	MVP _{FFF} (N_1, L)
2) $r_j(k' + i) = r_j(k' + i - 1) + x(k' + i)x(k' + i - j),$	(5.30)	$2(i + p - 1)$
$-x(k' - L + i)x(k' - L + i - j), j = 0, 1, \dots, i + p - 2$		
$\rho_{i+p-1}(k' + i) = \sum_{l=0}^i x(k' + l)x(k' - i - p + 1 + l)$	(5.30)	$i + 1$
$r_{i+p-1}(k' + i) = r_{i+p-1}(k' - 1)$	(5.31)	2
$+ \rho_{i+p-1}(k' + i) - \rho_{i+p-1}(k' - L + i)$		
3) $e(k' + i) = y(k' + i) - \hat{y}'(k' + i)$	(5.1)	$i + p$
4) $\mathbf{e}(k' + i) = \begin{bmatrix} e(k' + i) \\ (1 - \mu(k' + i - 1))\mathbf{e}(k' + i - 1) \end{bmatrix} \Big _p^1$	(2.37)	$p - 1$
5) $r_j(k' + i - l)$ から $\mathbf{R}(k' + 1), (j, l = 0, 1, \dots, p - 1)$ を構成し , $\mathbf{R}(k' + 1)\mathbf{g}(k' + i) = \mathbf{e}(k' + i).$ を解く .	(5.2)	FTF 法なら $15p$
6) $\mathbf{s}_{i+p}(k' + i) = \begin{bmatrix} 0 \\ \mathbf{s}_{i+p}(k' + i - 1) \end{bmatrix} + \mu(k' + i) \begin{bmatrix} \mathbf{g}(k' + i) \\ \mathbf{0} \end{bmatrix}$	(5.27)	p
7) $k = k + 1$		
8) $\mathbf{z}(k' + N_2 - 1) = \mathbf{z}(k' - 1)$	(5.43)	MVP _{FFF} (L, N_2)
$+ \mathbf{X}_{L, N_2}(k' - p + N_2)\mathbf{s}_{p+N_2-1}^p(k' + N_2 - 1)$		
9) $k' = k' + N_2$		

アルゴリズム 5.1: 逐次等価ブロックアフィン射影法

5.3.4 演算量と他の逐次等価ブロック法との比較

提案法をアルゴリズム 4.1 に示す . 提案法は三部分からなる : フィルタリング部ではフィルタ出力を FFF を用いて計算する (アルゴリズム 4.1 の 1) , 補正部では補正項を再帰的に

計算しフィルタ出力に加え，逐次法の出力と一致させる（アルゴリズム 4.1 の 2-7），フィルタ更新部ではフィルタを FFF を使って更新する（アルゴリズム 4.1 の 8）．1 サンプル時刻当たりの演算量は以下のように見積もることができる．

- フィルタリング部（アルゴリズム 4.1 の 1）

$$C_f = \text{MVP}_{\text{FFF}}(N_1, L)/N_1 \quad (5.44)$$

- 補正部（アルゴリズム 4.1 の 2-7）

もし，プレフィルタ $g(k)$ を FTF を用いて $15p$ の演算量で求めると [39]，補正部の演算量は次のように見積もられる．

$$\begin{aligned} C_c &= \sum_{i=0}^{\max(N_1, N_2)-1} (4i + 20p - 2)/\max(N_1, N_2) \\ &\approx 2\max(N_1, N_2) + 20p \end{aligned} \quad (5.45)$$

- フィルタ更新部（アルゴリズム 4.1 の 8）

$$C_u = \text{MVP}_{\text{FFF}}(L, N_2)/N_2 \quad (5.46)$$

合計すると，提案法の演算量は次式で見積もることができる．

$$\begin{aligned} C_T &= C_f + C_c + C_u \\ &= \text{MVP}_{\text{FFF}}(N_1, L)/N_1 + 2\max(N_1, N_2) + 20p + \text{MVP}_{\text{FFF}}(L, N_2)/N_2 \end{aligned} \quad (5.47)$$

表 5.1 は提案法の演算量 C_{BEFAP} と逐次 AP 法 C_{FAP} ，高速 block exact LMS 法 C_{FELMS} ，block exact 高速ニュートントランスバーサルフィルタ C_{BEFNTE} と FSU-RLS 法 C_{FSURLS} と比較するものである．ただし，これらの手法の演算量は以下の式により求めた．

$$C_{\text{FAP}} = 2L + 20p \quad (5.48)$$

$$C_{\text{FELMS}} = \text{MVP}_{\text{FFF}}(N, L)/N + \text{MVP}_{\text{FFF}}(L, N)/N - C_{f\cap c} + 2N \quad (5.49)$$

$$\begin{aligned} C_{\text{BEFNTE}} &= \text{MVP}_{\text{FFF}}(N, L)/N + \text{MVP}_{\text{FFF}}(L, N)/N - C_{f\cap c} + \\ &3N + 12p + \text{MVP}_{\text{FFF}}(N)/N + 3\text{MVP}_{\text{FFF}}(p)/p \end{aligned} \quad (5.50)$$

$$C_{\text{FSURLS}} = 4(\text{MVP}_{\text{FFF}}(N, L)/N + \text{MVP}_{\text{FFF}}(L, N)/N - C_{f_{\text{nc}}}) + 5.5N \quad (5.51)$$

ここで、 $C_{f_{\text{nc}}}$ は $\text{MVP}_{\text{FFF}}(N, L)$ と $\text{MVP}_{\text{FFF}}(L, N)$ に共通する N 次の行列の差分が $2N$ 点の FFT に関わる量であり、また、(5.47) では $p = 8, 48$, $N_1 = N_2 = N$ を、(5.48) と (5.50) では $p = 8, 48$ を用いた。(5.16) による $\text{MVP}_{\text{FFF}}(N)$ の計算では、 $n_0 = 2$ と $q = 3$ とした。表 5.1 では、幾つかのフィルタ長 L とブロック長 N の組について (5.47)–(5.51) を評価している。この表より、提案法の演算量がフィルタ長が $L > 1024$, $p = 8$ で逐次的高速 AP 法に比べ 50%以上少なく、FELMS 法とほぼ同程度であること、また、他の逐次等価ブロック法である BEFNTEF 法や FSU-RLS 法に比べても少ないことがわかる。第 2 章の音声に対する計算機シミュレーション条件である $L = 1024$, $p = 48$ では、高速 AP 法では 3008、それに比べて本章で提案した BEAP 法ではブロック長 $N = 128$ で 1980 であり、約 2/3 に演算量を低減できる。

5.3.5 従来の AP 法との収束曲線の比較

本項では、提案したブロック AP 法、元の逐次 AP 法と従来のブロック AP 法の収束曲線を比較する。実験は計算機シミュレーションで音響エコーキャンセラの構成で行った。未知系はスピーカとマイクロホンの間の音響経路である。実験条件は：未知系と適応フィルタのインパルス長 h は、ともに $L = 1024$ 、射影の次数は $p = 8$ 、ブロック長は $N_1 = N_2 = 8$ 、入力信号は音声であり、帯域は 0.3-3.4 kHz、サンプリング周波数は 8 kHz である。白色雑音をマイクロホンに音声の平均レベルに対し -40 dB で加えた。適応フィルタ（音響エコーキャンセラ）の性能の指標として次式で定義する二乗平均誤差（mean squared error, MSE）を用いた。

$$\text{MSE} = 10 \log_{10} |\mathbf{x}^T(k)\mathbf{h}(k) - \hat{y}(k)|^2 \text{ dB} \quad (5.52)$$

図 5.1 に計算機シミュレーションの結果を示す。同図で、縦軸は MSE で、横軸は時間であり、1 s でインパルス応答を変えている。収束曲線 1, 2, 3 は順に、元の逐次 AP 法、提案したブロック AP 法、そして従来のブロック AP 法のものであり、50 回の試行の平均であ

Projection order p	Filter length L	Block length N	FAP	BEFAP (proposed)	FELMS	BEFNTF	FSU-RLS
8	128	16	416	389	229	379	877
8	512	64	1184	782	622	841	2330
8	1024	128	2208	1177	1017	1318	3749
8	2048	256	4256	1833	1673	2128	6052
48	128	16	1216	1189	229	880	877
48	512	64	1984	1582	622	1342	2330
48	1024	128	3008	1977	1017	1819	3749
48	2048	256	5056	2633	1673	2630	6052

表 5.1: 他の逐次等価ブロック法との演算量の比較．提案法の演算量ががフィルタ長が $L > 1024$, $p = 8$ で逐次の高速 AP 法に比べ 50%以上少なく, FELMS 法とほぼ同程度である．また, 他の逐次等価ブロック法である BEFNTF 法や FSU-RLS 法に比べても少ない．FAP : 高速 AP 法 (第 4 章), FELMS : 高速ブロック exact LMS 法, BEFNTF : ブロック exact 高速ニュートントランスバーサルフィルタ法, FSU-RLS : 高速サブサンプル RLS

る．収束曲線 1 と 2 は同一であり，このことから提案法の収束特性が逐次 AP 法と同じであることがわかる．また，同図からは提案法は従来のブロック AP 法より，収束が速いこともわかる．

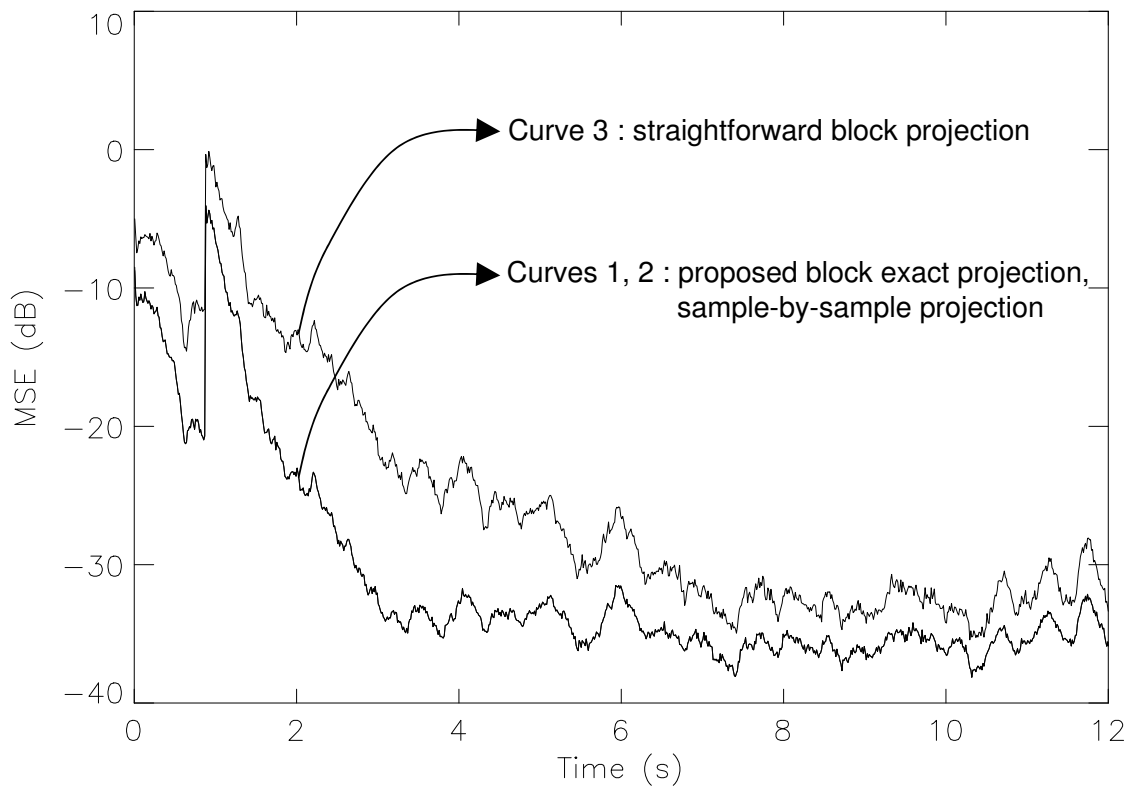


図 5.1: 従来法と提案法の収束特性の比較．逐次 AP 法と提案した逐次等価な高速 AP 法の収束曲線は一致する．また，従来のブロック処理手法によるブロック AP 法は逐次法と提案法に比べ収束が遅く定常エコーも多い．フィルタ長：1024，射影次数 p ：8，ブロック長 $N_1 = N_2 = 8$ ，入力信号：音声，サンプリング周波数：8 kHz，エコー雑音パワー比：40 dB

5.4 ブロック長の選択

適応フィルタの設計にあたっては，出力遅延時間と演算量には制限があり，その制限条件の下で，フィルタの長さを最長にするようにブロック長 N_1 と N_2 を決定する必要がある

る．本節では，提案法における出力遅延時間とフィルタ長の関係をブロック長 N_1 と N_2 の関数として表わす．

- フィルタ長

もし，演算量がある一定の値であれば，(5.47) に (5.17) を代入して変形し主要な項についてのみ残すと，次式のように，フィルタ長 L とブロック長 N_1 と N_2 の関係が得られる．

$$L = \frac{C_T - 2\max(N_1, N_2) - 20p}{\text{MVP}(N_1)/N_1^2 + \text{MVP}(N_2)/N_2^2} \quad (5.53)$$

- 遅延

フィルタリング部実行最中の入力信号は次のフィルタリングまで待つことになる．最も長く待たされる入力信号はフィルタリング部が始まって直ぐの入力信号であり，この入力に対応する出力信号は次のフィルタリング部の直後の補正部で計算される．補正部の処理時間が短いことを考慮にいと，最大の遅延時間はフィルタリング部の開始時間と次のフィルタリング部の終了時刻の区間で最長のものとして評価できる．図 5.2 は最大遅延時間を $N_1 > N_2$ と $N_1 \leq N_2$ の二つの場合について示している．同図より遅延時間 D は次式で評価できる．

$$D = \begin{cases} N_1 + N_1 C_f / C_T & (N_1 > N_2) \\ (2N_1 C_f + \lambda N_2 C_c + N_2 C_u) / C_T & (N_1 \leq N_2) \end{cases} \quad (5.54)$$

ここで， λ は補正部の直前の修正部の演算量の N_2 内全体の修正部の演算量に対する比を表わし，次式で与えられる．

$$\begin{aligned} \lambda &= \frac{\sum_{i=N_2-N_1}^{N_2-1} (4i + 20p - 2)}{\sum_{i=0}^{N_2-1} (4i + 20p - 2)} \\ &= N_1(-2N_1 + 4N_2 + 20p - 4) / [N_2(2N_2 + 20p - 4)] \end{aligned} \quad (5.55)$$

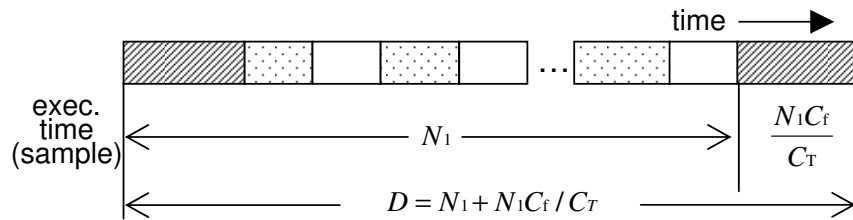
(5.53) と (5.54) で， N_1 と N_2 に値を代入するとで，遅延時間とフィルタ長の関係が得られ，これより，あらかじめ決められた遅延時間で最長のフィルタ長とブロック長がわかる．図 5.3 はフィルタ長と遅延時間の関係の一例を示す図であって，総演算量は $C_T = 2160$ ，射影の

次数は $p = 8$ を仮定している．同図から，提案法は逐次の高速 AP 法に比べ，ブロック長が $(N_1, N_2) = (8, 32)$ のときに，16 サンプルの遅延でフィルタ長を 20% 長くできることがわかる．また，数百サンプルの遅延時間が許容されるなら，ブロック長が $(N_1, N_2) = (128, 128)$ のときに，提案法は逐次の高速 AP 法に比べフィルタ長を倍にできることもわかる．

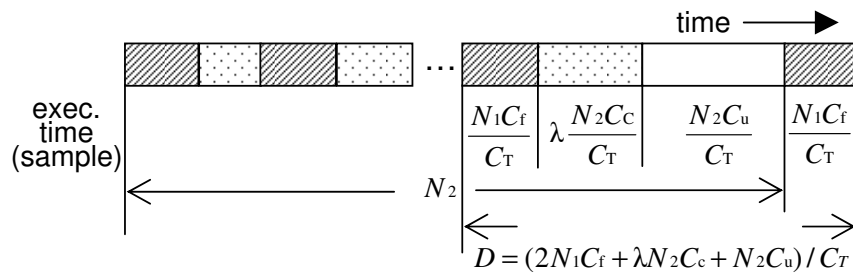
5.5 本章の結論

逐次等価ブロック高速 AP 法を導出した．この手法は従来のブロック処理にあった収束速度の低下という問題を克服し，その収束速度は逐次 AP 法と同じであるという特徴がある．これは (1) 従来のブロック AP 法の出力と逐次の AP 法の差を補正する項の導入したこと，(2) 高速 FIR フィルタリング手法をフィルタの出力の計算とフィルタの更新に適用することで実現した．また，フィルタ出力とフィルタ更新のブロック長を異なる値にすることができるので出力遅延の短いブロック処理が可能なることも本提案手法の特徴である．

総演算量が固定されている場合に最長のフィルタ長を与えるブロック長を選ぶ手順も示した．一例を挙げると，もし，数百サンプルの遅延が許されれば提案したブロック AP 法は高速 AP 法に比べ同じ演算量でフィルタ長を約 1000 からその倍にすることができる．



(a) $N_1 > N_2$



(b) $N_1 \leq N_2$

▨ : filtering ▤ : correction □ : update

図 5.2: 提案法の処理順序 . (a) $N_1 > N_2$, フィルタリングのブロック長が適応フィルタ更新のブロック長より長い場合, フィルタリングの後, 補正と更新が交互に繰り返される . 最長の出力遅延が生じるのは, フィルタリング処理開始時に入力した信号に対する補正が行われる場合である . (b) $N_1 \leq N_2$, フィルタリングのブロック長が適応フィルタ更新のブロック長より短い場合, フィルタリングと補正を交互に繰り返した後, 更新を行う . 最長の出力遅延が生じるのは最も補正までの時間が経った入力に対する補正が行われるときである .

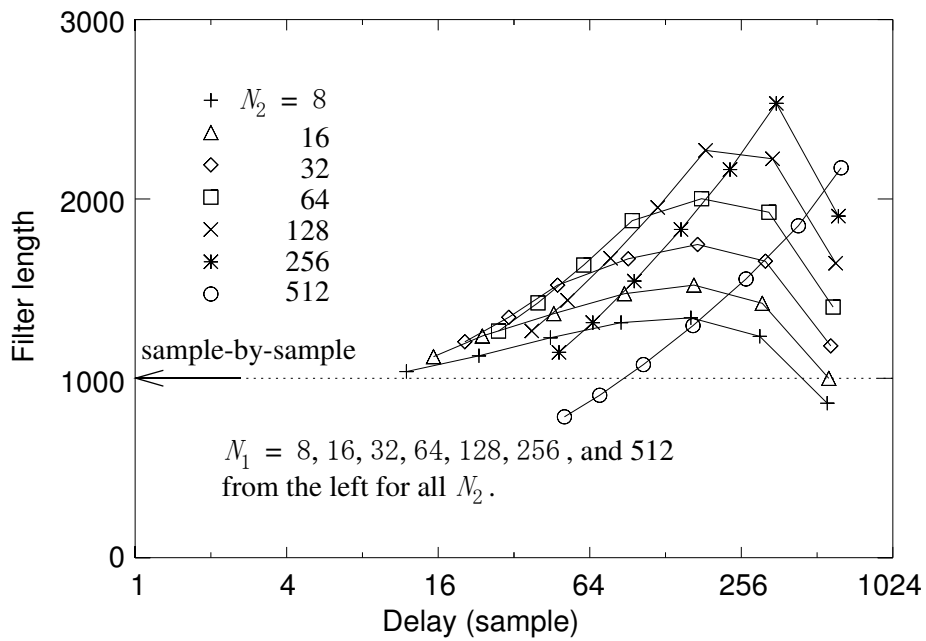


図 5.3: フィルタ長と遅延時間の関係の一例．提案法は逐次の高速 AP 法に比べ，ブロック長が $(N_1, N_2) = (8, 32)$ のときに，16 サンプルの遅延でフィルタ長を 20% 長くできることがわかる．また，数百サンプルの遅延時間が許容されるなら，ブロック長が $(N_1, N_2) = (128, 128)$ のときに，提案法は逐次の高速 AP 法に比べフィルタ長を倍にできる．総演算量 C_T : 2160，射影次数 p : 8

第6章 騒音制御のための高速アフィン射影法

6.1 はじめに

騒々しいオフィスや乗り物の中で通話をする際などに、ある空間的領域だけ騒音を低減したい場合がある。このような場合、騒音源を收音するマイクロホン、收音した騒音を通過させるフィルタ、フィルタを通過した騒音を騒音を抑圧したい制御点に向けて放射するスピーカ（2次音源ともいう）、そして、制御点で騒音の消去具合を観測するマイクロホンからなる *filtered-x* と呼ばれる系を用いる。スピーカから制御点までの伝達系が時間的に変化する場合は、エコーキャンセラと同様にその伝達系を適応的に推定する必要がある。しかしながら、エコーキャンセラの回線からの入力信号である音声と同様に騒音は通常有色信号であり、かつ、時間的にレベルと有色性（スペクトル形状）が変化するため、スピーカと制御点間の伝達系の推定に従来法である NLMS 法や LMS 法を用いたのでは収束速度が遅いという問題点がある [46, 47]。RLS 法 [48] やアフィン射影 (AP) 法 [4] などを用いれば、収束速度の問題は軽減するものの、大きな演算量が問題となる。

近年、この演算量を低減する手法が提案されてきている。例を挙げると、Cioffi が高速 RLS 法を [14]、その数値的に安定な改良版を Slock が提案している [15]。また、田中は丸山の手法 [32] の拡張として高速 AP 法を提案している。Gay もそれとは独立に高速 AP 法を提案している [38, 39]。これらの高速適応フィルタの手法は、前章までに検討の対象としてきた未知系推定に対して考案されたもので、音場制御に現れる *filtered-x* 系にはそのままでは適用できない。本章では、*filtered-x* 系のための高速 AP 法を示し、自己相関による収束速度劣化の問題を従来法より少ない演算量で解決する。

第2節では、AP 法の考え方を述べる。第3節では、*filtered-x* 系のための高速 AP 法を

提案する．第4節では，提案法により多入力 filtered-x 系の収束速度が改善されることを計算機シミュレーションにより確認する．

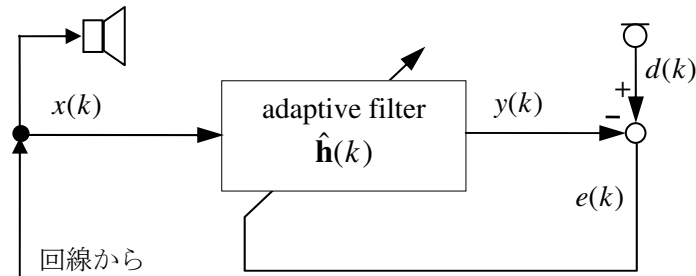


図 6.1: 一入力一出力の適応フィルタの構成

6.2 AP 法

この節では AP 法の基本的な考え方と定式化と高速計算手法を振り返る．

6.2.1 考え方

図 6.1 に，一入力，一出力の騒音制御のための適応フィルタを構成図を示す．同図で適応フィルタの出力 $y(k)$ は次式により計算される．

$$y(k) = \hat{\mathbf{h}}^H(k-1)\mathbf{x}(k) \quad (6.1)$$

ただし， k は離散時間を \cdot^H は複素共役を表わす．また， $\mathbf{x}(k)$ と $\hat{\mathbf{h}}(k)$ はそれぞれ入力信号と適応フィルタの係数からなるベクトルであり，次式で定義される．

$$\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-L+1)]^T \quad (6.2)$$

$$\hat{\mathbf{h}}(k) = [h_1(k), h_2(k), \dots, h_L(k)]^T \quad (6.3)$$

ただし $x(k)$ は騒音源信号観測用マイクロホンの出力として得られる入力信号であり， L は適応フィルタの係数の数である． \cdot^T は転置を表す． $h_1(k), h_2(k), \dots, h_L(k)$ は時刻 k にお

けるフィルタ係数である．適応フィルタはその出力 $y(k)$ が目的信号 $d(k)$ にできるだけ近くなるように修正される．適応フィルタの修正を一般的に表すと，

$$\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-1) + \mu \Delta \hat{\mathbf{h}}(k) \quad (6.4)$$

となる．ここで， $\Delta \hat{\mathbf{h}}(k)$ を修正ベクトルと呼び，修正の大きさを調整する μ をステップサイズと呼ぶ．AP法では修正ベクトルは，(6.4) で $\mu = 1$ として修正したフィルタ係数 $\hat{\mathbf{h}}(k) (= \hat{\mathbf{h}}(k-1) + \Delta \hat{\mathbf{h}}(k))$ が次式のように最近の p 時刻の目的信号を生成するように決定される．

$$\begin{aligned} d(k) &= [\hat{\mathbf{h}}(k-1) + \Delta \hat{\mathbf{h}}(k)]^H \mathbf{x}(k) \\ d(k-1) &= [\hat{\mathbf{h}}(k-1) + \Delta \hat{\mathbf{h}}(k)]^H \mathbf{x}(k-1) \\ &\vdots \\ d(k-p+1) &= [\hat{\mathbf{h}}(k-1) + \Delta \hat{\mathbf{h}}(k)]^H \mathbf{x}(k-p+1) \end{aligned} \quad (6.5)$$

ここで， $p (\leq L)$ は射影次数と呼ばれる．一般に p が大きければ適応フィルタの収束は速くなるが，より多くの演算量を必要とする．(6.5) を行列形式でかくと，

$$\mathbf{d}_{[p]}^T(k) = [\hat{\mathbf{h}}(k-1) + \Delta \hat{\mathbf{h}}(k)]^H \mathbf{X}_{[L,p]}(k) \quad (6.6)$$

または，等価的に以下のようにかける．

$$\mathbf{e}_{[p]}^H(k) = \Delta \hat{\mathbf{h}}^H(k-1) \mathbf{X}_{[L,p]}(k) \quad (6.7)$$

ただし，

$$\mathbf{d}_{[p]}(k) = [d(k), d(k-1), \dots, d(k-p+1)]^T \quad (6.8)$$

$$\mathbf{X}_{[L,p]}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-p+1)] \quad (6.9)$$

$$\mathbf{e}_{[p]}^H(k) = \mathbf{d}_{[p]}^T(k) - \hat{\mathbf{h}}^H(k-1) \mathbf{X}_{[L,p]}(k) \quad (6.10)$$

ここで，下付き添え字 $_{[p]}$ と $_{[L,p]}$ はそれぞれ，ベクトルの要素数が p であること，行列が L 行 p 列であることを示す（ただし，ベクトルの要素が $\hat{\mathbf{h}}$ や $\Delta \hat{\mathbf{h}}$ のように L 個のときにかぎり下付き添え字を省略する）． $\Delta \hat{\mathbf{h}}(k)$ の係数の個数 L が式の数 p より大きいので，これら

の連立方程式の解は不定となる．解は無限にあるが，AP 法では，収束が保証されているという理由で最小ノルム解を用いる．(6.7) の最小ノルム解は次式で与えられる．

$$\Delta \hat{\mathbf{h}}(k) = \mathbf{X}_{[L,p]}(k) [\mathbf{X}_{[L,p]}^H(k) \mathbf{X}_{[L,p]}(k)]^{-1} \mathbf{e}_{[p]}(k) \quad (6.11)$$

(6.10) , (6.11) と (6.4) が AP 法による適応フィルタの $\hat{\mathbf{h}}(k-1)$ から $\hat{\mathbf{h}}(k)$ への更新式となる．

射影次数が $p = 1$ の場合，AP 法はよく知られた正規化 LMS 法 (normalized least-mean-square, NLMS) と同一である．NLMS 法での入力ベクトル $\mathbf{x}(k)$ が AP 法では入力信号行列 $\mathbf{X}_{[L,p]}(k)$ に置き換わっているので，AP 法は NLMS 法の一般化と考えることができる．この一般化において，NLMS 法での入力信号ベクトル $\mathbf{x}(k)$ を正規化するその自乗ノルムは，AP 法では (6.11) にあるように， $[\mathbf{X}_{[L,p]}^H(k) \mathbf{X}_{[L,p]}(k)]^{-1}$ なる逆行列に置き換わっている．この逆行列は，正規化の他に， $\mathbf{X}_{[L,p]}(k)$ の列に関して入力信号の脱相関化という重要な役割がある．相関を除去した信号を修正ベクトル $\Delta \hat{\mathbf{h}}(k)$ に用いることで，AP 法は NLMS 法より速く収束する．

6.2.2 AP 法の演算低減法

AP 法の演算量の低減法については，4 章においても述べたが，便利のため改めて簡単に触れておく．

まず，(6.10) のベクトル $\mathbf{e}_{[p]}(k)$ の時間更新は次式で実行可能である．

$$\mathbf{e}_{[p]}(k) = \left[e(k), (1 - \mu) \mathbf{e}_{[p-1]}^H(k-1) \right]^H \quad (6.12)$$

ここで，

$$e(k) = d(k) - y(k) \quad (6.13)$$

であり， $\mathbf{e}_{[p-1]}(k-1)$ は $\mathbf{e}_{[p]}(k-1)$ の最初の $(p-1)$ 個の要素からなるベクトルである．このように，行列とベクトルの積を含む (6.10) は $(1 - \mu)$ を乗じて要素を一つ移動するだけの (6.13) と (6.12) に置き換えられた．

次に，(6.11) と (6.4) の演算量の低減を考える． $\mathbf{g}_{[p]}(k)$ と $\mathbf{R}_{[p,p]}(k)$ という変数を導入して，(6.11) を以下のように書きなおす．

$$\Delta \hat{\mathbf{h}}(k) = \mathbf{X}_{[L,p]}(k) \mathbf{g}_{[p]}(k) \quad (6.14)$$

ただし，

$$\mathbf{g}_{[p]}(k) = \mathbf{R}_{[p,p]}^{-1}(k) \mathbf{e}_{[p]}(k) \quad (6.15)$$

$$\mathbf{R}_{[p,p]}(k) = \mathbf{X}_{[L,p]}^H(k) \mathbf{X}_{[L,p]}(k) \quad (6.16)$$

$\mathbf{g}_{[p]}(k)$ をプレフィルタリングベクトルまたは単にプレフィルタと呼ぶものである．(6.14) は，プレフィルタリングベクトルを用いて， $\mathbf{X}_{[L,p]}(k)$ の列ベクトル $\mathbf{x}(k), \dots, \mathbf{x}(k-p+1)$ をフィルタリングした結果が補正ベクトル $\Delta \hat{\mathbf{h}}(k)$ となることを意味している．このプレフィルタリングにより，入力信号 $x(k)$ の時間的な相関が低減され，適応フィルタの収束が速まるという効果がある．

プレフィルタ $\mathbf{g}_{[p]}(k)$ は (6.15) の行列方程式をコレスキー法で解いても求められるが，逆行列の補助定理に基づいた方法 [49] や線形予測係数を利用した方法 [38] により逐次的に更新する方が演算量が少ない．どの方法を用いるにしても，(6.14) は行列とベクトルの積なので， Lp 回の積和演算が必要となる．しかし，この計算は次式のベクトル $\mathbf{z}(k)$ [32, 38] を導入することで回避できる．

$$\mathbf{z}(k-1) = \hat{\mathbf{h}}(k-1) - \sum_{i=1}^{p-1} s_i(k-1) \mathbf{x}(k-i) \quad (6.17)$$

ただし，

$$s_i(k-1) = \mu(g_1(k-i) + g_2(k-i+1) + \dots + g_i(k-1)) \quad (6.18)$$

であり，ベクトル $\mathbf{s}_{[p]}(k)$

$$\mathbf{s}_{[p]}(k) = [s_1(k), s_2(k), \dots, s_p(k)]^T \quad (6.19)$$

は次式により再帰的に更新される．

$$\mathbf{s}_{[p]}(k) = \begin{bmatrix} 0 \\ \mathbf{s}_{[p-1]}(k-1) \end{bmatrix} + \mu \mathbf{g}_{[p]}(k) \quad (6.20)$$

ベクトル $\mathbf{z}(k)$ は (6.17) が示すように，フィルタ係数 $\hat{\mathbf{h}}(k)$ から最近 $p-1$ 時刻の入力信号ベクトル $\mathbf{x}(k-i), i=1, \dots, p-1$ をさし引いたものであり，変形フィルタ係数と呼ぶ．この変形フィルタ係数は次式により再帰的に更新できる．

$$\mathbf{z}(k) = \mathbf{z}(k-1) + s_p(k)\mathbf{x}(k-p+1) \quad (6.21)$$

出力 $y(k)$ は陽にフィルタ係数ベクトルを計算することなく， $\mathbf{z}(k-1)$ から求めることができる．具体的には (6.17) を (6.1) に代入した以下の式による．

$$\begin{aligned} y(k) &= \hat{\mathbf{h}}^H(k-1)\mathbf{x}(k) \\ &= \mathbf{z}^H(k-1)\mathbf{x}(k) + \sum_{i=1}^{p-1} \bar{s}_i(k-1)\mathbf{x}^H(k-i)\mathbf{x}(k) \\ &= \mathbf{z}^H(k-1)\mathbf{x}(k) + \mathbf{s}_{[p-1]}^H(k-1)\mathbf{r}_{[p-1]}(k) \end{aligned} \quad (6.22)$$

ただし， $\bar{\cdot}$ は複素共役を意味し， $\mathbf{r}_{[p-1]}(k)$ は次式で定義される．

$$\mathbf{r}_{[p-1]}(k) = [\mathbf{x}^H(k-1)\mathbf{x}(k), \mathbf{x}^H(k-2)\mathbf{x}(k), \dots, \mathbf{x}^H(k-p+1)\mathbf{x}(k)]^T \quad (6.23)$$

この相関ベクトル $\mathbf{r}_{[p-1]}(k)$ は次式で更新できる．

$$\mathbf{r}_{[p-1]}(k) = \mathbf{r}_{[p-1]}(k-1) + \bar{\mathbf{x}}_{[p-1]}(k-1)x(k) - \bar{\mathbf{x}}_{[p-1]}(k-L-1)x(k-L) \quad (6.24)$$

以上のように，出力 (6.22) の $y(k)$ はベクトル $\mathbf{z}(k-1)$ ， $\mathbf{x}(k)$ ， $\mathbf{r}_{[p-1]}(k)$ ，と $s_{[p-1]}(k-1)$ から計算でき， $\mathbf{z}(k-1)$ ， $\mathbf{r}_{[p-1]}(k)$ ，と $s_{[p-1]}(k-1)$ はそれぞれ (6.21)，(6.24) と (6.20) により更新できるので，(6.11) の行列ベクトル積を計算する必要はなくなる．

これまで示した計算方法により，射影アルゴリズムの演算量は $2L + \alpha(p)$ （ほとんどが積和演算）となる． $2L$ の演算は (6.22) の右辺第一項と変形ベクトル $\mathbf{z}(k)$ の更新 (6.21) によるものである． $\alpha(p)$ は (6.22) の右辺第二項，ベクトル $\mathbf{r}_{[p-1]}(k)$ ， $\mathbf{e}_{[p]}(k)$ ，と $s_{[p]}(k)$ の

それぞれ (6.24), (6.12), と (6.20) による更新, そして, 行列方程式 (6.15) を解くことに
 対応する量である. (6.22) の右辺第二項とベクトル $\mathbf{r}_{[p-1]}(k)$, $\mathbf{e}_{[p]}(k)$, と $s_{[p]}(k)$ の更新に
 は, それぞれ, p と $4p$ の演算が必要である. (6.15) を解くのに必要となる演算量は, 解法
 に依存する. コレスキー法, 逆行列の補助定理, 線形予測係数を用いる方法では, それぞ
 れ, $(1/6)p^3 + p^2$, $5p^2$, $15p$ の演算量が必要となる.

以上より, コレスキー法を用いた場合の総演算量は, $2L + (1/6)p^3 + p^2 + 5p$ となる. 射
 影の次数 p がフィルタ長 L より十分小さい場合には, $\alpha(p)$ も $2L$ より小さくなり, AP 法
 の演算量は NLMS 法の $2L$ と同程度である.

以上, 6.2.1 と 6.2.2 では, 一入力, 一出力の適応フィルタ系における, AP 法の基本的
 な考え方と高速算法を示した.

6.3 FILTERED-X 系のための射影アルゴリズム

Filtered-x は音場, 騒音制御での基本的な適応フィルタの構成である. この節では, こ
 の filtered-x 系に対応した高速な AP 法を導出する. まず, 最も簡単な一入力一出力系につ
 いて導出し, その拡張し多入力多出力系の高速 AP 法を導出する.

6.3.1 Case 1: 一入力一出力系

図 6.2 に示した filtered-x 系を考える. ここでは, 入力, 二次音源 (secondary source,
 SS), そして, 制御点 (control point, CP) の数がそれぞれ一つである. 同図で, \mathbf{c} は二次音
 源 SS から制御点 CP の間のインパルス応答を表わし, この \mathbf{c} は適応フィルタの動作の前
 に推定しておく. 入力信号 $x(k)$ は \mathbf{c} でフィルタリングされ, 適応フィルタへ供給される $u(k)$
 となる. この filtered-x 信号 $u(k)$ は次式のように入力信号に制御点と二次音源の間のイン
 パルス応答を畳み込みとして表現される.

$$u(k) = \mathbf{c}_{[L_c]}^H \mathbf{x}_{[L_c]}(k) \quad (6.25)$$

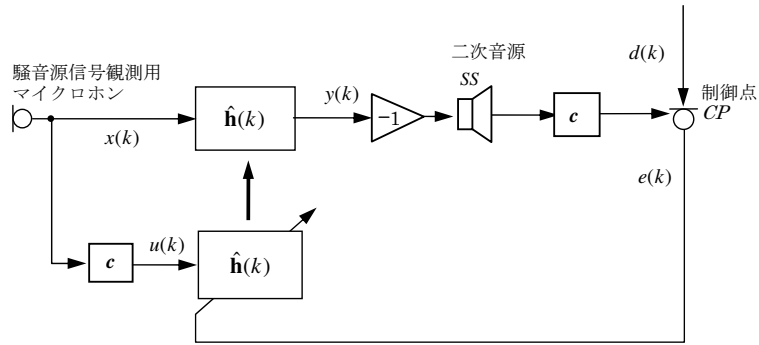


図 6.2: 一入力一出力 filtered-x 適応フィルタの構成

ただし， L_c は c の要素数である．二次音源からの出力信号 $y(k)$ は，(6.1) に示した通常の適応フィルタと同様に，騒音源信号観測用マイクロホンの出力として得られる入力信号 $x(k)$ とフィルタ $\hat{\mathbf{h}}(k)$ とを畳み込むことで得られる．制御点で観測される二次音源からの信号と $u(k)$ を $\hat{\mathbf{h}}(k)$ でフィルタリングした信号は等しくなるから，適応フィルタの修正ベクトルが満たすべき連立方程式は，(6.5) で入力信号 \mathbf{x} を filtered-x 信号 \mathbf{u} で置き換えた次式となる．

$$\begin{aligned}
 d(k) &= [\hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k)]^H \mathbf{u}(k) \\
 d(k-1) &= [\hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k)]^H \mathbf{u}(k-1) \\
 &\vdots \\
 d(k-p+1) &= [\hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k)]^H \mathbf{u}(k-p+1)
 \end{aligned} \tag{6.26}$$

ただし，

$$\mathbf{u}(k) = [u(k), u(k-1), \dots, u(k-L+1)]^T \tag{6.27}$$

この連立方程式 (6.26) が filtered-x AP 法を導出する出発点となる．(6.26) から，filtered-x 系での $\Delta\hat{\mathbf{h}}(k)$ は 6.2.1 で $\mathbf{x}(k)$ を $\mathbf{u}(k)$ で置き換えることで同様に導出ができる．(6.26) を行列形式でかくと，

$$\mathbf{d}_{[p]}^T(k) = [\hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k)]^H(k) \mathbf{U}_{[L,p]}(k) \tag{6.28}$$

または，等価的に

$$\mathbf{e}_{[p]}^H(k) = \Delta \hat{\mathbf{h}}^H(k) \mathbf{U}_{[L,p]}(k) \quad (6.29)$$

とかける．ただし，

$$\mathbf{d}_{[p]}(k) = [d(k), d(k-1), \dots, d(k-p+1)]^T \quad (6.30)$$

$$\mathbf{U}_{[L,p]}(k) = [\mathbf{u}(k), \mathbf{u}(k-1), \dots, \mathbf{u}(k-p+1)] \quad (6.31)$$

$$\mathbf{e}_{[p]}^H(k) = \mathbf{d}_{[p]}^T(k) - \hat{\mathbf{h}}^H(k-1) \mathbf{U}_{[L,p]}(k) \quad (6.32)$$

である．(6.32)の右辺において，制御点で残留騒音として観測できる信号は， $\mathbf{e}(k)$ の一番目の要素 $e(k)$ だけであり，他の $\mathbf{e}(k)$ の要素は過去の誤差信号から計算する必要がある．

さて，修正ベクトル $\Delta \hat{\mathbf{h}}(k)$ は前項と同様に (6.29) の最小ノルム解として次式で与えられる．

$$\Delta \hat{\mathbf{h}}(k) = \mathbf{U}_{[L,p]}(k) \mathbf{g}_{[p]}(k) \quad (6.33)$$

ただし，プレフィルタ $\mathbf{g}_{[p]}(k)$ は次式で定義される．

$$\mathbf{g}_{[p]}(k) = \mathbf{R}_{[p,p]}^{uu}{}^{-1}(k) \mathbf{e}_{[p]}(k) \quad (6.34)$$

$$\mathbf{R}_{[p,p]}^{uu}(k) = \mathbf{U}_{[L,p]}^H(k) \mathbf{U}_{[L,p]}(k) \quad (6.35)$$

前章でみたような演算量の低減は filtered-x 系にも適用可能である．以下にその演算量低減手法を示す． $\mathbf{e}(k)$ の更新は 6.2.2 の (6.12) で導出したように，次式で行われる．

$$\mathbf{e}_{[p]}(k) = \left[e(k), (1-\mu) \mathbf{e}_{[p-1]}^H(k-1) \right]^H \quad (6.36)$$

次式で $\hat{\mathbf{h}}(k)$ をから，変形フィルタベクトルが定義される．

$$\mathbf{z}(k-1) = \hat{\mathbf{h}}(k-1) - \sum_{i=1}^{p-1} s_i(k-1) \mathbf{u}(k-i) \quad (6.37)$$

ここで，filtered-x 信号 \mathbf{u} がフィルタ係数 $\hat{\mathbf{h}}(k-1)$ から引かれている．前節の AP 法の場合は，(6.17) で入力信号 \mathbf{x} が引かれていた．平滑化プレフィルタ係数 $s_i(k-1)$ は前節と同

じく, (6.18), (6.19) で定義され, (6.20) により更新される. 変形フィルタベクトル $\mathbf{z}(k)$ は次式で更新される.

$$\mathbf{z}(k) = \mathbf{z}(k-1) + s_p(k)\mathbf{u}(k-p+1) \quad (6.38)$$

そして, 出力 $y(k)$ は, この変形フィルタベクトル $\mathbf{z}(k)$ を用いて次式で計算される.

$$\begin{aligned} y(k) &= \hat{\mathbf{h}}^H(k-1)\mathbf{x}(k) \\ &= \mathbf{z}^H(k-1)\mathbf{x}(k) + \sum_{i=1}^{p-1} \bar{s}_i(k-1)\mathbf{u}^H(k-i)\mathbf{x}(k) \\ &= \mathbf{z}^H(k-1)\mathbf{x}(k) + \mathbf{r}_{[p-1]}^{ux}(k)^T \bar{\mathbf{s}}_{[p-1]}(k-1) \end{aligned} \quad (6.39)$$

ただし,

$$\mathbf{r}_{[p-1]}^{ux}(k) = [\mathbf{u}^H(k-1)\mathbf{x}(k), \mathbf{u}^H(k-2)\mathbf{x}(k), \dots, \mathbf{u}^H(k-p+1)\mathbf{x}(k)]^T \quad (6.40)$$

ここで, $y(k)$ を求めるのに $x(k)$ と $u(k)$ の相互相関行列が必要であるが, 前章の AP 法では (6.22) のように $x(k)$ の自己相関が必要であった. 相互相関ベクトル $\mathbf{r}_{[p-1]}^{ux}(k)$ も次式により再帰的に更新できる.

$$\mathbf{r}_{[p-1]}^{ux}(k) = \mathbf{r}_{[p-1]}^{ux}(k-1) + \bar{\mathbf{u}}_{[p-1]}(k-1)x(k) - \bar{\mathbf{u}}_{[p-1]}(k-L-1)x(k-L) \quad (6.41)$$

アルゴリズム 6.1 にまとめたように, 一入力一出力制御点の filtered-x 系での高速 AP 法は (6.25), (6.41), (6.39), (6.36), (6.34), (6.20), と (6.38) であり, これらの手順に要する演算量は, $2L + L_c + \alpha(p)$ であり, filtered-x NLMS の $2L + L_c$ とほぼ同程度である. ここで, 通常の高速度 AP 法の $2L$ に加え, L_c が filtered-x 信号 $u(k)$ を (6.25) により求めるために必要となる.

演算	式番号	演算量
1) $u(k) = \mathbf{c}_{[L_c]}^H \mathbf{x}_{[L_c]}(k)$	(6.25)	L_c
2) $\mathbf{r}_{[p-1]}^{ux}(k) = \mathbf{r}_{[p-1]}^{ux}(k-1) + \bar{\mathbf{u}}_{[p-1]}(k-1)x(k) - \bar{\mathbf{u}}_{[p-1]}(k-L-1)x(k-L)$	(6.41)	$2p$
3) $y(k) = \mathbf{z}^H(k-1)\mathbf{x}(k) + \mathbf{r}_{[p-1]}^{uxT}(k)\bar{\mathbf{s}}_{[p-1]}(k-1)$	(6.39)	$L+p$
4) $\mathbf{e}_{[p]}(k) = [e(k), (1-\mu)\mathbf{e}_{[p-1]}^H(k-1)]^H$	(6.36)	p
5) $\mathbf{g}_{[p]}(k) = \mathbf{R}_{[p,p]}^{uu-1}(k)\mathbf{e}_{[p]}(k)$	(6.34)	$O(p^3)$
6) $\mathbf{s}_{[p]}(k) = \begin{bmatrix} 0 \\ \mathbf{s}_{[p-1]}(k-1) \end{bmatrix} + \mu\mathbf{g}_{[p]}(k)$	(6.20)	p
7) $\mathbf{z}(k) = \mathbf{z}(k-1) + s_p(k)\mathbf{u}(k-p+1)$	(6.38)	L

アルゴリズム 6.1: 一入力一出力制御点の filtered-x 系での高速射影法

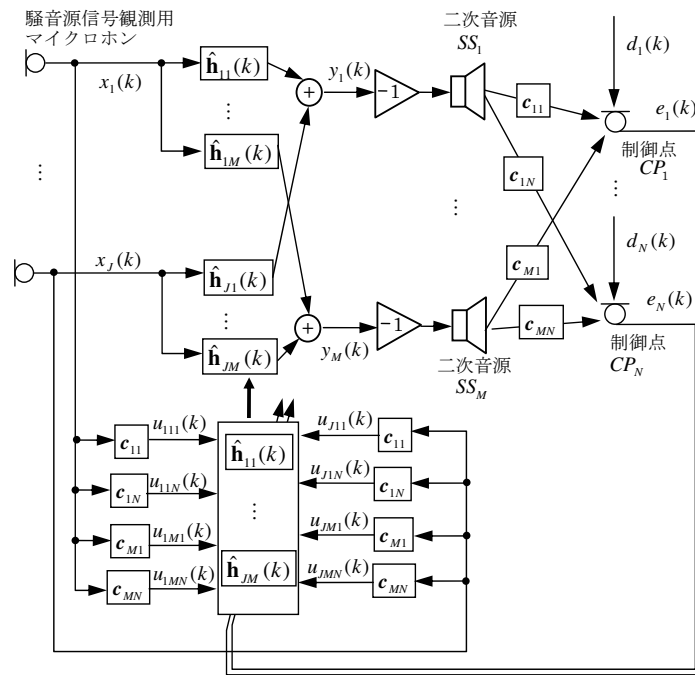


図 6.3: 複数入力複数二次音源複数出力 filtered-x 適応フィルタの構成

6.3.2 Case 2: 複数入力，複数二次音源，複数制御点

本項では前項の結果を入力，出力（二次音源）と制御点の数が全て複数ある一般的な場合に拡張する．図 6.3 に J 個の入力， M 個の二次音源， N 個の制御点からなる filtered-x 系を示す．同図で， \mathbf{c}_{mn} ($m = 1, 2, \dots, M, n = 1, 2, \dots, N$) は二次音源 SS_m から制御点 CP_n の間のインパルス応答を表わし， u_{jmn} ($j = 1, 2, \dots, J$) は次式にあるように x_j をインパルスレスポンス \mathbf{c}_{mn} で畳み込んだ filtered-x 信号を表わす．

$$u_{jmn}(k) = \mathbf{c}_{mn,[L_c]}^H \mathbf{x}_{j,[L_c]}(k) \quad (6.42)$$

ただし， $\mathbf{x}_{j,[L_c]}(k)$ は次式で定義される j 番目の騒音源信号観測用マイクロホンの出力として得られる入力信号ベクトルを意味する．

$$\mathbf{x}_{j,[L_c]}(k) = [x_j(k), x_j(k-1), \dots, x_j(k-L_c+1)]^T \quad (6.43)$$

二次音源の出力 $y_m(k)$ は次式で計算される．

$$\mathbf{y}_{[M]}(k) = \hat{\mathbf{H}}_{[JL,M]}^H(k-1) \mathbf{x}_{[JL]}(k) \quad (6.44)$$

ただし，

$$\mathbf{y}_{[M]}(k) = [y_1(k), y_2(k), \dots, y_M(k)]^T \quad (6.45)$$

$$\hat{\mathbf{H}}_{[JL,M]}(k-1) = \begin{bmatrix} \hat{\mathbf{h}}_{11}(k-1) & \hat{\mathbf{h}}_{12}(k-1) & \cdots & \hat{\mathbf{h}}_{1M}(k-1) \\ \hat{\mathbf{h}}_{21}(k-1) & \ddots & & \hat{\mathbf{h}}_{2M}(k-1) \\ \vdots & & \ddots & \vdots \\ \hat{\mathbf{h}}_{J1}(k-1) & \hat{\mathbf{h}}_{J2}(k-1) & \cdots & \hat{\mathbf{h}}_{JM}(k-1) \end{bmatrix} \quad (6.46)$$

$$\mathbf{x}_{[JL]}(k) = [\mathbf{x}_1^T(k), \mathbf{x}_2^T(k), \dots, \mathbf{x}_J^T(k)]^T \quad (6.47)$$

ここで， $\hat{\mathbf{h}}_{jm}(k-1)$ は j 番目の入力と m 番目の出力の間にあるフィルタ係数ベクトルを表わす．filtered-x 信号 $u_{jmn}(k)$ は通常時間とチャンネル j, m, n に関して相関があり，これらの時間的，チャンネル間の相関が既存の NLMS 法や MELMS (multiple error filtered-x LMS) 法 [47] 法の収束特性を劣化させている．

AP法では、修正ベクトル $\Delta\hat{\mathbf{h}}(k)$ をステップサイズ $\mu = 1$ で更新したフィルタ係数 $\hat{\mathbf{h}}(k) (= \hat{\mathbf{h}}(k-1) + \Delta\hat{\mathbf{h}}(k))$ が過去 p 時刻の間、 N 個の制御点で目的信号を生成するように修正する。これを連立方程式で表わすと次式となる。

$$\begin{aligned}
d_1(k) &= [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{u}_{1,[JML]}(k) \\
d_2(k) &= [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{u}_{2,[JML]}(k) \\
&\vdots \\
d_N(k) &= [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{u}_{N,[JML]}(k) \tag{6.48} \\
d_1(k-1) &= [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{u}_{1,[JML]}(k-1) \\
&\vdots \\
d_N(k-p+1) &= [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{u}_{N,[JML]}(k-p+1)
\end{aligned}$$

ここで、フィルタ係数 $\hat{\mathbf{h}}_{[JML]}(k)$ と修正ベクトル $\Delta\hat{\mathbf{h}}_{[JML]}(k)$ を以下のように再定義する。

$$\hat{\mathbf{h}}_{[JML]}(k) = [\hat{\mathbf{h}}_{11}^T(k), \hat{\mathbf{h}}_{21}^T(k), \dots, \hat{\mathbf{h}}_{J_1}^T(k), \hat{\mathbf{h}}_{12}^T(k), \dots, \hat{\mathbf{h}}_{JM}^T(k)]^T \tag{6.49}$$

$$\Delta\hat{\mathbf{h}}_{[JML]}(k) = [\Delta\hat{\mathbf{h}}_{11}^T(k), \Delta\hat{\mathbf{h}}_{21}^T(k), \dots, \Delta\hat{\mathbf{h}}_{J_1}^T(k), \Delta\hat{\mathbf{h}}_{12}^T(k), \dots, \Delta\hat{\mathbf{h}}_{JM}^T(k)]^T \tag{6.50}$$

filtered-x 信号 $\mathbf{u}_{n,[JML]}(k)$ は次式で定義される。

$$\mathbf{u}_{n,[JML]}(k) = [\mathbf{u}_{11n}^T(k), \mathbf{u}_{21n}^T(k), \dots, \mathbf{u}_{J_1n}^T(k), \mathbf{u}_{12n}^T(k), \dots, \mathbf{u}_{JMn}^T(k)]^T \tag{6.51}$$

$$\mathbf{u}_{jmn}(k) = [u_{jmn}(k), u_{jmn}(k-1), \dots, u_{jmn}(k-L+1)]^T \tag{6.52}$$

また、 $pN \leq JML$ を仮定している。(6.48) と単一入力、単一出力、単一制御点の場合の(6.26)を比較すると、(6.48)には N 倍の方程式があり、 $\hat{\mathbf{h}}_{[JML]}(k)$ と(6.48)の $\Delta\hat{\mathbf{h}}_{[JML]}(k)$ の要素の数は JM 倍である。

(6.48) を行列形式でかくと、

$$\mathbf{d}_{[Np]}^T(k) = [\hat{\mathbf{h}}_{[JML]}(k-1) + \Delta\hat{\mathbf{h}}_{[JML]}(k)]^H \mathbf{U}_{[JML,Np]}(k) \tag{6.53}$$

あるいは、等価的に

$$\mathbf{e}_{[Np]}^H(k) = \Delta\hat{\mathbf{h}}_{[JML]}^H(k) \mathbf{U}_{[JML,Np]}(k) \tag{6.54}$$

となる．ただし，

$$\mathbf{d}_{[Np]}(k) = [d_1(k), d_2(k), \dots, d_N(k), d_1(k-1), \dots, d_N(k-p+1)]^T \quad (6.55)$$

$$\mathbf{U}_{[JML, Np]}(k) = [\mathbf{U}_{[JML, N]}(k), \mathbf{U}_{[JML, N]}(k-1), \dots, \mathbf{U}_{[JML, N]}(k-p+1)] \quad (6.56)$$

$$\mathbf{U}_{[JML, N]}(k) = [\mathbf{u}_1(k), \mathbf{u}_2(k), \dots, \mathbf{u}_N(k)] \quad (6.57)$$

$$\mathbf{e}_{[Np]}^H(k) = \mathbf{d}_{[Np]}^T(k) - \hat{\mathbf{h}}_{[JML]}^H(k-1)\mathbf{U}_{[JML, Np]}(k) \quad (6.58)$$

(6.58) の左辺では， $\mathbf{e}_{[Np]}(k)$ の先頭の N 個の要素 $e_n(k)$ だけが制御点で残留騒音として観測され，残りの要素は以下の段落で述べるように，以前の誤差から計算する必要がある．

前項と同様に， $\Delta \hat{\mathbf{h}}(k)$ は (6.54) の最小ノルム解として次式で与えられる．

$$\Delta \hat{\mathbf{h}}_{[JML]}(k) = \mathbf{U}_{[JML, Np]}(k)\mathbf{g}_{[Np]}(k) \quad (6.59)$$

ここで，プレフィルタ $\mathbf{g}_{[Np]}(k)$ は次式で定義される．

$$\mathbf{g}_{[Np]}(k) = \mathbf{R}_{[Np, Np]}^{uu}{}^{-1}(k)\mathbf{e}_{[Np]}(k) \quad (6.60)$$

ただし， $\mathbf{R}_{[Np, Np]}^{uu}(k)$ は $\mathbf{u}_{n, [JML]}(k)$ 間の相互相関行列であり，次式で定義される．

$$\mathbf{R}_{[Np, Np]}^{uu}(k) = \mathbf{U}_{[JML, Np]}^H(k)\mathbf{U}_{[JML, Np]}(k) \quad (6.61)$$

以上で複数入力複数出力複数制御点の filtered-x 系のための射影アルゴリズムが導出された．

前項と同様に，本項で導出した AP 法の演算量を低減を考える． $\mathbf{e}_{[Np]}(k)$ は再帰的に次式のように導出される．

$$\mathbf{e}_{[Np]}^H(k) = [e_1(k), \dots, e_N(k), (1-\mu)\mathbf{e}_{[N(p-1)]}^H(k-1)]^T \quad (6.62)$$

ただし， $\mathbf{e}_{[N(p-1)]}(k-1)$ はベクトル $\mathbf{e}_{[Np]}(k-1)$ の最初の $N(p-1)p$ 個の要素からなるベクトルを表す．変形係数ベクトルを以下のように再定義する．

$$\mathbf{z}_{[JML]}(k-1) = \hat{\mathbf{h}}_{[JML]}(k-1) - \sum_{i=1}^{p-1} \sum_{n=1}^N s_{i,n}(k-1)\mathbf{u}_{n, [JML]}(k-i) \quad (6.63)$$

ただし ,

$$\mathbf{z}_{[JML]}(k) = [\mathbf{z}_{11}^T(k), \mathbf{z}_{21}^T(k), \dots, \mathbf{z}_{J1}^T(k) \mathbf{z}_{12}^T(k), \dots, \mathbf{z}_{JM}^T(k)]^T \quad (6.64)$$

ここで , $\mathbf{z}_{jm}(k)$ が $\hat{\mathbf{h}}_{jm}(k)$ に対応する変形係数ベクトルである . (6.63) を (6.37) と比較すると , (6.63) の右辺の総和がすべての制御点にわたるようになっていることがわかる .

平滑化プレフィルタ係数 $s_{i,n}(k-1)$ とその時間更新式は以下ようになる .

$$s_{i,n}(k) = \mu (g_{1,n}(k-i) + g_{2,n}(k-i+1) + \dots + g_{i,n}(k-1)) \quad (6.65)$$

$$\mathbf{s}_{[Np]}(k) = \begin{bmatrix} 0_{[N]} \\ \mathbf{s}_{[(N-1)p]}(k-1) \end{bmatrix} + \mu \mathbf{g}_{[Np]}(k) \quad (6.66)$$

ただし ,

$$\mathbf{s}_{[Np]}(k) = [s_{1,1}(k), s_{2,1}(k), \dots, s_{p,1}(k), s_{2,2}(k), \dots, s_{p,N}(k)]^T \quad (6.67)$$

ここで , $s_{i,n}$ と $g_{i,n}$ の下付き添え字 i,n は各々が対応するベクトル $\mathbf{s}_{[Np]}(k)$ と $\mathbf{g}_{[Np]}(k)$ の $(i-1)N+n$ 番目の要素であることを表わす . $\mathbf{z}_{[JML]}(k)$ の時間更新式は次式のように導出できる .

$$\mathbf{z}_{[JML]}(k) = \mathbf{z}_{[JML]}(k-1) + \mathbf{U}_{[JML,N]}(k-p+1)\mathbf{s}_{p,[N]}(k) \quad (6.68)$$

ただし , $\mathbf{s}_{p,[N]}(k)$ は $\mathbf{s}_{[Np]}(k)$ の最後の N 要素を表わす . ベクトル形式の (6.63) を行列形式に書き直すと ,

$$\mathbf{Z}_{[JL,M]}(k-1) = \hat{\mathbf{H}}_{[JL,M]}(k-1) - \sum_{i=1}^{p-1} \sum_{n=1}^N s_{i,n}(k-1) \mathbf{U}_{n,[JL,M]}(k-i) \quad (6.69)$$

となる . ただし , $\hat{\mathbf{H}}_{[JL,M]}(k)$ は (6.46) で定義され ,

$$\mathbf{Z}_{[JL,M]}(k) = \begin{bmatrix} \mathbf{z}_{11}(k) & \mathbf{z}_{12}(k) & \cdots & \mathbf{z}_{1M}(k) \\ \mathbf{z}_{21}(k) & \ddots & & \mathbf{z}_{2M}(k) \\ \vdots & & \ddots & \vdots \\ \mathbf{z}_{J1}(k) & \mathbf{z}_{J2}(k) & \cdots & \mathbf{z}_{JM}(k) \end{bmatrix} \quad (6.70)$$

$$\mathbf{U}_{n,[JL,M]}(k) = \begin{bmatrix} \mathbf{u}_{11n}(k) & \mathbf{u}_{12n}(k) & \cdots & \mathbf{u}_{1Mn}(k) \\ \mathbf{u}_{21n}(k) & \ddots & & \mathbf{u}_{2Mn}(k) \\ \vdots & & \ddots & \vdots \\ \mathbf{u}_{J1n}(k) & \mathbf{u}_{J2n}(k) & \cdots & \mathbf{u}_{JMn}(k) \end{bmatrix} \quad (6.71)$$

である．(6.69) を (6.44) に代入すると， $\mathbf{y}_{[M]}(k)$ は以下のように計算される．

$$\begin{aligned} \mathbf{y}_{[M]}(k) &= \mathbf{Z}_{[JL,M]}^H(k-1)\mathbf{x}_{[JL]}(k) + \\ &\quad \sum_{i=1}^{p-1} \sum_{n=1}^N \bar{s}_{i,n}(k-1)\mathbf{U}_{n,[JL,M]}^H(k-i)\mathbf{x}_{[JL]}(k) \\ &= \mathbf{Z}_{[JL,M]}^H(k-1)\mathbf{x}_{[JL]}(k) + \sum_{i=1}^{p-1} \sum_{n=1}^N \bar{s}_{i,n}(k-1)\mathbf{r}_{i,n,[M]}(k) \\ &= \mathbf{Z}_{[JL,M]}^H(k-1)\mathbf{x}_{[JL]}(k) + \mathbf{R}_{[M,N(p-1)]}^{ux}(k)\bar{\mathbf{s}}_{[N(p-1)]}(k-1) \end{aligned} \quad (6.72)$$

ただし，

$$\mathbf{r}_{i,n,[M]}(k) = [r_{1,i,n}^{ux}(k), r_{2,i,n}^{ux}(k), \dots, r_{M,i,n}^{ux}(k)]^T \quad (6.73)$$

$$r_{m,i,n}^{ux}(k) = \sum_{j=1}^J \mathbf{u}_{jmn}^H(k-i)\mathbf{x}_j(k) \quad (6.74)$$

$$\mathbf{R}_{[M,N(p-1)]}^{ux}(k) = [\mathbf{r}_{1,1}(k), \mathbf{r}_{1,2}(k), \dots, \mathbf{r}_{1,N}(k), \mathbf{r}_{2,1}(k), \dots, \mathbf{r}_{p-1,N}(k)] \quad (6.75)$$

チャンネル間相関 $r_{m,i,n}^{ux}$ すなわち， $\mathbf{R}_{[M,N(p-1)]}^{ux}(k)$ の m 行 $((i-1)N+n)$ 列の要素は次式により更新される．

$$r_{m,i,n}^{ux}(k) = r_{m,i,n}^{ux}(k-1) + \sum_{j=1}^J [\bar{u}_{jmn}(k-i)x_j(k) - \bar{u}_{jmn}(k-L-i)x_j(k-L)] \quad (6.76)$$

以上のように，出力ベクトル $\mathbf{y}(k)$ は $\mathbf{Z}(k-1)$ ， $\mathbf{x}(k)$ ， $\mathbf{s}(k-1)$ と $\mathbf{R}^{ux}(k)$ から求めることができる．

本項で導出した AP 法の手順をアルゴリズム 6.2 に示した．演算量は約 $JML + JMNL_c + \beta(p, J, M, N)$ であり，その JML ， $JMNL$ と $JNML_c$ はそれぞれ (6.72) の右辺の第一項，(6.68) による変形フィルタ係数 $\mathbf{z}(k)$ ，そして (6.42) により filtered-x 信号 $u_{jmn}(k)$ を計算するためのものである．また， $\beta(p, J, M, N)$ は (6.72) の右辺第二項，(6.76) による

$\mathbf{R}_{[M,N(p-1)]}^{ux}(k)$ の更新, (6.62) による $\mathbf{e}_{[Np]}(k)$ の更新, (6.66) による $\mathbf{s}_{[Np]}(k)$ の更新, そして行列方程式 (6.60) を解くことに対応している.

(6.72) の右辺第二項とベクトル, 行列の更新にはそれぞれ $MN(p-1)$ と $2JMN(p-1) + 2N(p-1)$ の演算が必要である. (6.60) を解くのに必要な演算量は解法に依存する. 例えば, コレスキー法, 逆行列の補助定理を基づく方法, 線形予測係数を利用する方法では, それぞれ, $1/6(Np)^3 + (Np)^2$, $(4JM+1)(Np)^2$, $(14N^2p+Np)$ の演算が必要である. 以上より, コレスキー法を用いるなら, $\beta(p, J, M, N) = MN(p-1) + 2JMN(p-1) + 2N(p-1) + 1/6(Np)^3 + (Np)^2$. となる.

p, J, M と N は通常, フィルタ長 L に比べ小さいので, $\beta(p, J, M, N)$ は $(1+N)JML$ より小さい. 従って, AP 法の演算量は従来法である, MELMS 法の $(1+N)JML + JMN L_c$ に近い.

この節では, filtered-x のための AP 法を導出した. まず, 通常の AP 法を単一入出力制御点系の AP 法に変更し, 次に, 多入出力制御点系に対応した AP 法に拡張した. これら提案法は, 元の高速度 AP 法の低演算量でかつ収束が速いという特性を継承している. 収束特性については次節で計算機シミュレーションにより確認する.

演算	式番号	演算量
1) $u_{jmn}(k) = \mathbf{c}_{mn,[L_c]}^H \mathbf{x}_{j,[L_c]}(k)$	(6.42)	$JMNL_c$
2) $r_{m,i,n}^{ux}(k) = r_{m,i,n}^{ux}(k-1) + \sum_{j=1}^J [\bar{u}_{jmn}(k-i)x_j(k) - \bar{u}_{jmn}(k-L-i)x_j(k-L)]$	(6.76)	$JMNp$
3) $\mathbf{y}_{[M]}(k) = \mathbf{Z}_{[JL,M]}^H(k-1)\mathbf{x}_{[JL]}(k) + \mathbf{R}_{[M,N(p-1)]}^{ux}(k)\bar{\mathbf{s}}_{[N(p-1)]}(k-1)$	(6.72)	$JLM + JMN(p-1)$
4) $\mathbf{e}_{[Np]}^H(k) = [e_1(k), \dots, e_N(k), (1-\mu)\mathbf{e}_{[N(p-1)]}^H(k-1)]^T$	(6.62)	$N(p-1)$
5) $\mathbf{g}_{[Np]}(k) = \mathbf{R}_{[Np,Np]}^{uu}(k)^{-1}\mathbf{e}_{[Np]}(k)$	(6.60)	$O((Np)^3)$
6) $\mathbf{s}_{[Np]}(k) = \begin{bmatrix} 0_{[N]} \\ \mathbf{s}_{[(N-1)p]}(k-1) \end{bmatrix} + \mu\mathbf{g}_{[Np]}(k)$	(6.66)	Np
7) $\mathbf{z}_{[JML]}(k) = \mathbf{z}_{[JML]}(k-1) + \mathbf{U}_{[JML,N]}(k-p+1)\mathbf{s}_{p,[N]}(k)$	(6.68)	$JMLN$

アルゴリズム 6.2: 多入出力制御点系での高速射影法

6.4 計算機シミュレーション

6.4.1 実験条件

提案法の効果を見るために、以下の三つの系 (a), (b) と (c) での計算機シミュレーションを行った。(a) 最初の系は図 6.1 にある通常の適応フィルタリングの系であり、AP 法の効果だけを見るためのものである。入力信号は二番目の系の filtered-x 信号 $u(k)$ である。(b) 二番目の系は (1-2-1)、すなわち、一入力、二出力、一制御点の filtered-x 系である。(c) 最後、三番目の系は (1-6-2) すなわち、一入力、六出力、二制御点の filtered-x 系である。この系は、制御点に関するチャンネル間相関の影響を見るための系である。チャンネル間相関は制御点間の距離に依存すると予想されるので制御点間の距離が 30 cm と 70 cm の二つの場合の計算機シミュレーションを行った。

騒音源（一次音源）と二次音源から制御点までのインパルス応答は大きさが 3.0(D) x 4.3(W) x 3.0(H) m の部屋で実測したものをを用いた。(c) でのマイクロホン（制御点）とスピーカ（一次、二次騒音源）の配置を図 6.4 に示した。サンプリング周波数は 1000 Hz で、80-400 Hz の帯域通過フィルタを用いた。残響時間は通過帯域で 400 ms である。計算機シミュレーションでは、60 タップ (60ms) を切り出して用いた。適応フィルタの条件としては、適応フィルタ長さは全て 60 とし、ステップサイズは収束が最速になる値を実験的に求めてそれをを用いた。騒音源信号は IIR フィルタ $x(k) = 0.9(x(k-1) + x(k-2)) + w(k)$ により生成した有色雑音である。ただし、 $w(k)$ は白色ガウス系列である。

6.4.2 実験結果

図 6.5-6.8 に誤差信号のパワーの収束曲線を示した。零初期条件、すなわち、 $\hat{h}(k) = u(k) = 0$ は、非零初期条件より速く収束することが多く、適応フィルタの追従性能を見るのには不適切であるので、これを避けるためにインパルス応答を時刻 $k = 1000$ で変えている。

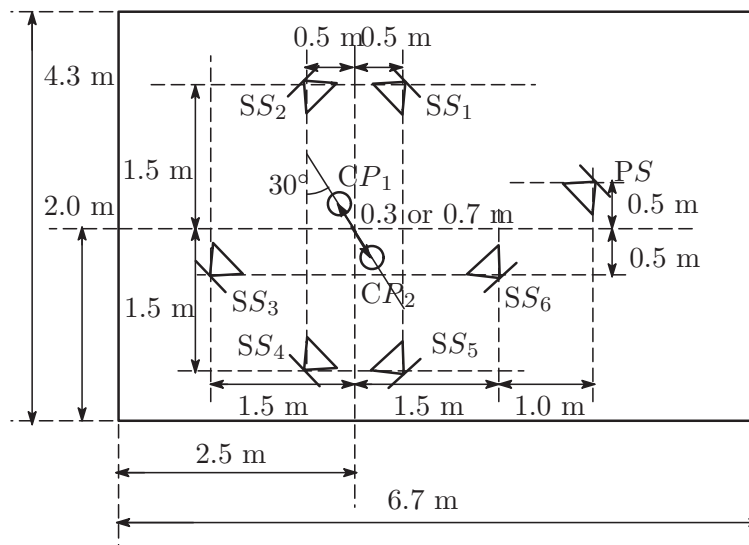


図 6.4: 騒音源, 2 次音源, 制御点の配置. PS: 騒音源, SS: 2 次音源 (出力), CP: 制御点

(a) 通常の適応フィルタ系

図 6.5 に実験結果を示す. 同図より, $p \geq 4$ では若干収束速度が向上することがわかる. この向上は時間的な脱相関の効果である.

(b) (1-2-1) filtered-x 系

図 6.6 に実験結果を示す. 同図では, (a) の場合とは異なり, AP 法の効果が顕著である. 例えば, 20 dB 誤差が減衰するのに, NLMS 法では約 2000 サンプル必要なのに, 提案した AP 法では, $p = 4$ で, NLMS 法の場合の半分の約 1000 サンプルしかかからない. 必要な演算量を比較すると, NLMS 法では 360 で, $p = 4$ でコレスキー法を用いた AP 法は 410 である. つまり, 提案法は従来の NLMS 法に比べ, 1.2 倍の演算量で二倍の収束速度を得ている.

(c) (1-6-2) filtered-x 系

図 6.7 と 6.8 に結果を示す. 図 6.7 が制御点間距離が 30 cm の場合の結果であり, 図 6.8 は制御点間距離が 70 cm の場合の結果である. これら二つの結果は射影次数 p を増加することに関して異なった振る舞いをみせる. この違いはチャネル相関の制御点間距離依存性を反

映している。図 6.7 の制御点間距離が 30 cm の場合、射影次数 p を大きくすることによる効果が小さい。一方、図 6.8 では、NLMS($p = 1$) に比べて射影次数が ($p > 2$) の方が収束が速く、AP 法の効果が顕著である。

実験結果をまとめると、

- もし、filtered-x 信号に時間的な自己相関があるなら、提案した AP 法は収束特性を改善する（小さな、filtered-x 信号の時間的自己相関が収束速度に大きく劣化させることがある。）
- AP 法による収束速度向上の効果は制御点間の距離に大きく依存する。

6.5 本章のまとめ

音場再生や能動騒音制御で用いられる多入力多出力多制御点の filtered-x 適応系のための高速 AP 法を導出した。この提案法は従来法である NLMS 法に比べフィルタの収束速度が速く、かつ、演算量の増加も僅かであるという利点を有する。計算機シミュレーションによる能動騒音制御の実験では、提案法が NLMS 法に対し、二倍の収束速度を 1.2 倍の演算量で得られることがわかり、提案法の有効性が確認された。

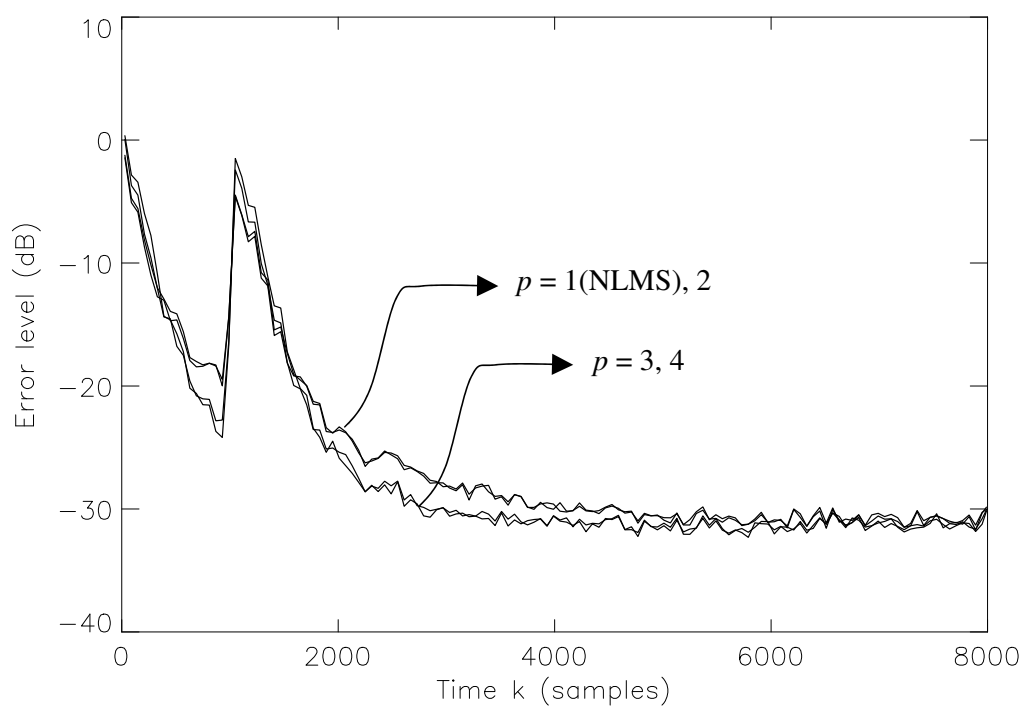


図 6.5: 通常の適応フィルタ系での誤差レベルの収束曲線.

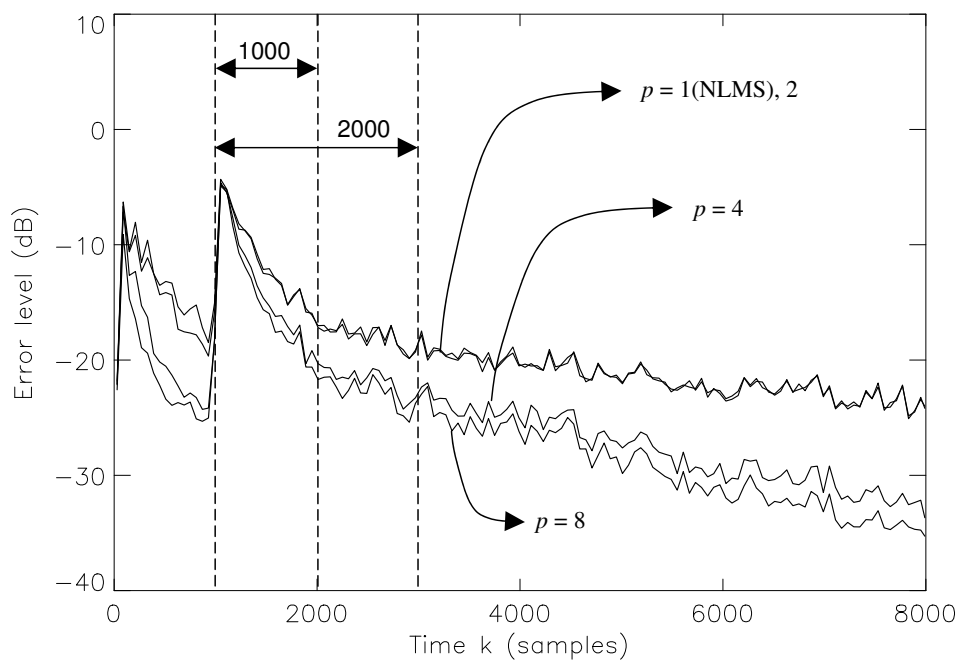


図 6.6: (1-2-1) 系での誤差レベルの収束曲線.NLMS 法に比べて提案した AP 法では収束が速い. 20 dB 誤差が減衰するのに, NLMS 法では約 2000 サンプル必要なのに対し, 提案した AP 法では, $p = 4$ で, NLMS 法の場合の半分の約 1000 サンプルしかかからない.

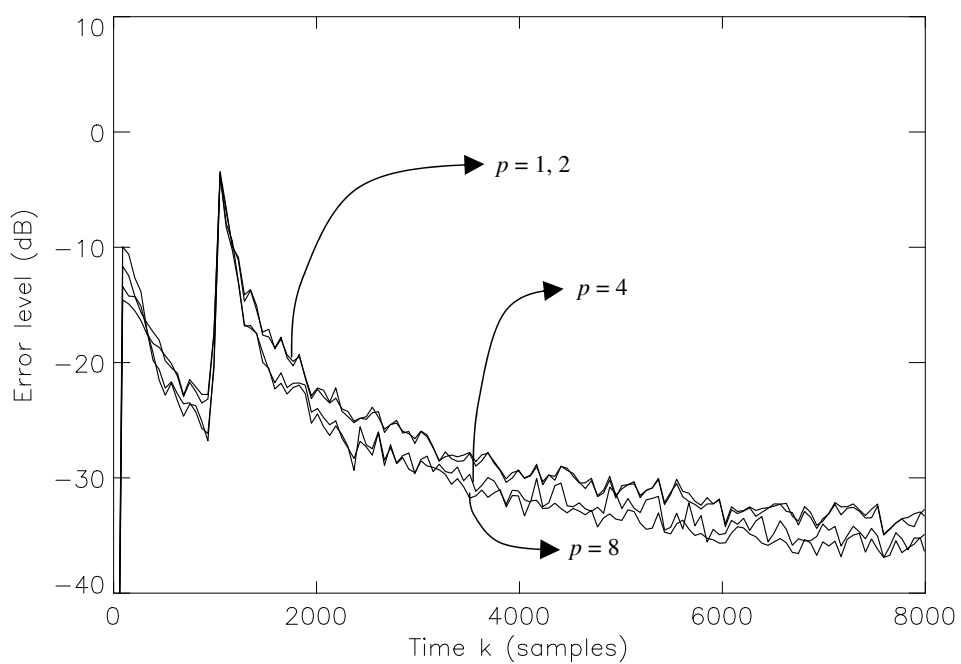


図 6.7: (1-6-2) 系での誤差レベルの収束曲線. 制御点間の距離は 30 cm. 提案した AP 法と NLMS 法はほぼ同じ収束特性を示す .

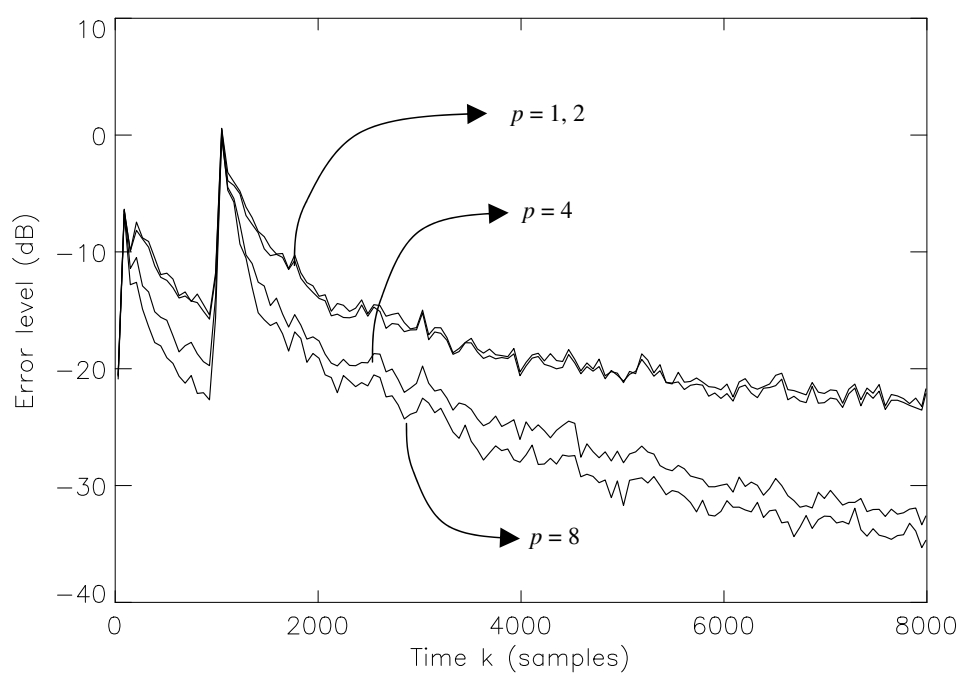


図 6.8: (1-6-2) 系での誤差レベルの収束曲線. 制御点間の距離は 70 cm. 提案した AP 法は NLMS 法より収束速度が速い.

第7章 結論

本論文は、拡声通話系における音響エコーキャンセラへを主な適用として、低演算量でかつ収束の速い適応フィルタアルゴリズムを開発を目指してきた著者の研究の成果をまとめたものである。この研究の背景には、テレビ会議などの拡声通話系が通信の大容量化とともに利用が拡大すると予想されることと、半導体技術の進歩により、複雑で高精度な処理が小型の装置で実現できるようになってきたことが挙げられる。さらに、音声のような有色信号に対しても速い収束速度をもつアフィン射影法 (AP) 法が、その演算量の大きさゆえに省みられておらず、低演算化とアルゴリズムの収束特性への理解を深めることにより、エコーキャンセラへの用途を超えて適応制御が必要な一般的な用途にもインパクトがあると期待されることが研究の動機となっている。

第1章では、研究の背景として、スピーカとマイクロホンを用いた拡声通話系でのエコーの発生と知覚そしてエコーを消去する方式について述べた。

第2章では、エコーを消去するための代表的適応アルゴリズムとして (N)LMS 法、AP 法、RLS 法等の概要を述べ、計算機シミュレーションによりそれらの白色雑音と有色雑音等に対する収束特性を比較し、白色雑音信号入力に対しては、NLMS 法、AP 法と RLS 法のどの適応フィルタアルゴリズムでも同様の収束が得られること、有色雑音信号と音声入力に対しては NLMS 法と AP 法では収束が遅くなるが AP 法では射影次数を上げることで収束速度が RLS 法並みに改善することをみた。

第3章では、AP 法が前章でみたような収束特性を有することの定性的、定量的でかつ容易な理解を目的として AP 法の収束特性の解析を行った。提案した解析の特徴は、従来

の特異値分解に基づくフィルタ長個の固有モードに分解するアプローチではなく、射影次数個の NLMS 法の収束特性に分解するアプローチを取ったことである。

まず、AP 法を、入力と参照信号を白色化する同一のフィルタバンクと射影次数分の NLMS フィルタの集まりとみなす複合 NLMS モデルを提示した。次に、複合 NLMS モデルに基づいて、AP 法の過渡および定常状態の自乗係数誤差を NLMS のそれを用いた次数再帰的に近似式を示した。結果として、AP 法の過渡誤差は、入力信号の予測残差を入力とする NLMS 法を繰り返し適用することで近似されることが示され、AP 法が収束が NLMS に比べて速い理由とその程度が理解可能となった。また、定常誤差は NLMS フィルタで生じる定常誤差の和として近似するという解釈と近似式を示した。四つの入力信号に対して計算機シミュレーションを行い、これらの近似の有効性を確認した。特に自己相関の強い信号では、従来の解析ではよい収束特性の近似が得られなかったが、提案法では良好な収束特性の近似ができることが分かった。

AP 法の収束特性をより簡単で、研究の進んでいる NLMS 法に基づいて解釈することで、これまでよりも単純明快な理解が可能となったといえる。

第 4 では、AP 法の演算量低減手法を提案した。提案した手法により、AP 法の演算量は $(p+1)L + O(p^3)$ から $2L + 20p$ (L はフィルタ長、 p は射影次数) に低減した。低減化は二つの部分からなり、一つは従来法で $O(p^3)$ 要していたプレフィルタの導出を再帰計算により約 $15p$ で行うというものであり、他方は変形フィルタベクトルを導入することにより、従来、 pL の演算を要したフィルタの更新とプレフィルタリングを $L + 3p$ の変形フィルタベクトルの更新に置換えるというものである。音響エコーキャンセラを想定したフィルタ長 $L = 1024$ 、射影次数 $p = 48$ の条件の場合、演算量は従来計算法に比べ $1/20$ 程度まで削減可能である。

第 2 章の計算機シミュレーションでみたように、高次の AP 法は FTF (高速 RLS) 法と同程度の収束速度をもつことから、提案した高速 AP 法により、音声のような自己相関の強い信号に対しても、FTF 法以下の演算で同等の収束速度を得ることが可能となったとい

える．

5章では，逐次等価ブロック高速 AP 法を導出した．この手法は従来のブロック処理にあった収束速度の低下という問題を克服し，その収束速度は逐次 AP 法と同じであるという特徴がある．これは (1) 従来のブロック AP 法の出力と逐次の AP 法の差を補正する項の導入したこと，(2) 高速 FIR フィルタリング手法をフィルタの出力の計算とフィルタの更新に適用することで実現した．また，フィルタ出力とフィルタ更新のブロック長を異なる値にすることができるので出力遅延の短いブロック処理が可能なることも本提案手法の特徴である．音響エコーキャンセラを想定したフィルタ長 $L = 1024$ ，射影次数 $p = 48$ の条件の場合，演算量は前章の計算法に比べ $1/2$ 程度まで削減可能である．

総演算量が固定されている場合に最長のフィルタ長を与えるブロック長を選ぶ手順も示した．一例を挙げると，もし，数百サンプルの遅延が供されれば提案したブロック射影法は高速 AP 法に比べ同じ演算量でフィルタ長を約 1000 からその倍にすることができる．

第 6 章では，音場再生や能動騒音制御で用いられる多入出力多制御点 filtered-x 系へ AP 法を応用した手法を提案した．この提案法は従来法である NLMS 法に比べフィルタの収束速度が速く，かつ，演算量の増加も僅かであるという利点を有する．計算機シミュレーションによる能動騒音制御の実験では，提案法が NLMS 法に対し，二倍の収束速度を 1.2 倍の演算量で得られることがわかり，提案法の有効性が確認された．

以上の研究により，AP 法の演算量が大幅に低減しこれまで利用できなかった高次の AP 法を利用すること可能となり，それによりエコーキャンセラの収束速度が改善し拡声通話系の品質が向上することが期待される．また，AP 法の収束特性の解釈や音場再生や能動騒音制御への適用を示し，今後，様々な応用への道も開いた．

今後の課題としては，本研究等により比較的低演算量で収束速度の速い適応フィルタア

ルゴリズムの理論的側面が明らかになったので、音響エコーキャンセラとして利用するに当たっての実際の側面からのものが重要となる。一つには、安定性に関する課題であり、高速 AP 法に含まれる高速 RLS を固定小数点表現で演算したときの数値的安定化や送話と受話が同時に存在する同時通話に対する適応フィルタの耐性改善がこれに含まれる。また、エコー除去性能の改善に関する課題として、雑音下での除去量の改善、音声スイッチなどの送話信号の振幅を制御する方法との連携によるエコー除去方法の開発などが挙げられる。

付録A NLMS法の定常誤差(3.11)の導出

まず, (3.6) の二乗ノルムの期待値をとる. 射影行列の性質, $\mathbf{P}_x^T(k) = \mathbf{P}_x(k)$, $\mathbf{P}_x(k)\mathbf{P}_x(k) = \mathbf{P}_x(k)$, $\mathbf{P}_x(k)\mathbf{x}(k) = \mathbf{x}(k)$ に注意すると,

$$\begin{aligned}
 E \left[\|\boldsymbol{\epsilon}(k)\|^2 \right] &= \\
 & E \left[\boldsymbol{\epsilon}^T(k-1) [\mathbf{I} - \mu(2-\mu)\mathbf{P}_x(k)] \boldsymbol{\epsilon}(k-1) \right] + \\
 & E \left[\boldsymbol{\epsilon}^T(k-1) [\mathbf{I} - \mu\mathbf{P}_x(k)] \mu \tilde{\mathbf{x}}(k) n(k) \right] + \\
 & E \left[\mu \tilde{\mathbf{x}}^T(k) n(k) [\mathbf{I} - \mu\mathbf{P}_x(k)] \boldsymbol{\epsilon}(k-1) \right] + \\
 & E \left[\mu^2 n^2(k) / \|\mathbf{x}(k)\|^2 \right] \tag{A.1} \\
 &= \\
 & E \left[\boldsymbol{\epsilon}^T(k-1) [\mathbf{I} - \mu(2-\mu)\mathbf{P}_x(k)] \boldsymbol{\epsilon}(k-1) \right] + \\
 & E \left[2\mu(1-\mu) \boldsymbol{\epsilon}^T(k-1) \tilde{\mathbf{x}}(k) n(k) \right] + \\
 & E \left[\mu^2 n^2(k) / \|\mathbf{x}(k)\|^2 \right]
 \end{aligned}$$

(A.1) の右辺の第二項にある期待値は以下の仮定の下で0となる. (3.6) より $\boldsymbol{\epsilon}(k-1)$ は過去 (時刻 $k-1$ 以前の) 入力信号と過去の雑音から成っている. この時, 現時刻 k の雑音 $n(k)$ と入力信号の系列 x および過去の雑音が独立だと仮定すると, $E \left[\boldsymbol{\epsilon}^T(k-1) \tilde{\mathbf{x}}(k) n(k) \right] = E \left[\boldsymbol{\epsilon}^T(k-1) \tilde{\mathbf{x}}(k) \right] E \left[n(k) \right]$ となる. さらに, $n(k)$ の期待値が0と仮定すると, (A.1) の右辺第二項にある期待値は0となる.

(A.1) の右辺第三項は分子と分母が独立事象なので個別に期待値をとることができる. 以

上を考慮すると, (A.1) は,

$$E \left[\|\epsilon(k)\|^2 \right] = E \left[\|\epsilon(k-1)\|^2 \right] - \mu(2-\mu)E \left[\|\mathbf{P}_x(k)\epsilon(k-1)\|^2 \right] + \mu^2 E \left[n^2(k) \right] / E \left[\|\mathbf{x}(k)\|^2 \right] \quad (\text{A.2})$$

となる. ここで, (A.2) に定常状態を意味する条件

$$E \left[\|\epsilon(k)\|^2 \right] = E \left[\|\epsilon(k-1)\|^2 \right] \quad (\text{A.3})$$

を代入し, 定常状態であることを明確にするために, $k \rightarrow \infty$ とすると,

$$\begin{aligned} \lim_{k \rightarrow \infty} E \left[\|\mathbf{P}_x(k)\epsilon(k-1)\|^2 \right] &= \frac{\mu}{2-\mu} \lim_{k \rightarrow \infty} \frac{E[n^2(k)]}{E[\|\mathbf{x}(k)\|^2]} \\ &= \frac{\mu}{2-\mu} \frac{\sigma_n^2}{L\sigma_x^2} \end{aligned} \quad (\text{A.4})$$

の関係が得られる.

(A.4) の左辺を詳しくみるために, (A.4) に (3.7) と (3.8) を代入すると,

$$\begin{aligned} \lim_{k \rightarrow \infty} E \left[\|\mathbf{P}_x(k)\epsilon(k-1)\|^2 \right] &= \\ \lim_{k \rightarrow \infty} E \left[\frac{\epsilon^T(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\epsilon(k-1)}{\|\mathbf{x}(k)\|^2} \right] &= \\ \lim_{k \rightarrow \infty} E \left[\frac{\left(\sum_{i=0}^{L-1} x(k-i)\epsilon_i(k-1) \right)^2}{\|\mathbf{x}(k)\|^2} \right] &= \\ \lim_{k \rightarrow \infty} E \left[\frac{\sum_{i=0}^{L-1} x^2(k-i)\epsilon_i^2(k-1)}{\|\mathbf{x}(k)\|^2} \right] + \\ \lim_{k \rightarrow \infty} E \left[\frac{\sum_{i=0}^{L-1} \sum_{\substack{j=0 \\ j \neq i}}^{L-1} x(k-i)x(k-j)\epsilon_i(k-1)\epsilon_j(k-1)}{\|\mathbf{x}(k)\|^2} \right] & \end{aligned} \quad (\text{A.5})$$

となる. ただし, $\epsilon_i(k-1)$ は, $\epsilon(k-1)$ の i 番目の要素である. 定常状態においては入力 $x(k)$ と係数誤差ベクトル $\epsilon(k-1)$ の各要素が独立であり, また, 係数誤差ベクトル $\epsilon(k)$ の各要素も互いに独立で期待値が 0 であると仮定できるので [45], (A.5) の右辺第二項は 0 となる. また, $x^2(k-i)/\|\mathbf{x}(k)\|^2$ と $\epsilon_i^2(k-1)$ の独立性が仮定できると, (A.5) の右辺第一項は, $E \left[\|\mathbf{x}(k)\|^2 \right] = LE \left[x^2(k) \right]$, $E \left[\sum_{i=0}^{L-1} \epsilon_i^2(k-1) \right] = E \left[\|\epsilon(k-1)\|^2 \right]$ であることから,

$\lim_{k \rightarrow \infty} E \left[\|\epsilon(k-1)\|^2 \right] / L$ となる．これらより，

$$\lim_{k \rightarrow \infty} E \left[\|\mathbf{P}_x(k)\epsilon(k-1)\|^2 \right] L = \lim_{k \rightarrow \infty} E \left[\|\epsilon(k-1)\|^2 \right] \quad (\text{A.6})$$

以上，(A.3)，(A.4)，(A.6) より，定常状態での係数二乗誤差を表す (3.11) が以下のとおり導かれる．

$$\begin{aligned} J_{\text{NLMS},x}(\infty) &\equiv \lim_{k \rightarrow \infty} E \left[\|\epsilon(k)\|^2 \right] = \lim_{k \rightarrow \infty} E \left[\|\epsilon(k-1)\|^2 \right] \\ &= \lim_{k \rightarrow \infty} E \left[\|\mathbf{P}_x(k)\epsilon(k-1)\|^2 \right] L \\ &= \frac{\mu}{2-\mu} \frac{\sigma_n^2}{\sigma_x^2} \end{aligned} \quad (\text{A.7})$$

謝辞

本研究を遂行するにあたり、数多くの方々にご指導とご援助を賜りました。特にお世話になった方々をここに記し、深い感謝の意を表します。

北脇信彦 筑波大学大学院システム情報工学研究科教授には、本論文をまとめるにあたり構成や内容についてご教示とご指導を賜りました。また、椎名 毅 同大学大学院システム情報工学研究科教授、水谷 孝一 同大学大学院システム情報工学研究科教授、山本 幹雄 同大学大学院システム情報工学研究科助教授ならびに、山田 武志 同大学大学院システム情報工学研究科助教授の諸先生方には、ご多忙のところ本研究に関する的確なご指導とご助言を賜りました。

今村明弘 NTT サイバースペース研究所プロジェクトリーダーには、上長として本研究にご理解頂き、本研究の機会を与えて頂きました。片岡俊章 NTT サイバースペース研究所グループリーダーには、上長として本研究にご理解頂き、また、本論文をまとめるにあたり暖かい激励と数多くのご配慮を頂きました。

金田豊 東京電機大学工学部教授には本研究を進め、まとめる上で、音響、信号処理に関する数多くのご指導とご助言を頂きました。牧野昭二 NTT コミュニケーション基礎研究所 部長には研究のきっかけを与えて頂くとともに多くのご指導とご助言を頂きました。

古井貞熙 東京工業大学 情報理工学研究科教授、小林幸雄 千葉工業大学電気電子情報工学科教授、西野 豊 NTT エレクトロニクス BB システム・デバイス事業本部 エレクトロニクス事業部長、山森和彦 NTT アドバンステクノロジー 経営企画部 統括部長、小原 永 NTT アドバンステクノロジー システムソリューション事業本部 情報ナビゲーションシステム事業

ユニット長，東山三樹夫 早稲田大学大学院情報生産システム研究科教授，小島順治 NTT
アドバンステクノロジーメディアインテグレーション事業ユニット長には歴代の研究部上長
として本研究の機会を与えて頂きました。

また，NTT サイバースペース研究所音声情報研究部，NTT サイバースペース研究所音
声言語メディア処理プロジェクトの方々には活発な討論を頂きました。学会，研究会を通
して多くの方からも多くのご助言を頂きました。

本研究は，このような方々の存在があって成し得たものであります。重ねて感謝申し上
げます。

参考文献

- [1] J. M. Cioffi and T. Kailath: “Windowed fast transversal filters adaptive algorithms with normalization”, IEEE Trans. on Acoust., Speech, Signal Processing, **ASSP-33**, 3, pp. 607–625 (1985).
- [2] H. Yasukawa, M. Ogawa and M. Nishino: “Echo return loss required for acoustic echo controller based on subjective assessment”, IEICE Trans., **E-74**, 4, pp. 692–705 (1991).
- [3] 雛元, 前川: “拡張された学習同定法”, 信学論 (C), **95**, 10, pp. 227–234 (1975).
- [4] 尾関, 梅田: “アフィン部分空間への直交射影を用いた適応フィルタ・アルゴリズムとその諸性質”, 信学論 (A), **J67-A**, 2, pp. 126–132 (1984).
- [5] D. M. et al.: “General transmission considerations in telephone conference systems”, IEEE Trans. on Communication Technology, **16**, 1, p. 163 (1968).
- [6] 山森, 牧野, 牧野: “マイクロプロセッサ制御を用いた拡声電話機の構成法”, 信学技法, **EA84-41**, (1984.9).
- [7] S. G. Sankaran and A. A. L. Beex: “Convergence behavior of affine projection algorithms”, IEEE Trans. on Signal Processing, **SP-48**, 4, pp. 1086–1096 (2000).
- [8] D. T. M. Slock: “Underdetermined growing and sliding window covariance fast transversal filter RLS algorithm”, Proc. EUSIPCO-92, pp. 1169–1172 (1992).

- [9] M. Montazeri and P. Duhamel: “A set of algorithms linking NLMS and block RLS algorithms”, *IEEE Trans. on Signal Processing*, **SP-43**, 2, pp. 444–453 (1995).
- [10] D. R. Morgan and S. G. Kratzer: “On a class of computationally-efficient, rapidly-converging, generalized NLMS algorithms”, *IEEE Signal Processing Lett.*, **3**, 8, pp. 245–247 (1996).
- [11] 黒沢, 古沢: “適応アルゴリズムの幾何学的解釈”, *信学論 (A)*, **J71-A**, 2, pp. 343–347 (1988).
- [12] S. Haykin: “Adaptive Filter Theory”, chapter 15, pp. 570–585, Prentice-Hall, second edition (1991).
- [13] E. Eleftheriou and D. D. Falconer: “Tracking properties and steady state performance of RLS algorithms”, *IEEE Trans. on Acoust., Speech, Signal Processing*, **ASSP-34**, pp. 1097–1110 (1986).
- [14] J. M. Cioffi and T. Kailath: “Fast, recursive-least-squares transversal filters for adaptive filtering”, *IEEE Trans. on Acoust., Speech, Signal Processing*, **ASSP-32**, 2, pp. 304–337 (1984).
- [15] D. T. Slock and T. Kailath: “Numerically stable fast transversal filters for recursive least squares adaptive filtering”, *IEEE Trans. on Acoust., Speech, Signal Processing*, **ASSP-39**, 1, pp. 92–114 (1991).
- [16] S. L. Gay: “Dynamically regularized fast RLS with application to echo cancellation”, *Proc. ICASSP-96*, pp. 957–960 (1996).
- [17] G. A. Clark, S. K. Mitra and S. R. Parker: “Block implementation of adaptive digital filters”, *IEEE Trans. on Acoust., Speech, Signal Processing*, **ASSP-29**, 3 (1981).

- [18] 古川, 久保田, 辻井: “ブロック適応信号処理における直交射影アルゴリズムとその諸性質”, 信学論 (A), **J71-A**, 5, pp. 2138–2146 (1988).
- [19] J. Benesty and P. Duhamel: “A fast exact least mean square adaptive algorithm”, IEEE Trans. on Signal Processing, **40**, 12, pp. 2904–2920 (1992).
- [20] A. Gilloire: “Experiments with sub-band echo cancellers for teleconferencing”, Proc. ICASSP-87, pp. 2144–2144 (1987).
- [21] W. Kellerman: “Analysis and design of multirate systems for cancellation of acoustical echoes”, Proc. ICASSP-88, pp. 2570–2573 (1988).
- [22] B. Hatty: “Recursive least squares algorithms using multirate systems for cancellation of acoustic echoes”, Proc. ICASSP-90, pp. 1145–1148 (1990).
- [23] S. Makino, Y. Kaneda and N. Koizumi: “Exponentially weighted step-size NLMS adaptive filter based on the statistics of a room impulse response”, IEEE Trans. on Speech and Audio, **1**, 1, pp. 101–108 (1993).
- [24] S. Makino and Y. Kaneda: “Exponentially weighted step-size projection algorithm for acoustic echo cancellers”, IEICE Trans.(EA), **E75-A**, 11, pp. 1500–1508 (1992).
- [25] Y. Haneda, S. Makino and Y. Kaneda: “Common acoustical pole and zero modeling of room transfer functions”, IEEE Trans. on Speech and Audio, **2**, 2, pp. 320–328 (1994).
- [26] 高松, 谷萩: “学習同定法と直交射影算法の収束速度について”, 信学論 (A), **J73-A**, 3, pp. 652–655 (1990).
- [27] 金田: “白色雑音入力時における RLS アルゴリズムと学習同定法の等価性について”, 信学全大, pp. A–88 (1992).

- [28] B. Widrow and S. D. Stearns: “Adaptive signal processing”, Prentice-Hall (1985).
- [29] J. Nagumo and A. Noda: “A learning method for system identification”, IEEE Trans. Automatic Control, **AC-12**, 3, pp. 282–287 (1967).
- [30] 古川, 久保田, 辻井: “高速なブロック適応アルゴリズムとその性能評価”, 信学論 (A), **J72-A**, 7, pp. 1069–1076 (1989).
- [31] 福本, 久保田, 辻井: “雑音の影響を考慮した共役こう配法によるブロック適応アルゴリズムの一提案とその性能評価”, 信学論 (A), **J77-A**, 1, pp. 16–23 (1994).
- [32] 丸山: “射影アルゴリズムの高速算法”, 信学全大, pp. B-744 (1990).
- [33] K. Berberidis and S. Theodoridis: “An efficient block newton-type algorithm”, Proc. ICASSP-95, pp. 1133–1136 (1995).
- [34] K. Berberidis and S. Theodoridis: “A new fast block adaptive algorithm”, IEEE Trans. on Signal Processing, **47**, 1, pp. 75–87 (1999).
- [35] D. T. M. Slock and K. Maouche: “The fast subsampled-updating recursive least-square (FSU-RLS) algorithm for adaptive filtering based on displacement structure and FFT”, Signal Processing, **40**, 1, pp. 5–20 (1994).
- [36] R. E. Blahut: “Fast algorithms for digital signal processing”, Addison-Wesley (1987).
- [37] Z. J. Mou and P. Duhamel: “Fast FIR filtering: algorithms and implementation”, Signal Processing, **13**, 4, pp. 377–384 (1987).
- [38] S. L. Gay: “A fast converging, low complexity adaptive filtering algorithm”, proceedings of 3rd International Workshop on Acoustic Echo Control, pp. 223–226 (1993).
- [39] S. L. Gay and S. Tavathia: “The fast affine projection algorithm”, Proc. ICASSP-95, pp. 3023–3026 (1995).

- [40] S. L. Gay: “Fast projection algorithms with application to voice echo cancellation”, PhD thesis, New Brunswick, New Jersey (1994).
- [41] B. Baykal, O. Tanrikulu and A. G. Constantinides: “Asymptotic analysis of the underdetermined recursive least-squares algorithm”, Proc. EUSIPCO-96 (1996).
- [42] H. Besbes, Y. B. Jemaa and M. Jaidane: “Exact convergence analysis of affine projection algorithm: the finite alphabet input case”, Proc. ICASSP-99, pp. 1441–1444 (1999).
- [43] 福本, 久保田, 辻井: “安定性を保証した確率的最速な学習同定法”, 信学論 (A), **J78-A**, 7, pp. 778–787 (1995). (in Japanese).
- [44] S. Haykin: “Adaptive filter theory”, Prentice-Hall, third edition (1996). New Jersey.
- [45] T. K. J. A. Bucklew and W. A. Sethares: “Weak convergence and local stability properties of fixed stepsize algorithms”, IEEE Trans. on Information Theory, **IT-39**, pp. 966–978 (1993).
- [46] S. Haykin: “Adaptive filter theory”, Prentice-Hall, second edition (1991). New Jersey.
- [47] S. Elliott, I. M. Stothers and P. A. Nelson: “Multiple error LMS algorithm and its application to the active noise control of sound and vibration”, IEEE Trans. on Acoust., Speech, Signal Processing, **ASSP-35**, 10, pp. 1423–1434 (1987).
- [48] S. Haykin: “Adaptive Filter Theory”, chapter 13, Prentice-Hall, second edition (1991).
- [49] Y. Kaneda, M. Tanaka and J. Kojima: “An adaptive algorithm with fast convergence for multi-input sound control”, Proc. ACTIVE 95, pp. 100–103 (1995).

研究業績目録

査読のある学術誌に掲載のフルペーパー

1. 田中 雅史, 金田 豊, 小島 順治: 「音源方向推定法の室内残響下での性能評価」, 日本音響学会誌, vol. 50, no.7, pp.540–548, (1994)
2. 田中 雅史, 牧野 昭二, 金田 豊: “A fast projection algorithm for adaptive filtering”, 電子情報通信学会論文誌, vol. E78-A, no.10, pp.1355–1361 (1995)
3. 田中 雅史, 金田 豊, 小島 順治: “A fast converging adaptive algorithm for filtered-x systems”, 日本音響学会英文誌, vol.(E)17, no.6, pp.311–321 (1996)
4. 田中 雅史, 牧野 昭二, 小島 順治: “A block exact fast affine projection algorithm”, IEEE Trans. on Acoust., Speech, Signal Processing, vol.7, no.1, pp.79–86 (1999)
5. 中島 弘史, 田中 雅史, 東山 三樹夫: 「逐次ベクトル射影法による信号表現とその逆フィルタ処理への応用」, 電子情報通信学会論文誌 (A), vol. J83-A, no.4, pp.353–360, (2000)
6. 阪内 澄宇, 羽田 陽一, 金田 豊, 田中 雅史: “A subjective assesment of the desired echo return loss for subband acoustic echo cancellers”, 電子情報通信学会論文誌 (EA), vol. E83-A, no.12, pp.2633–2639 (2000)
7. 阪内 澄宇, 羽田 陽一, 田中 雅史, 佐々木 潤子, 片岡 章俊: 「雑音抑圧及びエコー抑圧機能を備えた音響エコーキャンセラ」, 電子情報通信学会論文誌 (A), vol. J87-A, no. 4, pp. 448–457 (2004)
8. 田中 雅史, 金田 豊: 「複合 Normalized LMS(NLMS) 法モデルを用いたアフィン射影法の収束特性の検討」, 日本音響学会誌, vol.62, no.6, pp.443–453 (2006)

査読のある学術雑誌に掲載のショートペーパー，テクニカルノート

1. 田中 雅史，金田 豊：“Performance of sound source direction estimation methods under reverberant conditions”，音響学会英文論文誌， vol.14, no.4, pp.291–292, (1991)

査読のある国際会議に掲載のフルペーパー

1. 田中 雅史，牧野 昭二，金田 豊：“Fast projection algorithm and its step size control”，Proc. International Conference on Acoustics, Speech, and Signal Processing 95 (ICASSP95), pp.945–948 (1995)
2. 金田 豊，田中 雅史，小島 順治：“An adaptive algorithm with fast convergence for multi-input sound control”，Proc. 1995 International Symposium on Active Control of Sound and Vibration (ACTIVE95), pp.993–1004 (1995)
3. F.Giron，南 泰浩，田中 雅史，古家 賢一：“Compensation of speaker directivity in speech recognition using HMM composition”，Proc. International Conference on Acoustics, Speech, and Signal Processing 98(ICASSP98), pp.253–256 (1998)
4. 田中 雅史，金田 豊：“An interpretation of the affine projection algorithm (APA) as the normalized least mean squares (NLMS) algorithm complex”，Proc. International Workshop on Acoustic Echo and Noise Control-99 (IWAENC-99), pp.136–139 (1999)

解説論文

1. 田中 雅史，金田 豊，小島 順治：“マイクロホンアレーを用いた音源方向推定”，NTT R&D, vol.44, no.1, pp.59–64 (1995)

2. 田中 雅史 , 山森 和彦 : 「高速アフィンブロック射影法」, NTT R&D, vol.50, no.4, pp.231-239 (2001)
3. 阪内 澄宇 , 田中 雅史 , 羽田 陽一 , 山森 和彦 : 「短時間スペクトラル振幅推定を用いた周囲雑音と残留エコーの抑圧」, NTT R&D, vol.50, no.4, pp.246-252 (2001)

研究会資料

1. 田中 雅史 , 金田 豊 : 「室内での音源方向推定に及ぼす虚像音源の影響」, 信学技報, vol.94, no.20(EA94 1-5), pp.1-8 (1994)
2. 田中 雅史 : 「マイクロホンアレー処理技術」, 信学技報, vol.95, no.318(SP95 62-65), pp.1-8 (1995)
3. 中島 弘史 , 東山 三樹夫 , 田中 雅史 : 信学技報, vol.95, no.587(EA95 95-100), pp.39-44 (1996)

学会講演

1. 田中 雅史, 金田 豊 : 「近距離音源の方向探索における平面波近似の有効性について」, 平成 3 年日本音響学会春季大会講演論文集, 2-7-13, pp.445-446 (1991)
2. 田中 雅史, 金田 豊 : 「残響下における音源方向推定法について」, 平成 4 年日本音響学会春季大会講演論文集, 1-5-8, pp.413-414 (1992)
1. 田中 雅史, 牧野 昭二 : 「音声入力に対する射影法の次数と収束特性について」, 平成 4 年度日本音響学会秋季大会講演論文集, 1-4-14, pp.489-490 (1992)
3. 田中 雅史, 金田 豊, 牧野 昭二 : 「高次の射影アルゴリズムの演算量削減について」, 平成 5 年度電子情報通信学会秋季大会講演論文集, A-101, p.1-103 (1993)
4. 田中 雅史, 金田 豊 : 「室内での音源方向推定におよぼす虚像音源の影響について」, 平成 5 年日本音響学会秋季大会講演論文集, 1-5-7, pp.461-462 (1993)
5. 田中 雅史, 金田 豊 : 「適応フィルタを用いた移動物体の検出について」, 平成 6 年日本音響学会秋季大会講演論文集, 2-6-4, pp.629-630 (1994)
6. 羽田 陽一, 牧野 昭二, 田中 雅史, 島内 末廣, 小島 順治 : 「音響エコーキャンセラ用デュオフィルタコントロールシステム」, 平成 7 年電子情報通信学会春季大会講演論文集, A-350, p.350 (1995)
7. 田中 雅史, 金田 豊, 小島 順治 : 「Filtered-X 構成に対する高速射影アルゴリズムの適用について」, 平成 7 年日本音響学会秋季大会講演論文集, 2-6-6, pp.545-546 (1995)
8. 田中 雅史, 牧野 昭二, 金田 豊 : 「高速 FIR フィルタリング法を用いた射影法」, 平成 7 年電子情報通信学会秋季大会講演論文集, A-79, p. 81 (1995)
9. 牧野 昭二, 羽田 陽一, 田中 雅史, 金田 豊, 小島 順治 : 「E S 射影アルゴリズムの音響エコーキャンセラへの適用」, 平成 7 年電子情報通信学会秋季大会講演論文集, A-349, p.349 (1995)

10. 島内 末廣, 田中 雅史, 牧野 昭二:「高速射影アルゴリズムの多チャンネル系への適用」, 平成 8 年電子情報通信学会春季大会講演論文集, A-168, p. 170 (1996)
11. 中島 弘史, 東山 三樹夫, 田中 雅史:「最小 2 乗近似フィルタの逐次的解法」, 平成 8 年音響学会春季大会講演論文集, 1-6-16, p.519-520 (1996)
12. 清原 健司, 田中 雅史:「雑音抑圧マイクロホンアレーのマイクロホン配置の検討」, 平成 8 年音響学会春季大会講演論文集, 2-6-18, p.577-578 (1996)
13. 田中 雅史, 牧野 昭二, 金田 豊:「ブロック長を独立にしたブロック高速射影法」, 平成 9 年電子情報通信学会秋季大会講演論文集, TA-2-2, pp.554-555 (1997)
14. 田中 雅史, 金田 豊:「射影アルゴリズムの収束特性について」, 平成 10 年日本音響学会秋季大会講演論文集, 1-P-2, pp.599-600 (1998)
15. 佐々木 潤子, 田中 雅史:「雑音低減方式における主観的な最適原音付加率と客観量の対応について」, 平成 11 年日本音響学会春季大会講演論文集, 1-6-2, p.433-434 (1999)
16. 阪内 澄宇, 田中 雅史, 羽田 陽一, 山森 和彦:「多チャンネル拡声系へのエコーリダクションの適用」, 平成 13 年日本音響学会秋季大会講演論文集, 1-6-4, p.539-540 (2001)

特許

1. 田中 雅史，羽田 陽一，牧野 昭二：「適応の未知系出力推定方法」，特許第 3147207 号，平成 13 年 (2001)3 月 19 日 登録
2. 牧野 昭二，田中 雅史，金田 豊，羽田 陽一：「反響消去装置」，特許第 3180543 号，平成 13 年 (2001)6 月 25 日 登録
3. 羽田 陽一，牧野 昭二，田中 雅史，清水 潤子，小島 順治：「背景雑音平均レベル予測方法及び装置」，特許第 3244252 号，平成 14 年 (2002)1 月 7 日 登録
4. 羽田 陽一，金田 豊，牧野 昭二，田中 雅史：「反響消去装置」，特許第 3248550 号，平成 14 年 (2002)1 月 21 日 登録
5. 羽田 陽一，金田 豊，牧野 昭二，田中 雅史：「反響消去装置」，特許第 3248551 号，平成 14 年 (2002)1 月 21 日 登録
6. 金田 豊，牧野 昭二，羽田 陽一，田中 雅史，妹尾 達也：「短時間パワー計算装置」，特許第 3264347 号，平成 14 年 (2002)3 月 11 日 登録
7. 羽田 陽一，牧野 昭二，金田 豊，田中 雅史：「反響消去装置及び方法」，特許第 3268572 号，平成 14 年 (2002)3 月 25 日 登録
8. 牧野 昭二，金田 豊，田中 雅史，羽田 陽一：「反響消去装置」，特許第 3293706 号，平成 14 年 (2002)6 月 17 日 登録
9. 田中 雅史，金田 豊，羽田 陽一，小島 順治，牧野 昭二：「適応的伝達関数推定方法及びそれを使った推定装置」，特許第 3303898 号，平成 14 年 (2002)7 月 22 日 登録
10. 島内 末廣，牧野 昭二，田中 雅史，小島 順治：「多チャンネル音声通信会議用反響消去方法」，特許第 3339612 号，平成 14 年 (2002)10 月 28 日 登録

11. 金田 豊, 田中 雅史 :「適応的制御方法」, 特許第 3435675 号, 平成 15 年 (2003)8 月 11 日 登録
12. 田中 雅史, 牧野 昭二, 小島 順治 :「適応的制御方法」, 特許第 3452339 号, 平成 15 年 (2003)9 月 29 日 登録
13. 田中 雅史, 牧野 昭二, 小島 順治 :「適応的制御方法」, 特許第 3475318 号, 平成 15 年 (2003)12 月 8 日 登録
14. 阪内 澄宇, 田中 雅史, 羽田 陽一, 山森 和彦 :「反響抑圧方法、反響抑圧装置及び反響抑圧プログラム記憶媒体」, 特許第 3507020 号, 平成 16 年 (2004)3 月 15 日 登録
15. 阪内 澄宇, 田中 雅史, 金田 豊 :「反響抑圧における所望エコー抑圧量導出方法及びその装置」, 特許第 3579622 号, 平成 16 年 (2004)10 月 20 日 登録
16. 田中 雅史, 中川 朗, 阪内 澄宇, 羽田 陽一 :「多チャンネルエコー消去方法、その装置及びプログラム記録媒体」, 特許第 3583998 号, 平成 16 年 (2004)11 月 4 日 登録
17. 阪内 澄宇, 田中 雅史, 羽田 陽一, 山森 和彦 :「多チャンネル反響抑圧方法、その装置、そのプログラム及びその記録媒体」, 特許第 3607625 号, 平成 17 年 (2005)1 月 5 日 登録

本研究の関係発表論文

本論文の内容は，以下の論文において公表済みである．

1. 田中 雅史，牧野 昭二，金田 豊：“A fast projection algorithm for adaptive filtering”，電子情報通信学会論文誌， vol. E78-A, no.10, pp.1355–1361 (1995)
2. 田中 雅史，金田 豊，小島 順治：“A fast converging adaptive algorithm for filtered-x systems”，日本音響学会英文誌， vol.(E)17, no.6, pp.311–321 (1996)
3. 田中 雅史，牧野 昭二，小島 順治：“A block exact fast affine projection algorithm”，IEEE Trans. on Acoust., Speech, Signal Processing, vol.7, no.1, pp.79–86 (1999)
4. 田中 雅史，金田 豊：「複合 Normalized LMS(NLMS) 法モデルを用いたアフィン射影法の収束特性の検討」，日本音響学会誌， vol.62, no.6, pp.443–453 (2006)