

Studies on block orthogonal projection algorithm for adaptive signal processing and improvements of its performance

吉本, 定伸

東京工業高等専門学校情報工学科 : 助教授 : 情報通信工学, 適応信号処理

<https://doi.org/10.11501/3148621>

出版情報 : 九州芸術工科大学, 1998, 博士 (工学), 課程博士
バージョン :
権利関係 :

氏名・本籍(国籍) 吉本定伸 (福岡県)
学位の種類 博士(工学)
学位記番号 甲第 33 号
学位授与の日付 平成 11 年 3 月 18 日
学位授与の要件 学位規則第 4 条第 1 項該当
学位論文題目 適応信号処理におけるブロック直交射影アルゴリズムとその性能改善に関する研究
審査委員会 幹事 教授 浦濱喜一
委員 教授 瀧山龍三
委員 教授 福島重廣

論文内容の要旨

近年における半導体技術はめざましい発展を続けており、今後も益々その技術革新は続くものと予想される。そのような中、半導体の集積度の向上により、高速で高精度の演算が可能となり、デジタル信号処理技術は様々な分野で不可欠なものとなっている。今日、デジタル信号処理の対象となる環境や情報は多種多様であり、信号の統計的性質が変化するような場合、あるいは未知システムの特徴が変化するような場合など、固定係数フィルタによる信号処理では期待した結果が得られないということが多い。このような状況下において有効な信号処理が適応信号処理である。適応信号処理では、システムの特徴を必要に応じて変化させるために適応フィルタが用いられ、信号の先験的な情報が不十分であっても、入力される情報から未知システムの特徴を逐次的に求めることにより、環境に応じたシステムの推定が可能である。この適応フィルタの係数を更新する重要な部分は適応アルゴリズムと呼ばれ、従来より様々な方式が提案されてきた。適応アルゴリズムには主に、対象となる信号に関係なく高速に収束することと、1 サンプル当たりの演算量が少ないという性能が要求される。これらは一般的にトレードオフの関係にあり、現在これらの性能をバランスよく満足させる方式としてブロック直交射影アルゴリズムが提案されているが、これを更に効率よく運用する方式が期待される。また、雑音がシステムのパラメータ推定に影響を及ぼす場合においても良好な推定精度が得られることが望まれる。

本論文は、適応信号処理において適応フィルタの係数を更新するための適応アルゴリズムについて研究した結果をまとめたものであり、以下の 7 章から構成されている。

第 1 章では、まず本研究を行うに至るまでの背景について述べ、次いで本論文の概要について述べる。

第 2 章では、適応信号処理における未知システムのパラメータ推定の重要性について述べ、本研究の基礎となるブロック直交射影アルゴリズムについての説明を行う。また、適応信号処理の適用例として、適応ノイズキャンセラなどに代表されるキャンセラシステムをとりあげ、適応アルゴリズムの必要性について述べる。

第 3 章では、ブロック直交射影アルゴリズムの一実現法として、次数更新型 UD 分解を用いた方式を提案する。ブロック直交射影アルゴリズムは Moore-Penrose 型一般逆行列により表されるが、この逆行列を特異値分解により計算する方法では、多くの演算量が必要となりハードウェア構成上困難になると考えられる。提案する方式は Moore-Penrose 型一般逆行列が入力状態行列による自己相関行列の逆行列により生成されることに着目し、これを逐次 UD 分解法により効率よく計算することにより、ブロック直交射影アルゴリズムの演算量の軽減化を図っている。

第 4 章では、未知系出力信号に観測雑音が重畳する場合においても良好な推定精度が得られる方式を提案する。ブロック直交射影アルゴリズムは比較的収束速度と演算量のバランスがとれた方式として提案されているが未知系出力信号に観測雑音が重畳する場合、推定精度が劣化するということが指摘されている。推定精度の劣化を防ぐために、ステップゲイン、係数ベクトルの平均化などの操作により推定精度の劣化を防ぐ処置がとられるが、適用する方法によっては収束速度の劣化を招いたり、計算に雑音の分散が必要であり、また白色信号入力時の解析に基づいているなどの問題点があげられる。提案法は Moore-Penrose 型一般逆行列で表される直交射影アルゴリズムの基本型を対象として、修正方向ベクトルに含まれる観測信号から構成される成分の時間平均を行うことにより、良好な推定精度を実現した方式である。提案法はブロック直交射影アルゴリズムと同等の高速な収束速度を得られる方式であり、雑音の分散や入力信号の有色性を考慮する必要がないという利点を有している。

第 5 章および第 6 章では、ブロック直交射影アルゴリズムの一実現法として提案されている勾配法に基づく高速な適応アルゴリズムの改良法を提案する。この方式はよく知られている共役勾配法を用いる方式に対しアルゴリズムの繰り返し演算処理を途中で打ち切った場合においても良好な特性が得られる方式である。従ってこの方式を個々の目的に合わせ改良することは有用であると考えられる。

まず 5 章では、収束速度の高速性という観点から、勾配法に基づく高速な適応アルゴリズムのデータ運用法をブロック処理ではなくアフィン射影算法と同様の逐次処理とし、この処理により増加した 1 サンプル当たりの演算量を、勾配ベクトルの算出法を改良することにより軽減する方式を提案する。

適応アルゴリズムには上記のような性能以外に、有色信号入力時において高速な収束特性を有することと共に、観測雑音に対してもロバストであることが要求される。そこで、第 6 章では、勾配法に基づく高速な適応アルゴリズムを改良し、観測雑音が存在する場合でも良好な精度が得られる方式を提案する。本提案法は勾配法に基づく高速な適応アルゴリズムの修正量に含まれる観測雑音の影響を軽減することにより、観測雑音が存在する場合において従来法よりも良好な推定精度を得ることができる方式である。

最後に第 7 章では、本研究のまとめを行い、今後の話題を述べる。

論文審査の結果の要旨

最近の通信システムを取り巻く環境は多種多様であり、従来のような固定係数フィルタによる信号処理では満足な処理性能が得られない場合が多くなってきている。このためフィルタ特性を環境に応じて変化させる適応信号処理技術が注目され、実用化が進められている。デジタル信号処理の基盤となる LSI 技術の発展はめざましいが、適応信号処理では大量の計算を高速に行うことが要求されるので、適応フィルタの係数を更新する適応アルゴリズムの一層の性能向上が望まれている。適応アルゴリズムに要求される性能としては、1)対象となる信号の性質に依存することなく高速なパラメータ推定が可能である、すなわち収束速度の向上、2)パラメータ更新に要する演算量が少ない、すなわち信号のサンプル当たり計算量の軽減化、3)観測に雑音が混入する場合においても良好なパラメータ推定が可能である、すなわち推定精度の向上、などがあげられる。これらのうちの最初の二つの性能の改善を目指して、ブロック直交射影アルゴリズムが開発され、成果をあげている。これは複数データをブロック単位として取り扱い、それらを用いてフィルタの係数修正を行う方式である。

本論文は、このブロック直交射影アルゴリズムについて、上記 1 と 2 の性能の更なる改善と、特に 3 番目の要求に答えるアルゴリズムの開発とを行った研究成果をまとめたものである。

第 1 章と第 2 章では適応信号処理を概括し、本研究の基礎となるパラメータ推定と直交射影アルゴリズムについてまとめている。

第 3 章ではブロック直交射影アルゴリズムを少ない演算量で効率よく実行する新しい演算方式を提案している。ブロック直交射影アルゴリズムで使われる一般化逆行列に入力信号の自己相関行列の逆行列が含まれることに着目し、この逆行列を次数更新型 UD 分解法を用いて計算することにより、演算量を軽減化している。

第 4 章では、出力信号に雑音が重畳する場合の推定精度劣化を防ぐアルゴリズムを提案している。本方法は、修正方向ベクトルの時間的な平滑化に基礎を置いており、ブロック直交射影アルゴリズムの高速な収束特性を維持しつつ良好な精度が得られる方式でもある。また、雑音の分散や入力信号の白色性・有色性を考慮する必要がないという汎用性も有しており、実験によりその有効性が示されている。

第 5 章と第 6 章ではブロック直交射影アルゴリズムの一実現法である勾配法に基づく適応アルゴリズムについて、収束速度を向上させ、演算量を軽減させた方式を提案し、更に修正方向ベクトルの平滑化により推定精度を向上させた方式へ発展させ、実際に本方法が従来法よりも良好な精度を示すことを計算機シミュレーションにより実証している。

最後に第 7 章では、本研究で得られた成果を総括し、今後の課題について検討を加えている。

以上のように、本論文はブロック直交射影に基づく適応信号処理アルゴリズムについて、収束速度を向上させ、信号のサンプル当たりの計算量を軽減化した新しい演算法を提案し、

観測に雑音が混入する場合においても良好なパラメータ推定が可能な適応アルゴリズムを提案したものであり、計算機による多くの実験によって提案方式の有効性が実証されており、今後デジタル通信分野の研究開発に寄与しうると期待され、博士（工学）の学位論文に値するものと審査委員全員一致で認めるものである。

最終試験の結果の要旨

本論文の内容について著者の説明を受けた後、審査委員全員で口頭により試問を行った。入力信号と観測信号の周波数特性の影響、他の適応信号処理アルゴリズムとの収束速度の比較、入力自己相関行列の正則化係数の設定法、修正方向ベクトルの平滑化の範囲などについて質問があり、何れに対しても著者から明快な回答が得られた。

次に公開発表会が開催され、本学内研究室、他大学及び地元企業等から多数の出席者があった。著者の発表の後の質疑討論では、信号及び雑音の周波数特性、入力相関行列の正則化、修正方向ベクトルの平滑化の効果、推定精度の実用化レベル、アルゴリズムの各パラメータの設定法などについて活発な質問がなされ、著者の説明によって何れに対しても十分な理解が得られた。以上の結果から、著者は最終試験に合格したものと審査委員全員一致で認定した。

Studies on Block Orthogonal Projection Algorithm for Adaptive Signal Processing and Improvements of Its Performance

Sadanobu Yoshimoto

Conventional signal processing by using fixed coefficient filters works unsatisfactorily for the cases where statistical properties of input signals or the characteristics of unknown systems vary with time. Adaptive signal processing techniques are resorted to in such time varying environments. Adaptive signal processing is performed by using adaptive filters whose parameters are adjusted by adaptive algorithms. Adaptive algorithms are required to be converge fast and its computational cost is desired to be low. Block orthogonal projection algorithms have been proposed as a method satisfying these two conflicting demands. Improvement of this effective algorithm is currently advanced in many research groups.

This thesis presents some results of my researches on efficient adaptive algorithms based on the block orthogonal projection for adaptive signal processing systems. The thesis is composed of seven chapters. In chapter one, background of the present research is reviewed. Chapter two reviews the block orthogonal projection algorithms and surveys some application areas of adaptive algorithms. In chapter three, a novel algorithm based on the UD decomposition is proposed. In chapter four, a method is developed which gives stable solutions for the cases where observation noises are added to output signals of unknown systems. Chapters five and six are addressed to a fast adaptive algorithm based on the gradient method which has been used for block orthogonal projection algorithms. At first in chapter five, a method is introduced where data are processed sequentially similarly to affine projection methods and secondly in chapter six a gradient algorithm is modified to give high precision solutions even under moderate observation noises. This proposed method reveals estimation resolution higher than that of previous algorithms based on the gradient methods. Finally in chapter seven, the algorithms developed in this research are summarized and some future directions are illustrated for further studies.

目次

1	序論	4
1.1	背景	4
1.2	概要	8
2	パラメータ推定問題と直交射影アルゴリズム	10
2.1	まえがき	10
2.2	パラメータ推定問題	10
2.3	アフィン射影アルゴリズム	19
2.4	ブロック直交射影アルゴリズム	24
2.4.1	共役こう配法を適用した方式	28
2.4.2	直交射影行列を用いた方式	31
2.5	適応信号処理の適用例	32
2.5.1	適応ノイズキャンセラ	32
2.5.2	自動等化器	34
2.5.3	適応エコーキャンセラ	35
2.5.4	アクティブノイズコントロール	37
2.6	むすび	38
3	UD 分解法によるブロック直交射影アルゴリズム	40
3.1	まえがき	41
3.2	準備	43
3.3	逐次 UD 分解法を用いた BOP アルゴリズム	47

3.3.1	アルゴリズムの導出	47
3.3.2	演算量の比較	53
3.4	具体例	53
3.4.1	シミュレーション結果	56
3.5	むすび	57
	付録 式 (3:31) と BOP アルゴリズムの等価性に関する略証	63
4	観測雑音に対しロバスト性を実現する方向ベクトル分割平均化法によるブロッ ク適応アルゴリズム	65
4.1	まえがき	66
4.2	準備	67
4.3	方向ベクトルの分割平均化	69
4.4	具体例	73
4.4.1	シミュレーション結果	74
4.5	むすび	75
	付録 有限区間における平均化	83
5	こう配法に基づく高速な適応アルゴリズムにおけるデータ運用法の改良に関 する一考察	86
5.1	まえがき	86
5.2	高速な適応アルゴリズム	87
5.2.1	準備	87
5.2.2	データの運用法を改良した適応アルゴリズム	91
5.3	コンピュータ・シミュレーション	95
5.4	むすび	99
6	観測雑音にロバストなこう配法に基づく適応アルゴリズム	103
6.1	まえがき	103
6.2	準備	104

6.2.1	提案する方式	106
6.3	コンピュータ・シミュレーション	108
6.4	むすび	109
7	結論	113
	謝辞	116
	参考文献	117
	図目次	125
	表目次	127

第 1 章

序論

日常生活において，例えば電話中雑音がひどく相手の声が聞き取りにくいのもっと鮮明な音声で通話したいという要求，あるいは通信において，信頼性における情報の伝送を行いたいというような要求にかられることをよく経験するが，このような場面においてデジタル信号処理は有用である．現在，デジタル信号処理技術は急速な勢いで発展し，多くの分野で必要不可欠になってきている．本論文ではデジタル信号処理における重要分野の 1 つである適応信号処理の心臓部ともいえる適応アルゴリズムの研究を行っている．本章では，まず本研究を行うに至った背景について述べ，次いで本論文の概要を示す．

1.1 背景

近年における半導体技術の進歩はめまぐるしく，VLSI(Very Large Scale Integration) や ULSI(Ultra LSI) に代表されるような論理ゲート数が非常に多いデジタル IC が開発されており，その集積度はこれからもなお高密度になるであろうことが予想される．デジタル信号処理装置において，ハード量が最も多く，処理速度を決定するのが乗算器であり，この性能がリアルタイム処理への可能性を左右し，コストにも影響を与える [1]．一方，従来において短い時間内に乗算・加算などのデジタル演算を多量に行うためにリアルタイム処理が困難であった複雑な処理が，IC 回路の集積度が高くなることにより，高速で高精度な演算が可能になりつつある．そ

のため様々な分野でデジタル信号処理技術 [2]~[10] の有用性が益々高まってきている。

デジタル信号処理は、現在では工学のみにとどまらず医学、天文学、地震学など多くの分野で応用されている。また、工学の分野におけるデジタル信号処理関連分野についても、通信のみならず画像、情報、制御などの幅広い分野で適用されている [11][12]。さらに、近年におけるマルチメディア技術 [13] の発達もこれを支える技術の 1 つであるデジタル信号処理技術の発達に拍車をかけていると思われる。このようなことから、デジタル信号処理は更に多くの分野で重要な位置を占めることが予想される。

デジタル信号処理の中でも近年注目を集めている分野に適応信号処理 [14] がある。デジタル信号処理の研究初期には、Wiener 規範による平均 2 乗誤差を最小とする最適フィルタ (Wiener filter)[15][16] が確立されが、これは信号の定常性と観測が無限であるというを仮定に基づいており、先験的に統計的な性質が必要となることが知られている。しかしながら、適応信号処理では、処理する信号の性質が十分に得られない場合においても、フィルタの特性を最適フィルタの特性に近づけるということが要求される。これには適応的に最適フィルタの推定を行う、すなわちフィルタ係数の調整が可能な適応フィルタが有効である。適応フィルタはこのような場合に対応できるように、何らかのアルゴリズムを用いて係数を修正する機能、いわば一種の学習機能を持つインテリジェントシステムと位置づけることができよう。上記のアルゴリズムは一般に適応アルゴリズムとよばれるが、適応信号処理において適応フィルタの係数を更新するための適応アルゴリズムを導出することは重要な問題であると言える。

適応信号処理は 1960 年 Widrow と Hoff により研究が始められて以来、デジタル通信の分野で特に重要な位置を占めるようになってきた。Widrow と Hoff は適応スイッチング回路の研究で Widrow-Hoff の LMS アルゴリズム (Least Mean Square Algorithm, 以下 LMS アルゴリズム) とよばれる適応アルゴリズムの提案を行っている [17]。LMS アルゴリズムは最急降下法 (山下り法) に基づき、平均 2 乗誤差を最小にするようフィルタ係数を修正する方式であり、特に演算量が少ないという理由

で現在もなお代表的なアルゴリズムとしての地位を保ち続けている。

一方、1967年に野田と南雲は学習同定法の提案を行った [18][19]。学習同定法は LMS アルゴリズムの係数修正項をフィルタの入力状態ベクトルのノルムで正規化したもので別名 Normalized LMS アルゴリズムともよばれている。このアルゴリズムは正規化を行うため、演算量の点においては LMS アルゴリズムよりやや多くなるものの収束特性や演算量などの観点から優れた適応アルゴリズムといえることができる。

これらのアルゴリズムは信号の統計的性質が未知の場合、あるいは推定すべきパラメータが時間とともに変動するような場合に際してもパラメータの推定が可能であるという特徴を有している。しかしながら、広く知られているようにこれらの適応アルゴリズムは有色信号が入力された場合、収束速度が著しく劣化するという欠点がある。

このように、入力信号が有色である場合においても高速な収束特性を示すアルゴリズムとして、逐次最小 2 乗法 (Recursive Least Square, 以下 RLS)[20][21] が知られている。このアルゴリズムは対象となる信号の定常性とエルゴード性の仮定の下に成り立つ。また、未知システムのパラメータが変化しない場合、RLS アルゴリズムはカルマンフィルタ (Kalman filter)[22]~[24] と等価となる [25]。カルマンフィルタはウィナーフィルタを時間領域についての定式化に基づいて変革したものであり、多くの研究がなされている [26][27]。RLS アルゴリズムは推定すべきパラメータの個数を N とすれば、1 サンプル当たりの演算量は N の 2 乗に比例 ($O(N^2)$) するためハードウェア化は困難であるが、未知パラメータが時間的に不変であるとき非常に良好な収束特性を示す。このように演算量の軽減化と収束速度の高速化は一般にトレードオフの関係にあるといえることができる。ところで、未知パラメータが変動する場合、忘却係数の導入により追従性を確保できることが知られている。一方、単精度などの有限語長演算を行った場合、数値的不安定性が生ずる可能性を有することが指摘されており、実際の適用に関しては留意する必要がある。

ところで、適応アルゴリズムの応用に関する適用例はノイズキャンセラ、自動等化器など従来より数々報告されており [28]~[36]、この応用に際しては、特に有色信号入力時においても良好な収束特性を示し、演算量が少なくハードウェア構成が容

易である。また、パラメータの変動にも十分追従できる適応アルゴリズムが望ましいといえる。

この演算量の軽減と収束速度の高速化という要求を効果的に解決する手法としてブロック適応信号処理があげられる。入力信号をブロック化することに基礎をおいているブロック適応信号処理は、ブロックで最適なフィルタ係数を求める方式でありブロックごとにフィルタ係数を修正する。Clarkら [37][38] は、LMS アルゴリズムにこのブロック処理の概念を導入した Block LMS アルゴリズム [38]~[40] を提案し、LMS アルゴリズムの特性をある程度維持しながら実行が速いことを示している。しかしながら、Block LMS アルゴリズムは LMS アルゴリズム同様、入力される信号によって収束速度が劣化することから、収束速度を改善した OBA アルゴリズム [41] が Micheal らにより提案されている。また、古川らによりブロック学習同定法が提案されており [42]、Block LMS アルゴリズムよりも良好な収束速度を得ることができ

一方、複数の入力信号を処理速度の高速化に利用するのではなく、収束速度の向上に利用したのが雛元、前川の提案した拡張された学習同定法 [43]、尾関、梅田が提案したアフィン射影算法 [44] である。この2つの手法は多次元部分空間への直交射影演算に基づいた算法であり、有色信号入力時においても高速な収束特性を得られるが1サンプル当たりに要する演算量が非常に多くなるという問題点がある。古川らは、演算量の軽減を図るために文献 [43][44] の処理にブロック処理の概念を導入し、その一般的な表現として BOP(Block Orthogonal Projection) アルゴリズムを提案した [45]。BOP アルゴリズムではフィルタ係数の修正を入力データブロックごとに行うため、演算量の軽減化が図られている。BOP アルゴリズムについての研究は既に数々の方式が提案されており [46]~[48]、現在もなお活発に研究が進められている。また、古川らによりオーバーラップ長という新しい概念を導入した多次元空間への直交射影アルゴリズムも提案されている [49][50]。

本論文では、このような背景を鑑みて直交射影アルゴリズムを演算量、収束速度、観測雑音が存在する場合などの観点から考察し、それらに基づきブロック直交射影アルゴリズムの提案、あるいは改良を行い、その基本的性質及び有効性について論

ずる.

1.2 概要

以下第2章では、まず適応信号処理の基礎ともいえるパラメータ推定についての説明を行い、以後の議論を行う準備として直交射影アルゴリズム、ブロック直交射影アルゴリズムの説明を行う。また、適応信号処理の適用例として、適応ノイズキャンセラ、自動等化器、適応エコーキャンセラ、アクティブノイズキャンセラのモデルを用いてその必要性について述べる。

第3章では、BOP アルゴリズムを実現する一手法として、次数更新型 UD 分解を用いた BOP アルゴリズムを提案する。BOP アルゴリズムは Moore-Penrose 型一般逆行列 [51] により表現することができるが、この逆行列を特異値分解により計算する場合 $O(N^3)$ の演算量が必要となり、ハードウェア構成上困難になると考えられる (但し、 N はフィルタのタップ長である)。本章で議論する手法は、Moore-Penrose 型一般逆行列を効率よく計算するため、この Moore-Penrose 型一般逆行列が入力状態行列と入力状態行列の自己相関行列の逆行列により生成されることに着目し、自己相関行列の逆行列を逐次 UD 分解法により計算し、1 回のフィルタ修正に要する演算量を $O(Nr)$ にまで軽減している (但し、 r はブロック長である)。また、このアルゴリズムの性能を評価するために、直交射影行列を用いた BOP アルゴリズム [48] との比較検討を行い、本アルゴリズムの優位性を示す。

第4章では、未知系出力信号に観測雑音が重畳する場合においても良好な推定精度が得られる方式の提案を行う。BOP アルゴリズムは比較的収束速度と演算量のバランスがとれた方式として提案されているが未知系からの出力信号に観測雑音が存在する場合、推定精度が雑音に影響されやすいという欠点がある。推定精度の劣化を防ぐため、ステップゲインを小さくとり、係数ベクトルの平均化を行うなどの操作により推定精度の劣化を防ぐ処置がとられが、適用する方法によっては収束速度の劣化を招く、雑音の分散が必要である、白色信号入力時の解析に基づいているなどの問題点があげられる。提案法は Moore-Penrose 型一般逆行列で表される直交

射影アルゴリズムの一般論としての観点から、修正方向ベクトルに含まれる観測信号から構成される成分の時間平均を行うことにより、良好な推定精度を得られる方式である。本方式は信号と観測雑音が無相関であり、雑音の平均値が零であるという条件のみで適用が可能である。また、シミュレーションにより収束特性の検討を行う。

第5章および第6章ではBOPアルゴリズムの新しい実現法として古川らにより提案されている勾配法に基づく高速な適応アルゴリズム [52][53] の改良法を提案する。この方式はよく知られている共役勾配法を用いたBOPアルゴリズムに対しアルゴリズムの繰り返し演算処理を途中でうち切った場合においても良好な特性が得られる方式である。従ってこの方式をある目的に合わせ改良することは有用であると考えられる。

まず第5章では、収束速度の高速性を失うことのないように勾配法に基づく高速な適応アルゴリズムをブロック処理ではなくアフィン射影算法と同様の逐次処理となるようデータ運用法を考慮し、この処理により増加した1サンプル当たりの演算量を軽減する方式を提案する。

一方、適応アルゴリズムには、有色信号入力時において高速な収束特性を有することと共に、観測雑音に対してもロバストであることが要求される。従って、第6章では、勾配法に基づく高速な適応アルゴリズムを改良し、観測雑音が存在する場合でも良好な精度が得られる方式を提案する。提案法はこう配法に基づく高速な適応アルゴリズムの修正量に含まれる観測雑音の影響を軽減することにより、観測雑音が存在する場合において従来法よりも良好な収束特性を示す。また、計算機によるシミュレーションを行うことにより提案法の有効性を検討する。

最後に第7章では、本研究のまとめを行い、今後の課題について言及する。

第2章

パラメータ推定問題と直交射影アルゴリズム

前章において本論文の背景及び概要を述べた。本章では、以後の章で議論を行う際に必要な準備を行う。

2.1 まえがき

本章では、適応信号処理における基礎的概念であるパラメータ推定について述べる。次いで本論文の基礎となるアフィン射影アルゴリズム、ブロック直交射影アルゴリズムを図を用いて説明する。また、適応信号処理の応用例として適応ノイズキャンセラ、自動等化器、適応エコーキャンセラ、アクティブノイズキャンセラの図を用いてその必要性を述べる。なお、本論文を通じて $(N \times r)$ の行列 A を $A_{r,N}$ 、 $(N \times 1)$ のベクトル b を b_N と表記する。また、 $[\cdot]^T$ は行列あるいはベクトルの転置を表す。

2.2 パラメータ推定問題

入出力データからその未知システムの構造とパラメータを推定することはシステム同定と呼ばれている。本論文では未知システムの構造を FIR(有限インパルス応答: Finite Impulse Response) 型と仮定し、パラメータ (インパルス応答) 推定問題として取り扱う。

ここで $x(k)$, $y(k)$, $d(k)$ をそれぞれ時刻 k における適応フィルタ (Adaptive Filter) の入力信号, 出力信号及び所望信号とする. 入力信号 $\{x(k), -\infty < k < \infty\}$ に対して以下のモデル

$$d(k) = \sum_{i=0}^{N-1} w(i) \cdot x(k-i) \quad (2.1)$$

で定まる信号 $\{d(k), -\infty < k < \infty\}$ を出力する未知システムを考える. 但し N はインパルス応答長である. また, $w(i) (i = 0, \dots, N-1)$ は推定すべきインパルス応答であり

$$W_N \triangleq [w(0), w(1), \dots, w(N-1)]^T \quad (2.2)$$

と定義する. 一方, FIR デジタルフィルタの入出力関係は

$$y(k) = \sum_{i=0}^{N-1} h(i)^{(k)} \cdot x(k-i) \quad (2.3)$$

で与えられる. $h(i)^{(k)}$ は時刻 k における適応フィルタの係数で

$$h_N(k) \triangleq [h(0)^{(k)}, h(1)^{(k)}, \dots, h(N-1)^{(k)}]^T \quad (2.4)$$

と定義する. また出力誤差信号 $e(k)$ は所望信号及び推定システムの出力信号より

$$e(k) = d(k) - y(k) \quad (2.5)$$

と表される. ここで, 入力信号をベクトルで表し

$$x_N(k) \triangleq [x(k), x(k-1), \dots, x(k-N+1)]^T \quad (2.6)$$

と定義すれば適応フィルタの出力信号は

$$y(k) = h_N(k)^T \cdot x_N(k) \quad (2.7)$$

となる.

さて, 式 (2.1) で表されているようにインパルス応答が有限である場合, FIR デジタルフィルタと未知システムの関係が

$$w(i) = h(i)^{(k)} \quad (i = 0, 1, \dots, N-1) \quad (2.8)$$

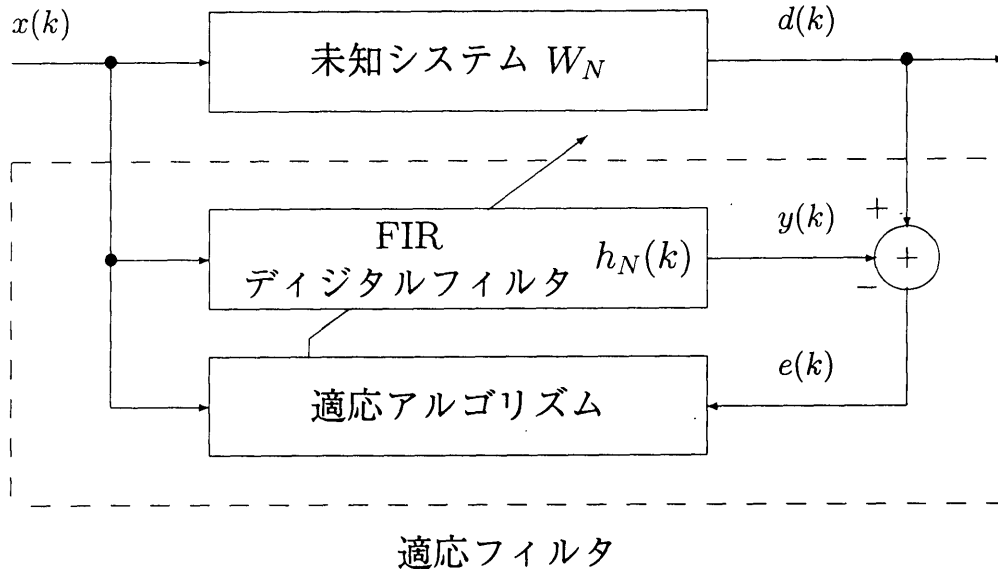


図 2.1: システム同定のモデル

であれば出力は等しくなる。従って、未知システムのインパルス応答が有限で、インパルス応答長がわかっていると仮定すれば、同じ入力に対して完全に等しい出力を得る FIR デジタルフィルタを得られる可能性がある。

一方、未知システムのインパルス応答が無限に続く場合、同じ入力に対して等しい出力を得る FIR デジタルフィルタは得られない。しなしながら、インパルス応答が時間とともに減衰する場合を考慮すると、インパルス応答の最初の N 個がわかればよい適用例も多い。また、 N を可能な限り大きくすれば、未知システムとの入出力関係に非常に近い FIR デジタルフィルタが得られる。このようなことから、前述の FIR 型と仮定した妥当性が理解できよう。このような未知システムに非常に近い FIR デジタルフィルタは、後述する評価量を最小にすることにより得ることができる。

図 2.1 にシステム同定のモデルを示す。一般に、係数修正のためのアルゴリズムを適応アルゴリズム、適応アルゴリズムを含む FIR デジタルフィルタは適応フィルタあるいは推定システムと呼ばれる。

係数修正のために使用する評価量 J はパラメータ推定の目的からいえば，未知システムと推定システムのパラメータの距離を直接評価量とすることが重要であるが未知システムのパラメータが未知であるため，一般には未知システムと推定システムの差の 2 乗平均値

$$\begin{aligned} J &= E[\{d(k) - y(k)\}^2] \\ &= E[e(k)^2] \end{aligned} \quad (2.9)$$

が評価量として使用される．この評価量 J を適応フィルタ h_N に関する 2 次関数で表すと

$$J = h_N^T E[x_N(k)x_N(k)^T] h_N - 2h_N^T E[x_N(k)d(k)] + E[d(k)^2] \quad (2.10)$$

が得られる．従って， J を最小にする最適係数ベクトルを $h_N(opt)$ とし，評価量 J を h_N で偏微分することにより

$$\frac{\partial J}{\partial h_N} = 2E[x_N(k)x_N(k)^T] h_N - 2E[x_N(k)d(k)] \quad (2.11)$$

となり，最適係数ベクトル $h_N(opt)$ は

$$h_N(opt) = E[x_N(k)x_N(k)^T]^{-1} E[x_N(k)d(k)] \quad (2.12)$$

となる．これはウィナーホッフの解とよばれている．但し， $E[x_N(k)x_N(k)^T]$ は正則であると仮定する．しかしながらウィナーホッフの解では平均操作と逆行列演算を必要としリアルタイム処理には不向きである．また，逆行列の補題を用い，この平均操作と逆行列演算を逐次的に効率よく行う RLS アルゴリズムも提案されているが，これらの方法は相関行列の正則性が必要となり場合によっては数値的不安定性を招くおそれが生じる．

一方，Widrow は最急降下法に基づき瞬時自乗誤差 $e(k)^2$ を評価量とした LMS アルゴリズムを提案した．ここで最急降下法について簡単に説明する．式 (2.9) で与えられている J に関する任意の h_N におけるこう配ベクトルを

$$g_N = 2E[x_N(k)x_N(k)^T] h_N - 2E[x_N(k)d(k)] \quad (2.13)$$

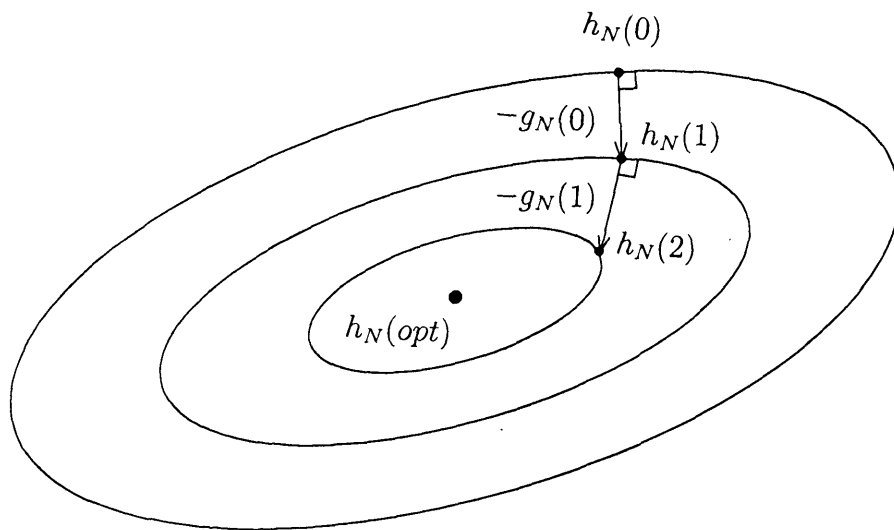


図 2.2: J に関する等高線

とする. 図 2.2は最急降下法による $N = 2$ の場合のパラメータ更新のようすを示している. 図における等高線は係数 $h_N(j)$ ($j = 0, 1, 2, \dots$) の変化に対し等しく, 同一の等高線上において J の値は一定である. 但し, $h_N(j)$ は h_N の j 番目の修正値である. こう配ベクトル $g_N(j)$ は係数 $h_N(j)$ におけるこう配であり, 周知のように等高線上の法線方向を向いている. また, 式 (2.10) はパラメータ h_N の 2 次形式となっていることから, 評価量 J を最小にする h_N は唯一存在する ($h_N(opt)$). 従って, 図からわかるように任意の点 $h_N(0)$ を初期値とすれば, $h_N(0)$ を $-g_N(0)$ 方向に移動することにより $h_N(1)$ における J を $h_N(0)$ における J よりも小さくすることができる. すなわち J が最小になる方向に進むことにより $h_N(j)$ を $h_N(opt)$ に近づけることができる. 以上のことを考慮すると係数修正式は

$$h_N(j) = h_N(j-1) - \mu(j)g_N(j) \quad (2.14)$$

となる. ここで $\mu(j)$ はステップゲインと呼ばれている.

上記のように最急降下法ではパラメータ推定を行うために信号の 2 次統計量 (入

力自己相関行列 $E[x_N(k)x_N(k)^T]$, 相互相関ベクトル $E[x_N(k)d(k)]$ が必要である。しかしながらこれらの統計量を用いる場合, 前述のようにリアルタイム処理には適していない。従って, (2.9) における平均操作を取り除き, 瞬時自乗誤差を評価量 J

$$\begin{aligned} J &= [d(k) - y(k)]^2 \\ &= h_N(k)^T x_N(k) x_N(k)^T h_N(k) - 2h_N(k)^T x_N(k) d(k) + d(k)^2 \end{aligned} \quad (2.15)$$

として用いることにより, リアルタイム処理への対応という問題を解決した LMS アルゴリズムが提案された。

この評価量 J に対する勾配ベクトル $g_N(k)$ は式 (2.15) を $h_N(k)$ に関して偏微分することにより

$$\begin{aligned} g_N(k) &= 2x_N(k)x_N(k)^T h_N(k) - 2x_N(k)d(k) \\ &= -2x_N(k)[d(k) - x_N(k)^T h_N(k)] \\ &= -2x_N(k)e(k) \end{aligned} \quad (2.16)$$

で与えられることから, LMS アルゴリズムの係数更新式は

$$h_N(k+1) = h_N(k) + \mu \cdot x_N(k)e(k) \quad (2.17)$$

と表される。式 (2.17) からわかるように, LMS アルゴリズムは演算量が少なくハードウェアの構成が容易であるために現在でも広く利用されている。

また, これとは別に野田, 南雲により学習同定法が提案されている。学習同定法は NLMS(Normalized LMS) と言われるように LMS アルゴリズムの修正方向ベクトルを入力信号ベクトルのノルムの自乗で正規化することにより得られる。従って学習同定法の係数更新式は

$$h_N(k+1) = h_N(k) + \alpha \frac{x_N(k)}{\|x_N(k)\|^2} e(k) \quad (2.18)$$

と表すことができる。但し α はステップゲインであり, $\|\cdot\|$ はベクトルのユークリッドノルムを表し, 2乗和の平方根と定義する。学習同定法は LMS アルゴリズムに比べやや演算量が増えるものの, 比較的演算量が少なくハードウェア構成が容易であ

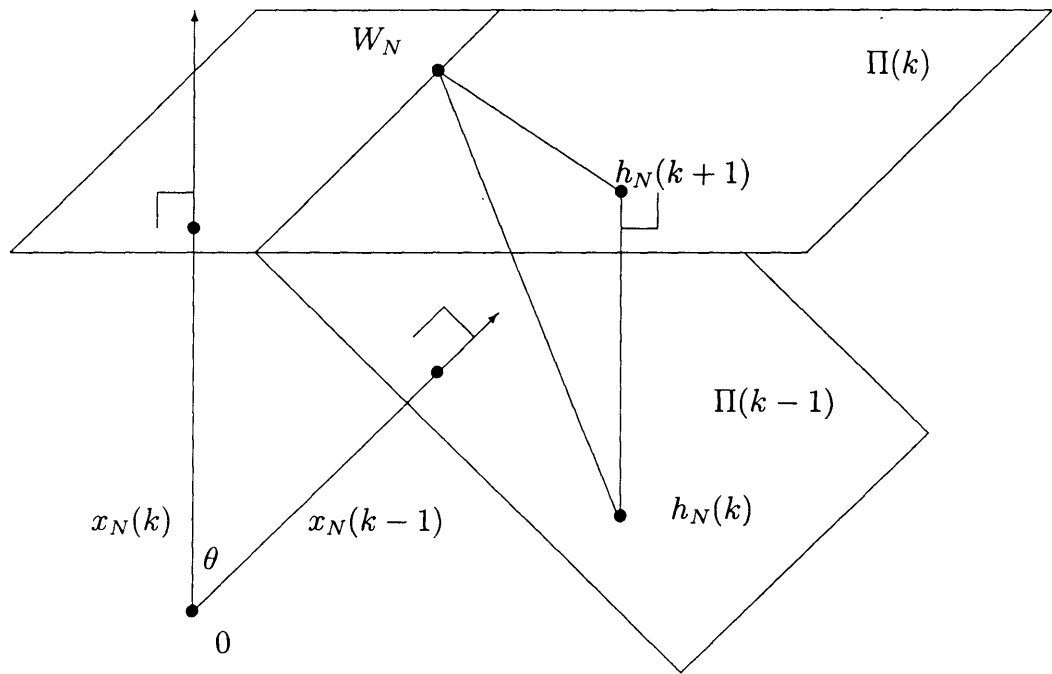


図 2.3: 学習同定法の幾何学的関係

るという理由から、LMS アルゴリズムと同様に広く利用されている適応アルゴリズムの 1 つである。また、学習同定法は LMS アルゴリズムに対し修正量のみが異なることから、等価な方式であるとも考えることができる。しかしながら (2.18) 式中のステップゲインを $\alpha = 1$ と置くと、直交射影原理に基づく適応アルゴリズムという非常に興味深い解釈ができる。すなわち (2.18) 式は

$$h_N(k+1) = h_N(k) + x_N(k)[x_N(k)^T x_N(k)]^{-1} x_N(k)^T [W_N - h_N(k)] \quad (2.19)$$

と表される。これはベクトル $[W_N - h_N(k)]$ を $x_N(k)$ の張る空間へ直交射影することにより、係数ベクトル $h_N(k+1)$ を得ることを示している。

ここで、学習同定法について追記する。図 2.1において時刻 k における適応フィル

タ出力 $y(k)$ が所望信号 $d(k)$ と等しいとすると

$$d(k) = x_N(k)^T h_N \quad (2.20)$$

を得る。明らかに未知システムのパラメータ W_N (真値ベクトル) は式 (2.20) を満足する。しかし、適応フィルタのパラメータ h_N (推定ベクトル) は W_N と等しいとは限らないが、すべての入力信号に対し式 (2.20) が成立するとき $W_N = h_N$ となる。このように、式 (2.20) を満足する h_N は真値ベクトルを含む解集合である。いま、式 (2.20) を満足する h_N の代表ベクトル $h_N(k+1)$ を図 2.3 に示すように $h_N(k)$ から解集合 $\Pi(k)$ に下ろした垂線の足と考える。図 2.3 からわかるように、ある時刻に対応する入力ベクトルと解集合は直交の関係にある。さらに、 W_N はこの解集合に含まれるので $h_N(k+1)$ は $h_N(k)$ から $x_N(k)$ 方向に係数を修正したとき W_N に最も近い点であることがわかる。従って、図 2.3 に示す時刻 $k, k-1$ における解集合 $\Pi(k), \Pi(k-1)$ は所望信号 $d(k), d(k-1)$ と等しくなる適応フィルタ係数の集合である。また、真値ベクトル W_N はすべての状態ベクトル $x_N(k)$ ($-\infty < k < \infty$) に対し所望の信号 $d(k)$ となる点であることからすべての解集合の交点である。以上のことから

$$\begin{aligned} h_N(k+1) &= h_N(k) + [h_N(k+1) - h_N(k)] \\ &= h_N(k) + \frac{[W_N - h_N(k)]^T [h_N(k+1) - h_N(k)]}{\|h_N(k+1) - h_N(k)\|^2} \\ &\quad \cdot \frac{h_N(k+1) - h_N(k)}{\|h_N(k+1) - h_N(k)\|} \end{aligned} \quad (2.21)$$

となる。ここで

$$\frac{h_N(k+1) - h_N(k)}{\|h_N(k+1) - h_N(k)\|} = \frac{x_N(k)}{\|x_N(k)\|} \quad (2.22)$$

$$[W_N - h_N(k)]^T x_N(k) = d(k) - y(k) = e(k) \quad (2.23)$$

が成立する。従って、式 (2.21) は

$$h_N(k+1) = h_N(k) + \frac{x_N(k)}{\|x_N(k)\|^2} e(k) \quad (2.24)$$

のように変形でき、ステップゲインを掛けることにより既に示した式 (2.18) と表される。上式 (2.24) により適応フィルタ係数 $h_N(k+1)$ は初期値に関係なくフィルタ

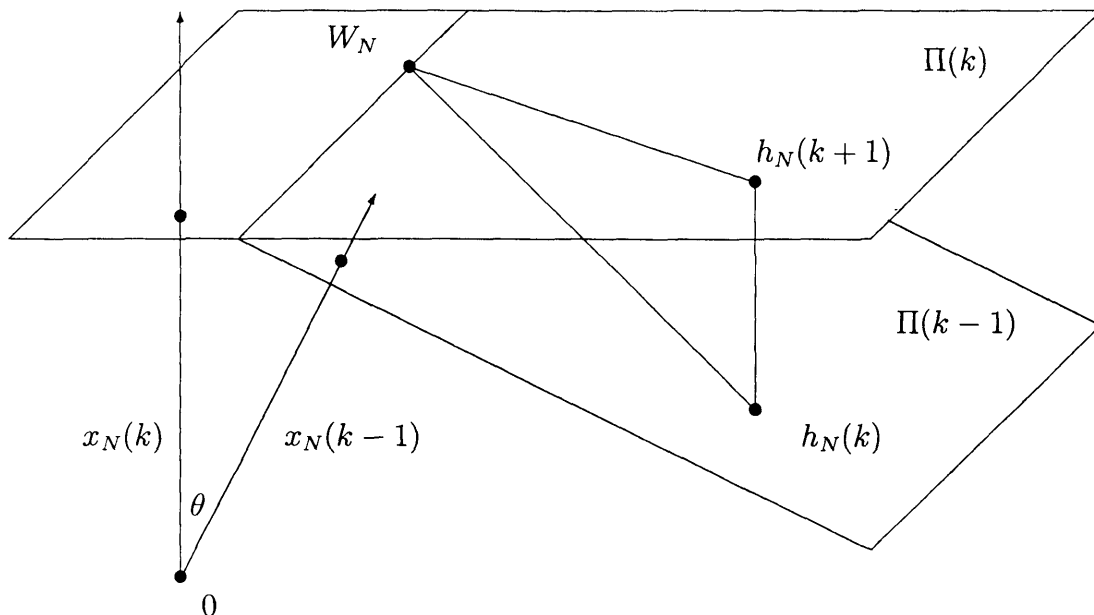


図 2.4: 信号の相関が強い場合

の更新に要した状態ベクトル $x_N(k)$ に対し所望の信号を与えるという特徴を有している。また、時刻 k においては未知システムのパラメータが変動してもこの特徴は保存される。

ところで、学習同定法における収束速度は入力信号の相関により大きく左右されることが指摘されている。図 2.3 における入力信号ベクトル $x_N(k)$ と $x_N(k-1)$ のなす角 θ は

$$\cos \theta = \frac{x_N(k)^T x_N(k-1)}{\|x_N(k)\| \cdot \|x_N(k-1)\|} \quad (2.25)$$

により求められる。信号ベクトル $x_N(k)$ と $x_N(k-1)$ の余弦が 1 に近いほど信号の相関が強く収束速度が劣化し、0 に近いほど収束速度が速くなると考えられる。このようすを図 2.4 及び図 2.5 に示す。図 2.4 は信号の相関が強い場合 ($0 << \cos \theta < 1$) であり、図 2.5 は信号の相関が弱い場合 ($0 < \cos \theta << 1$) である。これらの図を見比べればわかるように、ベクトル $[W_N - h_N(k+1)]$ のノルムは明らかに図 2.5 の信号の相関が弱い方が短く真値ベクトルに近い。このようなことから学習同定法は演算

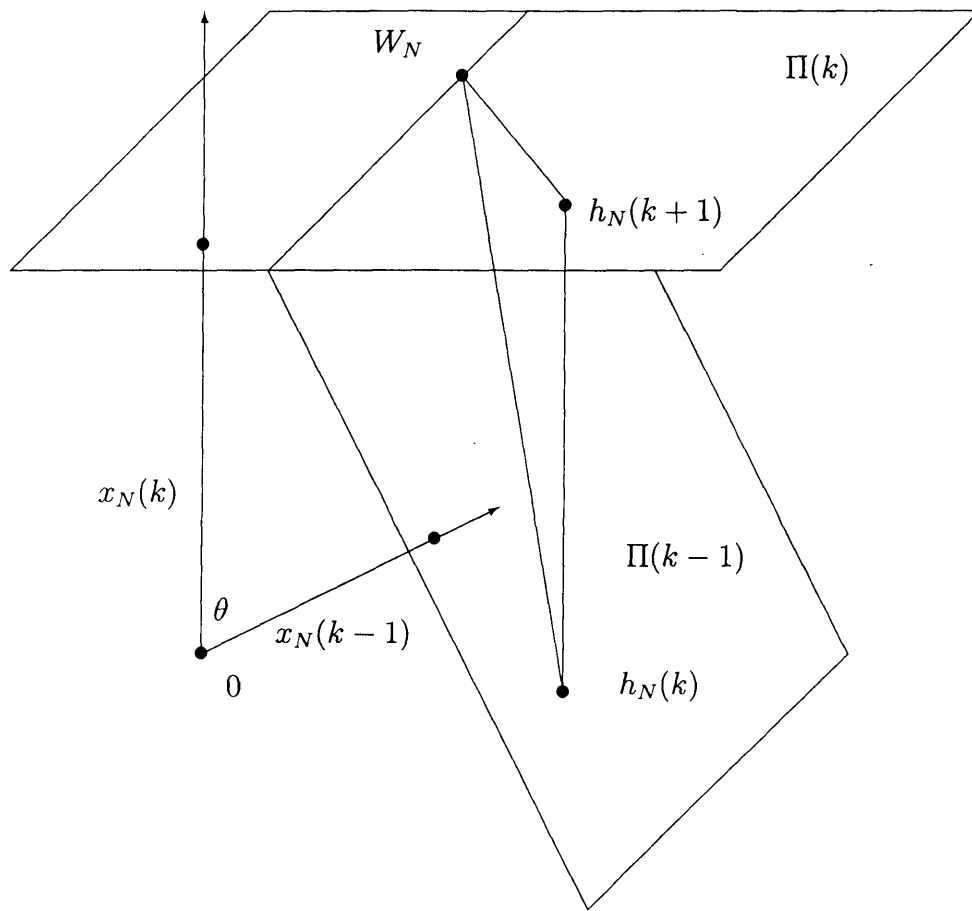


図 2.5: 信号の相関が弱い場合

量は少ないものの、信号の相関により収束速度が大きく左右される適応アルゴリズムであるといえる。

2.3 アフィン射影アルゴリズム

2.2 では、未知システムのパラメータを推定することの重要性を述べた。このパラメータ推定の際、推定システムに入力される信号の相関度が高い場合、入力信号の性質により収束特性は大きく左右される。このように入力信号が有色である場合においても高速な収束特性を得られるアルゴリズムとして、雛元らと尾関らは入力

信号ベクトルの張る部分空間に対する直交射影演算に基づいた適応アルゴリズムを提案している [43][44]. これらの幾何学的図を図 2.6, 図 2.7に示す. これらの図はともに2次のアフィン (2本の入力ベクトルを用いる) の場合を表している. 図 2.6のアフィン射影の場合, 真値ベクトルから入力信号ベクトル $x_N(k)$, $x_N(k-1)$ の張る空間 $S[x_N(k), x_N(k-1)]$ へ垂直におろした点が適応フィルタの係数となる. このとき学習同定法の係数ベクトルは, ベクトル $[W_N - h_N(k)]$ を信号ベクトル $x_N(k)$ に直交射影した点であり, 図より明らかにアフィン射影アルゴリズムの係数ベクトルの方が学習同定法のそれよりも真値ベクトル W_N に近いことがわかる. 他方, 拡張された学習同定法は, まず学習同定法と同じくベクトル $[W_N - h_N(k)]$ を信号ベクトル $x_N(k)$ に直交射影し (仮にこの点を $\hat{h}_N(k+1)$ とする), 次いで信号ベクトル $x_N(k-1)$ を $x_N(k)$ に直交化させ (このベクトルを $\hat{x}_N(k-1)$ と置く), 最後に先ほど直交射影した点 ($\hat{h}_N(k+1)$) からベクトル $\hat{x}_N(k-1)$ 上にベクトル $[W_N - \hat{h}_N(k+1)]$ を直交射影した点を係数ベクトルとする. 図 2.6, 図 2.7よりアフィン射影アルゴリズムと拡張された学習同定法の係数ベクトル $h_N(k+1)$ は等しく, この2方式は部分空間への直交射影に基づく等価なアルゴリズムであることが分かる. また, 本論文ではこれらのアルゴリズムを総称して AP アルゴリズム (Affine Projection Algorithm) と呼ぶこととする.

AP アルゴリズムは, 任意の時刻において適応フィルタの係数を修正するために過去の複数の入力信号ベクトルを使用することを述べた. これらのアルゴリズムは Moore-Penrose 型一般逆行列 $A_{r,N}(k)^+$ で表され

$$h_N(k+1) = h_N(k) + A_{r,N}(k)^+ \cdot e_r(k) \quad (2.26)$$

と表現できる. 但し, $h_N(k)$ は時刻 k における推定係数ベクトルであり, 入力状態行列 $A_{r,N}(k)$, 入力状態ベクトル $x_N(k)$ を

$$A_{r,N}(k) \triangleq [x_N(k-r+1), x_N(k-r+2), \dots, x_N(k)]^T \quad (2.27)$$

$$x_N(k) \triangleq [x(k), x(k-1), \dots, x(k-N+1)]^T \quad (2.28)$$

と定義し, 所望信号ベクトル $d_r(k)$, 出力誤差ベクトル $e_r(k)$ をそれぞれ

$$d_r(k) \triangleq A_{r,N}(k) \cdot W_N \quad (2.29)$$

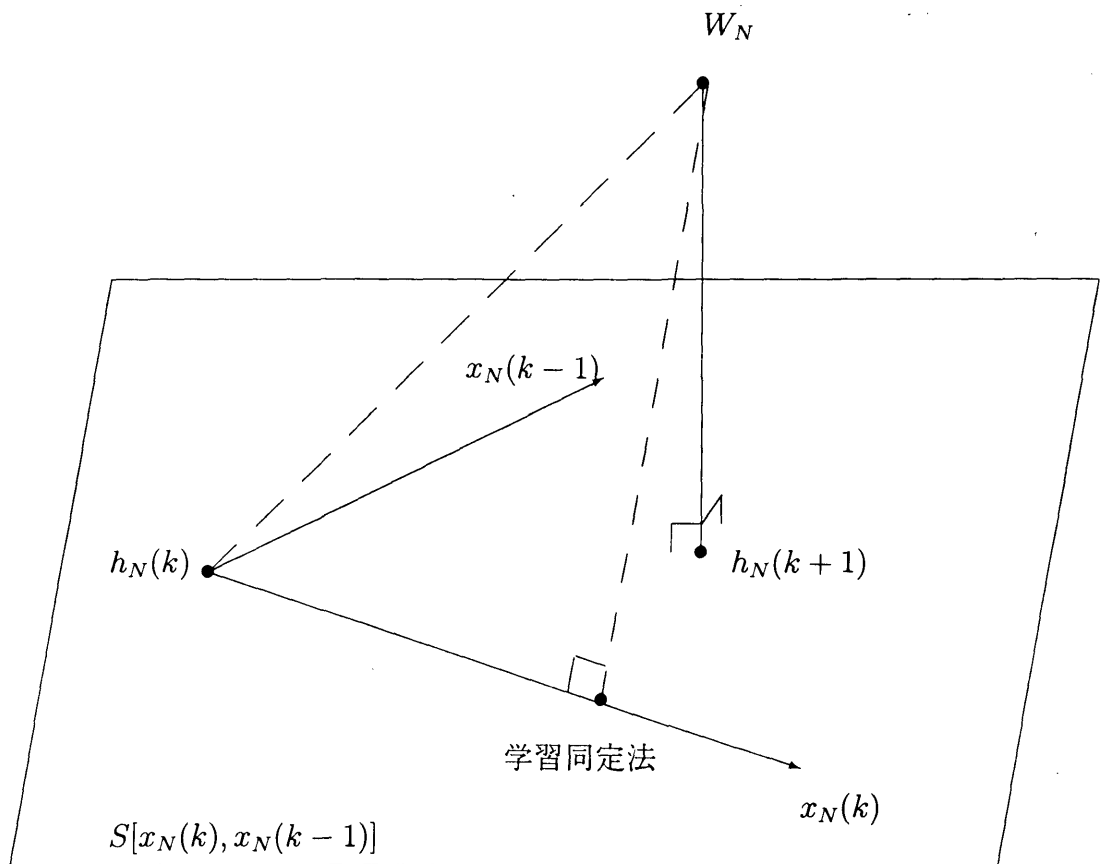


図 2.6: アフィン射影アルゴリズムの幾何学的図

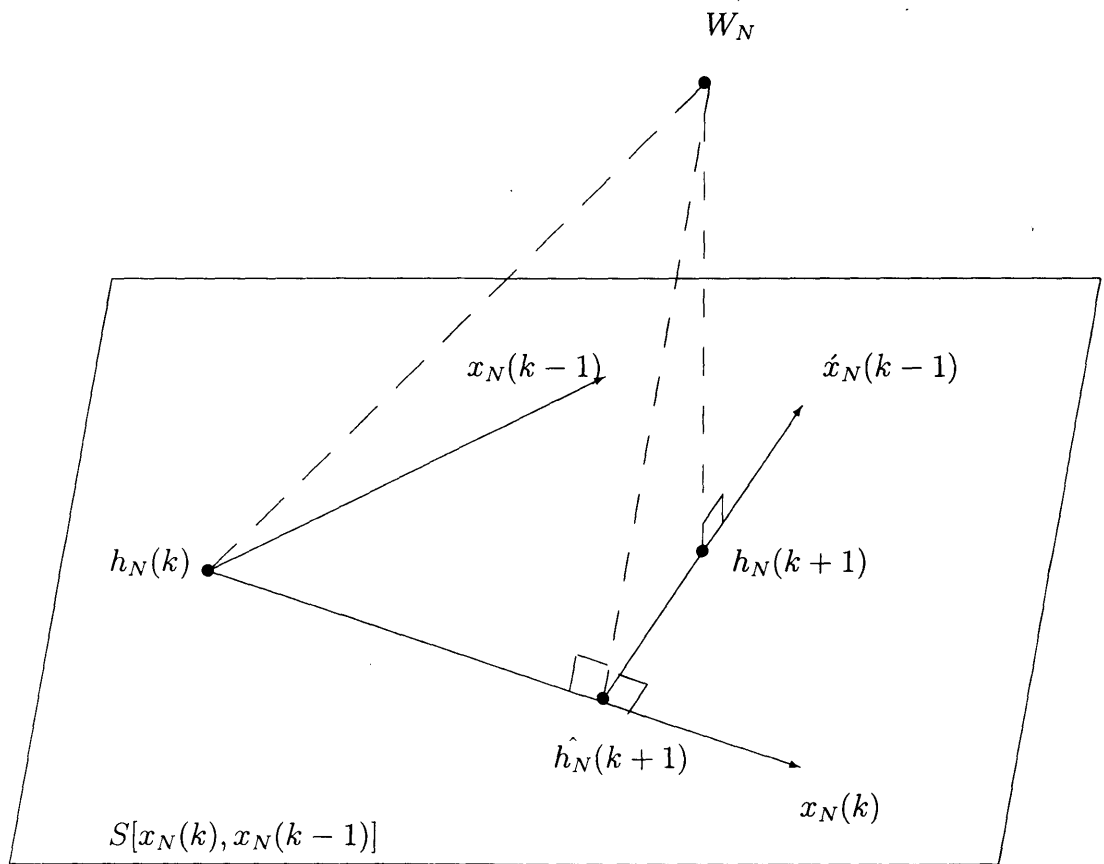


図 2.7: 拡張された学習同定法の幾何学的図

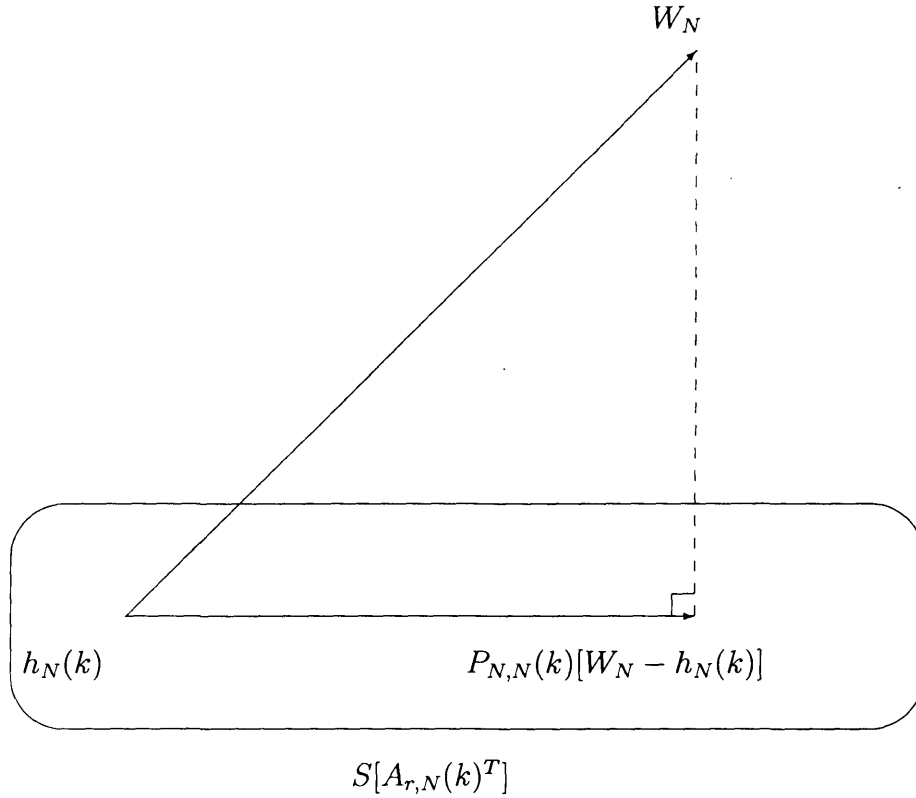


図 2.8: 直交射影行列による部分空間への直交射影

$$e_r(k) \triangleq d_r(k) - A_{r,N}(k) \cdot h_N(k) \quad (2.30)$$

と定義する。ここで、 r はブロック長であり、1回の係数修正に用いられる入力状態ベクトルの数である。すなわち処理はこのブロック長を1単位として行われる。また、 r はベクトルのサイズを超えないことを前提としている ($r \leq N$)。

ここで、行列 $A_{r,N}(k)$ の列ベクトルが張る空間 $S[A_{r,N}(k)^T]$ への直交射影行列を $P_{N,N}(k)$ とすると、 $P_{N,N}(k)$ は

$$P_{N,N}(k) = A_{r,N}(k)^T \cdot [A_{r,N}(k) \cdot A_{r,N}(k)^T]^{-1} \cdot A_{r,N}(k) \quad (2.31)$$

で与えられる。但し、 $A_{r,N}(k)$ の行ベクトルは一次独立とする。図 2.8に直交射影行列による部分空間 $S[A_{r,N}(k)^T]$ への直交射影の図を示す。また、直交射影行列 $P_{N,N}(k)$

を Moore-Penrose 型一般逆行列で表すと,

$$P_{N,N}(k) = A_{r,N}(k)^+ \cdot A_{r,N}(k) \quad (2.32)$$

となり, Moore-Penrose 型一般逆行列は

$$A_{r,N}(k)^+ = [A_{r,N}(k) \cdot A_{r,N}(k)^T]^{-1} \cdot A_{r,N}(k)^T \quad (2.33)$$

で表される. すなわち式 (2.26) は

$$h_N(k+1) = h_N(k) + A_{r,N}(k)^T \cdot [A_{r,N}(k) \cdot A_{r,N}(k)^T]^{-1} \cdot e_r(k) \quad (2.34)$$

と表すことができる.

式 (2.34) からわかるように, この方式は 1 サンプルごとに一般逆行列の計算を繰り返さなければならず, ブロック長 r が大きくなるにつれてフィルタ係数の修正を行うための演算量は膨大なものとなる.

2.4 ブロック直交射影アルゴリズム

AP アルゴリズムの演算量を軽減するという観点から, 古川らは AP アルゴリズムにおいてフィルタ係数を r サンプルごとに行う, ブロック直交射影アルゴリズム (以下, BOP アルゴリズム) を提案している. BOP アルゴリズムはブロックごとにフィルタ係数の修正を行う方式であるため, $k = (L+1)r - 1$ とおくことにより

$$h_N\{(L+1)r\} = h_N(Lr) + [A_{r,N}\{(L+1)r-1\}]^+ \cdot e_r\{(L+1)r-1\} \quad (2.35)$$

と表すことができる. 但し, 式 (2.35) の右辺第一項は $k = Lr$ とおいた. ここで再度 BOP アルゴリズムのブロック L における入力状態行列 $A_{r,N}^{(L)}$, 入力状態ベクトル $x_N^{(L)}$, 所望ベクトル $d_r^{(L)}$, 出力誤差ベクトル $e_r^{(L)}$ をそれぞれ

$$A_{r,N}^{(L)} \triangleq [x_N(1)^{(L)}, x_N(2)^{(L)}, \dots, x_N(r)^{(L)}]^T \quad (2.36)$$

$$x_N(i)^{(L)} \triangleq [x\{r(L-1)+i\}, x\{r(L-1)+i-1\}, \dots, x\{r(L-1)+i-N+1\}]^T \quad (i=1, 2, \dots, r) \quad (2.37)$$

$$d_r^{(L)} \triangleq A_{r,N}^{(L)} \cdot W_N \quad (2.38)$$

$$e_r^{(L)} \triangleq d_r^{(L)} - A_{r,N}^{(L)} \cdot h_N^{(L)} \quad (2.39)$$

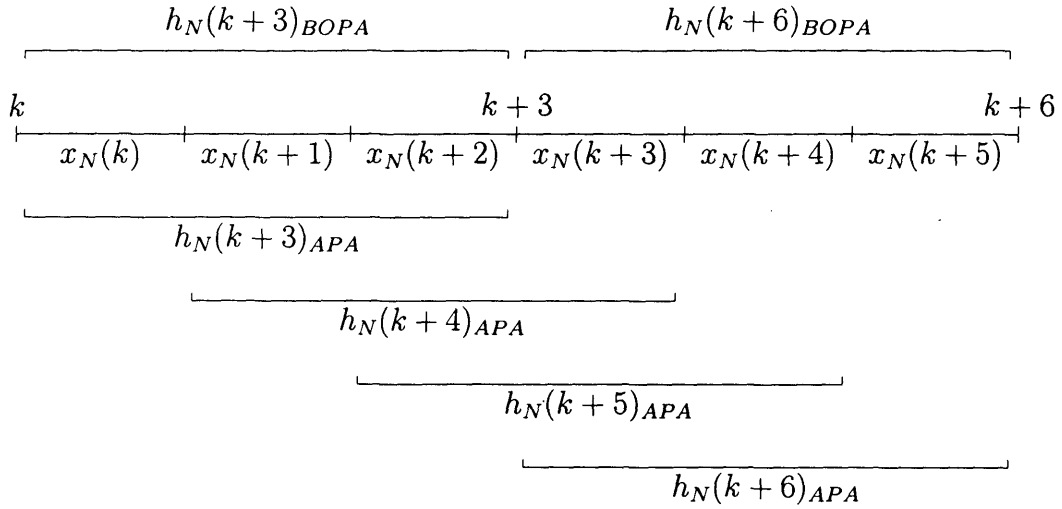


図 2.9: BOP アルゴリズムと AP アルゴリズムのデータの運用法

と定義する. 但し $h_N^{(L)}$ はブロック L におけるフィルタ係数である. これらの定義により式 (2.35) は

$$h_N^{(L+1)} = h_N^{(L)} + A_{r,N}^{(L)+} \cdot e_r^{(L)} \quad (2.40)$$

となる.

ここでブロック直交射影アルゴリズムのデータの運用法を説明するために図 2.9 を用いてブロック長 $r=3$ の場合を例に取り, 式 (2.26) と式 (2.40) の運用法の違いを簡単に説明する. 但し, 図 2.9 において式 (2.26) の方式を APA, 式 (2.40) の方式を BOPA と表記する. まず式 (2.26) の場合, 修正係数ベクトル $h_N(k+3)_{APA}$ は入力状態ベクトル $x_N(k)$, $x_N(k+1)$, 及び $x_N(k+2)$ により求められる. 更に時刻 $k+4$ における修正係数ベクトル $h_N(k+4)_{APA}$ は入力状態ベクトル $x_N(k+1)$, $x_N(k+2)$, $x_N(k+3)$ により得られる. 図からわかるように, 2 本の入力状態ベクトル $x_N(k+1)$, $x_N(k+2)$ が重複している. このように式 (2.26) の方式は 1 サンプルごとに 2 本のベクトルを重複させながら係数ベクトルを修正していく方式である.

次に, 式 (2.40) の場合について説明する. 式 (2.26) の場合と同じく係数ベクトル

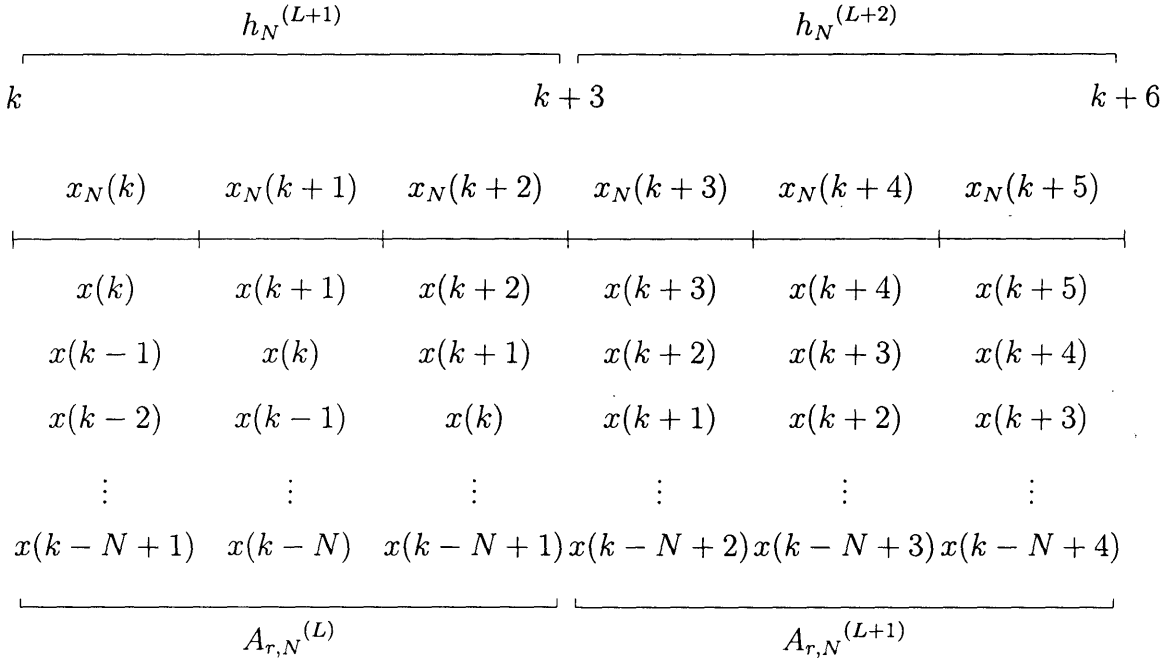


図 2.10: BOP アルゴリズムのデータの運用法

$h_N(k+3)_{BOPA}$ は入力状態ベクトル $x_N(k)$, $x_N(k+1)$, $x_N(k+2)$ により得られる。しかし、次の修正は3サンプル後であり、修正係数ベクトル $h_N(k+6)_{BOPA}$ は入力状態ベクトル $x_N(k+3)$, $x_N(k+4)$, $x_N(k+5)$ によって得られる。すなわち、式(2.40)の方式では入力状態ベクトルを r 個ずつブロック化し、入力状態ベクトルを重複させることなく r サンプルごとに係数の修正を行う方式である。

また、ブロック直交射影アルゴリズムにおける入力データ行列 $A_{r,N}^{(L)}$ の状態を詳しく図示したものが図 2.10である。図において $h_N(k+3)_{BOPA}$ を $h_N^{(L+1)}$ と置いている。

さて、式(2.40)で与えられる $h_N^{(L+1)}$ は両辺に $A_{r,N}^{(L)}$ を乗じることにより

$$A_{r,N}^{(L)} h_N^{(L+1)} = d_r^{(L)} \tag{2.41}$$

が成立し $A_{r,N}^{(L)}$ における行ベクトルのそれぞれに対し所望の出力を与える [42]。このことから h_N を未知数としたとき $A_{r,N}^{(L)} h_N = d_r^{(L)}$ を満足する式において、 $\|h_N^{(L)} -$

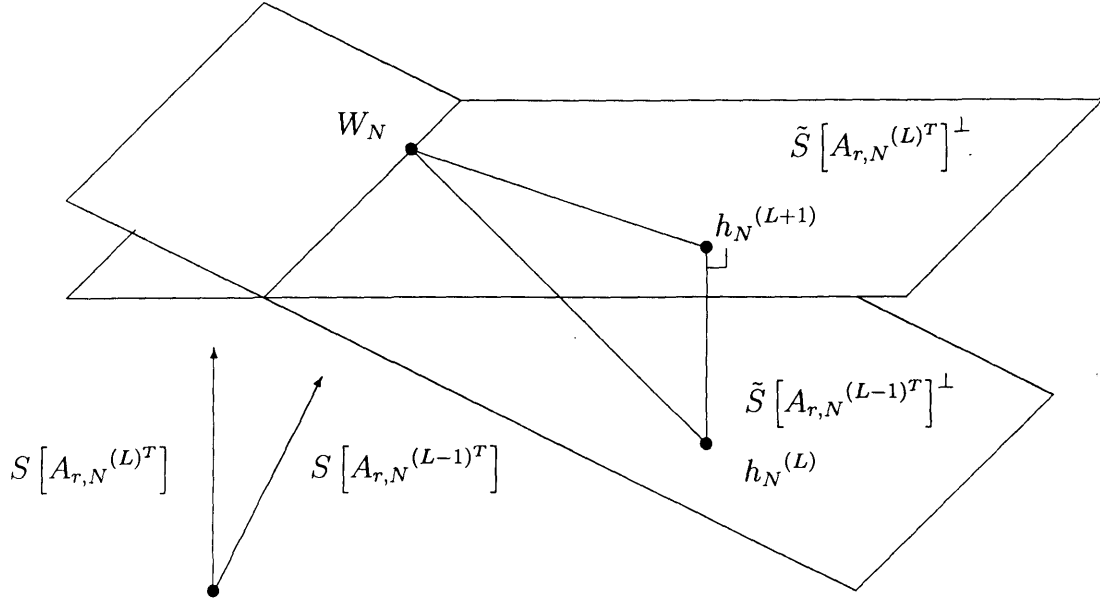


図 2.11: BOP アルゴリズムの幾何学的図

h_N を最小にすることがわかる。また、 $h_N^{(L+1)}$ は 1 ブロック内の誤差の 2 乗和 $J^{(L)}$

$$J^{(L)} = \|d_r^{(L)} - A_{r,N}^{(L)} \cdot h_N\|^2 \quad (2.42)$$

を最小にすることから、ブロック内における部分空間への直交射影演算は誤差 2 乗和最小化問題を解くことであることが理解できる。

BOP アルゴリズムを幾何学的にとらえると図 2.11 のように表現することができる。図 2.11 において、 $S[A_{r,N}^{(L)T}]$ は入力状態行列 $A_{r,N}^{(L)T}$ の列ベクトルの張る空間を示しており、 $\tilde{S}[A_{r,N}^{(L)T}]^\perp$ は $S[A_{r,N}^{(L)T}]$ の直交補空間に平行な超平面を表している。また他も同様である。これは図 2.3 において時刻 k をブロック L と置き換えることにより導くことができる。但し、図 2.3 における解集合 $\Pi(k)$ は図 2.11 における直交補空間に平行な超平面 $\tilde{S}[A_{r,N}^{(L)T}]^\perp$ である。図において $h_N^{(L+1)}$ はベクトル $W_N - h_N^{(L)}$ を入力状態ベクトル $A_{r,N}^{(L)T}$ の列ベクトルの張る部分空間へ直交射影した点である。このように、BOP アルゴリズムは直交射影演算に基礎をおいている。従って、図

2.11からわかるように任意の入力信号に対し

$$\|W_N - h_N^{(L+1)}\| \leq \|W_N - h_N^{(L)}\| \quad (2.43)$$

が成り立ち、推定誤差ベクトルのノルムの収束に関して単調非増加であることが保証される。

このBOPアルゴリズムの計算を具体的に実現する方式の例として、共役こう配法を適用した方式及び、直交射影行列を用いた方式の計算手順を列記する。なお、これらのアルゴリズムは以後の議論において計算機によるシミュレーションなどの比較対象として用いている。

2.4.1 共役こう配法を適用した方式

BOPアルゴリズムは次式

$$h_N^{(L+1)} = h_N^{(L)} + A_{r,N}^{(L)+} e_r^{(L)} \quad (2.44)$$

で与えられることは既に述べた。式(2.44)は式(2.38), (2.39)より, h_N を未知ベクトルとする連立方程式

$$A_{r,N}^{(L)} h_N = A_{r,N}^{(L)} A_{r,N}^{(L)+} d_r^{(L)} \quad (2.45)$$

の解である。ここで、式(2.45)の両辺に $A_{r,N}^{(L)T}$ を乗じると

$$\begin{aligned} A_{r,N}^{(L)T} A_{r,N}^{(L)} h_N &= A_{r,N}^{(L)T} A_{r,N}^{(L)} A_{r,N}^{(L)+} d_r^{(L)} \\ A_{r,N}^{(L)T} A_{r,N}^{(L)} h_N &= A_{r,N}^{(L)T} d_r^{(L)} \end{aligned} \quad (2.46)$$

となり、BOPアルゴリズムは式(2.46)を解くことにより得ることができる。従ってこの方式は、式(2.46)中の行列 $A_{r,N}^{(L)T} A_{r,N}^{(L)}$ の対称性に着目し、式(2.46)の解を共役勾配法により求める方式である。また図2.12に共役こう配法の幾何学的図を示す。図において $h_N^{(opt)(L+1)}$ はブロック L における最適解, $\theta_N(i)^{(L+1)}$ はブロック L のステップ i ($i = 1 \sim 2$) における近似解, $c_N(1)^{(L)}$ は共役ベクトルと呼ばれる修正方向

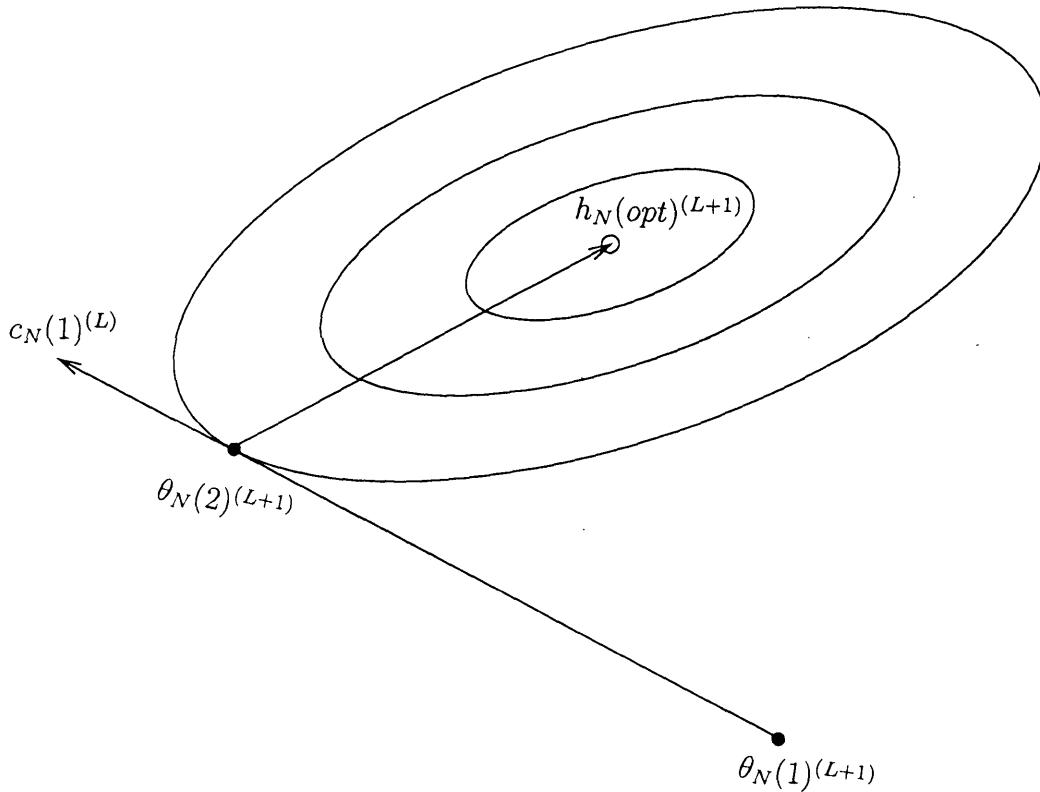


図 2.12: 共役こう配法の幾何学的図

ベクトルを表している．ここで最適解 $h_N(opt)^{(L+1)}$ は $\theta_N^{(L+1)}$ を変数ベクトルとする次の 2 次形式

$$J(\theta_N^{(L+1)}) = \frac{1}{2} \theta_N^{(L+1)T} A_{r,N}^{(L)T} A_{r,N}^{(L)} \theta_N^{(L+1)} - \theta_N^{(L+1)T} A_{r,N}^{(L)T} d_r^{(L)} + c \quad (2.47)$$

を最小にする． c は任意の定数である．従って，

$$J[h_N(opt)^{(L+1)}] \leq J(\theta_N^{(L+1)}) \quad (2.48)$$

が成り立つ．また，図における曲線は $J(\theta_N^{(L)})$ に関する等高線であり，共役こう配法における近似解 $\theta_N(i+1)^{(L+1)}$ は $J[h_N(opt)^{(L+1)}]$ における等高線のある接点に位置することがわかる．

共役こう配法を適用した方式の計算手順を以下に示す．

まず，ブロック L における初期値として

$$g_N(1)^{(L)} = A_{r,N}^{(L)T} A_{r,N}^{(L)} h_N^{(L)} - A_{r,N}^{(L)T} d_r^{(L)} \quad (2.49)$$

$$c_N(1)^{(L)} = -g_N(1)^{(L)} \quad (2.50)$$

$$\theta_N^{(L+1)} = h_N^{(L)} \quad (2.51)$$

を用いる．次に， $(i = 1 \sim m)$ において

$$p_r(i)^{(L)} = A_{r,N}^{(L)} c_N(i)^{(L)} \quad (2.52)$$

$$\alpha(i)^{(L)} = -\frac{g_N(i)^{(L)T} c_N(i)^{(L)}}{\|p_r(i)^{(L)}\|^2} \quad (2.53)$$

$$\theta_N(i+1)^{(L+1)} = \theta_N(i)^{(L+1)} + \alpha(i)^{(L)} c_N(i)^{(L)} \quad (2.54)$$

$$q_N(i)^{(L)} = A_{r,N}^{(L)T} A_{r,N}^{(L)} c_N(i)^{(L)} \quad (2.55)$$

$$g_N(i+1)^{(L)} = g_N(i)^{(L)} + \alpha(i)^{(L)} q_N(i)^{(L)} \quad (2.56)$$

$$\beta(i)^{(L)} = \frac{g_N(i+1)^{(L)T} q_N(i)^{(L)}}{\|p_r(i)^{(L)}\|^2} \quad (2.57)$$

$$c_N(i+1)^{(L)} = -g_N(i+1)^{(L)} + \beta(i)^{(L)} c_N(i)^{(L)} \quad (2.58)$$

を繰り返す．最後に係数ベクトルを

$$h_N^{(L+1)} = \theta_N(m+1)^{(L+1)} \quad (2.59)$$

とし，次のブロックに移る．ここで m は，ブロック L におけるデータ行列 $A_{r,N}^{(L)}$ の異なる特異値の数を示す．従って，共役こう配法 [54]~[56] を適用した方式では，ブロック L において上記手順を m 回繰り返すことにより，ブロックでの最適な係数ベクトルが得られる方式である [57]．

上記の計算手順からわかるように，共役こう配法を適用した方式は計算を行うにあたり，ブロックにおける入力状態行列 $A_{r,N}^{(L)}$ が必要となる．換言すれば，ブロックにおける入力状態行列の全てのデータ（入力状態ベクトル $x_N(i)^{(L)}$ ($i = 1 \sim r$)）が入力されるまで計算を実行することができない．すなわち，この方式はバッチ（一括）処理により計算される方式であるということが出来る．従って，リアルタイム処理を考慮した場合，データを蓄積する間は計算を行うことが不可能であり，全てのデータを得た後に一度に大量の演算を行わなくてはならない．このことから，ブロックにおいて全てのデータの入力を待たずに計算が可能な方式が期待される．この問題に対しては，次の直交射影行列を用いた方式により解決される．また，共役

こう配法を適用した方式は演算時間上，ハードウェア構成上の問題など，何らかの理由で計算をうち切った場合，収束速度が劣化するということが指摘される．この問題に対しては文献 [52][53] で議論されており，第 5 章において文献 [52][53] の方式の収束速度と演算量を考慮した方式について議論を行っている．

2.4.2 直交射影行列を用いた方式

この方式は，直交射影行列を用いて算出された $S[A_{r,N}^{(L)T}]$ の直交基底を修正方向とする学習同定法によって BOP アルゴリズムを実現する方式である．

直交射影行列を用いた方式の計算手順を以下に示す．

ブロック L における初期値として

$$P_{N,N}(1)^{(L)} = I_{N,N} \quad (2.60)$$

$$h_N(1)^{(L)} = h_N^{(L)} \quad (2.61)$$

とする．但し， $I_{N,N}$ は単位行列である．次に， $(i = 1 \sim r)$ において

$$P_{N,N}(i+1)^{(L)} = P_{N,N}(i)^{(L)} - \frac{P_{N,N}(i)^{(L)} x_N(i)^{(L)} x_N(i)^{(L)T} P_{N,N}(i)^{(L)}}{x_N(i)^{(L)T} P_{N,N}^{(L)} x_N(i)^{(L)}} \quad (2.62)$$

$$c_N(i)^{(L)} = \frac{P_{N,N}(i)^{(L)} x_N(i)^{(L)}}{x_N(i)^{(L)T} P_{N,N}^{(L)} x_N(i)^{(L)}} \quad (2.63)$$

$$e(i)^{(L)} = d(i)^{(L)} - h_N(i)^{(L)T} x_N(i)^{(L)} \quad (2.64)$$

$$h_N(i+1)^{(L)} = h_N(i)^{(L)} + \alpha e(i)^{(L)} c_N(i)^{(L)} \quad (2.65)$$

を繰り返す．最後に

$$h_N^{(L+1)} = h_N(r+1)^{(L)} \quad (2.66)$$

とし，次のブロックへ移る．

このように直交射影行列を用いた方式は，ブロックにおいて入力状態行列 $A_{r,N}^{(L)}$ の行ベクトル $x_N(i)^{(L)}$ ($i = 1 \sim r$) の全ての入力を待たずに，これら入力状態ベクトルが入力される度に計算の実行が可能な方式であり，共役こう配法を適用した方式よりも高速な計算が可能な方式である．また，この方式は行列とベクトルの積を必要

とするが、これらの演算はシストリックアレイを用いた並列計算によって高速かつ能率的に計算することができる [58]. 従って、この方式は並列計算アーキテクチャの使用を前提とした場合、短時間でフィルタ係数を更新することができる方式である. しかしながら直交射影行列を用いた方式は演算量が $O(N^2)$ であり、更なる演算量の軽減化が望まれる. この問題に関しては第 3 章において議論を行っている.

2.5 適応信号処理の適用例

現在において、適応信号処理の理論は様々な分野に利用されている. ここでは、代表的なものとして適応ノイズキャンセラ、自動等化器、適応エコーキャンセラ及びアクティブノイズコントロール (以下、ANC) を図を用いながら簡単に説明する. 但し $K(z)$, $H(z)$, $C(z)$ は離散時間伝達関数を表している.

2.5.1 適応ノイズキャンセラ

所望信号が雑音に埋もれているような場合、雑音の影響をできる限り少なくし、なるべく所望信号に近い信号を得るといふ信号処理は応用範囲が広く、現在でも研究が続けられている.

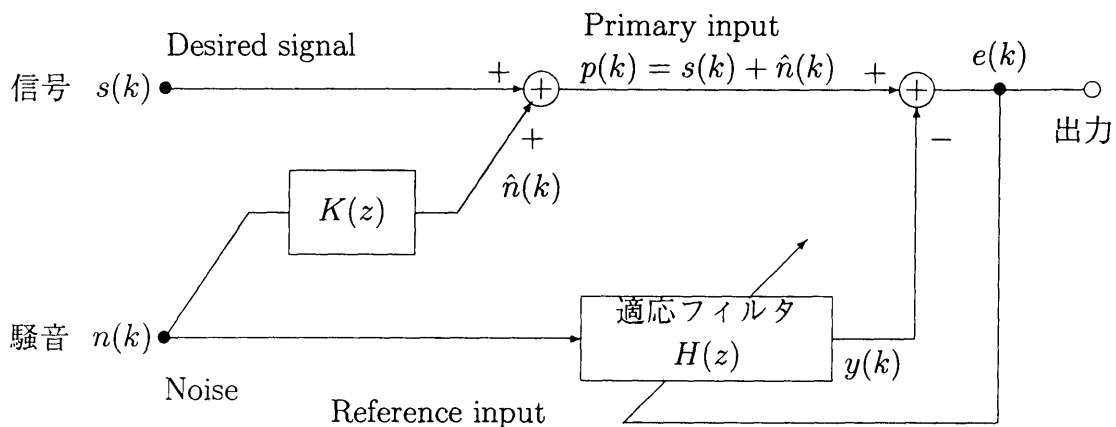


図 2.13: 適応ノイズキャンセラ

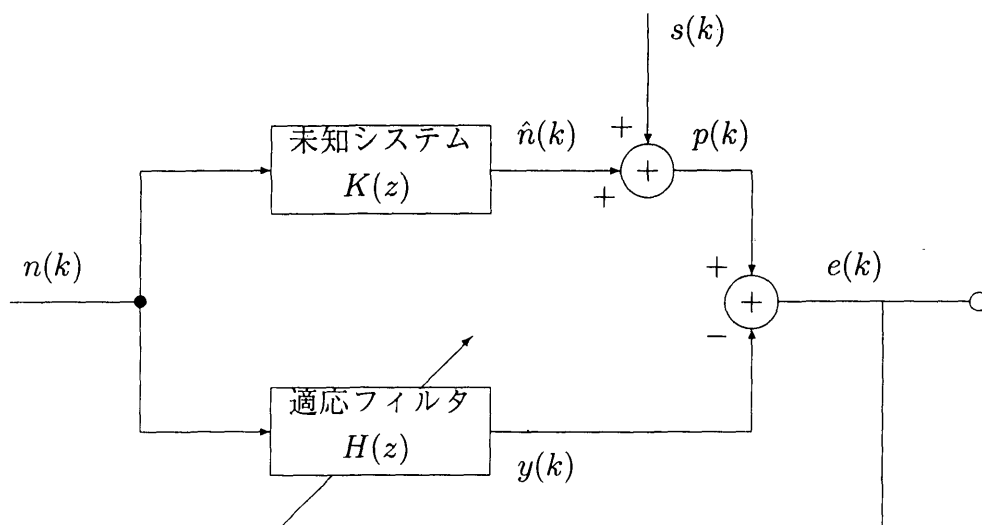


図 2.14: 適応ノイズキャンセラにおけるパラメータ推定
 $H(z) = K(z)$ なら $y(k) = \hat{n}(k)$

さて、図 2.13 は適応ノイズキャンセラの基本的なモデルである。もし図 2.13 のように雑音だけを観測することが可能であるならば、適応ノイズキャンセラによってかなり効率よく所望信号だけを取り出すことが可能である。また、図 2.13 は図 2.14 のようにパラメータ推定問題としてとらえることができる。主入力として、所望信号 $s(k)$ と雑音 $\hat{n}(k)$ の和、すなわち主入力信号 $p(k)$ が入力されるとする。所望信号 $s(k)$ が参照入力に混入しない場合 (雑音 $n(k)$ と無相関)、参照入力 $n(k)$ に対し適当な線形処理を施し、主入力信号 $p(k)$ から減算することにより所望信号 $s(k)$ のみを取り出すことができる。この場合、推定システムの伝達関数 $H(z)$ と未知システムの伝達関数 $K(z)$ が等しくなったとき推定システムの出力行号 $y(k)$ と主入力における雑音 $\hat{n}(k)$ が等しくなり完全に雑音を消去することができる。すなわち、あるシステム $K(z)$ の構造が既知 (FIR 型と仮定) であるならば、適応フィルタ $H(z)$ により $K(z)$ のパラメータ (インパルス応答) を推定することにより、 $y(k)$ を $\hat{n}(k)$ に近づけ $e(k)$ を最小にすることが可能である。従って、図 2.13 における $K(z)$ のパラメータ推定が重要な意味を持つことが理解できよう。

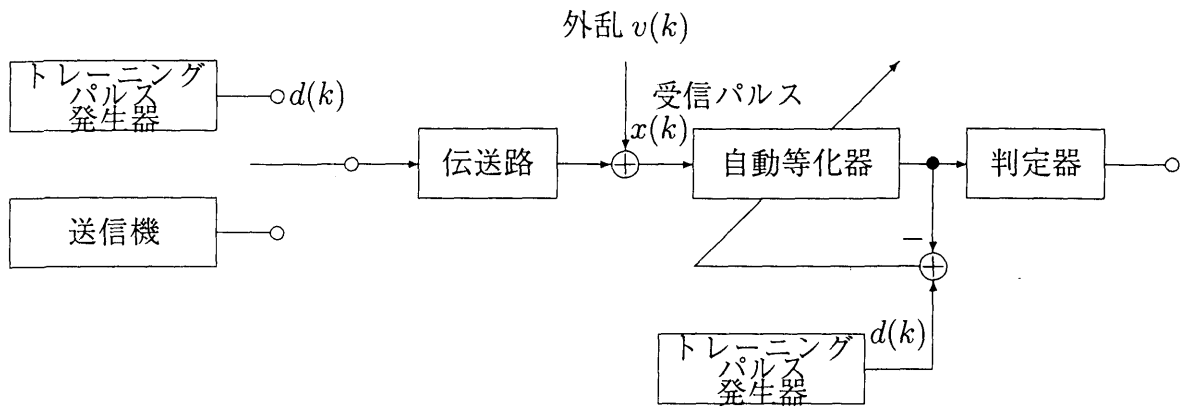


図 2.15: デジタル伝送系

2.5.2 自動等化器

通信を行う際、信号が受信点に到達するまでに、伝送路歪、雑音などの影響を受ける。このような状況下において、信号の劣化を補償するために自動等化器が用いられる。自動等化器では実際に通信を行う前(トレーニング期間)に、先だてて決められているパターン、すなわちトレーニング信号を送信する。従って、この期間にお

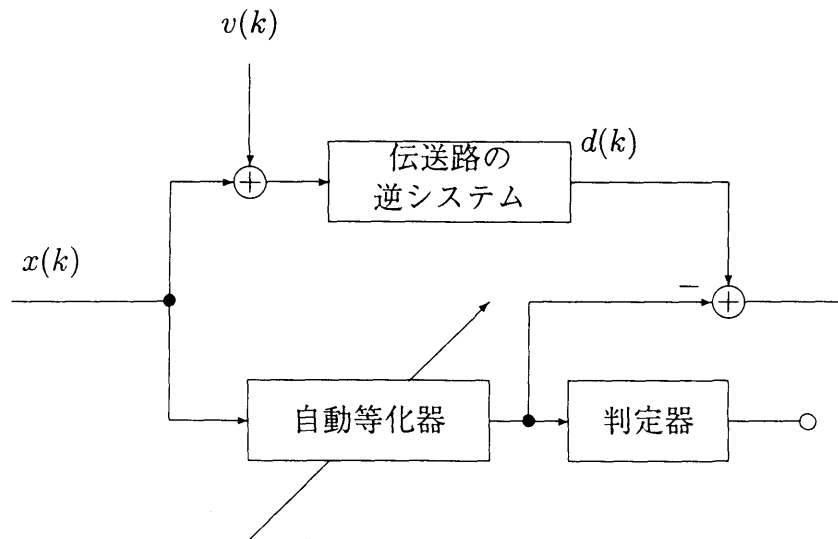


図 2.16: 自動等化器におけるパラメータ推定

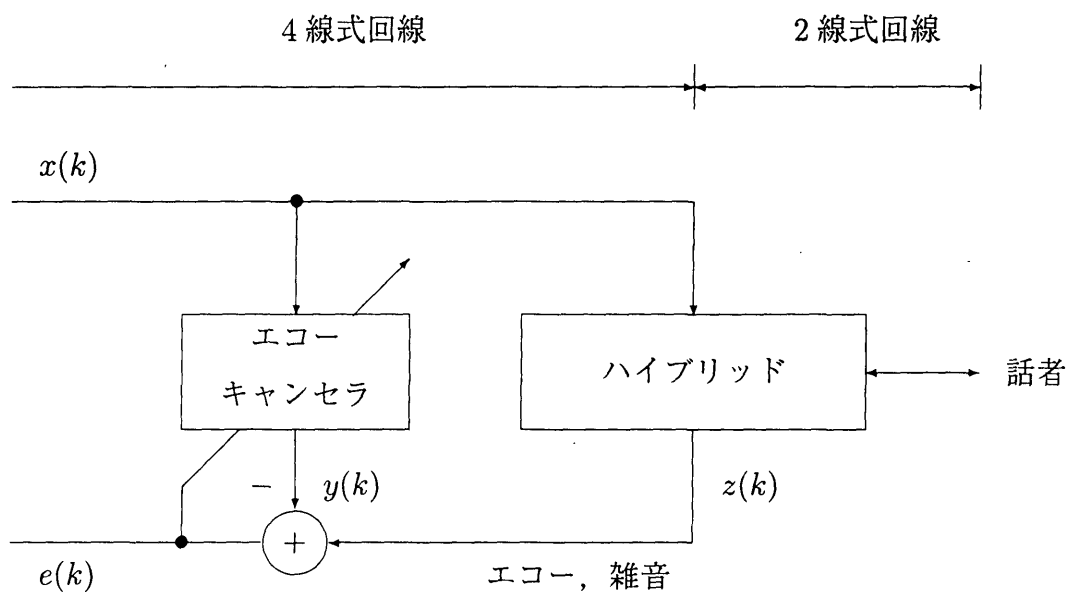


図 2.17: 適応エコーキャンセラの例 1

いては送信される信号が予めわかっているため、自動等化器からの出力信号と、トレーニング信号の差をとることにより、等化器の係数を調整する。図 2.15 にデジタル伝送系の図を示す。この自動等化器もノイズキャンセラ同様、パラメータ推定問題に帰着する。そのようすを図 2.16 に示す。

2.5.3 適応エコーキャンセラ

国際電話のような長距離回線では信号増幅などの理由から 4 線式回線が用いられているが、一般加入者回線のように比較的近距离なものには 2 線式回線が用いられる。2 線式回線と 4 線式回線の接続部にはハイブリッドが導入されインピーダンス整合が行われる。しかしながら、加入者側のインピーダンスは一般には異なっているので、全ての回線に対してインピーダンス整合を行うのは不可能である。このようなインピーダンスの不整合によりエコー（ある程度遅延された自分の声）が生じ、会話がしづらくなる。また、近年のマルチメディア化の発達においてテレビ会議システムに代表される議論も活発されるようになってきた [60]~[63]。このテレビ会議

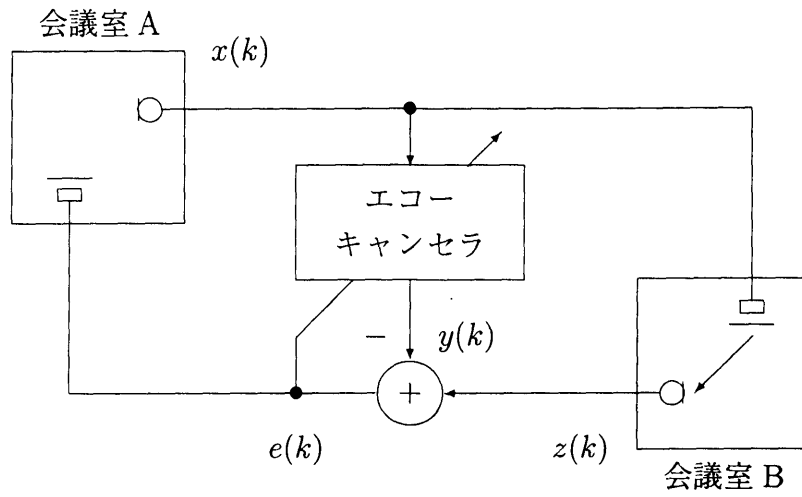


図 2.18: 適応エコーキャンセラの例 2

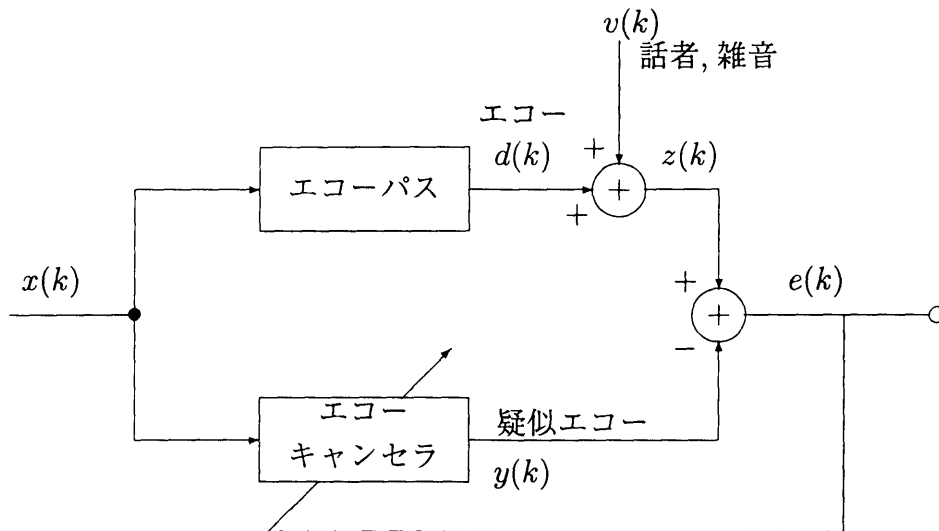


図 2.19: 適応エコーキャンセラにおけるパラメータ推定

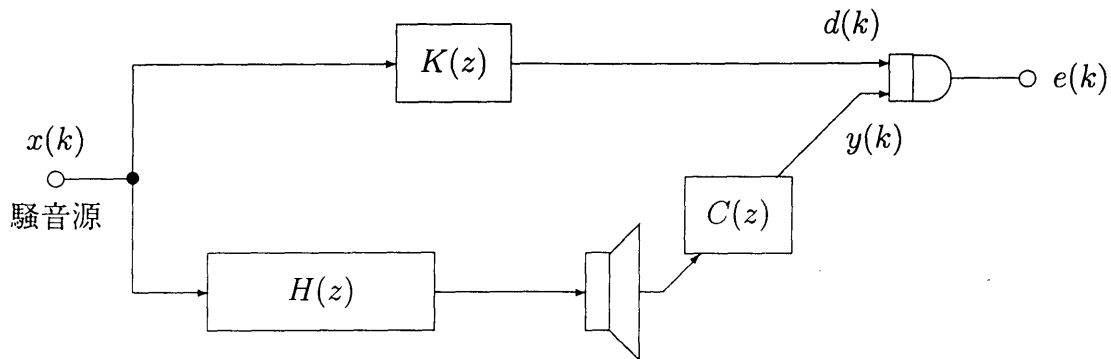


図 2.20: アクティブノイズコントローラ

システムにおいてもエコーによりスムーズな会議の進行が行えないなどの問題が生じることから、エコーの除去は重要な問題であるといえる。これらを図 2.17, 2.18 に示す。図 2.17, 2.18 もまたパラメータ推定問題に帰着する。この図を 2.19 に示す。エコーキャンセラは受信信号 $x(k)$ とエコー $d(k)$ からエコーパスを推定し、疑似エコー $y(k)$ を作りだし、実際のエコーから減算することによりエコーの除去を行う。

2.5.4 アクティブノイズコントロール

近年、生活環境における騒音を低減させ快適な空間を得るための研究が盛んに行われている。従来の吸音・共鳴現象などを利用した消音技術に対し、理論研究が中心であったアクティブノイズコントロール (Active Noise Controller: ANC) が活発に研究されるようになった。自動車の走行による騒音など、低周波数成分を含む騒音はかなり厚い遮へい物を必要とし現実的ではない。このようなことから、ANC は適応信号処理の技術を応用し、主として比較的 low 周波数成分からなる騒音と同振幅・逆位相の信号を人工的に作り出しこれにより騒音を低減させる方式とすることができる。特に、現在では自動車に代表されるような閉空間において、騒音低減を目的とする研究に応用されている [59][64]。

上記のように、ANC は騒音を除去したい場所にスピーカ (付加音源) とセンサ (受音点) を配置し、それらの間で騒音と逆相となる音を生成することに基礎をおいて

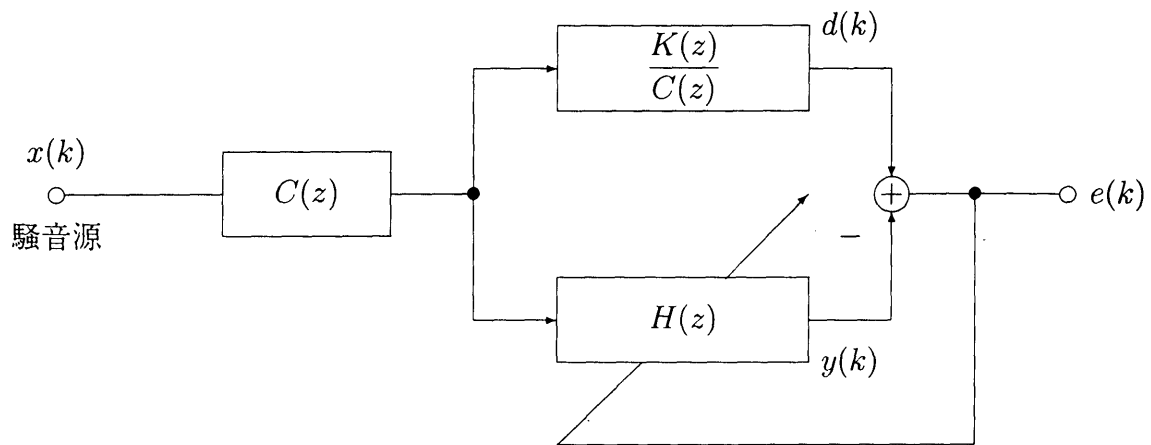


図 2.21: アクティブノイズコントローラにおけるパラメータ推定

いる。すなわち，図 2.20において $e(k)$ の自乗平均値を最小化することを目的とするパラメータ推定問題となる [65]。このようすを図 2.21に示す。図 2.21において，適応フィルタ $H(z)$ は $\frac{K(z)}{C(z)}$ を推定することにより，受信点における騒音を除去することができる。ここで，図中 $C(z)$ は何らかの方法で予め求めておく必要がある。また，適応フィルタに入力される信号は， $x(k)$ をフィルタリングした信号を用いることから，このような構成は Filtered-X と呼ばれている。

2.6 むすび

このように適応信号処理において，何らかの線形システムのインパルス応答を推定することにより必要となる信号を得ることが可能となる。パラメータの推定を行うに当たり，エコーキャンセラのように長い残響時間 (インパルス応答) に対応できる [13]，あるいは入力される信号が有色信号であるなどの場合，収束の高速な適応アルゴリズムが必要である。また，演算量が少なく高速な実行が可能であるということも要求される。さらに，未知システムからの出力信号に観測雑音が重畳しているような場合においても精度良く推定することも必要となる。従って以後の章では，パラメータ推定を行うために重要となる適応アルゴリズムについて，有用であると

考えられる方式に関する提案及び検討を行う。

第 3 章

UD 分解法によるブロック直交射影アルゴリズム

フィルタ係数を修正する適応アルゴリズムは適応信号処理において重要な役割を果たしており、これまでも各種の手法が提案されている。それらの中でも入力信号ベクトルが張る部分空間への直交射影演算に基づくアルゴリズムは、入力信号が相関を有する場合においても最適なフィルタ係数を高速に推定することができるという特長を有している。直交射影演算に基づく方式の中で、比較的演算量と収束特性のバランスがとれた方式として、ブロック直交射影アルゴリズムが知られている。これは Moore-Penrose 型一般逆行列を含んだ形で表現され、このアルゴリズムを具体的に実行する方式がいくつか提案されている。本章では自己相関行列の UD 分解とブロック逆行列に関する補題 [66] を用いることによって、従来方式よりも少ない演算量で Moore-Penrose 型一般逆行列を含むブロック直交射影アルゴリズムを実行できる手法について議論している。本手法は、任意のブロック中の i 番目のステップに対応する直交射影行列を、1 ステップ前における直交射影行列を利用して算出する点に特徴がある。また、計算機シミュレーションにより、本方式と従来から提案されている手法との比較検討を行い、本アルゴリズムの優位性について議論している。

3.1 まえがき

適応信号処理では，フィルタの係数を修正する適応アルゴリズムが重要な位置を占めており，演算量の軽減と収束速度の高速化の観点から，従来より様々な方式が提案されていることは既に述べた．

それらの中でも，LMS アルゴリズム [67] や学習同定法 [19] は演算量が少なくハードウェア構成が容易であるため，広く使用されている．しかしながら，これらの方式はシステム (フィルタ) への入力信号が有色である場合，収束速度が著しく劣化するという問題点を有している．このような収束速度劣化を改善する手法として，1 回の係数修正を行うために過去の複数個の入力信号ベクトルを用いる，拡張された学習同定法 [43]，アフィン射影算法 [44] が提案されている．これらは有色入力信号に対しても良好な収束特性を得ることができる反面，係数修正に用いる入力信号ベクトルの数が多いほど演算量の増加が顕著となり，ハードウェア構成が困難になるなどの問題点を有している．

周知のように適応アルゴリズムには，1) 1 サンプル当たりの演算量の軽減，2) 有色信号入力時における収束速度の劣化防止などが要求される．そこでこれらの要求に応える 1 つの方法として，ブロック直交射影アルゴリズム (Block Orthogonal Projection Algorithm, 以下 BOP アルゴリズム)[45] が提案されている．これは基本的に文献 [43]，[44] の方式と同様，入力信号ベクトルが張る多次元部分空間への直交射影演算に基づいており，フィルタ係数をブロック単位で修正する方式である．

良く知られているように，BOP アルゴリズムは Moore-Penrose 型一般逆行列を含んだ形で表わされる．この BOP アルゴリズムを具体的に実行する手法として，共役勾配法を適用した方式 (以下，CGM-BOPA)[47] や直交射影行列を用いた方式 (以下，OPM-BOPA)[48] などが提案されている．

文献 [48] によれば，OPM-BOPA はあるブロック内の任意のステップにおいて，1 本の入力信号ベクトルが入力されるごとにフィルタ係数を修正し，この手順を繰り返すことによって，対応するブロックでの最適係数ベクトルを求める方式である．また，未知系の出力に観測雑音が重畳している状況下で，OPM-BOPA と CGM-BOPA

を比較すると、前者は後者よりも安定した収束特性を得ることができるとされている。更に OPM-BOPA はその構造上、並列計算に適したアーキテクチャを使用することにより、高速な係数更新が可能であるとも報告されている。しかしながら、このアルゴリズムはサンプル当たりの演算量が $O(N^2)$ であり、これの軽減が望まれていた。但し、 N は未知系と推定系を FIR 型フィルタとしたときのフィルタタップ数である。

そこで本章では、OPM-BOPA よりも少ない演算量で BOP アルゴリズムを実行できる一手法を提案し、その基本的特性について議論する。本方式は、あるブロックにおける任意の計算ステップ i で対象となる $(i \times i)$ の自己相関行列を UD 分解することにその基礎をおいている。後述するように、本方式の係数修正のための入力信号の運用法は、OPM-BOPA の場合と同様である。

いま、あるブロック中の任意のステップ i における Moore-Penrose 型一般逆行列が、上記の自己相関行列の逆行列を含む形で表現され、この自己相関行列が UD 分解されているものとする。このとき、次のステップで対象となる $(i+1) \times (i+1)$ の自己相関行列の逆行列は、ブロック逆行列の補題を用いて容易に計算でき、その結果このステップにおける Moore-Penrose 型一般逆行列を求めることが可能となる。本方式は任意のブロックに対し、自己相関行列の UD 分解とブロック逆行列の補題を用いてステップごとに Moore-Penrose 型一般逆行列を更新することにより、そのステップでの係数ベクトルを算出し、この手順を一定回数繰り返すことによって対応するブロックでの最適係数ベクトルを求める手法 [68][69] である。

以下、**3.2** では、BOP アルゴリズムについて簡単に説明し、以後の議論で必要となる準備を行う。**3.3** では、前章での結果を利用して自己相関行列の UD 分解に基づく本アルゴリズムを導く。次いで演算量の観点から文献 [48] のアルゴリズムとの比較を行う。**3.4** では計算機シミュレーションにより、本アルゴリズムの基本的な収束特性を示すと共に従来方式との性能比較を行う。

3.2 準備

ここでは、問題設定及び以後の議論に必要となる準備を行う。なお、本章では特に断らない限り $(N \times i)$ の行列 A , $(N \times 1)$ のベクトル b をそれぞれ $A_{N,i}$, b_N と表記する。

さて、第2章で述べたように適応フィルタの問題は未知系の構造が既知であれば、パラメータ推定問題に帰着できる。従って、未知系が FIR 型のものを考え議論を行う。すなわち、定常な入力信号 $\{x(k), -\infty \leq k \leq \infty\}$ に対して

$$z(k) = \sum_{m=0}^{N-1} w(m) \cdot x(k-m) \quad (3.1)$$

で定まる観測信号 $\{z(k), -\infty < k < \infty\}$ を出力する未知系を考える。ここで、 $w(m) (m = 0 \sim N-1)$ は推定すべき FIR フィルタのインパルス応答であり、これを

$$W_N \triangleq [w(0), w(1), \dots, w(N-1)]^T \quad (3.2)$$

と定義する。一方、適応フィルタの入出力関係は

$$y(k) = \sum_{m=0}^{N-1} h(m)^{(k)} \cdot x(k-m) \quad (3.3)$$

で与えられる。ここで $h(m)^{(k)}$ は、時刻 k における $(m+1)$ 番目のフィルタ係数である。

問題はある評価量に基づき、適当なアルゴリズムを用いて、 $h(m)^{(k)}$ により W_N を推定することである。

この問題に対する一つのアプローチとして、既に古川らにより BOP アルゴリズム [45] が提案されている。BOP アルゴリズムは、次式

$$h_N^{(L+1)} = h_N^{(L)} + [A_{N,r}^{(L)T}]^+ \cdot e_r^{(L)} \quad (3.4)$$

により、ブロックごとにフィルタ係数を修正する方式である。ここで $L, r, h_N^{(L)}, A_{N,r}^{(L)}, [A_{N,r}^{(L)T}]^+$ および $e_r^{(L)}$ はそれぞれブロック番号、ブロック長、ブロック L におけるフィルタ係数、入力状態行列、 $A_{N,r}^{(L)T}$ の Moore-Penrose 型一般逆行列および出力誤差ベクトルである。また、各諸量は

$$A_{N,r}^{(L)} \triangleq [x_N(1)^{(L)}, x_N(2)^{(L)}, \dots, x_N(r)^{(L)}] \quad (3.5)$$

$$\begin{aligned} x_N(1)^{(L)} &\triangleq [x\{N + (L-1)r - 1\}, x\{N + (L-1)r - 2\}, \dots, x\{(L-1)r\}]^T \\ x_N(2)^{(L)} &\triangleq [x\{N + (L-1)r\}, x\{N + (L-1)r - 1\}, \dots, x\{(L-1)r + 1\}]^T \\ &\vdots \\ x_N(r)^{(L)} &\triangleq [x\{N + Lr - 2\}, x\{N + Lr - 3\}, \dots, x\{Lr - 1\}]^T \end{aligned} \quad (3.6)$$

$$\begin{aligned} e_r^{(L)} &\triangleq z_r^{(L)} - A_{N,r}^{(L)T} \cdot h_N^{(L)} \\ &= A_{N,r}^{(L)T} [W_N - h_N^{(L)}] \end{aligned} \quad (3.7)$$

で定義される。ただし、 $x_N(j)^{(L)} (j = 1 \sim r)$ 、 $z_r^{(L)}$ はそれぞれブロック L に対応する入力信号ベクトル、所望信号ベクトルである。さて、式 (3.4) は Moore-Penrose 型一般逆行列を含んでおり、式 (3.4) を具体的に実行する方式が文献 [47], [48] で提案されている。特に文献 [48] の方式 (OPM-BOPA) は、観測雑音が存在する状況下でも安定した収束特性を有するとされている。

本章の目的は、上記のような特徴を持つ OPM-BOPA よりも、演算量の少ない方式を提案することである。以下において、提案方式を導出する準備を行う。

まず、 i 個の入力信号ベクトル $x_N(1)^{(L)} \sim x_N(i)^{(L)}$ からなる $(N \times i)$ の行列 $A_{N,i}(i)^{(L)}$ を

$$\begin{aligned} A_{N,i}(i)^{(L)} &\triangleq [x_N(1)^{(L)}, x_N(2)^{(L)}, \dots, x_N(i)^{(L)}] \\ &= [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}] \end{aligned} \quad (3.8)$$

と定義する。式 (3.8) 中の i 個のベクトル $x_N(1)^{(L)}, x_N(2)^{(L)}, \dots, x_N(i)^{(L)}$ は式 (3.6) の定義に従う量であり、 $i = r$ としたときの $A_{N,i}(i)^{(L)}$ は式 (3.5) の $A_{N,r}^{(L)}$ に一致する。さて、行列 $A_{N,i}(i)^{(L)}$ の階数 (ランク) を

$$\text{Rank} [A_{N,i}(i)^{(L)}] = i \quad (\leq r \leq N) \quad (3.9)$$

と仮定すれば、 $[A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]$ は正則であり逆行列が存在する。そこで、この逆行列を $(i-1) \times (i-1)$ の行列 $[A_{N,i-1}(i-1)^{(L)T} \cdot A_{N,i-1}(i-1)^{(L)}]$ の逆行列から

UD 分解を用いて求めることを考えよう。これが本章で提案する方式を導出するための基礎となる。まず、行列 $[P_{i,i}(i)^{(L)}]^{-1}$ を次式により

$$[P_{i,i}(i)^{(L)}]^{-1} \triangleq A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)} \quad (3.10)$$

と定義する。

$i \geq 2$ の場合、式 (3.8) より

$$A_{N,i}(i)^{(L)} = [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}] \quad (3.11)$$

であるので、式 (3.10) は

$$\begin{aligned} [P_{i,i}(i)^{(L)}]^{-1} &= A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)} \\ &= [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}]^T \cdot [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}] \\ &= \left[\begin{array}{c|c} \widetilde{A_{i-1,i-1}}^{(L)} & a_{i-1}^{(L)} \\ \hline a_{i-1}^{(L)T} & \|x_N(i)^{(L)}\|^2 \end{array} \right] \\ &= \left[\begin{array}{c|c} [P_{i-1,i-1}(i-1)^{(L)}]^{-1} & a_{i-1}^{(L)} \\ \hline a_{i-1}^{(L)T} & \|x_N(i)^{(L)}\|^2 \end{array} \right] \end{aligned} \quad (3.12)$$

となる。但し、 $\widetilde{A_{i-1,i-1}}^{(L)}$ 、 $a_{i-1}^{(L)}$ はそれぞれ

$$\widetilde{A_{i-1,i-1}}^{(L)} = A_{N,i-1}(i-1)^{(L)T} \cdot A_{N,i-1}(i-1)^{(L)} \quad (3.13)$$

$$a_{i-1}^{(L)} = A_{N,i-1}(i-1)^{(L)T} \cdot x_N(i)^{(L)} \quad (3.14)$$

である。また、 $\|\cdot\|$ はユークリッドノルムを表す。ところで、 $[P_{i,i}(i)^{(L)}]^{-1}$ は式 (3.9)、(3.10) より正定値対称行列であるから、

$$[P_{i,i}(i)^{(L)}]^{-1} = L_{i,i}(i)^{(L)} \cdot D_{i,i}(i)^{(L)} \cdot L_{i,i}(i)^{(L)T} \quad (3.15)$$

のように下三角行列、対角行列、上三角行列の積に UD 分解できる。ここで、下三角行列 $L_{i,i}(i)^{(L)}$ 、対角行列 $D_{i,i}(i)^{(L)}$ は $(i \times i)$ の正方行列であり、それぞれ

$$L_{i,i}(i)^{(L)} = \left[\begin{array}{c|c} L_{i-1,i-1}(i-1)^{(L)} & \mathbf{0}_{i-1} \\ \hline \ell_{i-1}(i)^{(L)T} & 1 \end{array} \right] \quad (3.16)$$

$$D_{i,i}(i)^{(L)} = \left[\begin{array}{c|c} D_{i-1,i-1}(i-1)^{(L)} & \mathbf{0}_{i-1} \\ \hline \mathbf{0}_{i-1}^T & d(i)^{(L)} \end{array} \right] \quad (3.17)$$

と書ける. 但し, $l_{i-1}(i)^{(L)}$ は行列 $L_{i,i}(i)^{(L)}$ の i 行 1 列から i 行 $(i-1)$ 列までの要素を成分とする $(i-1) \times 1$ のベクトル, $d(i)^{(L)}$ は行列 $D_{i,i}(i)^{(L)}$ の i 行 i 列の要素である. また, $\mathbf{0}_{i-1}$ は $(i-1) \times 1$ のゼロベクトルである. 従って式 (3.12) ~ (3.17) から次の関係

$$\begin{aligned} & \left[P_{i-1,i-1}(i-1)^{(L)} \right]^{-1} \\ &= L_{i-1,i-1}(i-1)^{(L)} \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot L_{i-1,i-1}(i-1)^{(L)T} \end{aligned} \quad (3.18)$$

$$\begin{aligned} & A_{N,i-1}(i-1)^{(L)T} \cdot x_N(i)^{(L)} \\ &= L_{i-1,i-1}(i-1)^{(L)} \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot l_{i-1}(i)^{(L)} \end{aligned} \quad (3.19)$$

$$\begin{aligned} & x_N(i)^{(L)T} \cdot A_{N,i-1}(i-1)^{(L)} \\ &= l_{i-1}(i)^{(L)T} \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot L_{i-1,i-1}(i-1)^{(L)T} \end{aligned} \quad (3.20)$$

$$\begin{aligned} & d(i)^{(L)} \\ &= \|x_N(i)^{(L)}\|^2 - l_{i-1}(i)^{(L)T} \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot l_{i-1}(i)^{(L)} \end{aligned} \quad (3.21)$$

を得る. また, $P_{i,i}(i)^{(L)}$ は式 (3.15) より

$$P_{i,i}(i)^{(L)} = \left[L_{i,i}(i)^{(L)} \right]^{-T} \cdot \left[D_{i,i}(i)^{(L)} \right]^{-1} \cdot \left[L_{i,i}(i)^{(L)} \right]^{-1} \quad (3.22)$$

で与えられる. 但し, $\left[L_{i,i}(i)^{(L)} \right]^{-T}$ は $\left[L_{i,i}(i)^{(L)} \right]^{-1}$ の転置を表す. ここで, 式 (3.16), (3.17) にブロック逆行列の補題を適用すると, 下三角行列 $\left[L_{i,i}(i)^{(L)} \right]^{-1}$, 対角行列 $\left[D_{i,i}(i)^{(L)} \right]^{-1}$ はそれぞれ

$$\left[L_{i,i}(i)^{(L)} \right]^{-1} = \left[\begin{array}{c|c} \left[L_{i-1,i-1}(i-1)^{(L)} \right]^{-1} & \mathbf{0}_{i-1} \\ \hline -g_{i-1}(i)^{(L)T} & 1 \end{array} \right] \quad (3.23)$$

$$\left[D_{i,i}(i)^{(L)} \right]^{-1} = \left[\begin{array}{c|c} \left[D_{i-1,i-1}(i-1)^{(L)} \right]^{-1} & \mathbf{0}_{i-1} \\ \hline \mathbf{0}_{i-1}^T & \left[d(i)^{(L)} \right]^{-1} \end{array} \right] \quad (3.24)$$

と表現できる. 但し, $g_{i-1}(i)^{(L)}$ は

$$g_{i-1}(i)^{(L)} = \left[L_{i-1,i-1}(i-1)^{(L)} \right]^{-T} \cdot l_{i-1}(i)^{(L)} \quad (3.25)$$

であり, $l_{i-1}(i)^{(L)}$ は式 (3.19) より

$$l_{i-1}(i)^{(L)}$$

$$\begin{aligned}
&= [D_{i-1,i-1}(i-1)^{(L)}]^{-1} \cdot [L_{i-1,i-1}(i-1)^{(L)}]^{-1} \\
&\quad \cdot A_{N,i-1}(i-1)^{(L)T} \cdot x_N(i)^{(L)}
\end{aligned} \tag{3.26}$$

である. このように, 式(3.21) ~ (3.26) を用いて $P_{i-1,i-1}(i-1)^{(L)}$ から $P_{i,i}(i)^{(L)}$ を得ることができる. すなわち, $[L_{i-1,i-1}(i-1)^{(L)}]^{-1}$, $[D_{i-1,i-1}(i-1)^{(L)}]^{-1}$ から $[L_{i,i}(i)^{(L)}]^{-1}$, $[D_{i,i}(i)^{(L)}]^{-1}$ を計算することができる.

また, $i=1$ における $L_{i,i}(i)^{(L)}$, $D_{i,i}(i)^{(L)}$ の初期値はスカラ量であり, (1×1) の行列とみなすことができるので, これら及びその逆数は式(3.8), (3.10), (3.15) より

$$D_{1,1}(1)^{(L)} = \|x_N(1)\|^2 \tag{3.27}$$

$$\begin{aligned}
[D_{1,1}(1)^{(L)}]^{-1} &= \frac{1}{D_{1,1}(1)^{(L)}} \\
&= \frac{1}{\|x_N(1)^{(L)}\|^2}
\end{aligned} \tag{3.28}$$

$$L_{1,1}(1)^{(L)} = [L_{1,1}(1)^{(L)}]^{-1} = 1 \tag{3.29}$$

とおくことができる.

3.3 逐次 UD 分解法を用いた BOP アルゴリズム

3.3.1 アルゴリズムの導出

本節では, 前節での準備に基づいて OPM-BOPA よりも少ない演算量で式(3.4)を実行する方式について述べる. 本章で提案するアルゴリズムは, $A_{N,r}^{(L)}$ の列ベクトルの中で $i(1 \leq i \leq r)$ 個の入力信号ベクトル, $x_N(1)^{(L)} \sim x_N(i)^{(L)}$ が張る部分空間への直交射影演算を行うことにより, ブロック L 中のステップ i における係数ベクトル $h_N(i+1)^{(L)}$ を求め, その手順を $i=1 \sim r$ まで繰り返すことにより式(3.4)の $h_N^{(L+1)}$ を求める手法である. ただし, i はブロック L におけるステップ変数である.

さて, 式(3.9)の仮定より Moore-Penrose 型一般逆行列が

$$[A_{N,i}(i)^{(L)T}]^+ = A_{N,i}(i)^{(L)} \cdot [A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]^{-1} \tag{3.30}$$

で表されることに注意すると，本アルゴリズムのステップ i における係数修正手順は式 (3.30) を用いて，次式

$$\begin{aligned} h_N(i+1)^{(L)} &= h_N(i)^{(L)} + [A_{N,i}(i)^{(L)T}]^+ \cdot A_{N,i}(i)^{(L)T} [W_N - h_N(i)^{(L)}] \\ &= h_N(i)^{(L)} + A_{N,i}(i)^{(L)} \cdot [A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]^{-1} \cdot e_i(i)^{(L)} \end{aligned} \quad (3.31)$$

で与えられる。但し，

$$\begin{aligned} e_i(i)^{(L)} &= A_{N,i}(i)^{(L)T} W_N - A_{N,i}(i)^{(L)T} h_N(i)^{(L)} \\ &= z_i(i)^{(L)} - A_{N,i}(i)^{(L)T} h_N(i)^{(L)} \end{aligned} \quad (3.32)$$

であり， $z_i(i)^{(L)}$ は i 個の入力信号ベクトル $x_N(1)^{(L)} \sim x_N(i)^{(L)}$ に対する所望信号ベクトルである。なお，式 (3.9) の仮定が成立しない場合の対処法については後述する。

式 (3.31) からわかるように $h_N(i+1)^{(L)}$ は，ベクトル $[W_N - h_N(i)^{(L)}]$ を $A_{N,i}(i)^{(L)}$ の列ベクトルが張る部分空間 $S[A_{N,i}(i)^{(L)}]$ 上へ直交射影することにより得られる。この修正式においてステップ i を $i = 1, 2, \dots, r$ と順次変化させることにより得られる係数ベクトル，すなわち式 (3.31) の手順を r 回繰り返した後得られる係数ベクトル $h_N(r+1)^{(L)}$ が，式 (3.4) の $h_N^{(L+1)}$ に一致することは容易に理解できよう。この一致性に関する簡単な証明を付録に記す。なお，本章では特に断らない限り，式 (3.31) 中の行列 $[A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]$ を L ブロック中のステップ i に対応する自己相関行列と呼ぶことにする。そこで，前節で行った準備に基づきこの自己相関行列の逆行列を UD 分解することにより，式 (3.31) を実行する手法について述べる。本アルゴリズムのブロック L に対する各ステップにおける係数修正手順は以下のように導出される。

$i=1$ のとき式 (3.31) は式 (3.8)，(3.32) より

$$h_N(2)^{(L)} = h_N(1)^{(L)} + \frac{x_N(1)^{(L)}}{\|x_N(1)^{(L)}\|^2} e(1)^{(L)} \quad (3.33)$$

となる。但し

$$\begin{aligned} e(1)^{(L)} &= x_N(1)^{(L)T} [W_N - h_N(1)^{(L)}] \\ &= z(1)^{(L)} - x_N(1)^{(L)T} h_N(1)^{(L)} \end{aligned} \quad (3.34)$$

であり, $z(1)^{(L)}$ は入力信号ベクトル $x_N(1)^{(L)}$ に対する所望信号である.

次に $i \geq 2$ の場合を考えよう. 式 (3.31) を再記すると

$$\begin{aligned} h_N(i+1)^{(L)} &= h_N(i)^{(L)} + A_{N,i}(i)^{(L)} \cdot [A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]^{-1} \\ &\quad \cdot A_{N,i}(i)^{(L)T} \cdot [W_N - h_N(i)^{(L)}] \end{aligned} \quad (3.35)$$

である. 前述したように式 (3.31) あるいは式 (3.35) の $h_N(i+1)^{(L)}$ は, ベクトル $[W_N - h_N(i)^{(L)}]$ を $A_{N,i}(i)^{(L)}$ の列ベクトルが張る部分空間 $S[A_{N,i}(i)^{(L)}]$ へ直交射影することにより得られる. 従って, $[W_N - h_N(i+1)^{(L)}]$ は $S[A_{N,i}(i)^{(L)}]$ に直交している. それゆえ 1 ステップ前, すなわちステップ $(i-1)$ においては, $A_{N,i-1}(i-1)^{(L)}$ の列ベクトルが張る部分空間と $[W_N - h_N(i)^{(L)}]$ が直交するので, 次式

$$A_{N,i-1}(i-1)^{(L)T} \cdot [W_N - h_N(i)^{(L)}] = \mathbf{0}_{i-1} \quad (3.36)$$

が成り立つ. そこで, 式 (3.35) 中の右辺第 2 項は, 式 (3.8), (3.10), (3.36) を用いると

$$\begin{aligned} &A_{N,i}(i)^{(L)} \cdot P_{i,i}(i)^{(L)} \cdot [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}]^T \cdot [W_N - h_N(i)^{(L)}] \\ &= A_{N,i}(i)^{(L)} \cdot P_{i,i}(i)^{(L)} \cdot [\mathbf{0}_{i-1}^T, e(i)^{(L)}]^T \end{aligned} \quad (3.37)$$

となる. ここで

$$\begin{aligned} e(i)^{(L)} &= x_N(i)^{(L)T} \cdot [W_N - h_N(i)^{(L)}] \\ &= z(i)^{(L)} - x_N(i)^{(L)T} \cdot h_N(i)^{(L)} \end{aligned} \quad (3.38)$$

である. 但し, $z(i)^{(L)}$ は入力信号ベクトル $x_N(i)^{(L)}$ に対する所望信号である. 従って, 式 (3.37) は式 (3.11), (3.22)~(3.24) を用いて

$$\begin{aligned} &A_{N,i}(i)^{(L)} \cdot P_{i,i}(i)^{(L)} \cdot [\mathbf{0}_{i-1}^T, e(i)^{(L)}]^T \\ &= [A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)}] \cdot [L_{i,i}(i)^{(L)}]^{-T} \cdot [D_{i,i}(i)^{(L)}]^{-1} \cdot [L_{i,i}(i)^{(L)}]^{-1} \\ &\quad \cdot [\mathbf{0}_{i-1}^T, e(i)^{(L)}]^T \end{aligned}$$

$$\begin{aligned}
&= \begin{bmatrix} A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)} \\ \left[\begin{array}{c|c} B_{i-1,i-1}^{(L)} & -[d(i)^{(L)}]^{-1} \cdot b_{i-1}^{(L)} \\ \hline -[d(i)^{(L)}]^{-1} \cdot b_{i-1}^{(L)T} & [d(i)^{(L)}]^{-1} \end{array} \right] \end{bmatrix} \cdot [\mathbf{0}_{i-1}^T, e(i)^{(L)}]^T \\
&= \begin{bmatrix} A_{N,i-1}(i-1)^{(L)}, x_N(i)^{(L)} \\ -e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \cdot b_{i-1}^{(L)T}, e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \end{bmatrix}^T \\
&= -A_{N,i-1}(i-1)^{(L)} \cdot [L_{i-1,i-1}(i-1)^{(L)}]^{-T} \\
&\quad \cdot \ell_{i-1}(i)^{(L)} \cdot e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} + x_N(i)^{(L)} \cdot e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \\
&= -e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \\
&\quad \cdot \left\{ A_{N,i-1}(i-1)^{(L)} \cdot [L_{i-1,i-1}(i-1)^{(L)}]^T \cdot \ell_{i-1}(i)^{(L)} - x_N(i)^{(L)} \right\} \quad (3.39)
\end{aligned}$$

のように変形できる。但し、 $b_{i-1}^{(L)}$ 、 $B_{i-1,i-1}^{(L)}$ はそれぞれ

$$\begin{aligned}
&b_{i-1}^{(L)} \\
&= [L_{i-1,i-1}(i-1)^{(L)}]^{-T} \cdot \ell_{i-1}(i)^{(L)} \quad (3.40)
\end{aligned}$$

$$\begin{aligned}
&B_{i-1,i-1}^{(L)} \\
&= P_{i-1,i-1}(i-1)^{(L)} + [d(i)^{(L)}]^{-1} \cdot b_{i-1}^{(L)} \cdot b_{i-1}^{(L)T} \quad (3.41)
\end{aligned}$$

である。故に式(3.25)、(3.39)を用いると、式(3.35)は

$$\begin{aligned}
&h_N(i+1)^{(L)} \\
&= h_N(i)^{(L)} - \left\{ A_{N,i-1}(i-1)^{(L)} \cdot [L_{i-1,i-1}(i-1)^{(L)}]^{-T} \right. \\
&\quad \left. \cdot \ell_{i-1}(i)^{(L)} - x_N(i)^{(L)} \right\} \cdot e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \\
&= h_N(i)^{(L)} - [A_{N,i-1}(i-1)^{(L)} \cdot g_{i-1}(i)^{(L)} - x_N(i)^{(L)}] \\
&\quad \cdot e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1} \quad (3.42)
\end{aligned}$$

となる。

ところで、以上の議論は式(3.9)の仮定に基づいたものである。しかしながら、入力信号の性質によってはこの仮定が成立しない場合もありうる。例えば、入力信号の有色度が強く、入力自己相関行列が悪条件となるようなときである。いま、ステッ

プ ($i-1$) までは式 (3.9) の仮定を満たすが, ステップ i ではこの仮定が成立しない場合, 即ち $\text{Rank}[A_{N,i-1}(i-1)^{(L)}] = \text{Rank}[A_{N,i}(i)^{(L)}] = i-1$ である場合を考えてみよう. このとき, 式 (3.21), (3.26) より $d(i)^{(L)} = 0$ となり, 係数修正式である式 (3.42) が計算不可能となる. これは, 本アルゴリズムの基礎となっている式 (3.31) 中の逆行列 $P_{i,i}(i)^{(L)} = [A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)}]^{-1}$ の演算が実行できなくなることに起因している.

そこで, 上述した逆行列演算が可能となるように, 式 (3.10) で定義した $[P_{i,i}(i)^{(L)}]^{-1}$ の代わりに, 次式

$$[P_{i,i}(i)^{(L)}]^{-1} = A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)} + \delta \cdot I_{i,i} \quad (3.43)$$

を用いることにする. 但し $I_{i,i}$, δ はそれぞれ $(i \times i)$ の単位行列, 小さな正数である. このような手法は, 例えば文献 [70], [71] に示されているように, 高速カルマンフィルタにおけるカルマンゲイン $k(t)$ を計算するためにも用いられている. すなわち, $k(t)$ を求めるためには, 周知のように自己相関行列 $R(t)$ の正則性が要求されている. これらの文献ではこの条件が満たされない場合も考慮して, $R(t)$ の代わりに $[R(t) + \alpha_0 \cdot I]$ を用いることにより, $k(t)$ を算出している. 但し, α_0 , I はそれぞれ小さな正数, 単位行列である. 本章の式 (3.43) もこれと同様の意味で用いられている. 従ってこのような処理を施すことにより, 式 (3.9) の仮定が成立しない場合でも, 式 (3.43) で示した $[P_{i,i}(i)^{(L)}]^{-1}$ を用いることにより, 本アルゴリズムの実行が可能となる. また, 式 (3.43) を用いることにより, 式 (3.21) の $d(i)^{(L)}$ は

$$d(i)^{(L)} = \|x_N(i)^{(L)}\|^2 + \delta - l_{i-1}(i)^{(L)T} \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot l_{i-1}(i)^{(L)} \quad (3.44)$$

のように修正される. 表 3.1 に式 (3.43), (3.44) を考慮した本アルゴリズムを示す.

本表では, 観測雑音が存在している場合の対処法として, 表 3.1 中の (ii) 及び (iii) の手順 5) においてステップゲイン α ($0 < \alpha \leq 1$) を付加している. これまでの議論からわかるように, 本アルゴリズムでは任意のブロック L の各ステップにおける係数修正が, 式 (3.8) の行列の列ベクトルが張る部分空間への直交射影演算に基づいている. それゆえ, 観測雑音が存在する場合, ステップゲイン α が収束速度と推定精

表 3.1: 本章で提案するアルゴリズム

ブロック L に対して

(i) 準備

$$h_N(1)^{(L)} = h_N^{(L)}, \text{ 但し } L=1 \text{ のときのみ,}$$

$$h_N(1)^{(1)} = \mathbf{0}_N \text{ とおく.}$$

(ii) $i=1$ のとき

$$h_N(i+1)^{(L)} = h_N(i)^{(L)} + \alpha \frac{x_N(i)^{(L)}}{\|x_N(i)^{(L)}\|^2} \cdot e(i)^{(L)}$$

$$e(i)^{(L)} = z(i)^{(L)} - x_N(i)^{(L)T} \cdot h_N(i)^{(L)}$$

$$\text{また, } D_{i,i}(i)^{(L)} = \|x_N(i)^{(L)}\|^2$$

$$[D_{i,i}(i)^{(L)}]^{-1} = \frac{1}{\|x_N(i)^{(L)}\|^2}$$

$$L_{i,i}(i)^{(L)} = [L_{i,i}(i)^{(L)}]^{-1} = 1$$

とおく.

(iii) $i \geq 2$ に対して, 以下の手順 1)~6) を繰り返す.

$$1) \quad \ell_{i-1}(i)^{(L)} = [D_{i-1,i-1}(i-1)^{(L)}]^{-1} \cdot [L_{i-1,i-1}(i-1)^{(L)}]^{-1} \\ \cdot A_{N,i-1}(i-1)^{(L)T} \cdot x_N(i)^{(L)}$$

$$2) \quad d(i)^{(L)} = \|x_N(i)^{(L)}\|^2 + \delta - \ell_{i-1}(i)^{(L)T} \\ \cdot D_{i-1,i-1}(i-1)^{(L)} \cdot \ell_{i-1}(i)^{(L)}$$

$$3) \quad g_{i-1}(i)^{(L)} = [L_{i-1,i-1}(i-1)^{(L)}]^{-T} \cdot \ell_{i-1}(i)^{(L)}$$

$$4) \quad e(i)^{(L)} = z(i)^{(L)} - x_N(i)^{(L)T} \cdot h_N(i)^{(L)}$$

$$5) \quad h_N(i+1)^{(L)} = h_N(i)^{(L)} - \alpha [A_{N,i-1}(i-1)^{(L)} \\ \cdot g_{i-1}(i)^{(L)} - x_N(i)^{(L)}] \cdot e(i)^{(L)} \cdot [d(i)^{(L)}]^{-1}$$

6) **If** $i=r$ **then**

$$h_N^{(L+1)} = h_N(i+1)^{(L)} \text{ とおき}$$

ブロック番号 L を $L \rightarrow L+1$ として, (i) へ戻る.

else ($i < r$)

$$D_{i,i}(i)^{(L)} = \left[\begin{array}{c|c} D_{i-1,i-1}(i-1)^{(L)} & \mathbf{0}_{i-1} \\ \hline \mathbf{0}_{i-1}^T & d(i)^{(L)} \end{array} \right]$$

$$[L_{i,i}(i)^{(L)}]^{-1} = \left[\begin{array}{c|c} [L_{i-1,i-1}(i-1)^{(L)}]^{-1} & \mathbf{0}_{i-1} \\ \hline -g_{i-1}(i)^{(L)T} & 1 \end{array} \right]$$

$$[D_{i,i}(i)^{(L)}]^{-1} = \left[\begin{array}{c|c} [D_{i-1,i-1}(i-1)^{(L)}]^{-1} & \mathbf{0}_{i-1} \\ \hline \mathbf{0}_{i-1}^T & [d(i)^{(L)}]^{-1} \end{array} \right]$$

とおき, 1) へ戻る.

End if

表 3.2: 1 サンプル当たりの演算量の比較

アルゴリズム	乗算回数
本アルゴリズム	$\frac{r^2}{3} + Nr + 2N + \frac{r}{2}$
OPM-BOPA	$1.5N^2 + 4.5N + 1$

度に及ぼす効果は、学習同定法の場合と同様であると考えられる。すなわち、本アルゴリズムにおいても $\alpha = 1$ のとき、収束速度が最も速くなり、 α が 0 に近づくほど収束速度は遅くなるが、推定精度は向上するものと思われる。従って、要求される収束速度・推定精度に応じてステップゲインを $0 < \alpha \leq 1$ の範囲で選択すればよい。

3.3.2 演算量の比較

3.1 での議論からわかるように、本方式は OPM-BOPA 同様、任意のブロックに対して、入力信号が入力される度にステップ i におけるフィルタ係数を求め、その手順を基本的にブロック長回繰り返すことにより、対応するブロックでの最適係数ベクトルを求める方式である。従って、本アルゴリズムと OPM-BOPA のフィルタ係数修正を行うための入力信号の運用法は同等である。これを考慮して、両者の 1 サンプル当たりに換算した平均演算量を比較し、その結果を表 3.2 に示す。さらに、ブロック長を変化させたときのそれぞれの演算量を図 3.1 に示す。ただしここでは、演算量を係数修正に必要となる乗算回数で評価しており、 $N = 20$ としている。また、本アルゴリズムを UD-BOPA と表現している。

さて、OPM-BOPA は基本的に直交射影行列を用いて修正方向ベクトルを算出しているため、1 サンプルあたり演算量はブロック長に無関係で N に依存している。一方、本アルゴリズムの演算量は N を固定した場合、ブロック長 r に従い変化する。図 3.1 からわかるように、ブロック長を $1 \leq r \leq N$ の範囲で選択する場合には、本アルゴリズムの方が OPM-BOPA より演算量の点で有利となる。

3.4 具体例

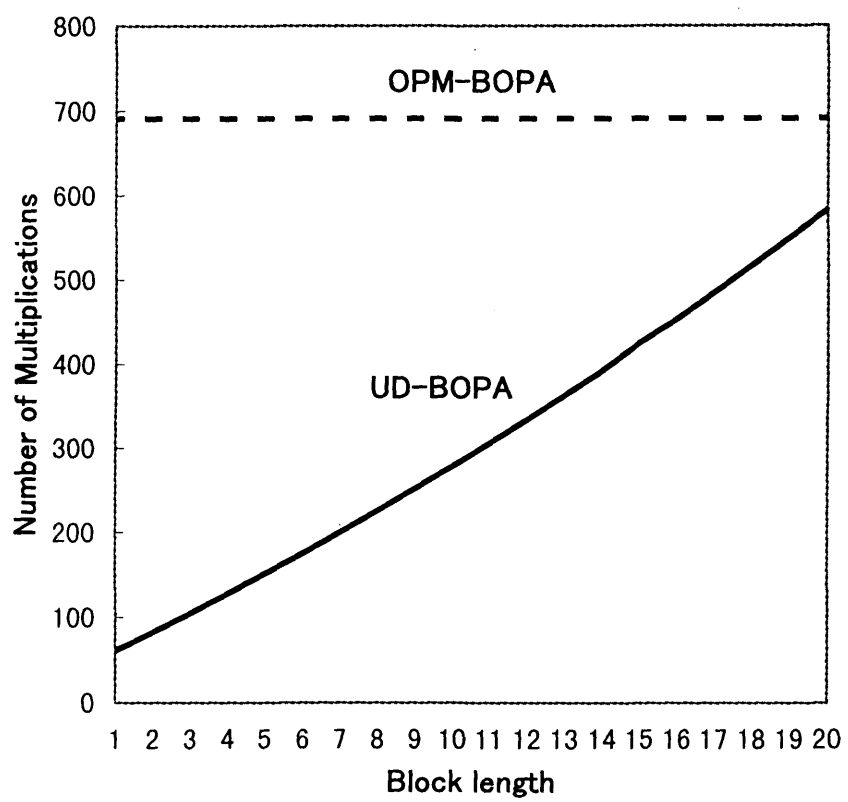


図 3.1: 演算量の比較

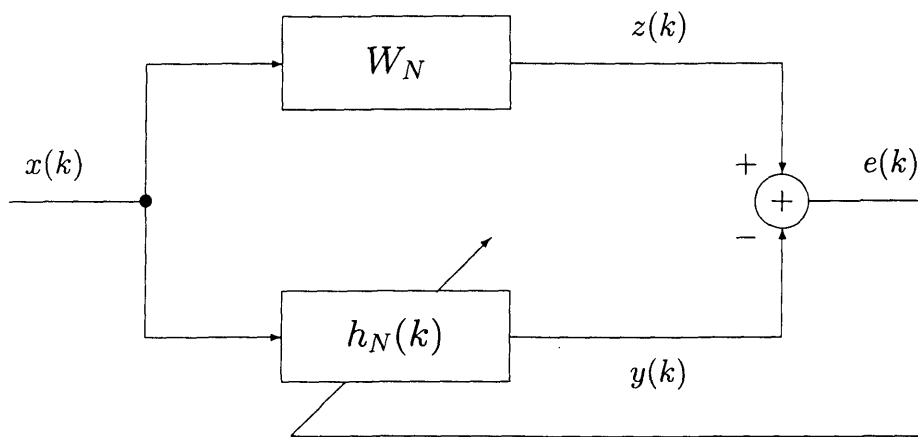


図 3.2: 適応フィルタによるパラメータ推定モデル

本節では，計算機シミュレーションにより本方式の性能評価を行うと共に従来方式との比較を行う．ここでは比較の対象として，本アルゴリズムと係数修正のための入力信号の運用法が同じ OPM-BOPA を採用した．本シミュレーションで用いたパラメータ推定モデルを図 3.2 に示し，シミュレーション条件を以下に列記する．

1. 未知系，推定系への入力信号 $x(k)$ は次のように設定した．

(a) $x(k)$ が平均値 0，分散 $1/48$ の正規乱数である場合．

(b) $x(k)$ が次式

$$x(k) = H(z) \cdot u(k) \quad (3.45)$$

で与えられる信号である場合．但し， $u(k)$ は (a) で規定した正規乱数である．また， $H(z)$ は伝達関数 $F(z)$ が次式

$$F(z) = 1 + 0.445z^{-1} + 0.202z^{-2} + 0.00907z^{-3} + 0.0408z^{-4} + 0.0183z^{-5} \quad (3.46)$$

で与えられる FIR 型フィルタを 4 個縦続接続したシステムの伝達関数である [53]．

2. 未知系, 推定系の次数はともに 19 次 ($N = 20$) とし, 未知系の伝達関数は次の周波数特性

$$A(\omega) = 1 + A_0 \cos(0.5\omega) \quad (3.47)$$

$$\gamma(\omega) = 1 + \gamma_0(\pi^2 - \omega^2)/\pi^2 \quad (3.48)$$

の 20 点 DFT により決定した. 但し $A(\omega)$, $\gamma(\omega)$ はそれぞれ振幅特性, 群遅延特性であり $A_0 = 0.5$, $\gamma_0 = 0.5$ とした.

3. アルゴリズムの性能を評価する量として, 次式で与えられる正規化パラメータ推定誤差 (Normalized Estimation Error : NEE)

$$NEE = 10 \log_{10} \frac{\|W_N - h_N^{(L)}\|^2}{\|W_N\|^2} \text{ [dB]} \quad (3.49)$$

及び平均自乗誤差 (Mean Square Error:MSE)

$$MSE = 10 \log_{10} \frac{1}{r} \|e_r^{(j)}\|^2 \text{ [dB]} \quad (3.50)$$

を採用した. 但し, $e(k)$ は未知系と推定系の出力誤差であり, $E[\cdot]$ は期待値を表す.

なお, 結果は全て 10 回の試行を行い, その平均を示している. また, 表 3.1 中の δ , ステップゲイン α はそれぞれ $\delta = 10^{-10}$, $\alpha = 1$ に設定した.

3.4.1 シミュレーション結果

図 3.3, 3.4 はそれぞれ条件 (a), (b) の下で, ブロック長を $r = 5, 10, 15, 20$ と変化させたときの本アルゴリズムの収束特性を示している. ここで, アルゴリズムの性能評価量として, 式 (3.49) を用いている. また, 条件 (a), (b) の入力信号に対する自己相関行列 Φ_{xx} の条件数はそれぞれ約 1.03, 1.8×10^3 であり, Φ_{xx} は

$$\Phi_{xx} = \frac{1}{M} \sum_{L=1}^M A_{N,r}^{(L)} \cdot A_{N,r}^{(L)T} \quad (3.51)$$

により計算している。但し、 $A_{N,r}^{(L)}$ は式(3.5)で定義されている量であり、式(3.51)の M は $M = 10^3$ とした。また、条件数は Φ_{xx} の最大固有値と最小固有値の比で与えている。

図3.3, 3.4からわかるように、入力信号に拘わらず本アルゴリズムはブロック長が長くなる程、収束速度が速くなり、ブロック長 r をフィルタ次数 N に選んだとき、すなわち $r = N (= 20)$ のとき1ブロックで最適なフィルタ係数が得られている。なお、BOPアルゴリズムでは、任意の N に対して $r = N$ としたとき、1ブロックで最適係数ベクトルが得られることは文献[48]で言及されている。また、式(3.51)で与えられる自己相関行列 Φ_{xx} が悪条件である場合、すなわち入力信号が条件(b)である場合、ブロック長を短くしたときの収束速度は白色信号入力時(条件(a)の場合)と比較して多少の劣化が認められるものの、ブロック長を長くするほど収束速度は、白色入力信号時のそれに近づいていることが確認できる。従って入力信号が有色である場合でも、本方式は適当にブロック長を選択することにより、良好な収束特性を得ることができる。

図3.5は条件(b)の下で、ブロック長を $r = 4$ 及び $r = 8$ としたときの本アルゴリズムとOPM-BOPAの収束特性を式(3.49)の観点から比較したものである。また図3.6は条件(b)で、ブロック長を $r = 10$ としたときの本アルゴリズムとOPM-BOPAの収束特性を式(3.50)の観点から比較したものである。ここで、UD-BOPAは本アルゴリズムを表す。これらの結果と3.3.2の議論から容易に推察できるように、本アルゴリズムはOPM-BOPAよりも少ない演算量で、これとほぼ同等の収束特性を得ることができる。

3.5 むすび

本章では、任意のブロック中の計算ステップ i に対応する自己相関行列のUD分解とブロック逆行列の補題を用いることによって、Moore-Penrose型一般逆行列を含むBOPアルゴリズムの一実現手法を提案し、その基本的性質について議論した。次いで、計算機シミュレーションにより本アルゴリズムの性能を評価した。以下、本

研究で得られた結果を示す。

- 有色信号入力時における本アルゴリズムの収束速度は、白色信号入力時と比較してブロック長が短い場合には多少の劣化が認められるものの、ブロック長を長くするほど白色信号入力時のそれに近い特性が得られる。また、係数修正を行うための入力信号の運用法が同じ OPM-BOPA と比べて、本アルゴリズムはこれよりも少ない演算量で BOP アルゴリズムを実行することができる。

今回は、表 3.1 中のパラメータ δ を一定にしたときの本アルゴリズムの基本的性質について議論した。この量は、入力として低周波部にエネルギーが集中している信号が用いられるような場合など、入力信号の性質によっては表 3.1 中の手順 (iii), 5) に含まれる、 $[d(i)^{(L)}]^{-1}$ の計算結果に影響を及ぼす。換言すれば本アルゴリズムの数値的安定性に影響を与える。従って、 δ の設定は重要な問題であるが、この設定法は入力信号の性質やアルゴリズムを実行する際に与えられる計算精度に依存するため、現段階で一般的な δ の設定法を議論することは困難である。

それゆえ今後は、入力信号に限定を加え、適当な計算精度を与えた上で δ が収束特性に及ぼす影響を、未知系に観測雑音が重畳している場合も含めて検討する予定である。さらに、任意の 1 ブロックにおける係数修正のための入力信号の運用法に着目して、本アルゴリズムを高速に処理できるアーキテクチャの開発についても併せて検討課題としたい。

また、未知系の出力信号に観測雑音が重畳している場合について、本章では提案法の具体的議論行わなかったが、次章において直交射影アルゴリズムの一般論としての観測雑音による推定精度への影響を軽減する方式についての議論を行う。

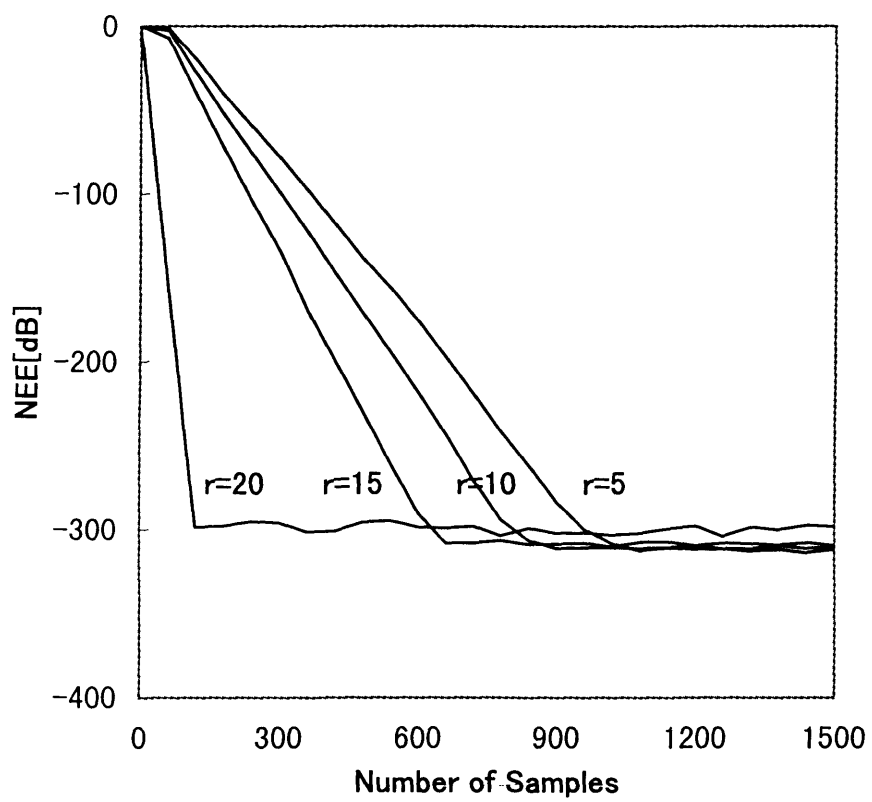


図 3.3: 本方式の収束特性 (条件 (a) の場合)

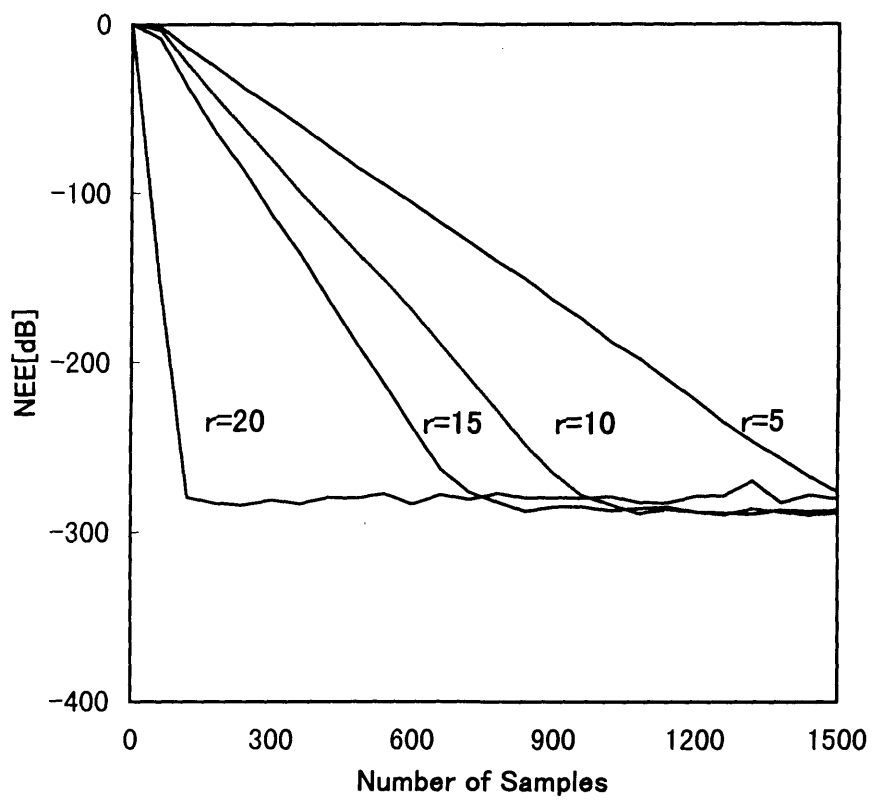


図 3.4: 本方式の収束特性 (条件 (b) の場合)

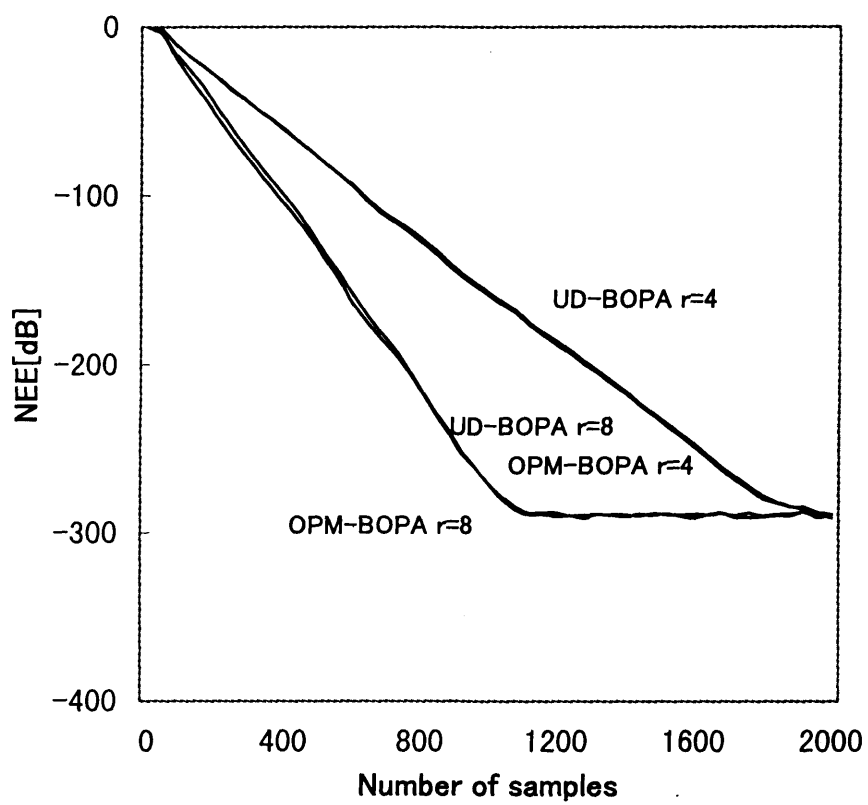


図 3.5: 本方式と OPM-BOPA の収束特性比較 (条件 (b) の場合)

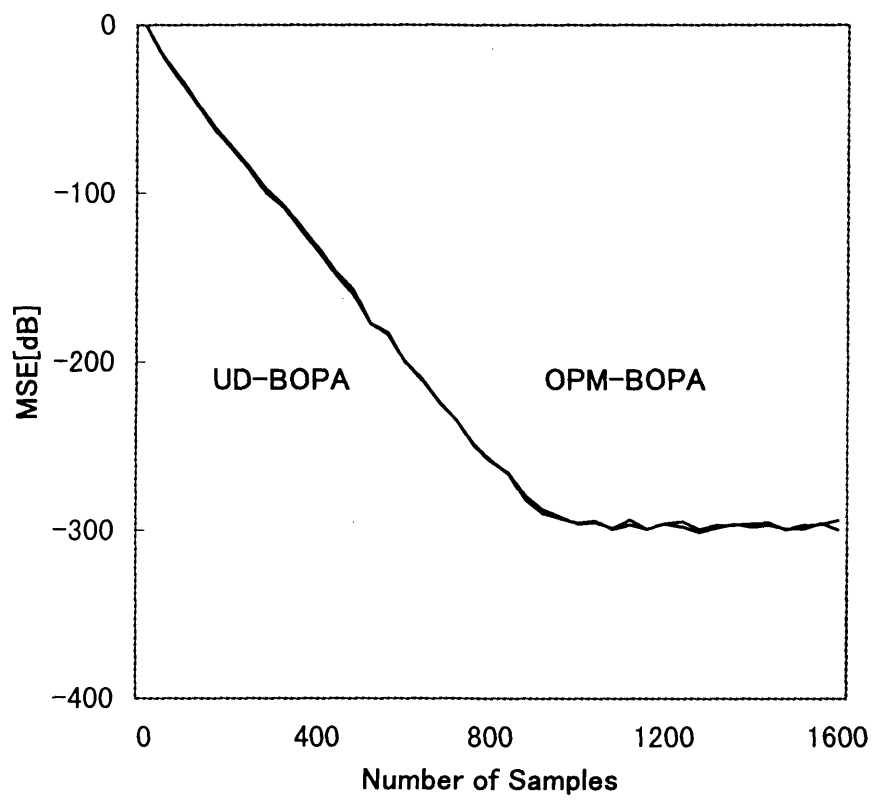


図 3.6: MSE 特性比較 (条件 (b) の場合)

付録 式 (3.31) と BOP アルゴリズムの等価性に関する 略証

本アルゴリズムの基礎となる係数修正式は、式 (3.31) で与えられており、これを再記すると

$$\begin{aligned} h_N(i+1)^{(L)} &= h_N(i)^{(L)} + A_{N,i}(i)^{(L)} \left[A_{N,i}(i)^{(L)T} \cdot A_{N,i}(i)^{(L)} \right]^{-1} \cdot e_i(i)^{(L)} \end{aligned} \quad (3.52)$$

である。ここで式 (3.52) の両辺に左から $A_{N,i}(i)^{(L)T}$ を乗じ、式 (3.32) を用いると

$$\begin{aligned} A_{N,i}(i)^{(L)T} \cdot h_N(i+1)^{(L)} &= A_{N,i}(i)^{(L)T} \cdot h_N(i)^{(L)} + e_i(i)^{(L)} \\ &= A_{N,i}(i)^{(L)T} \cdot W_N \\ &= z_i(i)^{(L)} \end{aligned} \quad (3.53)$$

となる。上記の式で $i = r$ とおけば

$$\begin{aligned} A_{N,r}(r)^{(L)T} \cdot h_N(r+1)^{(L)} &= A_{N,r}(r)^{(L)T} \cdot W_N \\ &= z_r^{(L)} \end{aligned} \quad (3.54)$$

を得る。但し、 $z_r^{(L)}$ は式 (3.7) で定義されている量である。式 (3.54) から

$$A_{N,r}(r)^{(L)T} [W_N - h_N(r+1)^{(L)}] = 0 \quad (3.55)$$

であるので、式 (3.8) より得られる関係

$$A_{N,r}(r)^{(L)} = A_{N,r}^{(L)} \quad (3.56)$$

を用いれば、ベクトル $[W_N - h_N(r+1)^{(L)}]$ は、 $A_{N,r}^{(L)}$ の列ベクトルが張る空間 $S[A_{N,r}^{(L)}]$ の直交補空間 $S[A_{N,r}^{(L)}]^\perp$ に属することがわかる。つまり

$$[W_N - h_N(r+1)^{(L)}] \in S[A_{N,r}^{(L)}]^\perp \quad (3.57)$$

である。

一方, 式 (3.52) より

$$\begin{aligned} h_N(r+1)^{(L)} &= h_N(1)^{(L)} + \sum_{i=1}^r A_{N,i}(i)^{(L)} \left[A_{N,i}(i)^{(L)T} A_{N,i}(i)^{(L)} \right]^{-1} \cdot e_i(i)^{(L)} \end{aligned} \quad (3.58)$$

であるから

$$[h_N(r+1)^{(L)} - h_N(1)^{(L)}] \in S[A_{N,r}^{(L)}] \quad (3.59)$$

が言える。また, 式 (3.58) より

$$\begin{aligned} W_N - h_N(1)^{(L)} &= W_N - h_N(r+1)^{(L)} \\ &\quad + \sum_{i=1}^r A_{N,i}(i)^{(L)} \left[A_{N,i}(i)^{(L)T} A_{N,i}(i)^{(L)} \right]^{-1} \cdot e_i(i)^{(L)} \end{aligned} \quad (3.60)$$

を得る。ここで, 式 (3.60) の両辺に関してノルムの二乗を計算すると, 式 (3.57), (3.59) より

$$\begin{aligned} \|W_N - h_N(1)^{(L)}\|^2 &= \|W_N - h_N(r+1)^{(L)}\|^2 + \|h_N(r+1)^{(L)} - h_N(1)^{(L)}\|^2 \end{aligned} \quad (3.61)$$

となる。これは, $h_N(r+1)^{(L)}$ が W_N を空間 $S[A_{N,r}^{(L)}]$ へ直交射影することにより得られることを示している。一方, 式 (3.4) の BOP アルゴリズムで与えられる $h_N^{(L+1)}$ は, W_N を $S[A_{N,r}^{(L)}]$ へ直交射影することによって得られることが知られている [45][47][48]。従って, 式 (3.31) を r 回繰り返して得られる $h_N(r+1)^{(L)}$ は, $h_N^{(L+1)} = h_N(r+1)^{(L)}$ を満足するので, 式 (3.31) と BOP アルゴリズムは等価であると言える。

(略証終)

第 4 章

観測雑音に対しロバスト性を実現する 方向ベクトル分割平均化法によるブロッ ク適応アルゴリズム

前章までの議論により，ブロック直交射影アルゴリズムは収束速度と演算量のバランスが比較的とれた方式であることが明らかとなった．しかしながら直交射影アルゴリズムは所望信号に観測雑音が重畳している場合，収束特性が雑音に影響されやすいことが指摘されている [77]．推定精度の劣化を防ぐため，一般には適応アルゴリズムにおけるステップゲインを小さくとる，あるいは係数ベクトルの平均化を行うなどといった操作により推定精度の劣化を防ぐ処置がとられる．しかしながら，これら適用する方法によっては，収束速度の劣化を招く，観測雑音の分散が必要である，白色信号入力時の解析に基づいているなどの問題点があげられる．本章では Moore-Penrose 型一般逆行列で表される直交射影アルゴリズムの基本型を対象として，修正方向ベクトルに含まれる観測信号から構成される成分の時間平均を行い，良好な推定精度を得られる方式を提案している．本方式は信号と観測雑音が無相関であり，雑音の平均値が零であるという条件のみで適用が可能である．また，計算機シミュレーションにより収束特性の検討を行っている．

4.1 まえがき

エコーキャンセラやノイズキャンセラなどに代表される適応システムには、アルゴリズムを実現するための演算量が少ないLMSアルゴリズム [17] や学習同定法 [19] が広く利用されている。しかしながら、相関を有する有色信号である場合、収束速度が著しく劣化するという欠点がある。そこで収束速度の劣化を防ぐ方式として、拡張された学習同定法 [43] やアフィン射影算法 [44] が提案されている。これらは直交射影原理に基づいた算法であり、1回のフィルタ修正に過去の複数の入力信号ベクトルを用いることにより、収束速度の高速化を実現している。しかしながら、1サンプル当たりの演算量が非常に多くなることから、リアルタイム処理での実現は困難であるという問題点を有している。これを解決する一手法として、ブロック直交射影アルゴリズム (以下、BOP アルゴリズム)[45] が提案されている。これは前述のアルゴリズム [43], [44] と同様、直交射影原理に基づいた算法であるが、ブロック処理の概念を導入し、フィルタ係数の修正をブロックごとに行うことにより、収束速度の高速性と1サンプル当たりの演算量のある程度両立させる優れた方式ということができる。

しかしながらBOPアルゴリズムは雑音の影響を受けやすいということが指摘されており、雑音が存在する状況での適用については何らかの改善が必要である。この問題に対する解決法として、直交射影アルゴリズムの範疇である学習同定法を対象として二つの適応フィルタを用いることにより、観測雑音の分散の推定値に基づいてステップゲインを調整する方法 [74], [75], 適応フィルタ係数の時間平均操作を行うことにより、雑音の影響を軽減する手法 (以下、RBAF)[76] などが知られている。また、誤差解析の結果を用いてBOPアルゴリズムの一実現法であるCGM-BOPアルゴリズム [47] の一ブロック内のアルゴリズム繰り返し回数を制御する方式 (以下、BCGM-OR) も提案されている [77][78]。文献 [74], [75] の手法は入力信号が白色であるという仮定の下で行った解析を基礎としており、有色信号入力の場合については、数値例による有効性の確認にとどまっている。また、BCGM-ORも白色信号入力時における前提の下での議論であり、観測雑音の分散が既知である必要がある。さら

に、この手法は前述したように BOP アルゴリズムの一実現法である CGM-BOP アルゴリズムの収束特性改善に関する方式であり、Moore-Penrose 型一般逆行列が陽に現れる BOP アルゴリズムそのものに対する改善策については言及されていない。

本章では、Moore-Penrose 型一般逆行列が陽に現れる直交射影アルゴリズムにおける観測雑音の影響を軽減するという観点に立ち、直交射影アルゴリズム全般に適用可能な方式について議論する。本方式は Moore-Penrose 型一般逆行列で表される直交射影アルゴリズムに適用可能であるが、本章では BOP アルゴリズムを例にとつて議論を進める。本方式は、修正方向ベクトルの時間平均に基礎をおく方式である。すなわち、時間平均を行う際に修正方向ベクトルを、観測信号を含む項により生成される方向ベクトルの成分と推定系出力信号による方向ベクトル成分に分割し、それぞれを独立に平均化することにより BOP アルゴリズムの収束の高速性を失うことなく、観測雑音の影響を軽減する方式である。さらに本方式は入力信号と観測雑音が無相関であり、雑音の平均値が零であれば入力信号の白色性・有色性を考慮する必要がないという特徴をも有している。

以下 4.2 では以後の議論に必要な準備を行い、4.3 では本方式の導出を行っている。また、4.4 ではコンピュータ・シミュレーションにより本方式の有効性を示している。

4.2 準備

本節では以後の議論で必要となる諸量の定義を行う。 k , N , r 及び j をそれぞれ時刻、フィルタのインパルス応答長、ブロック長及びブロック番号とする。

時刻 k における入力状態ベクトルを

$$x_N(k) \triangleq [x(k), x(k-1), \dots, x(k-N+1)]^T \quad (4.1)$$

と定義し、ブロック j における入力状態行列を

$$A_{r,N}^{(j)} \triangleq [x_N(jr), x_N(jr-1), \dots, x_N(jr-r+1)]^T \quad (4.2)$$

と定義する。ここで、ブロック j における推定系の係数ベクトルを $h_N^{(j)}$ とすれば推定

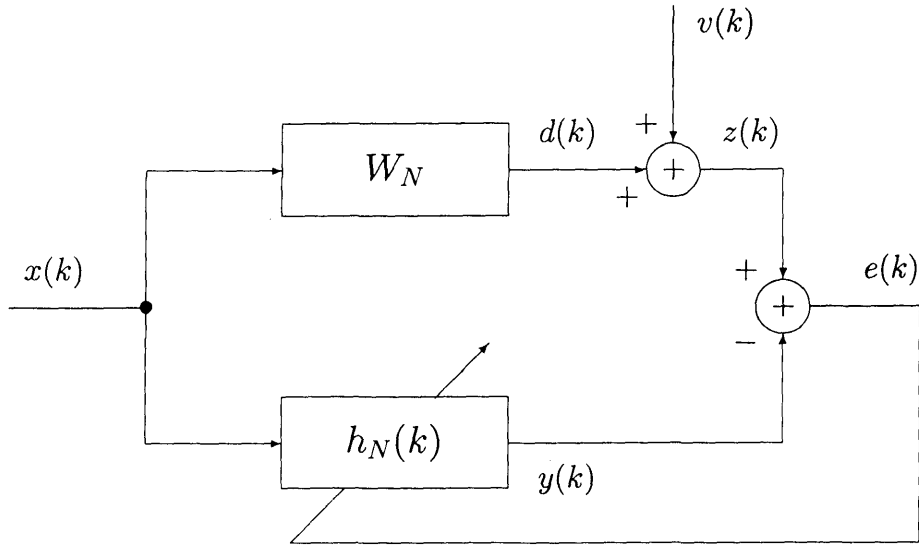


図 4.1: パラメータ推定のモデル

系出力信号ベクトルは

$$y_r^{(j)} = A_{r,N}^{(j)} h_N^{(j)} \quad (4.3)$$

で与えられる。一方、未知系出力信号ベクトルを

$$d_r^{(j)} = A_{r,N}^{(j)} W_N \quad (4.4)$$

とすれば出力誤差ベクトルは

$$e_r^{(j)} = z_r^{(j)} - y_r^{(j)} \quad (4.5)$$

となる。ここで $z_r^{(j)}$ は観測信号ベクトルであり

$$z_r^{(j)} = d_r^{(j)} + v_r^{(j)} \quad (4.6)$$

である。但し、 $v_r^{(j)}$ はブロック j における観測雑音ベクトルであり、入力信号 $x(k)$ と無相関であると仮定する。また、 W_N を推定すべき未知系のパラメータベクトルとする。図 4.1 にパラメータ推定のモデルを示す。上記の定義を用いれば BOP アルゴリズムのフィルタ修正式は一般に

$$h_N^{(j+1)} = h_N^{(j)} + \alpha \cdot A_{r,N}^{(j)+} e_r^{(j)} \quad (4.7)$$

と表される．ここで $A_{r,N}^{(j)+}$ は入力状態行列 $A_{r,N}^{(j)}$ の Moore-Penrose 型一般逆行列であり， α はステップゲインを表している．従って，ブロック j における BOP アルゴリズムの修正方向ベクトル $c_N^{(j)}$ は

$$c_N^{(j)} = A_{r,N}^{(j)+} e_r^{(j)} \quad (4.8)$$

と表される．なお，以後の議論においてすべての信号は定常かつエルゴード性を有するものと仮定する．

4.3 方向ベクトルの分割平均化

図 4.1 で示されるように，観測信号 $z(k)$ には観測雑音 $v(k)$ が重畳されており推定精度に影響を与えている．従って，本節では修正方向ベクトル $c_N^{(j)}$ を観測雑音 $v(k)$ を含む項と観測雑音を含まないフィルタ出力による項に分割し，それぞれの時間平均をとることにより，BOP アルゴリズムによる収束の高速性を維持しつつ，雑音の影響を軽減する方式を導出する．

まず，式 (4.5)，(4.8) により修正方向ベクトルは

$$\begin{aligned} c_N^{(j)} &= A_{r,N}^{(j)+} [z_r^{(j)} - y_r^{(j)}] \\ &= A_{r,N}^{(j)+} z_r^{(j)} - A_{r,N}^{(j)+} y_r^{(j)} \end{aligned} \quad (4.9)$$

と表される．また，式 (4.9) の右辺第一項を $c_N^{z(j)}$ ，右辺第 2 項を $c_N^{y(j)}$ とおき

$$c_N^{(j)} = c_N^{z(j)} - c_N^{y(j)} \quad (4.10)$$

とする．右辺第一項 $c_N^{z(j)}$ は式 (4.6) より観測雑音を含む項である．

ここで，式 (4.10) の右辺第 1 項の期待値をとると式 (4.6) より

$$\begin{aligned} E[c_N^{z(j)}] &= E[A_{r,N}^{(j)+} z_r^{(j)}] \\ &= E[A_{r,N}^{(j)+} (d_r^{(j)} + v_r^{(j)})] \\ &= E[A_{r,N}^{(j)+} d_r^{(j)}] + E[A_{r,N}^{(j)+} v_r^{(j)}] \end{aligned} \quad (4.11)$$

となる。式 (4.11) において入力信号と雑音が無相関であることと雑音の平均値を零と仮定すれば

$$E[c_N^{z(j)}] = E[A_{r,N}^{(j)+} d_r^{(j)}] \quad (4.12)$$

となり $E[A_{r,N}^{(j)+} z_r^{(j)}]$ は $E[A_{r,N}^{(j)+} d_r^{(j)}]$ に一致する。すなわち集合平均により理論上観測雑音の影響がなくなることを意味する。しかしながら実際のシステムでの運用を考えれば、実時間処理を考慮する必要がある、集合平均をとることは困難である。従って、本章では先に仮定した信号のエルゴード性を利用し、あるブロックまでに得られた観測信号に対して時間平均を行うことにより、推定精度に対する雑音の影響の軽減を行う方式について議論する。

はじめに、観測雑音を含む方向ベクトル成分 $c_N^{z(j)}$ の平均化の計算を行う。ブロック j における平均化された方向ベクトル成分 $\hat{c}_N^{z(j)}$ の算出は過去のブロックで算出される方向ベクトル成分の平均化を行うことにより、式 (4.9), (4.10) を用いて

$$\hat{c}_N^{z(j)} = \frac{1}{j} \sum_{i=1}^j A_{r,N}^{(i)+} z_r^{(i)} \quad (4.13)$$

となる。次に、式 (4.10) 右辺第二項 $c_N^{y(j)}$ の平均操作について考える。この量は観測雑音に影響されないので、入力信号に関する平均操作のみを考慮する。従って、このような意味で平均化された量 $\hat{c}_N^{y(j)}$ を j ブロックにおける推定系の係数ベクトル $h_N^{(j)}$ を用いて

$$\hat{c}_N^{y(j)} \triangleq \frac{1}{j} \sum_{i=1}^j A_{r,N}^{(i)+} A_{r,N}^{(i)} h_N^{(j)} \quad (4.14)$$

と定義する。一方、ブロック j における入力データ行列 $A_{r,N}^{(j)}$ の行ベクトルが張る空間への直交射影行列 $P_{N,N}^{(j)}$ は

$$P_{N,N}^{(j)} = A_{r,N}^{(j)+} A_{r,N}^{(j)} \quad (4.15)$$

で算出される。よって、方向ベクトル成分 $c_N^{y(j)}$ は

$$c_N^{y(j)} = P_{N,N}^{(j)} h_N^{(j)} \quad (4.16)$$

と表すことができる。従って、平均化された直交射影行列 $\hat{P}_{N,N}^{(j)}$ を

$$\hat{P}_{N,N}^{(j)} = \frac{1}{j} \sum_{i=1}^j P_{N,N}^{(i)} \quad (4.17)$$

とし、式(4.16)による計算を考慮すれば、 $\hat{c}_N^{y(j)}$ はブロック j におけるフィルタ係数 $h_N^{(j)}$ を用いることにより

$$\hat{c}_N^{y(j)} = \hat{P}_{N,N}^{(j)} h_N^{(j)} \quad (4.18)$$

と導出される。

従って、本方式における BOP アルゴリズムの修正方向ベクトル $\hat{c}_N^{(j)}$ は

$$\begin{aligned} \hat{c}_N^{(j)} &= \hat{c}_N^{z(j)} - \hat{c}_N^{y(j)} \\ &= \frac{1}{j} \left[\sum_{i=1}^j A_{r,N}^{(i)+} z_r^{(i)} - \sum_{i=1}^j P_{N,N}^{(i)} h_N^{(j)} \right] \\ &= \frac{1}{j} \sum_{i=1}^j [c_N^{z(i)} - c_N^{y(i)}] \end{aligned} \quad (4.19)$$

となる。ここで式(4.9), (4.15), (4.16)を用いた。

式(4.19)からわかるように、 j 番目のブロックにおける係数ベクトル $\hat{c}_N^{(j)}$ は1ブロックから j ブロックまでのそれを単純平均して得られる量である。しかしながら、上記の方法で $\hat{c}_N^{(j)}$ を算出した場合、未知パラメータが変動したときのパラメータ追従性に欠けることが予想される。そこで、そのような場合にもある程度のパラメータ追従性を確保するために、式(4.13), (4.14)で行っている演算に若干の工夫を加えた平均操作法について考察する。これは指数重みつき RLS アルゴリズム [79] で知られているように、過去のデータの影響を漸次軽減するという考えに基づくものである。このように、重みつき RLS アルゴリズムでは、入力信号に対して重みづけを行っており、本方式では修正方向ベクトルに対して同様の操作を行っている。

まず、観測雑音を含む方向ベクトル成分 $c_N^{z(j)}$ の $j=1$ から j にわたる平均値 $\hat{c}_N^{z(j)}$ を $0 < \lambda \leq 1$ なる λ を用いて

$$\hat{c}_N^{z(j)} \triangleq \frac{1}{j} \sum_{i=1}^j \lambda^{j-i} c_N^{z(i)} \quad (4.20)$$

と定義する．式 (4.20) より次式

$$\hat{c}_N^{z(j)} = \frac{1}{j} \left[\lambda(j-1) \hat{c}_N^{z(j-1)} + c_N^{z(j)} \right] \quad (4.21)$$

が成り立つ．同様に，他の方向ベクトル成分 $c_N^{y(i)}$ の $i = 1 \sim j$ までの平均値 $\hat{c}_N^{y(j)}$ を

$$\hat{P}_{N,N}^{z(j)} \triangleq \frac{1}{j} \sum_{i=1}^j \lambda^{j-i} P_{N,N}^{(i)} \quad (4.22)$$

と定義する．式 (4.22) より

$$\hat{c}_N^{y(j)} = \frac{1}{j} \left[\lambda(j-1) \hat{P}_{N,N}^{z(j-1)} + P_{N,N}^{(j)} \right] h_N^{(j)} \quad (4.23)$$

$$\hat{P}_{N,N}^{(j)} = \frac{1}{j} \left[\lambda(j-1) \hat{P}_{N,N}^{(j-1)} + P_{N,N}^{(j)} \right] \quad (4.24)$$

を得る．従って，本方式におけるフィルタ係数更新式は式 (4.21)，(4.23)，(4.24) を用いて

$$h_N^{(j+1)} = h_N^{(j)} + \hat{\alpha} \cdot \hat{c}_N^{(j)} \quad (4.25)$$

$$\hat{c}_N^{(j)} = \hat{c}_N^{z(j)} - \hat{c}_N^{y(j)} \quad (4.26)$$

で与えられる．但し， $\hat{\alpha}$ はステップゲインである．ここで， $\hat{c}_N^{z(j)}$ ， $\hat{P}_{N,N}^{(j)}$ の初期値 $\hat{c}_N^{z(0)}$ ， $\hat{P}_{N,N}^{(0)}$ は任意であるが，ここでは $\hat{c}_N^{z(0)} = \mathbf{0}_N$ ， $\hat{P}_{N,N}^{(0)} = \mathbf{0}_{N,N}$ とおいた．また，本方式における演算量はブロックあたり $O(N^2r)$ で表される．

パラメータの追従性を確保する方法としては，上記の方法以外にも，適当なデータ区間における平均操作も考えられる．この方法については付録に計算法を略記する．

ところで，Moore-Penrose 型一般逆行列を直接計算することなく $O(N^2r)$ よりも少ない演算量で BOP アルゴリズムを実現する方式が，文献 [47] 以外にもいくつか知られている [48]，[80]．これらの方式にはいずれも，1 ブロック内で有限回繰り返される処理手順中に式 (4.7) に対応する係数修正方向ベクトルが含まれている．従って，本方式における忘却係数を $\lambda = 1$ とすれば，ブロック内で単純時間平均操作を行うことになり，これらの方式にも本章で議論した手法が適用できると考えられる．一方，演算量については個々の方式によって異なる．既に述べたように，本章では

文献 [47], [48], [80] で提案されている手法の基礎となっている Moore-Penrose 型一般逆行列が陽に現れる BOP アルゴリズムの観測雑音による影響を軽減することに主眼をおいている。従って、ここではそれらの方式における演算量との比較を含めて、本方式の演算量に関する議論を行わないことにする。

4.4 具体例

本方式の有効性を示すためにコンピュータ・シミュレーションを行った。ここでは $A_{r,N}^{(j)+}$ を文献 [81] に従って、 $A_{r,N}^{(j)+} \simeq A_{r,N}^{(j)T} [A_{r,N}^{(j)} A_{r,N}^{(j)T} + \delta I_{r,r}]^{-1}$ として計算した。但し、 $I_{r,r}$ は $(r \times r)$ の単位行列であり、 δ は $\delta = 10^{-10}$ に設定した。また、式 (4.15), (4.19) の $P_{N,N}^{(j)}$ についても同様に計算した。

また、本方式に対する比較検討例として、本方式がブロック処理であることを考慮し、式 (4.7) においてステップゲインを変化させた方式、式 (4.7) に文献 [76] の手法を適用した方式 (RBAF) を採用した。但し RBAF は以下の式より算出している。

$$h_N^{*(j)} = \frac{(j-1)h_N^{*(j-1)} + h_N^{(j)}}{j} \quad (4.27)$$

ここで、 $h_N^{*(j)}$ は平均化されたフィルタ係数であり、ステップゲインは $\alpha = 1$ としている。

参考のために BCGM-OR アルゴリズム、UD-RLS アルゴリズム [72] の収束特性も図中に示す。BCGM-OR アルゴリズムでは高速な収束速度 (特に、推定初期段階) を保持するためにブロック長を $r = N$ としている。また、UD-RLS アルゴリズムでは忘却係数 λ を $\lambda = 0.99, 0.97$ としている。

パラメータの推定精度を表す評価量として

$$NEE = 10 \log_{10} \frac{\|W_N - h_N^{(j)}\|^2}{\|W_N\|^2} \text{ [dB]} \quad (4.28)$$

および

$$MSE = 10 \log_{10} \frac{1}{r} \|e_r^{(j)}\|^2 \text{ [dB]} \quad (4.29)$$

を用いた。信号対雑音比 (SNR) は

$$\text{SNR} = 10 \log_{10} \frac{E[d(k)^2]}{E[v(k)^2]} \quad (4.30)$$

により算出した。

また、入力信号 $x(k)$ として次の信号を用いた。

<1> 白色信号：平均値零，分散 $1/48$ の正規乱数

<2> 有色信号：白色信号をフィルタ $H(z)$

$$H(z) = \frac{1}{1 - 0.99z^{-1}} \quad (4.31)$$

に通した信号。

シミュレーションで使用した入力信号と観測雑音の関係は次の2通りである。

1)：入力信号：白色信号，観測雑音：白色信号

2)：入力信号：有色信号，観測雑音：白色信号

未知系，推定系のインパルス応答はともに $N = 20$ とし，式 (4.30) で定義している SNR は $30[\text{dB}]$ とした。また，以下のシミュレーション例においては，それぞれのアルゴリズムの収束特性の立ち下がり部分なるべく一致するように，ブロック長 r を RBAF，ステップゲインが $\alpha = 1$ の BOP アルゴリズム，本方式のいずれも $r = N/4 = 5$ とした。また，BCGM-OR では前述した理由で $r = N = 20$ と設定した。本方式におけるステップゲインは $\hat{\alpha} = 1$ としている。結果はすべて10回試行の平均を示している。

4.4.1 シミュレーション結果

図 4.2 は信号の条件が 2) の場合の下で λ を変化させたときの収束特性である。図のようにインパルス応答が変動した場合， λ をある程度小さく取ったほうが追従性が向上している。以後のシミュレーション結果においては，図 4.2 における追従性と推定精度の関係を考慮して，本アルゴリズムにおける λ を $\lambda = 0.97$ と固定する。

図 4.3，4.4 はそれぞれ信号 1)，2) の場合の下で本方式，RBAF，BCGM-OR 及び BOP アルゴリズムの特性を示している。これらの図より白色信号及び有色信号を

入力した場合においても比較対象より良好な収束特性が得られていることがわかる。本章では収束速度の劣化を防ぐために提案法のステップゲインを1に固定しているが、ステップゲインを小さくとることにより更に精度が向上すると考えられ、ステップゲインを自動的に変化させるようなインテリジェントアダプティブフィルタとの組み合わせることによって、更に推定精度を向上させることが期待されよう。

図 4.5, 4.6は図 4.3, 4.4と同条件での収束特性を示している。但し、評価量として(4.29)を用いている。また、MSEの初期値が0dBとなるように初期化を行っている。これらの図からわかるように、MSEを評価量として用いた場合においても本方式は比較対象方式よりも良好な特性が得られている。従って、パラメータ推定及び出力誤差を最小とするような系においても有効であると思われる。

図 4.7は信号の条件 2) の下で、UD-RLS アルゴリズムと本方式の忘却係数 λ の変化による収束特性を示している。図からわかるように、 $\lambda = 0.99$ 及び $\lambda = 0.97$ において本方式は初期の収束速度が若干遅いものの、推定精度については本方式の方が優れている。

4.5 むすび

本章では、Moore-Penrose 型一般逆行列で表される BOP アルゴリズムの立場から、修正方向ベクトルを観測信号からなる方向ベクトルと、フィルタの出力信号からなる方向ベクトルに分割し、それぞれを平均化する方式を提案した。シミュレーションで明らかのように、本方式は従来の BOP アルゴリズムと同等の収束速度を保持しているにもかかわらず、推定精度は初期段階でも約 10dB 改善されている。従って、本方式は BOP アルゴリズムの高速な収束速度を犠牲にすることなく良好な推定精度を実現する方式といえる。また、ステップゲインを自動的に制御する VS アルゴリズム [82]~[85] のようなインテリジェントアダプティブフィルタと組み合わせることにより、更に推定精度の向上が見込まれる。

また、本章ではその性質上提案法を行列演算で示していることから、本方式を効率よく運用するための方式、及びその演算量の検討、更に音声信号など非定常な信

号が入力された場合の検討などを併せて行う予定である。

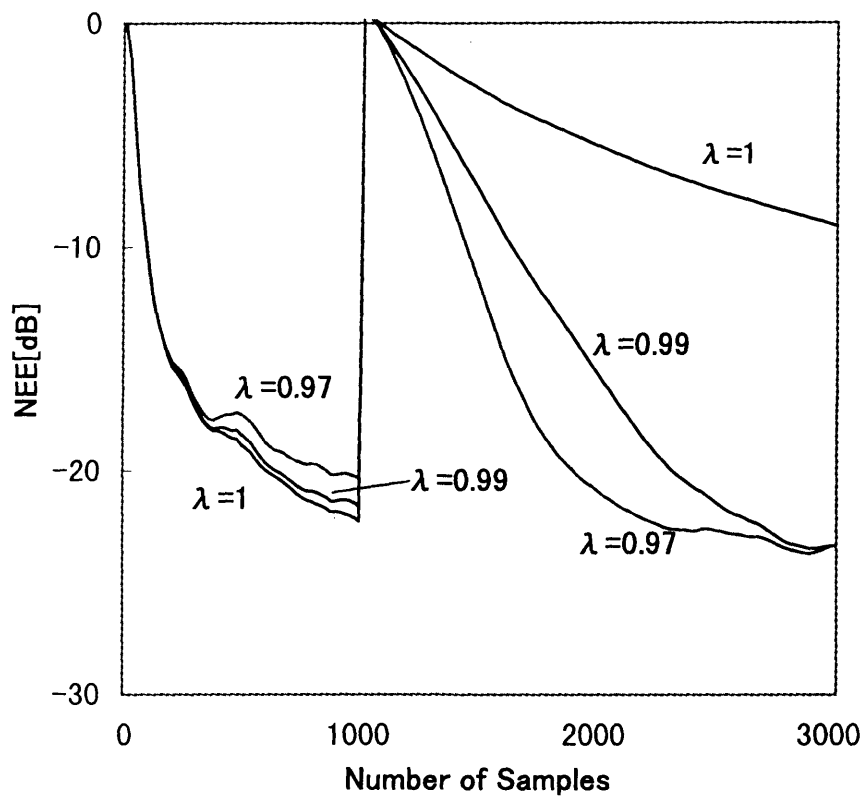


図 4.2: 収束特性の比較 (2) の場合)

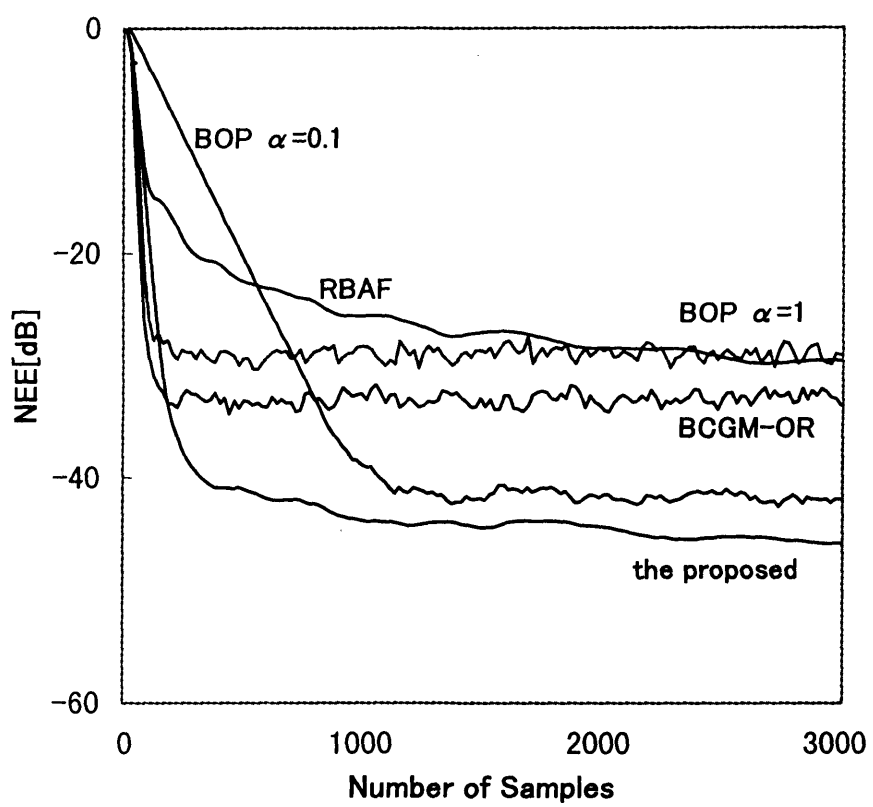


図 4.3: 収束特性の比較 (1) の場合)

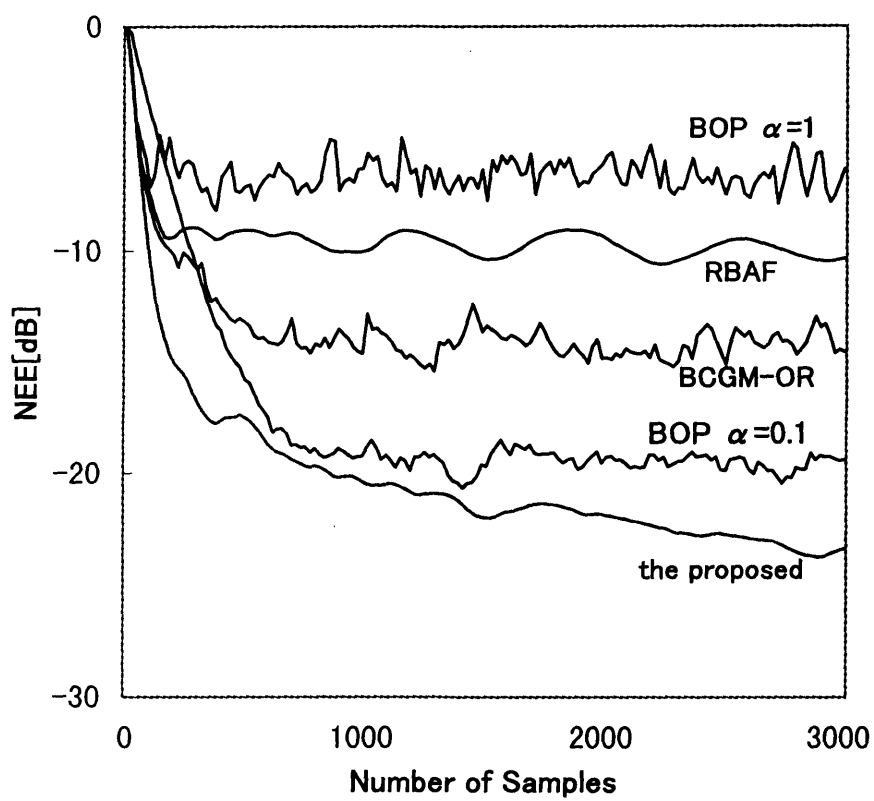


図 4.4: 収束特性の比較 (2) の場合)

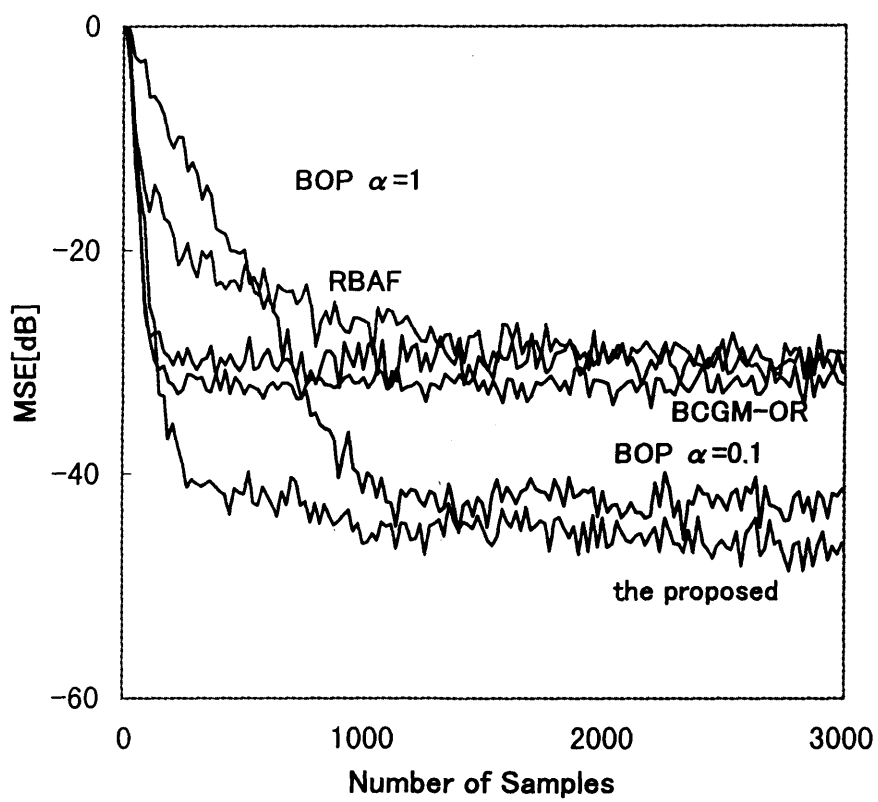


図 4.5: 収束特性の比較 (1) の場合)

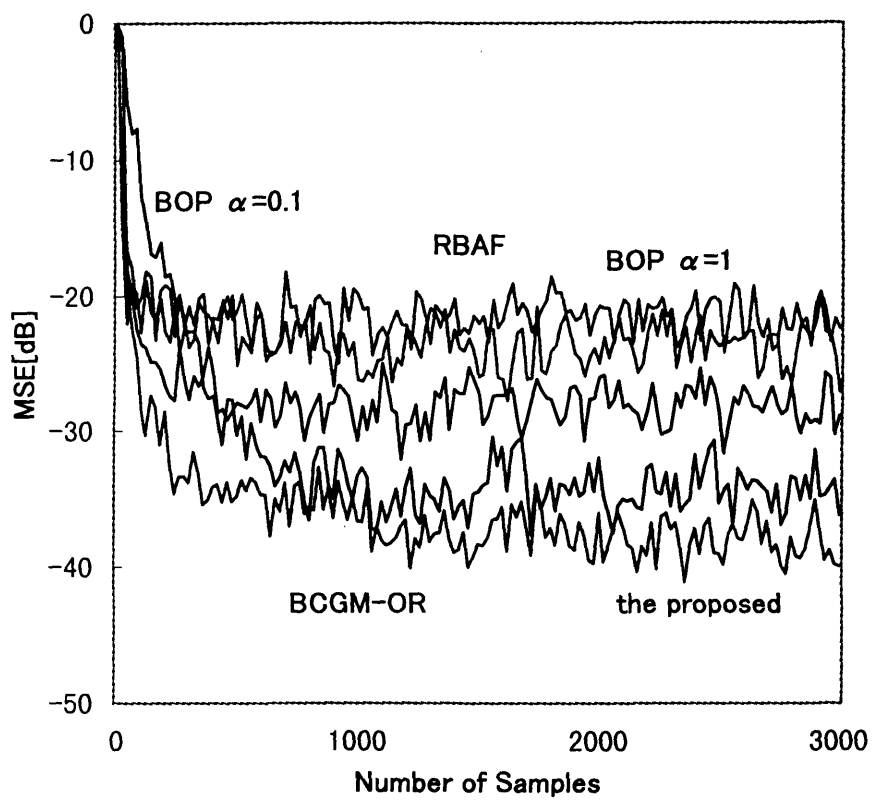


図 4.6: 収束特性の比較 (2) の場合)

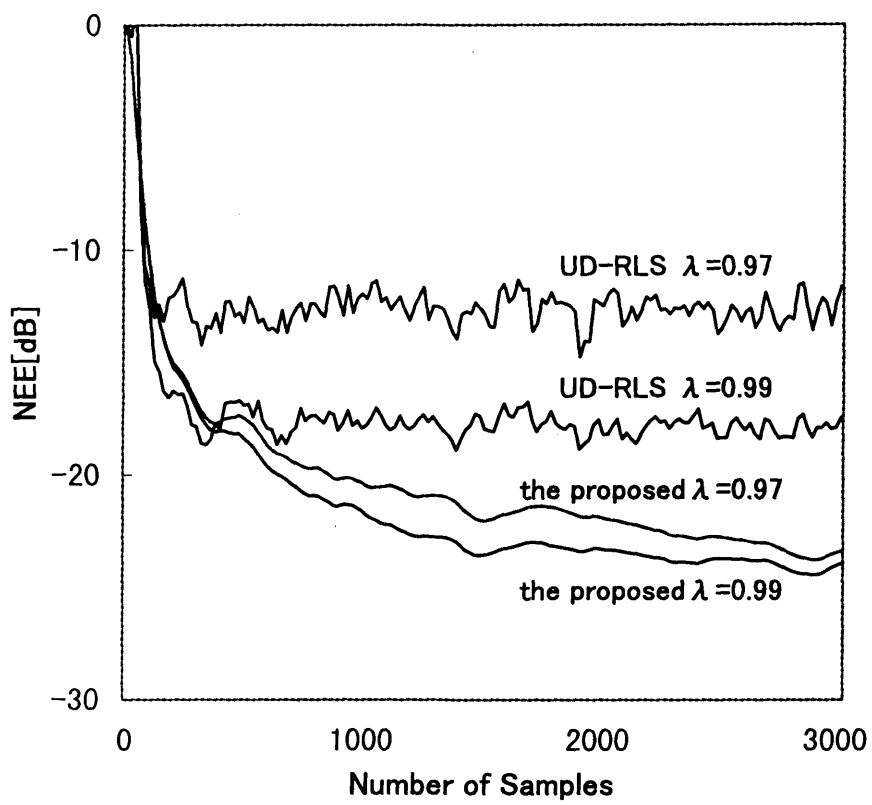


図 4.7: 収束特性の比較 (2) の場合)

付録 有限区間における平均化

あるデータ区間における平均化を行う場合の方法を示す。まず、 m を平均化に用いるデータの数とする。式 (4.19) で与えられる $\hat{c}_N^{(j)}$ は、 $j < m$ の場合と $j \geq m$ の場合に対してそれぞれ以下のように計算できる。

式 (4.13), (4.18) における $\hat{c}_N^{z(j)}$, $\hat{c}_N^{y(j)}$ で必要となる m 個の方向ベクトル成分が蓄積されるまでの初期状態 ($j < m$) はそれぞれ

$$\hat{c}_N^{z(j)} = \frac{1}{j} \sum_{i=1}^j c_N^{z(i)} \quad (j < m) \quad (4.32)$$

$$\hat{c}_N^{y(j)} = \frac{1}{j} \sum_{i=1}^j P_{N,N}^{(i)} h_N^{(j)} \quad (j < m) \quad (4.33)$$

とする。

i) $j < m$ の場合

$\hat{c}_N^{z(j)}$ は式 (4.32) より

$$\hat{c}_N^{z(j)} = \frac{1}{j} \sum_{i=1}^j [(j-1)\hat{c}_N^{z(j-1)} + c_N^{z(j)}] \quad (4.34)$$

となる。 $\hat{c}_N^{z(j)}$ の初期値 $\hat{c}_N^{z(0)}$ は $\mathbf{0}_N$ である。

また、式 (4.33) を参照すると $\hat{c}_N^{y(j)}$ は

$$\hat{c}_N^{y(j)} = \frac{1}{j} [(j-1)\hat{P}_{N,N}^{(j-1)} + P_{N,N}^{(j)}] h_N^{(j)} \quad (4.35)$$

で計算される。但し、 $\hat{P}_{N,N}^{(j)}$ は

$$\hat{P}_{N,N}^{(j)} = \frac{1}{j} [(j-1)\hat{P}_{N,N}^{(j-1)} + P_{N,N}^{(j)}] \quad (4.36)$$

により更新され、 $\hat{P}_{N,N}^{(j)}$ の初期値 $\hat{P}_{N,N}^{(0)}$ は $\mathbf{0}_{N,N}$ である。

ii) $j \geq m$ の場合

同様に $\hat{c}_N^{z(j)}$ は式(4.13)より

$$\begin{aligned}
 \hat{c}_N^{z(j)} &= \frac{1}{m} \sum_{i=j-m+1}^j c_N^{z(i)} \\
 &= \frac{1}{m} \left[\sum_{i=j-m}^{j-1} c_N^{z(i)} + c_N^{z(j)} - c_N^{z(j-m)} \right] \\
 &= \frac{1}{m} \left[m \cdot \hat{c}_N^{z(j-1)} + c_N^{z(j)} - c_N^{z(j-m)} \right]
 \end{aligned} \tag{4.37}$$

となる。但し、

$$c_N^{z(i)} = A_{r,N}^{(i)+} z_r^{(i)} \tag{4.38}$$

である。

また、式(4.17)、(4.18)より $\hat{c}_N^{y(j)}$ は

$$\begin{aligned}
 \hat{c}_N^{y(j)} &= \left[\frac{\sum_{i=j-m+1}^{j-1} P_{N,N}^{(i)} + P_{N,N}^{(j)}}{m} \right] h_N^{(j)} \\
 &= \left[\frac{\sum_{i=j-m}^{j-1} P_{N,N}^{(i)} + P_{N,N}^{(j)} - P_{N,N}^{(j-m)}}{m} \right] h_N^{(j)} \\
 &= \left[\frac{m \cdot \hat{P}_{N,N}^{(j-1)} + P_{N,N}^{(j)} - P_{N,N}^{(j-m)}}{m} \right] h_N^{(j)}
 \end{aligned} \tag{4.39}$$

となる。但し、式(4.17)の $\hat{P}_{N,N}^{(j)}$ は

$$\hat{P}_{N,N}^{(j)} = \frac{m \cdot \hat{P}_{N,N}^{(j-1)} + P_{N,N}^{(j)} - P_{N,N}^{(j-m)}}{m} \tag{4.40}$$

である。

本方法におけるデータの取り方を図4.8に示す。この方法では、データ数 m を少なくするほど追従性が向上するが推定精度は劣化すると考えられる。また m を多くした場合、過去のデータを記憶するためのメモリも m 比例して必要になると考えられる。

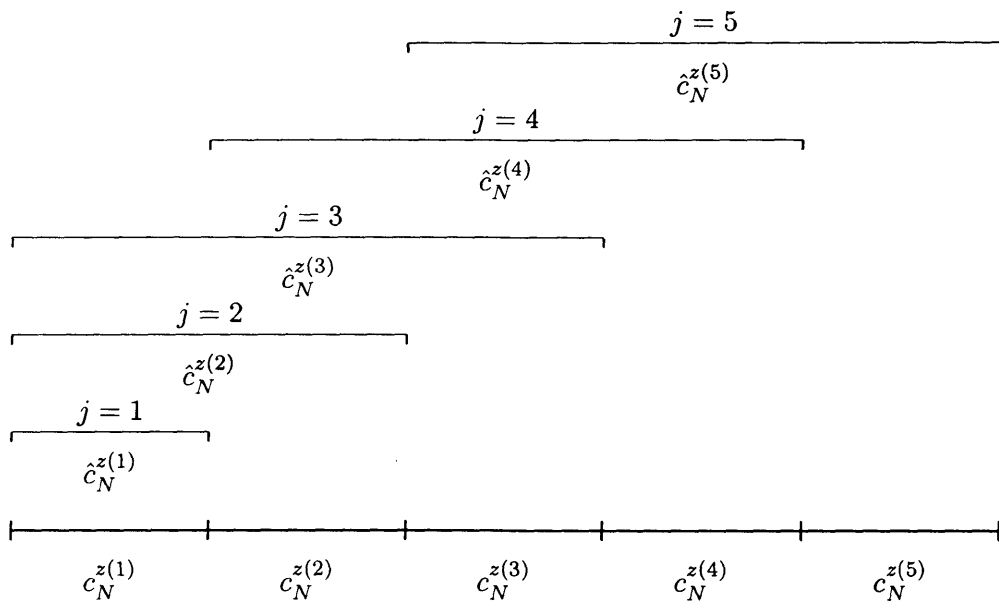


図 4.8: データの取り方 ($m = 3$ の時)

第5章

こう配法に基づく高速な適応アルゴリズムにおけるデータ運用法の改良に関する一考察

第2章で述べたように、有色信号入力時においても高速なパラメータ推定が可能な適応アルゴリズムとして、アフィン射影算法が提案されている。これは多次元部分空間への直交射影演算に基づいた方式であり、1サンプルごとに適応フィルタ係数の修正を逐次的に行う。従って、高速な収束を期待できるが、演算量は非常に多くなる。本章では古川らによってBOPアルゴリズムの一実現法として提案されているこう配法に基づく高速な適応アルゴリズムをアフィン射影算法と同様の逐次処理に対応させ、データの運用法を改良することにより収束速度と演算量を考慮した手法を提案している。また、シミュレーションを行うことにより本手法の有効性を検討している。

5.1 まえがき

適応アルゴリズムには1) 有色信号入力時においても高速な収束速度が得られること、2) 1サンプル当たりの計算に必要となる演算量が少ないこと、などの性能が要求されることは既に述べた。1)の要求を満たす手法としてアフィン射影算法(以下、APアルゴリズム)[44]が提案されている。APアルゴリズムは直交射影原理に基づいた算法であり、1サンプルごとにフィルタ係数の修正を行うため、信号の有色度

に関係なく高速な収束速度を示すが、ブロック長を長くした場合には1サンプル当たり演算量が非常に多くなるという欠点が生じる。そこでAPアルゴリズムの演算量軽減のためにフィルタ係数の修正をブロック単位で行うブロック直交射影アルゴリズム(以下、BOPアルゴリズム)[45]が提案されている。しかしながらBOPアルゴリズムではブロックごとにフィルタ係数の修正を行うため、APアルゴリズムと比べ収束の高速性は失われる。従って逐次処理による収束速度の高速性を欠くことなく、演算量の軽減化を図ることが重要となる。

そこで本章では、比較的演算量が少ない高速な逐次適応アルゴリズムについて議論する。本手法は、BOPアルゴリズムの一実現法である“こう配法に基づく高速な適応アルゴリズム”(以下、GM-BOPアルゴリズム)[52][53]を逐次処理へ変換した方式(以下、GM-APアルゴリズム)に基礎をおいている。GM-BOPアルゴリズムは、演算時間あるいはハードウェア構成上の制約などで一ブロック内の処理を途中でうち切っても良好な収束特性を得ることができるとされており、これを逐次処理変換したGM-APアルゴリズムも同様の特徴を有することができる。しかしながらその反面、演算量の増加が見込まれる。本章ではそれを避けるために、任意の時刻における修正方向ベクトル算出に用いたデータを次時刻における修正方向ベクトル算出に利用する適応アルゴリズムを提案する。本手法は、GM-APアルゴリズムにおいて算出される修正方向ベクトルの計算に工夫を加えることにより演算量の軽減を図っている。従って、本手法はGM-APアルゴリズムよりも少ない演算量で高速な収束特性を実現できることが期待される。

以下5.2では準備及び本手法の導出を行い、5.3ではコンピュータ・シミュレーションを行い本手法の有効性を示している。

5.2 高速な適応アルゴリズム

5.2.1 準備

まず、以後の議論で必要となる諸量の定義を行う。また、 k , N , r をそれぞれ時刻、フィルタのタップ長およびブロック長とする。

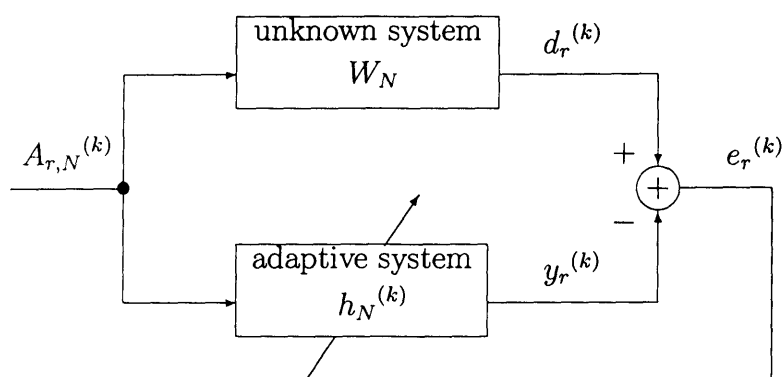


図 5.1: パラメータ推定のモデル

時刻 k における入力状態行列および入力状態ベクトルをそれぞれ

$$A_{r,N}^{(k)} \triangleq [x_N^{(k-r+1)}, x_N^{(k-r+2)}, \dots, x_N^{(k)}]^T \quad (5.1)$$

$$x_N^{(k)} \triangleq [x^{(k)}, x^{(k-1)}, \dots, x^{(k-N+1)}]^T \quad (5.2)$$

と定義し推定系の係数ベクトルを $h_N^{(k)}$ とすれば未知系の出力ベクトル $d_r^{(k)}$, 推定系出力ベクトル $y_r^{(k)}$ および出力誤差ベクトル $e_r^{(k)}$ はそれぞれ

$$d_r^{(k)} = A_{r,N}^{(k)} \cdot W_N \quad (5.3)$$

$$y_r^{(k)} = A_{r,N}^{(k)} \cdot h_N^{(k)} \quad (5.4)$$

$$e_r^{(k)} = d_r^{(k)} - y_r^{(k)} \quad (5.5)$$

で与えられる。但し, W_N は推定すべき未知系のパラメータである。図 5.1 に本章におけるパラメータ推定のモデルを示す。また表 5.1 は GM-AP アルゴリズムの適応フィルタの係数修正手順を表している。

ここで, 第 2 章において共役こう配法について述べたが, それをふまえながら再度 GM-AP アルゴリズムも取り混ぜて説明する。但し第 2 章とは定義した量など異なる部分があるので本章における表 5.1 を基準として行う。

まず,

$$A_{r,N}^{(k)T} A_{r,N}^{(k)} = A_{N,N}^{(k)} \quad (5.6)$$

$$A_{r,N}^{(k)T} d_r^{(k)} = b_N^{(k)} \quad (5.7)$$

表 5.1: GM-AP アルゴリズム

<p>For time k</p> <p>(a) Preparation</p> $\theta_N(1)^{(k)} = h_N^{(k)}$ $c_N(1)^{(k)} = g_N(1)^{(k)}$ <p>where $h_N^{(k)}$ is the optimum solution vector at time $(k - 1)$</p> <p>(b) Procedure</p> <p>for $i = 1$ to m</p> <p>1) $y_r(i)^{(k)} = A_{r,N}^{(k)} \cdot \theta_N(i)^{(k)}$</p> <p>2) $e_r(i)^{(k)} = d_r^{(k)} - y_r(i)^{(k)}$</p> <p>3) $g_N(i)^{(k)} = -2A_{r,N}^{(k)T} \cdot e_r(i)^{(k)}$</p> <p>4) $c_N(i)^{(k)} = g_N(i)^{(k)} - \sum_{\ell=1}^{i-1} \frac{c_N(\ell)^{(k)T} \cdot g_N(i)^{(k)}}{\ c_N(\ell)^{(k)}\ ^2} \cdot c_N(\ell)^{(k)}$</p> <p>5) $\theta_N(i+1)^{(k)} = \theta_N(i)^{(k)} - 2\alpha \frac{\ e_r(i)^{(k)}\ ^2}{\ c_N(i)^{(k)}\ ^2} \cdot c_N(i)^{(k)}$</p> <p>next i</p> <p>$h_N^{(k+1)} = \theta_N(m+1)^{(k)}$, and after replacing time index k by $k+1$, repeat the procedure 1) ~ 5) for time $(k+1)$ similarly.</p> <p>where i is the step number at time index k.</p>
--

と定義する．時刻 k におけるデータで得られる最適解を $h_N(opt)^{(k+1)}$ とすれば， $A_{N,N}^{(k)}$ の対称性を考慮して，共役こう配法により最適解を求めることができる．ここで最適解 $h_N(opt)^{(k+1)}$ は $\theta_N^{(k)}$ を変数ベクトルとする次の 2 次形式

$$J(\theta_N^{(k)}) = \frac{1}{2} \theta_N^{(k)T} A_{N,N}^{(k)} \theta_N^{(k)} - \theta_N^{(k)T} b_N^{(k)} + c \quad (5.8)$$

を最小にする． c は任意の定数である．従って，

$$J[h_N(opt)^{(k+1)}] \leq J(\theta_N^{(k)}) \quad (5.9)$$

が成り立つ．図 5.2 は GM-AP アルゴリズムと共役こう配法の幾何学的関係を示している．この図は $N = 2$ と置いたときの J に関する等高線群である．図において $c_N(1)^{(k)}$ は修正方向ベクトル， $\theta_N(2)^{(k)}$ は GM-AP アルゴリズムにおけるフィルタ係数ベクトル， $\hat{\theta}_N(2)^{(k)}$ は共役こう配法におけるフィルタ係数ベクトルである．但し，初期値 $\theta_N(1)^{(k)}$ は等しいとする．図 5.2 からわかるように， $\hat{\theta}_N(2)^{(k)}$ は修正方向ベクトル $c_N(1)^{(k)}$ 上の等高線に接している点である．これに対し， $\theta_N(2)^{(k)}$ は $h_N(opt)^{(k+1)}$ から $c_N(1)^{(k)}$ に直交射影した点であり $h_N(opt)^{(k+1)}$ に最も近い点であることがわかる．ここで，式 (5.9) に関する等高線が楕円であることを考える． $\theta_N(2)^{(k)}$ と $\hat{\theta}_N(2)^{(k)}$ は $c_N(1)^{(k)}$ が等高線の主軸にある場合か， $A_{N,N}^{(k)}$ の対角成分がすべて等しい対角行列でないとき一致しない．このことから $h_N(opt)^{(k+1)}$ に対する最良の近似解は $\theta_N(2)^{(k)}$ であることがわかる．従って何らかの理由によって繰り返し計算を途中でうち切った場合，GM-AP アルゴリズムの方がより最適な近似が得られることがわかる．

ここで $\|h_N(opt)^{(k+1)} - \theta_N(2)^{(k)}\|$ と $\|h_N(opt)^{(k+1)} - \hat{\theta}_N(2)^{(k)}\|$ の比は $A_{N,N}^{(k)}$ の最大固有値 λ_1 と最小固有値 λ_2 の比が大きいほど大きくなると考えられる．図 5.3 に等高線と固有値の関係を示す．実際の適用において $A_{N,N}^{(k)}$ の最大固有値と最小固有値比はかなり大きい場合も想定されるので，反復計算を途中でうち切った場合においても精度の良い近似解が期待される．以上のことより GM-AP アルゴリズムは共役こう配法を適用した方式に対し有用な方式であると考えられる．

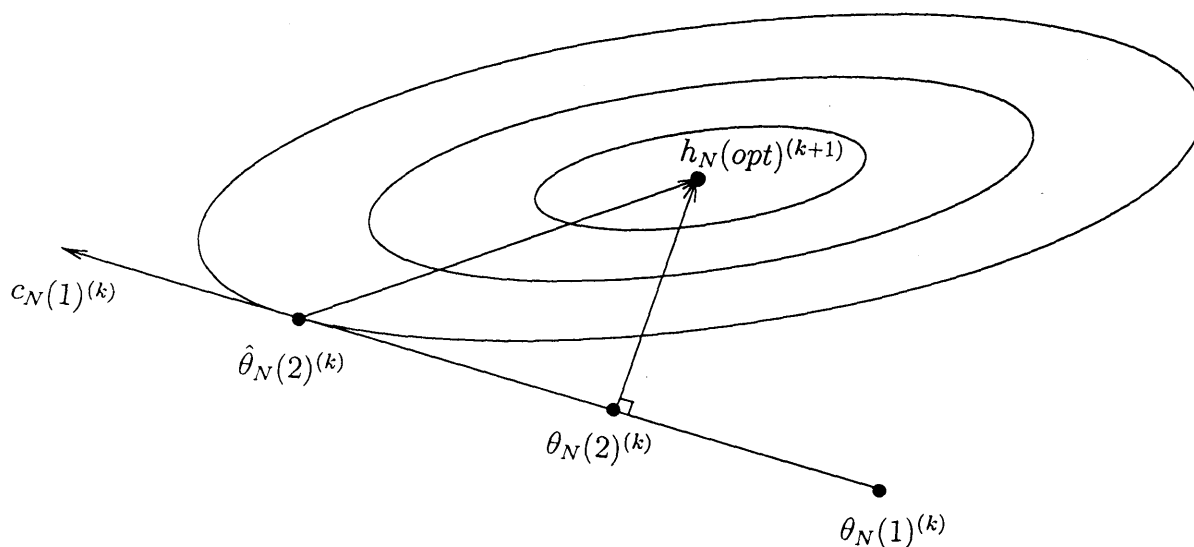


図 5.2: GM-AP アルゴリズムと共役こう配法の幾何学的関係

5.2.2 データの運用法を改良した適応アルゴリズム

本手法は、基本的に GM-AP アルゴリズムにおける修正方向ベクトルの算出に関するデータ運用法の改良に主眼をおいているため、基本となるアルゴリズムの流れは GM-AP アルゴリズムと同じである。従って本章では表 5.1 の 4) の修正方向ベクトル算出に際する式の変更点のみの導出を行う。但し、表 5.1 中の変数 i は任意の時刻におけるステップ変数である。ここでは説明の都合上、それ以外の手順を省略するが、本手法でも任意の時刻 k において表 5.1 の 4) を除く全ての手順が実行されているものとする。また本手法において、時刻 k のステップ i における修正方向ベクトルを $\widehat{g}_N(i)^{(k)} (i = 1, 2, \dots, m)$ と表記し、この修正方向ベクトルの流れを理解しやすいように修正方向ベクトルを行成分に持つ行列を便宜的に $\widehat{G}_{m,N}^{(k)}$ と表記する。但し、行列 $\widehat{G}_{m,N}^{(k)}$ は実際の計算には何ら寄与しないことに注意されたい。ここで、上記の m は時刻 k におけるデータ行列 $A_{r,N}^{(k)}$ の異なる特異値の数に対応するパラメータであり、これはブロック長 r を越えることはない ($m \leq r$)。ここでは時刻 k に関わ

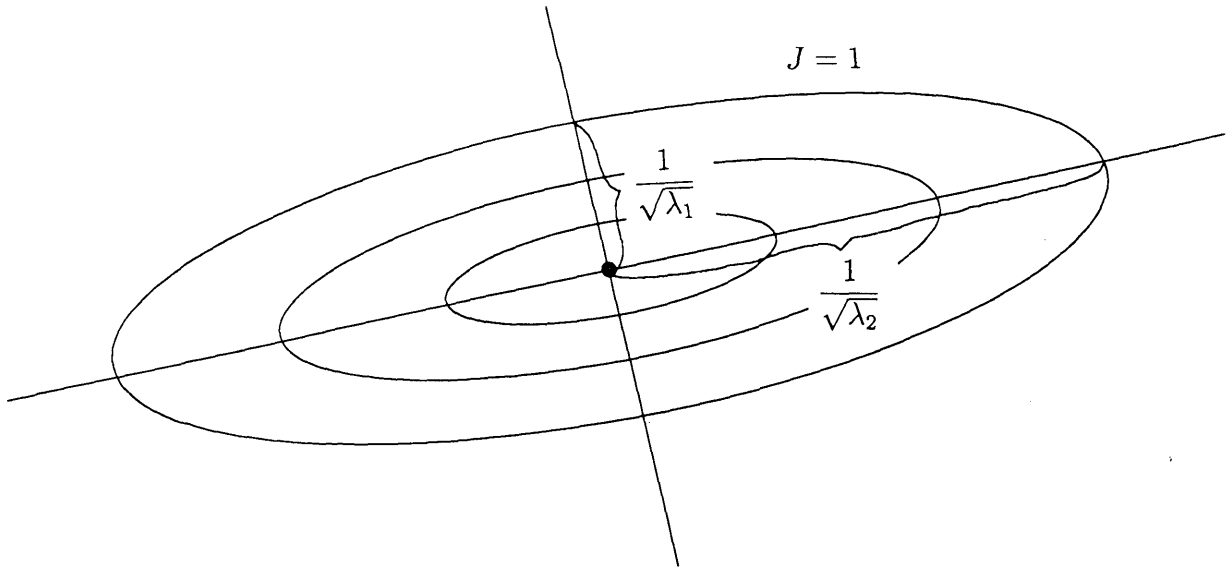


図 5.3: 等高線と固有値の関係

らずパラメータ m は変化しないと仮定する。また、 γ を正の整数とし、以後の議論の都合上時刻 k を $k = j + \gamma \cdot m (j = 1 \sim m)$ と表記する。さらに、時刻 k に対応する量、例えば $\widehat{g}_N(i)^{(j+\gamma \cdot m)}$ を表記の簡略化のため $\widehat{g}_N(i)^{(j)}$ と表すことにする。

第 1 段階 ($j = 1$ すなわち $k = 1 + \gamma \cdot m$)

この場合は GM-AP アルゴリズムと全く同じ方法で修正方向ベクトルを算出する。すなわち、この時刻においては m 本の修正方向ベクトル $\widehat{g}_N(i)^{(1)} (i = 1, 2, \dots, m)$ が算出される。これは Gram-Schmidt の直交化を用い

$$\widehat{g}_N(i)^{(1)} = g_N(i)^{(1)} - \sum_{\ell=1}^{i-1} \frac{g_N(i)^{(1)T} \cdot \widehat{g}_N(\ell)^{(1)}}{\|\widehat{g}_N(\ell)^{(1)}\|^2} \cdot \widehat{g}_N(\ell)^{(1)} \quad (5.10)$$

から容易に計算することが出来る。ただし $i = 1 \sim m$ である。またこのときの行列 $\widehat{G}_{m,N}^{(1)}$

$$\widehat{G}_{m,N}^{(1)} = [\widehat{g}_N(m)^{(1)}, \widehat{g}_N(m-1)^{(1)}, \dots, \widehat{g}_N(1)^{(1)}]^T \quad (5.11)$$

の行ベクトルは m 本の修正方向ベクトルであり、これを次の段階における修正方向

ベクトルの算出に使用する.

第2段階 ($j = 2$ すなわち $k = 2 + \gamma \cdot m$)

この時刻では, ステップ i が $i = 1 \sim (m-1)$ のときにおける修正方向ベクトル $\widehat{g}_N(i)^{(2)}$ を前時刻に得られている $\widehat{g}_N(i)^{(1)}$ とおく. すなわち,

$$\widehat{g}_N(i)^{(2)} = \widehat{g}_N(i)^{(1)} \quad [i = 1 \sim (m-1)] \quad (5.12)$$

として, $i = m$ のときの修正方向ベクトル $\widehat{g}_N(m)^{(2)}$ のみを算出する. これは $i = m$ におけるこう配ベクトル $g_N(m)^{(2)}$ と式 (5.12) の量を用いて Gram-Schmidt の直交化により

$$\widehat{g}_N(m)^{(2)} = g_N(m)^{(2)} - \sum_{\ell=1}^{m-1} \frac{g_N(m)^{(2)T} \cdot \widehat{g}_N(\ell)^{(2)}}{\|\widehat{g}_N(\ell)^{(2)}\|^2} \cdot \widehat{g}_N(\ell)^{(2)} \quad (5.13)$$

で与えられる. また, この時刻の各ステップ i における修正方向ベクトルを行ベクトルとする行列 $\widehat{G}_{m,N}^{(2)}$ を

$$\widehat{G}_{m,N}^{(2)} = \left[\widehat{g}_N(m)^{(2)}, \widehat{g}_N(m-1)^{(2)}, \dots, \widehat{g}_N(1)^{(2)} \right]^T \quad (5.14)$$

と表記する.

第3段階 ($j = 3$ すなわち $k = 3 + \gamma \cdot m$)

この時刻では2本の修正方向ベクトルをの算出を行う. ステップ i が $i = 1 \sim (m-2)$ のときにおける修正方向ベクトル $\widehat{g}_N(i)^{(3)}$ を前時刻に得られている $\widehat{g}_N(i)^{(2)}$ とおく. すなわち,

$$\widehat{g}_N(i)^{(3)} = \widehat{g}_N(i)^{(2)} \quad [i = 1 \sim (m-2)] \quad (5.15)$$

として, $i = (m-1)$ のときの修正方向ベクトル $\widehat{g}_N(m-1)^{(3)}$ および $i = m$ のときの修正方向ベクトル $\widehat{g}_N(m)^{(3)}$ を算出する. 手順はまず, $i = (m-1)$ におけるこう配ベクトル $g_N(m-1)^{(3)}$ と式 (5.15) の量を用いて Gram-Schmidt の直交化により, ステップ $i = (m-1)$ のときの修正方向ベクトル $\widehat{g}_N(m-1)^{(3)}$

$$\widehat{g}_N(m-1)^{(3)} = g_N(m-1)^{(3)} - \sum_{\ell=1}^{m-2} \frac{g_N(m-1)^{(3)T} \cdot \widehat{g}_N(\ell)^{(3)}}{\|\widehat{g}_N(\ell)^{(3)}\|^2} \cdot \widehat{g}_N(\ell)^{(3)} \quad (5.16)$$

を導出する.

次に, $i = m$ におけるこう配ベクトル $g_N(m)^{(3)}$ および式 (5.15), (5.16) の量を用い, Gram-Schmidt の直交化により, この時刻の最終的な修正方向ベクトル $\widehat{g}_N(m)^{(3)}$

$$\widehat{g}_N(m)^{(3)} = g_N(m)^{(3)} - \sum_{\ell=1}^{m-1} \frac{g_N(m)^{(3)T} \cdot \widehat{g}_N(\ell)^{(3)}}{\|\widehat{g}_N(\ell)^{(3)}\|^2} \cdot \widehat{g}_N(\ell)^{(3)} \quad (5.17)$$

を算出する. また, この時刻の各ステップ i における修正方向ベクトルを行ベクトルとする行列 $\widehat{G}_{m,N}^{(3)}$ を

$$\widehat{G}_{m,N}^{(3)} = [\widehat{g}_N(m)^{(3)}, \widehat{g}_N(m-1)^{(3)}, \dots, \widehat{g}_N(1)^{(3)}]^T \quad (5.18)$$

と表記する.

$j = 4 \sim (m-1)$ においても同様の操作を繰り返す.

最終段階 ($j = m$ すなわち $k = (1 + \gamma) \cdot m$)

この時刻では $m-1$ 本の修正方向ベクトルを新たに算出する. ステップ i が $i = 1$ のときにおける修正方向ベクトル $\widehat{g}_N(1)^{(m)}$ を前時刻に得られている $\widehat{g}_N(m)^{(m-1)}$ とおく. すなわち,

$$\widehat{g}_N(1)^{(m)} = \widehat{g}_N(1)^{(m-1)} \quad (5.19)$$

として, $i = (2 \sim m)$ のときの修正方向ベクトル $\widehat{g}_N(i)^{(m)}$ を算出する. まず, $i = 2$ におけるこう配ベクトル $g_N(2)^{(m)}$ と式 (5.19) の量を用い, ステップ $i = 2$ のときの修正方向ベクトル $\widehat{g}_N(2)^{(m)}$

$$\widehat{g}_N(2)^{(m)} = g_N(2)^{(m)} - \frac{g_N(2)^{(m)T} \cdot \widehat{g}_N(1)^{(m)}}{\|\widehat{g}_N(1)^{(m)}\|^2} \cdot \widehat{g}_N(1)^{(m)} \quad (5.20)$$

を算出する.

次に $i = 3$ におけるこう配ベクトル $g_N(3)^{(m)}$ と式 (5.19), (5.20) の量を用い, ステップ $i = 3$ のときの修正方向ベクトル $\widehat{g}_N(3)^{(m)}$

$$\widehat{g}_N(3)^{(m)} = g_N(3)^{(m)} - \sum_{\ell=1}^2 \frac{g_N(3)^{(m)T} \cdot \widehat{g}_N(\ell)^{(m)}}{\|\widehat{g}_N(\ell)^{(m)}\|^2} \cdot \widehat{g}_N(\ell)^{(m)} \quad (5.21)$$

を算出する.

⋮

以下 $\widehat{g}_N(i)^{(m)}(i = 4, 5, \dots, m-1)$ についても同様に算出する.

⋮

最終的に算出される修正方向ベクトル $\widehat{g}_N(m)^{(m)}$ はステップ $i = m$ におけるこう配ベクトル $g_N(m)^{(m)}$ と式 (5.19) およびステップ $i = 2 \sim m-1$ で算出された修正方向ベクトル $\widehat{g}_N(i)^{(m)}$ を用い

$$\widehat{g}_N(m)^{(m)} = g_N(m)^{(m)} - \sum_{\ell=1}^{m-1} \frac{g_N(m)^{(m)T} \cdot \widehat{g}_N(\ell)^{(m-1)}}{\|\widehat{g}_N(\ell)^{(m-1)}\|^2} \cdot \widehat{g}_N(\ell)^{(m-1)} \quad (5.22)$$

で与えられる.

以上で述べたように, 本アルゴリズムでは $k = 1 + \gamma \cdot m \sim (1 + \gamma) \cdot m$ ($\gamma = 0, 1, 2, \dots$) までの一連の流れを繰り返すことによりその時刻における修正方向ベクトルを算出し適応フィルタの係数修正を行う. この時刻が本手法における手順の区切りとなり, 次の時刻, すなわち $j = m$ における処理を満了時刻を終えると第1段階の処理($k \equiv 1 \pmod{m}$)へ戻る. この一連の手順の流れをまとめたものが表 5.2である.

また図 5.4は本手法における時刻 k での修正方向ベクトルの算出に伴うデータの流れを示したものである. 図の左側の点線で囲まれた部分は計算済みの結果であり, 右側はその時刻において新たに計算が必要となる修正方向ベクトルである.

表 5.3では本方式と GM-AP アルゴリズムとの演算量の比較を行っている. $m = 1$ の場合, 本アルゴリズムの計算手順は GM-AP アルゴリズムのそれと全く同じであるため演算量も等しくなる.

図 5.5は表 5.3に基づきブロック長 r を変化させたときの両方式の演算量が変化する様子を図示したものである. また, それぞれの方式の1ブロックにおける処理の繰り返し回数 m は両方の演算量が最大となる場合, すなわち $m = r$ と設定している. この図からブロック長 r が大きくなる程, GM-AP 法と本手法の演算量の差が大きくなることがわかる.

5.3 コンピュータ・シミュレーション

本手法の有効性を示すためにコンピュータ・シミュレーションを行う. 但し, 本章では外乱が存在しない場合の基本的特性のみを示す. 推定精度を表す評価量とし

表 5.2: 提案する手法

<p>For time k ($k = 1, 2, \dots$)</p> <p>(a) Preparation</p> <p>If $k \equiv 1 \pmod{m}$ then $p = 1 : sum = 0$ (γ is a positive number of constant) else $p = m - sum + 1$ $\theta_N(1)^{(k)} = h_N^{(k)}$ $\widehat{g}_N(1)^{(1)} = g_N(1)^{(1)}$</p> <p>(b) Procedure</p> <p>for $i = p$ to m</p> <p>1) $y_r(i)^{(k)} = A_{r,N}^{(k)} \cdot \theta_N(i)^{(k)}$ 2) $e_r(i)^{(k)} = d_r^{(k)} - y_r(i)^{(k)}$ 3) $g_N(i)^{(k)} = -2A_{r,N}^{(k)T} \cdot e_r(i)^{(k)}$ 4) $\widehat{g}_N(i)^{(k)} = g_N(i)^{(k)} - \sum_{\ell=1}^{p-1} \frac{\widehat{g}_N(\ell)^{(k)T} \cdot g_N(i)^{(k)}}{\ \widehat{g}_N(\ell)^{(k)}\ ^2} \cdot \widehat{g}_N(\ell)^{(k)}$ 5) $\theta_N(i+1)^{(k)} = \theta_N(i)^{(k)} - 2\alpha \frac{\ e_r(i)^{(k)}\ ^2}{\ \widehat{g}_N(i)^{(k)}\ ^2} \cdot \widehat{g}_N(i)^{(k)}$</p> <p>next i $h_N^{(k+1)} = \theta(m+1)^{(k)}$ $sum \leftarrow sum + 1$ next time</p>
--

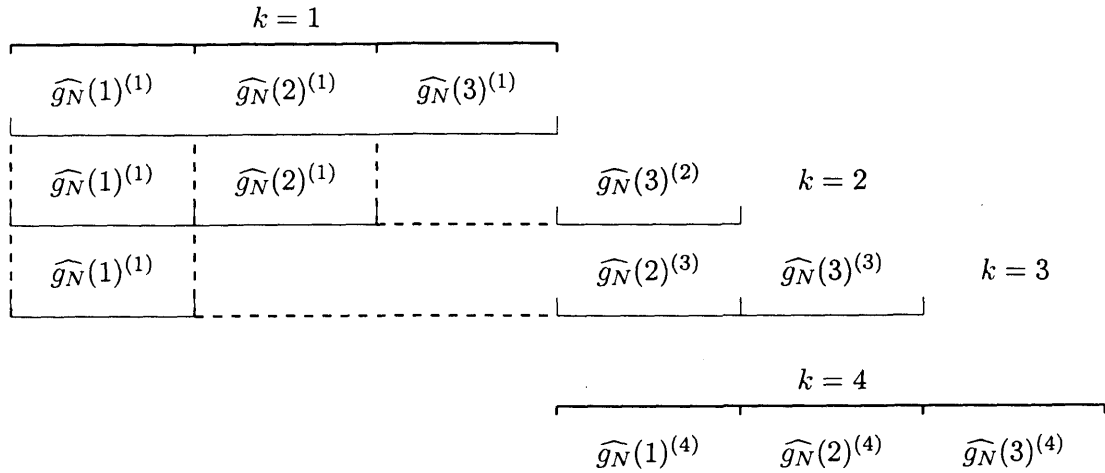


図 5.4: 時刻 k における修正方向ベクトルの計算 ($m = 3$ の時)

表 5.3: 演算量の比較

algorithm	number of multiplications
GM-AP	$2Nm \left(r + \frac{m+1}{2} \right) + m(r+1)$
the proposed	$\frac{1}{6} \{ (6N+3)(rm+r+m) + 2N(2m^2+1) + 3 \}$

て (NEE:Normalized Estimation Error)

$$NEE = 10 \log_{10} \frac{\|W_N - h_N^{(k)}\|^2}{\|W_N\|^2} \quad [\text{dB}] \quad (5.23)$$

を用いた。また入力信号 $x(k)$ として平均値零，分散 $1/48$ の正規乱数をフィルタ $F(z)$

$$F(z) = 1 + 0.445z^{-1} + 0.202z^{-2} + 0.0907z^{-3} + 0.0408z^{-4} + 0.0183z^{-5}. \quad (5.24)$$

を 7 個従属接続したシステムの伝達関数で表される FIR フィルタに通した信号 [86] を用いた。この信号の条件数は約 5.04×10^5 である。また，図 5.6, 5.7, 5.8 において両方

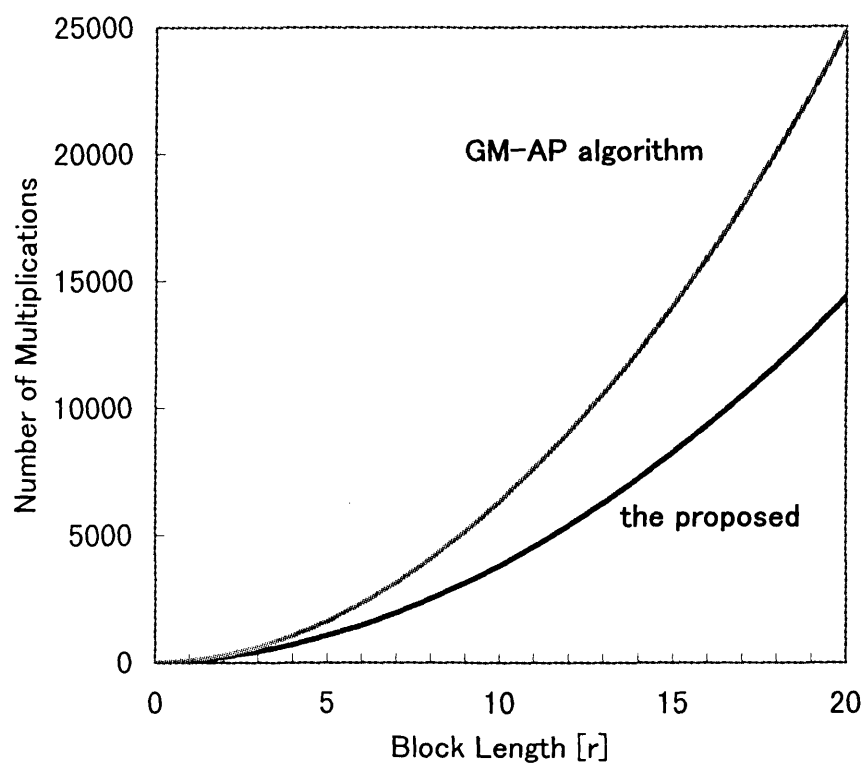


図 5.5: 演算量の比較

式ともブロック長 r , 1 ブロックの処理繰り返し回数 m はそれぞれ $m = r = 9, 12, 15$ である.

図 5.6 から図 5.8 は GM-AP 法と本手法の収束特性を示している. これらの図からわかるように, ブロック長 r 及び m をある程度大きく取ったとき, 本手法は GM-AP アルゴリズムよりも高速な収束特性を示す. 従って, 図 5.5 の結果を考慮すると, 本手法の方が GM-AP アルゴリズムと比較して, 少ない演算量で高速な収束速度を実現できることが確認できる.

5.4 むすび

本章では, ブロック処理である GM-BOP アルゴリズムを逐次処理に対応させた方式に基づき, 修正方向ベクトルの算出手順を改良することにより, 演算量の軽減化を実現した手法を提案し, 観測雑音を無視した場合の基本的特性について議論した. 本手法は GM-AP アルゴリズムと比べ, 少ない演算量で高速な収束速度の実現が可能であることが確認できた. 本手法は r が大きくなる程, 収束速度が速くなることがシミュレーションで確認されている. これを論理的に検証することが今後の課題である. さらに, 観測雑音が存在する場合の本手法の収束特性の解析も併せて今後の課題としたい.

次章では, 本章で議論の対象としたこの配法に基づく高速なアルゴリズムを観測雑音が存在する場合に適用したときに, 観測雑音による影響を軽減する方式について議論を行う.

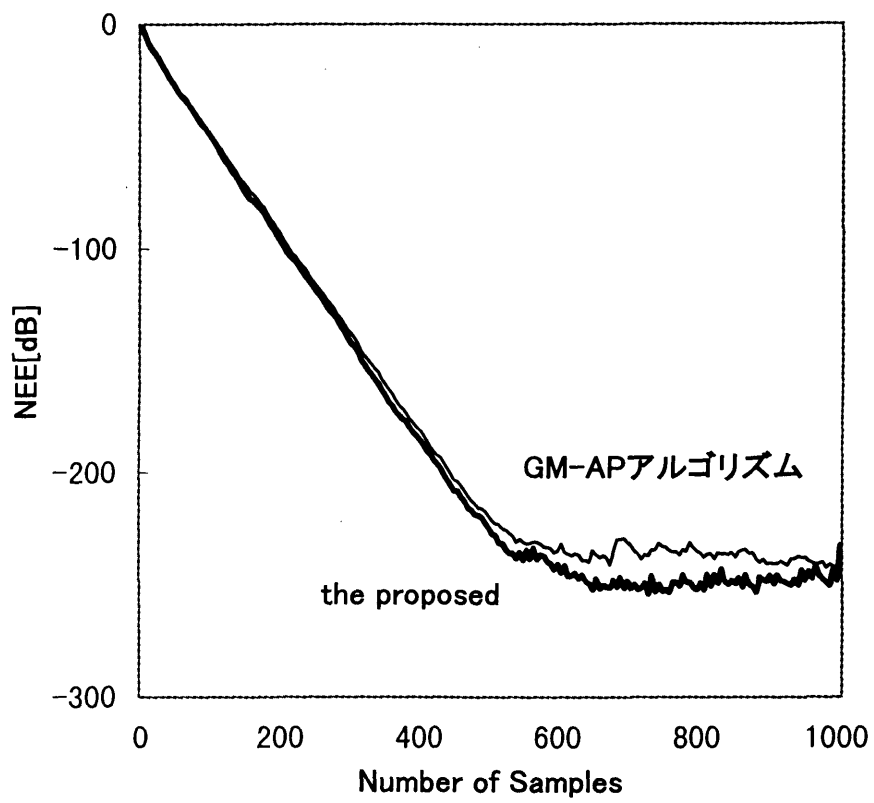


図 5.6: 収束特性の比較 ($N = 20, r = 9, m = 9$)

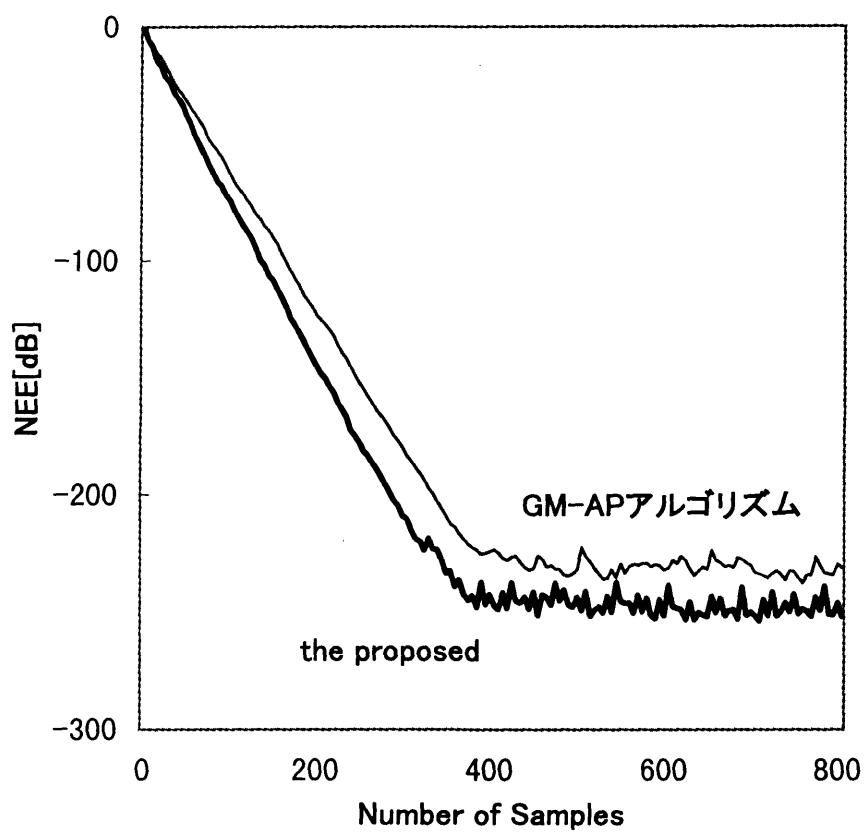


図 5.7: 収束特性の比較 ($N = 20, r = 12, m = 12$)

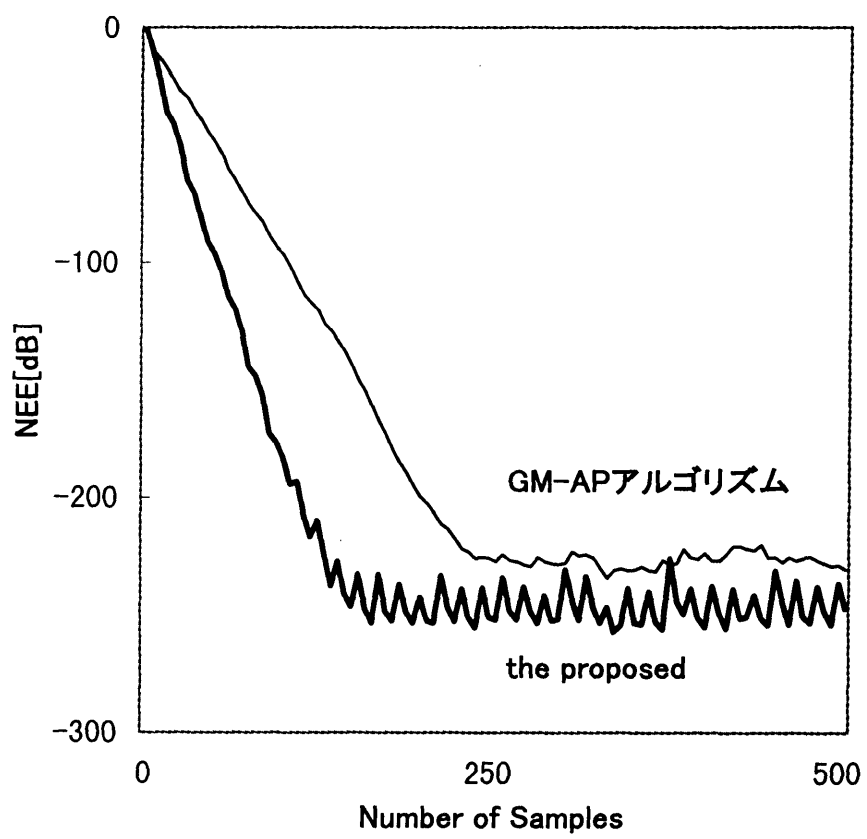


図 5.8: 収束特性の比較 ($N = 20, r = 15, m = 15$)

第6章

観測雑音にロバストなこう配法に基づく適応アルゴリズム

適応アルゴリズムには，有色信号入力時において高速な収束特性を有することと共に，観測雑音に対してもロバストであることが要求される．本章では前章において議論の対象としたこう配法に基づく高速な適応アルゴリズムを改良し，観測雑音に対しロバストな方式を提案している．本方式はこう配法に基づく高速な適応アルゴリズムの修正量に含まれる観測雑音の影響を軽減することにより，観測雑音が存在する場合において従来法よりも良好な推定精度を得ることができる．

6.1 まえがき

BOP アルゴリズムは演算量の軽減化と収束速度の高速化を比較的両立させた優れた方式とすることができるが，BOP アルゴリズムを含む直交射影アルゴリズムは観測雑音の影響を受けやすく，推定精度が劣化することが指摘されていることは既に述べた．

ところで，BOP アルゴリズムの一実現法としてこう配法に基づく高速な適応アルゴリズム (以下，GM-BOP アルゴリズム)[52][53] が提案されている．GM-BOP アルゴリズムはフィルタ修正における計算手順の繰り返し回数を途中でうち切った場合においても良好な収束特性が得られる方式である．GM-BOP アルゴリズムでは，こう配ベクトルが重要な役割を果たしており，周知のように，こう配ベクトルは任

意の一ブロック内において、入力データ行列のブロック長 (r) 本の状態ベクトルと、それら状態ベクトルに対応する r 個の誤差信号により計算される。さて、観測雑音が存在する状況下において、GM-BOP アルゴリズムを適用した場合、入力信号と観測雑音が無相関で観測雑音の平均を零と仮定すれば、こう配ベクトルの生成の際に、ブロック内で観測雑音 r 個による平均化の操作が行われることになり、ブロック長を長くすれば雑音による収束特性への影響がある程度緩和されると考えられる。その一方で、GM-BOP アルゴリズムは係数更新の際、出力誤差ベクトルのノルムの自乗を使用しているため、観測雑音が存在する場合、推定精度が劣化する。これは、後述するように誤差ベクトルの自乗が観測雑音の分散を含む形で与えられることに起因するためと考えられる。

そこで本章では、GM-BOP アルゴリズムのフィルタ係数更新式中で与えられる出力誤差ベクトルのノルムの自乗項の代わりに、こう配ベクトルを含む量を用いた適応アルゴリズムを提案する。本方式によればブロック長を長くすることにより、観測雑音が存在する場合には GM-BOP アルゴリズムよりも良好な推定精度が得られると期待される。以下 6.2 では議論に必要な準備を行った後、本方式の導出を行っている。また、6.3 ではコンピュータ・シミュレーションにより本方式の有効性を示している。

6.2 準備

本節では以後の議論で必要となる諸量の定義及び、GM-BOP アルゴリズムの計算手順を示す。 k , N , r 及び j をそれぞれ時刻、フィルタのインパルス応答、ブロック長及びブロック番号とする。時刻 k における入力状態ベクトルを

$$x_N(k) = [x(k), x(k-1), \dots, x(k-N+1)]^T \quad (6.1)$$

と定義し、ブロック j における入力状態行列を

$$\begin{aligned} A_{r,N}^{(j)} &= [x_N(jr), x_N(jr-1), \dots, x_N(jr-r+1)]^T \\ &= [x_N(1)^{(j)}, x_N(2)^{(j)}, \dots, x_N(r)^{(j)}]^T \end{aligned} \quad (6.2)$$

と定義する。ブロック j における推定系の係数ベクトルを $h_N^{(j)}$ とすれば推定系出力信号ベクトルは

$$y_r^{(j)} = A_{r,N}^{(j)} h_N^{(j)} \quad (6.3)$$

で与えられる。一方、未知系出力信号ベクトルを

$$d_r^{(j)} = A_{r,N}^{(j)} W_N \quad (6.4)$$

とすれば出力誤差ベクトルは

$$e_r^{(j)} = z_r^{(j)} - y_r^{(j)} \quad (6.5)$$

となる。ここで $z_r^{(j)}$ は観測信号ベクトルであり

$$\begin{aligned} z_r^{(j)} &= d_r^{(j)} + v_r^{(j)} \\ &= [z(1)^{(j)}, z(2)^{(j)}, \dots, z(r)^{(j)}] \end{aligned} \quad (6.6)$$

である。但し、 $v_r^{(j)}$ はブロック j における観測雑音ベクトルであり、 W_N は推定すべき未知系のパラメータとする。また、パラメータ推定のモデルは図 4.1 を用いる。

上記の定義に基づき GM-BOP アルゴリズムの計算手順は、次のように表される。ブロック j において以下の手順で計算を行う。

準備として

$$\theta_N(1)^{(j)} = h_N^{(j)} \quad (6.7)$$

$$c_N(1)^{(j)} = g_N(1)^{(j)} \quad (6.8)$$

とおく。ここで $h_N^{(j)}$ はブロック $(j-1)$ の最適係数ベクトルである。

以下の式を $i = 1 \sim m (m \leq r)$ 回繰り返す。

$$y_r(i)^{(j)} = A_{r,N}^{(j)} \theta_N(i)^{(j)} \quad (6.9)$$

$$e_r(i)^{(j)} = z_r^{(j)} - y_r(i)^{(j)} \quad (6.10)$$

$$g_N(i)^{(j)} = -2A_{r,N}^{(j)T} e_r(i)^{(j)} \quad (6.11)$$

$$c_N(i)^{(j)} = g_N(i)^{(j)} - \sum_{\ell=1}^{i-1} \frac{\langle c_N(\ell)^{(j)}, g_N(i)^{(j)} \rangle}{\langle c_N(\ell)^{(j)}, c_N(\ell)^{(j)} \rangle} c_N(\ell)^{(j)} \quad (6.12)$$

$$\theta_N(i+1)^{(j)} = \theta_N(i)^{(j)} - 2 \frac{\langle e_r(i)^{(j)}, e_r(i)^{(j)} \rangle}{\langle c_N(i)^{(j)}, c_N(i)^{(j)} \rangle} c_N(i)^{(j)} \quad (6.13)$$

最後に $h_N^{(j+1)} = \theta_N(m+1)^{(j)}$ と置き次のブロックへ進む。但し、 $\langle a_N, b_N \rangle \triangleq a_N^T b_N$ である。

6.2.1 提案する方式

入力信号の定常性及びエルゴード性を仮定し、観測雑音の期待値が $E[v(k)] = 0$ とする。また、入力信号と観測雑音は無相関とする。ここで式 (6.11) の観測信号を含む項を $g_N^z(i)^{(j)}$ とおけば、これは

$$\begin{aligned} g_N^z(i)^{(j)} &= -2A_{r,N}^{(j)T} z_r^{(j)} \\ &= -2 \sum_{\ell=1}^r x_N(\ell)^{(j)} z(\ell)^{(j)} \\ &= -2 \sum_{\ell=1}^r x_N(\ell)^{(j)} d(\ell)^{(j)} - 2 \sum_{\ell=1}^r x_N(\ell)^{(j)} v(\ell)^{(j)} \end{aligned} \quad (6.14)$$

と表すことができる。上記の仮定に基づき、 $r \rightarrow \infty$ とすれば式 (6.14) の右辺第 2 項は零に収束する。このことから、ブロック内における r 個のデータの平均により

$$\sum_{\ell=1}^r x_N(\ell)^{(j)} z(\ell)^{(j)} \simeq \sum_{\ell=1}^r x_N(\ell)^{(j)} d(\ell) \quad (6.15)$$

を仮定することができる。従って、

$$A_{r,N}^{(j)T} z_r^{(j)} \simeq A_{r,N}^{(j)T} d_r^{(j)} \quad (6.16)$$

が成り立つことも仮定できる。よって、式 (6.16) を考慮すれば GM-BOP アルゴリズムの計算手順 (6.7)~(6.13) において観測雑音の影響を受けるのは式 (6.13) の $\|e_r(i)^{(j)}\|^2$ である。ここで $\|e_r(i)^{(j)}\|^2$ を展開すると

$$\begin{aligned} \|e_r(i)^{(j)}\|^2 &= \|d_r^{(j)} + v_r^{(j)} - y_r(i)^{(j)}\|^2 \\ &= \|d_r^{(j)}\|^2 + \|v_r^{(j)}\|^2 + \|y_r(i)^{(j)}\|^2 \\ &\quad + 2[d_r^{(j)T} v_r^{(j)} - d_r^{(j)T} y_r(i)^{(j)} - v_r^{(j)T} y_r(i)^{(j)}] \end{aligned} \quad (6.17)$$

となる。式 (6.17) の期待値をとると収束特性に対して観測雑音の分散 $\|v_r^{(j)}\|^2$ が影響するものと考えられる。そこで出力誤差 $\|e_r(i)^{(j)}\|^2$ を直接計算に用いるのではなく、

これの代わりにこう配ベクトル $g_N(i)^{(j)}$ を含む量を用いたアルゴリズムについて考察を行う。

ここで、ある行列 $P_{N,N}$ を用いてベクトル a_N と b_N の内積を

$$\langle a_N, b_N \rangle_{P_{N,N}} \triangleq a_N^T P_{N,N} b_N \quad (6.18)$$

と定義し [87], 式 (6.13) の $\|e_r(i)^{(j)}\|^2$ を直接計算しない形を考える。

そこで GM-BOP アルゴリズムの計算手順における式 (6.12) の右辺第二項を

$$r_N(i)^{(j)} = - \sum_{\ell=1}^{i-1} \frac{\langle c_N(\ell)^{(j)}, g_N(i)^{(j)} \rangle_{P_{N,N}}}{\langle c_N(\ell)^{(j)}, c_N(\ell)^{(j)} \rangle_{P_{N,N}}} c_N(\ell)^{(j)} \quad (6.19)$$

とおく。但し, $P_{N,N} = A_{r,N}^{(j)T} A_{r,N}^{(j)}$ である。また, 式 (6.13) はの右辺第二項は

$$s_N(i)^{(j)} = -2 \frac{c_N(i)^{(j)} c_N(i)^{(j)T}}{c_N(i)^{(j)T} c_N(i)^{(j)}} [W_N - \theta_N(i)^{(j)}] \quad (6.20)$$

で与えられる [52][53]。ここで上式の内積の計算を式 (6.18) にしたがって行うと式 (6.20) の $s_N(i)^{(j)}$ は

$$s_N(i)^{(j)} = -2 \frac{\langle c_N(i)^{(j)}, W_N - \theta_N(i)^{(j)} \rangle_{P_{N,N}}}{\langle c_N(i)^{(j)}, c_N(i)^{(j)} \rangle_{P_{N,N}}} c_N(i)^{(j)} \quad (6.21)$$

となる。従って, 式 (6.19), 式 (6.21) はそれぞれ

$$\begin{aligned} r_N(i)^{(j)} &= - \sum_{\ell=1}^{i-1} \frac{c_N(\ell)^{(j)T} A_{r,N}^{(j)T} A_{r,N}^{(j)} g_N(i)^{(j)}}{c_N(\ell)^{(j)T} A_{r,N}^{(j)T} A_{r,N}^{(j)} c_N(\ell)^{(j)}} c_N(\ell)^{(j)} \\ &= - \sum_{\ell=1}^{i-1} \frac{p_r(\ell)^{(j)T} q_r(i)^{(j)}}{\|p_r(\ell)^{(j)}\|^2} c_N(\ell)^{(j)} \end{aligned} \quad (6.22)$$

$$\begin{aligned} s_N(i)^{(j)} &= -2 \frac{c_N(i)^{(j)} c_N(i)^{(j)T} A_{r,N}^{(j)T} A_{r,N}^{(j)}}{c_N(i)^{(j)T} A_{r,N}^{(j)T} A_{r,N}^{(j)} c_N(i)^{(j)}} [W_N - \theta_N(i)^{(j)}] \\ &= \frac{p_r(i)^{(j)T} e_r(i)^{(j)}}{2\|p_r(i)^{(j)}\|^2} c_N(i)^{(j)} \end{aligned} \quad (6.23)$$

となる。但し,

$$p_r(i)^{(j)} = A_{r,N}^{(j)} c_N(i)^{(j)} \quad (6.24)$$

$$q_r(i)^{(j)} = A_{r,N}^{(j)} g_N(i)^{(j)} \quad (6.25)$$

である。式 (6.23), 式 (6.24) より, フィルタ修正式における誤差ベクトルのノルムの自乗の計算が, こう配ベクトルを用いた式で与えられることがわかる。従って式 (6.13) 及び式 (6.12) の右辺第 2 項をそれぞれ式 (6.22), 式 (6.23) で代用することにより本方式が得られる。

6.3 コンピュータ・シミュレーション

本手法の有効性を示すためにコンピュータ・シミュレーションを行う。推定精度を表す評価量として

$$NEE = 10 \log_{10} \frac{\|W_N - h_N^{(j)}\|^2}{\|W_N\|^2} \quad [\text{dB}] \quad (6.26)$$

を用いた。また S/N 比は他章と同様に

$$\text{SNR} = 10 \log_{10} \frac{E[d(i)^2]}{E[v(i)^2]} \quad (6.27)$$

により算出し, SNR は 30dB としている。

入力信号 $x(k)$ として平均値零, 分散 1/48 の正規乱数を白色信号, 白色信号をフィルタ $H(z)$

$$H(z) = \frac{1}{1 - 0.9z^{-1}} \quad (6.28)$$

に通した信号を有色信号として用いた。条件 a, c は入力信号として白色信号を, 条件 b は有色信号を入力し, 観測雑音には白色信号を用いている。未知系, 推定系のインパルス応答はともに $N = 20$ とし, ブロック長及び繰り返し回数は $r = 20, m = 3$ とした。但し, 条件 c ではブロック長を $r = 5, r = 40$ と変化させた場合の収束特性を示している。図中における結果は 10 回試行の平均を示している。

図 6.1 及び図 6.2 より, 白色信号入力時において, 本方式は GM-BOP アルゴリズムとほぼ同等の収束速度を得られているが, 有色信号入力時では, やや収束速度が劣化していることがわかる。しかしながら, 推定精度においては本方式の方が GM-BOP アルゴリズムよりも良好な特性が得られている。また, 図 6.3 より本方式はブロック長 r が長いほど推定精度が向上していることがわかる。

6.4 むすび

本章では、GM-BOP アルゴリズムの係数更新式における出力誤差ベクトルのノルムの自乗に、観測雑音の分散が含まれるれていることに着目し、これの代わりに観測雑音の平均化が行われるこう配ベクトルを含む量を用いることにより、収束特性への影響を軽減し従来法よりも良好な収束特性を得られる方式を提案した。本方式では、ブロック長を長くすることにより、こう配ベクトルの生成による観測雑音の平均化に用いられるサンプル数が多くなり、推定精度が向上する。

今後は、本手法による推定精度向上に対する理論解析を行うと共に、有色信号入力時における収束速度の改善について検討を行う予定である。

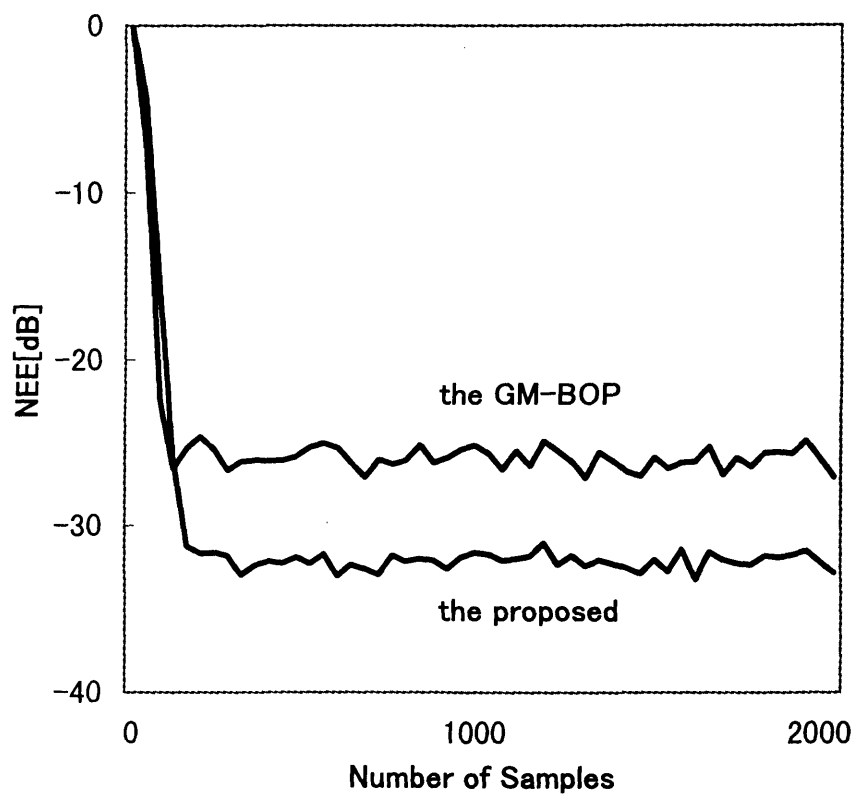


図 6.1: 収束特性の比較 (条件 a の場合)

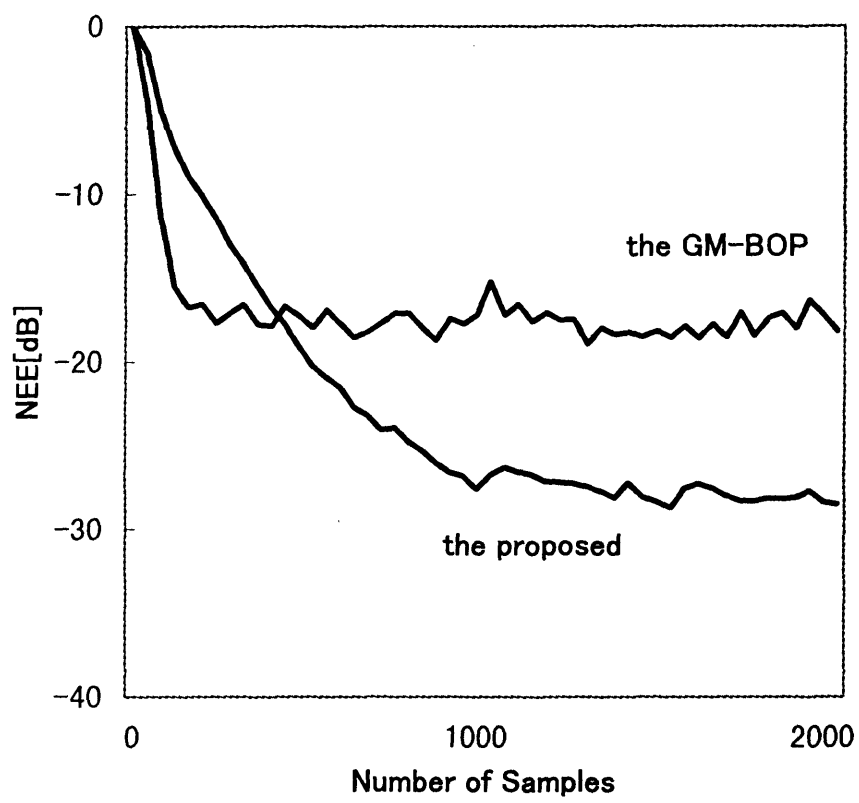


図 6.2: 収束特性の比較 (条件 b の場合)

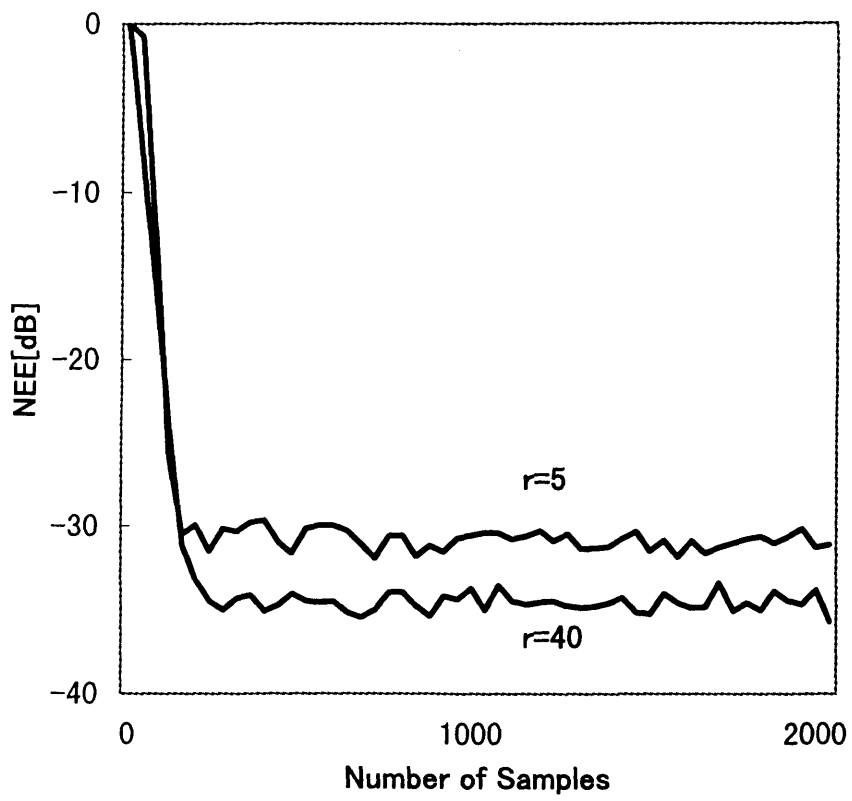


図 6.3: 収束特性の比較 (条件 c の場合)

第7章

結論

本論文は、適応信号処理において適応フィルタの係数修正を行う適応アルゴリズムの性能向上という点に主眼をおいて、その研究結果をまとめたものである。

適応信号処理では適応フィルタの修正を行う適応アルゴリズムに、有色信号が入力された場合においても高速な収束速度を示す、あるいは未知系出力に観測雑音が重畳されている場合においても良好な推定精度が得られるという性能を要求する。また、ハードウェア構成上高速な実行が可能である、構成が比較的シンプルであるということも望まれる。本論文ではこのような背景を鑑みて上記のような性能を実現するための適応アルゴリズムについて提案した。

まず、ブロック直交射影アルゴリズムが Moore-Penrose 型一般逆行列で表されることに着目し、この逆行列を効率よく解く手法として、次数更新型 UD 分解法によるブロック直交射影アルゴリズムを提案した。Moore-Penrose 型一般逆行列に含まれる入力状態行列の自己相関行列の逆行列を UD 分解により逐次的に解くことにより、大幅な演算量の軽減化を図ることができた。Moore-Penrose 型一般逆行列を特異値分解により解けば、必要となる演算量は $O(N^3)$ であるが、本方式は $O(Nr)$ である。また、シミュレーションにより本方式と直交射影行列を用いた方式との比較を行い、本方式の方が演算量の点で有利であることを示した。更に、本方式は入力信号に対して逐次的な処理が可能であり、本方式を高速に処理できるアーキテクチャの開発が期待される。

次に、ブロック直交射影アルゴリズムは比較的収束速度と演算量のバランスがと

れた方式として提案されているが、直交射影アルゴリズムは未知系出力に観測雑音が存在する場合、推定精度にその影響を受けやすいということが知られていることから、Moore-Penrose 型一般逆行列で表される直交射影アルゴリズムの一般論として、特にこの逆行列が陽に表れる直交射影アルゴリズムの雑音が存在する場合における推定精度を改善する方式の提案を行った。本方式は直交射影アルゴリズム全般に適用可能であると考えられるが、本論文ではブロック直交射影アルゴリズムについて定式化を行った。また、一般に適応アルゴリズムにおける雑音の影響を軽減する目的として提案されている方式は、白色信号に対しての解析に基づいている、雑音の分散が必要であるという条件が必要であったりするが、本方式は入力信号の白色性、有色性を考慮する必要が無く、入力信号と観測雑音が無相関で、雑音の平均値が零であるという条件のみで適用可能である。提案法は修正方向ベクトルの時間平均に基礎をおく方式であり、観測雑音を含む成分とそれ以外の成分に分割し、それぞれを独立に平均化することにより良好な精度が得られることを確認した。

続いて、ブロック直交射影アルゴリズムの一実現法である“こう配法に基づく高速な適応アルゴリズム”の性能改善という観点から、収束速度の向上と演算量の軽減を考慮した方式、及び観測雑音に対しロバストな方式をそれぞれ独立に提案した。前者においては、収束の高速性を低減させぬようブロック処理として定式化されたこう配法に基づく高速な適応アルゴリズムをアフィン射影算法と同様の逐次処理に変換した。また、この処理により1サンプル当たりの演算量が増大したが、データの運用法を改良することにより、従来法よりも演算量の軽減化が図られたことが確認できた。また、後者においては、こう配法に基づく高速な適応アルゴリズムのフィルタ係数更新式中で与えられている出力誤差ノルムの自乗をこう配ベクトルを含む量により表した。これにより、修正量に含まれる観測雑音の影響を軽減され、従来法よりも良好な推定精度が得られることを示した。本方式におけるブロック長を長くとした場合に、より推定精度が向上することが確認できた。

以上、本論文では適応信号処理に有用である適応アルゴリズムの提案及び改良を行い議論した。今後は、これらの方式を適応ノイズキャンセラなどのアプリケーションに用いた場合の特性の解析や、シストリックアレイを含むこれらの適応アルゴリ

ズムを効率的に実行するアーキテクチャの開発などを行う予定である。また、ニューラルネットやブラインド等化などへの応用についても今後の研究課題としたい。

謝辞

本研究をまとめるにあたり，終始懇切なご指導をいただいた九州芸術工科大学芸術工学部画像設計学科浦濱喜一教授に対し心から感謝の意を表す。

また，筆者の福岡工業大学在学中にデジタル信号処理に関するテーマを与えて下さって以来，本研究を進めるにあたり有益なご教示を賜った福岡工業大学情報工学部情報システム工学科古川利博教授に深く感謝の意を表す。

特に本研究は浦濱喜一教授，古川利博教授のご指導の下に成り立ったものであり，両教授のご援助なしでは成しえなかったと考えられる。

また，日頃から多くの貴重な御教示を賜った千葉工業大学工学部電気工学科久保田一教授に深く感謝の意を表す。

更に，本論文をまとめるに当たり，有益な御教示を頂いた九州芸術工科大学芸術工学部画像設計学科滝山龍三教授，福島重廣教授に深く感謝の意を表す。

加えて，九州芸術工科大学芸術工学部画像設計学科坂本博康講師，小野直樹助手には有益なご助言ご討論を頂きここに感謝の意を表す。また，井上光平氏，北岡義弘氏をはじめとする九州芸術工科大学芸術工学部画像設計学科画像設計研究室，福岡工業大学情報工学部情報システム工学科古川研究室の方々には研究を進めるにあたりご協力して頂いたことに感謝の意を表す。

最後に，本研究を遂行するに際してご指導，ご協力頂いた方々に深く感謝の意を表す。

参考文献

- [1] 井上伸雄, 辻井重男, 他, “デジタル処理の応用,” 電子通信学会, 1981.
- [2] B. Gold and C. M. Rader, “*Digital Processing of Signals*,” McGraw-Hill, 1969.
- [3] L. R. Rabiner and C. M. Rader, “*Digital Signal Processing*,” IEEE Press, 1972.
- [4] A. V. Oppenheim and R. W. Schaffer, “*Digital Signal Processing*,” Prentice-Hall, 1975.
- [5] L. R. Labiner and B. Gold, “*Theory and Applications of Digital Signal Processing*,” McGraw-Hill, 1975.
- [6] 宮川洋, 城戸健一, 他, “デジタル信号処理,” 電子通信学会, 1975.
- [7] 木村英紀, “デジタル信号処理と制御,” 昭晃堂, 1982.
- [8] 古井貞熙, “デジタル音声処理,” 東海大出版会, 1985.
- [9] 辻井重男, 久保田一, “デジタル信号処理,” オーム社, 1986.
- [10] 雨宮好文, 佐藤幸男, “信号処理入門,” オーム社, 1987.
- [11] 有本卓, “信号・画像のデジタル処理,” 産業図書, 1980.
- [12] 谷萩隆嗣, “デジタル信号処理と基礎理論,” コロナ社, 1996.
- [13] 谷萩隆嗣, “マルチメディアとデジタル信号処理,” コロナ社, 1997.

- [14] S. Haykin, "Introduction to Adaptive Filters," Macmillan Publishing Company, 1984.
- [15] N. Wiener, "Extrapolation, Interpolation and Smoothing of Stationary Time Series," MIT Press, 1949.
- [16] N. Wiener, "Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications," Wiley, 1949.
- [17] Widrow B. and Hoff M. E., "Adaptive switching circuit," IREWESCON Conv. Rec., pp.96-104, 1960.
- [18] J. Nagumo and A. Noda, "A Learning method for system identification," IEEE Trans. Autom. Control, vol.AC-12, no.3, pp.282-287, 1967.
- [19] 野田淳彦, 南雲仁一, "システムの学習的同定法," 計測制御, 7-9, pp.597-605, Sept. 1968.
- [20] G.J. Bierman, "Factorization Methods for Discrete Sequential Estimation," Academic Press. New York, 1977.
- [21] L. Ljung and T. Soderstrom, "Theory and Practice of Recursive Identification," MIT Press, 1983.
- [22] R. E. Kalman, "A new approach to linear filtering and prediction problems," J. Basic Eng., Trans. ASME, vol.82D, 1, pp.35-45, March 1960.
- [23] R. E. Kalman and R.S. Bucy, "New results in linear filtering and prediction theory approach to linear filtering and prediction problems," J. Basic Eng., Trans. ASME, vol.83D, 1, pp.95-108, March 1961.
- [24] R. E. Kalman, "New methods in Wiener filtering theory," Proc. 1st Symposium on Engineering Applications of Random Function Theory and Probability, J. L. Bogdanoff and F. Kozin (eds.), pp.270-288, Wiley 1963.

- [25] 相良節夫, 秋月影雄, 中溝高好, 片山徹, “システム同定,” 計測自動制御学会, 1981.
- [26] 有本草, “カルマン・フィルター,” 産業図書, 1977.
- [27] 中野道雄, 西山清, “パソコンで解くカルマンフィルタ,” 丸善株式会社, 1993.
- [28] R. W. Lucky, “Automatic equalization for digital communication,” Bell Sys. Tech. J., vol.44, 4, pp.547-588, April 1965.
- [29] R. W. Lucky, “Techniques for adaptive equalization of digital communication systems,” Bell Sys. Tech. J., vol.45, 2, pp.255-286, Feb. 1966.
- [30] M. M. Sondhi, “An adaptive echo canceler,” Bell Sys. Tech. J., vol.46, pp.497-511, 1967.
- [31] A. Gersho, “Adaptive equalization of highly dispersive channels for data transmission,” Bell Sys. Tech. J., vol.48, pp.55-71, 1969.
- [32] B. Widrow and J. R. Glover, “Adaptive noise cancelling : principles and applications,” Proc. IEEE, vol.63, no.12, pp.1962-1716, Dec. 1975.
- [33] 久保田一, 古川利博, 板倉秀清, “前処理を含むノイズキャンセラのアプローチとその性能評価,” 信学論 (A), vol.J69-A, no.5, pp.584-591, May 1986.
- [34] 大石邦夫, 久保田一, “周波数領域における自動等価器適応アルゴリズムの簡略化について,” 信学論 (A), vol.J70-A, no.3, pp.570-572, march 1987.
- [35] 大石邦夫, 久保田一, “周波数領域における評価量を用いた自動等価器高速アルゴリズム,” 信学論 (A), vol.J71-A, no.3, pp.867-874, march 1988.
- [36] 古川利博, 久保田一, 松本浩樹, “高信頼性を有する信号処理システムの提案とその性能評価,” 信学論 (A), vol.J73-A, no.1, pp.26-34, Jan. 1991.

- [37] G. A. Clark, S. K. Mitra and S. R. Parker, "Block implementation of adaptive digital filters," *IEEE Trans. Circuit & Systems*, vol.CAS-28, pp.548-592, 1981.
- [38] G. A. Clark, S. K. Mitra and S. R. Parker, "A unified approach to time and frequency-domain realization of FIR adaptive digital filters," *IEEE Trans. Acoust., Speech & Signal Process.*, ASSP-31, 5, pp.1073-1083, Oct. 1983.
- [39] T. Walzman and M. Schwartz, "Automatic equalization using the discrete frequency domain," *IEEE Trans., Inf. Theory*, IT-19, 1, pp.59-68, Jan. 1973.
- [40] G. Picchi and G. Prati, "Self-orthogonalizing adaptive equalization in the discrete frequency domain," *IEEE Trans. Commun.*, COM-32, 4, pp.371-379, April 1984.
- [41] W. B. Mikhael and F. H. Wu, "Fast algorithm for block FIR adaptive digital filtering," *IEEE Trans. Circuit & Systems*, vol.CAS-34, pp.1152-1160, 1987.
- [42] 古川利博, 久保田一: "ブロック学習同定法の提案と射影アルゴリズムの統一的理解," *信学技報*, vol.CAS-87-221, Dec. 1987.
- [43] 雛元孝夫, 前田禎男, "拡張された学習同定法," *電学論*, vol.95-C, pp.227-234, Oct. 1975.
- [44] 尾関和彦, 梅田哲夫, "アフィン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)*, vol.J67-A, no.2, pp.126-132, Feb. 1984.
- [45] 古川利博, 久保田一, 辻井重男, "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)*, vol.J71-A, no.12, pp.2132-2146, Dec. 1988.
- [46] 大石邦夫, 久保田一, 小野田真穂樹, "高速なシステム同定のためのブロック適応アルゴリズム," *信学論 (A)*, vol.J71-A, no.11, pp.2018-2026, Nov. 1988.
- [47] 古川利博, 久保田一, 辻井重男, "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)*, vol.J72-A, no.7, pp.1069-1076, July 1989.

- [48] 大石邦夫, 福本昌弘, 久保田一, “直交射影行列を用いた安定なブロック適応アルゴリズム,” 信学論 (A), vol.J75-A, no.9, pp.1450-1457, Sept. 1992.
- [49] 古川利博, 久保田一, 木村祐司, “多次元部分空間への直交射影アルゴリズムとその運用に関する一考察,” 信学技報, DSP89-24, pp.23-29, 1989.
- [50] 古川利博, 久保田一, 木村祐司, “多次元部分空間への直交射影アルゴリズムの諸性質と運用に関する考察,” 信学論 (A)
- [51] 柳井晴夫, 竹内啓, “射影行列・一般逆行列・特異値分解,” 東京大学出版会, 1983.
- [52] 古川利博, 前田常憲, 久保田一, “連立方程式の新しい反復解法と適応信号処理への応用に関する考察,” 信学論 (A), vol.J78-A, no.2, pp.214-226, Feb.1995.
- [53] 古川利博, 前田恒憲, 久保田一, “勾配法と直交射影行列に基づく高速な適応アルゴリズム,” 電学論 (C), vol.115-C, 3, pp.345-353, March 1995.
- [54] R. Fletcher and C. Reeves, “Function minimization by conjugate gradient method,” *Computer Journal*, 7, pp.149-154, 1964.
- [55] 清水清孝, “制約条件のないシステムの最適化,” 計測と制御, 9, 6, June 1970.
- [56] 戸川隼人, “共役勾配法,” 教育出版, 1977.
- [57] 茨木俊秀, 福島雅夫, “情報数学講座第 14 巻最適化の手法,” 共立出版, 1993.
- [58] Y. S. Kung, “*VLSI Array Processors*,” Prentice-Hall, 1988.
- [59] 辻井重男, 久保田一, 古川利博, 趙晋輝, “適応信号処理,” 昭晃堂, 1995.
- [60] 谷口隆行, 南重信, “多地点通信会議用疑似ステレオエコーキャンセラの実現,” 第 10 回 DSP シンポジウム講演論文集, pp.129-134, Nov. 1995.
- [61] 木許雅則, 古川利博, “入力信号の直交化に基づくマルチチャンネルエコーキャンセラ,” 第 11 回 DSP シンポジウム講演論文集, pp.365-368, Nov. 1996.

- [62] 木許雅則, 古川利博, “高速な収束速度を有する適応アルゴリズムを用いたマルチチャンネルエコーキャンセラの一提案,” 第12回DSPシンポジウム講演論文集, pp.527-530, Nov. 1997.
- [63] 松藤秀樹, 古川利博, “補助フィルタを用いたコンパクトステレオエコーキャンセラ,” 第12回DSPシンポジウム講演論文集, pp.537-541, Nov. 1997.
- [64] 足立修一, 佐野久, “能動騒音制御におけるシステム同定の役割,” システム/制御/情報, vol41, no.2, pp.64-72, 1997.
- [65] S. J. Elliot and P. A. Nelson, “Active noise control,” IEEE Signal Processing Magazine, pp.13-35, Oct. 1993.
- [66] 小郷寛, 美多勉, “システム制御理論入門,” 実教出版株式会社, 1979.
- [67] C. F. N. Cowan and P. M. Grant, “Adaptive Filters,” Prentice-Hall, 1985.
- [68] S. Yoshimoto, T. Furukawa and H. Kubota, “A consideration on block orthogonal projection algorithm using order recursive UD factorization method,” Proc. IEEE Inter. Workshop on ISPACS, pp.179-184, Oct. 1993.
- [69] T. Furukawa, S. Yoshimoto and H. Kubota, “A new block adaptive algorithm using order recursive UD factorization method,” Proc. IEEE Inter. ISCAS, vol.2, pp.369-372, May 1994.
- [70] L. Ljung, M. Morf and D. Falconer, “Fast calculation of gain matrices for recursive estimation schemes,” Int.J.Control, vol.27, no.1, pp.1-19, Jan. 1978.
- [71] 谷萩隆嗣, “デジタル信号処理の理論 3,” コロナ社, 1986.
- [72] 坪川宏, 久保田一, 辻井重男, “UD分解を用いたRLSアルゴリズムの有限語長浮動小数点演算誤差解析,” 信学論(A), vol.J73-A, 6, pp.1075-1082, June 1990.

- [73] 菊池義浩, 齊藤恒雄, “入力信号相関行列の特異値分解に基づく適応フィルタ,” 信学論 (A), vol.J71-A, 6, pp.1266-1274, June 1988.
- [74] 高橋潔, 笹瀬巖, 森真作, “収束遅延を改善するための観測雑音の推定方法と可変ステップアルゴリズムの提案,” 信学論 (A), vol.J76-A, 12, pp.1704-1713, Dec. 1993.
- [75] J. Tanpreeyachaya, I. Takumi and M. Hata, “Performance improvement of variable stepsize NLMS,” IEICE Trans., vol.E78-A, no.8, pp.905-914, Aug. 1995.
- [76] S. Kinjo and H. Ochi, “A new robust block adaptive filter for colored signal input,” IEICE Trans., vol.E78-A, no.3, pp.437-439, March 1995.
- [77] 福本昌弘, 久保田一, 辻井重男, “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), vol.J77-A, no.1, pp.16-23, Jan. 1994.
- [78] 福本昌弘, 金井剛志, 久保田一, 辻井重男, “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における安定性の改善,” 信学論 (A), vol.J81-A, no.3, pp.318-327, March. 1998.
- [79] 酒井英昭, 他, “信号処理,” オーム社, 1998.
- [80] 古川利博, 吉本定伸, 久保田一, “逐次 UD 分解による直交射影アルゴリズム,” 信学論 (A), vol.J80-A, no.11, pp.1910-1921, Nov. 1997.
- [81] 田島稔, 小牧和雄, “最小二乗法の理論とその応用,” 東洋書店, 1986.
- [82] R. W. Harris, D. M. Chabries and F. A. Bishop, “A variable step (VS) adaptive filter algorithm,” IEEE Trans. Acoust., Speech & Signal Process., ASSP-34, 2, pp.309-316, April 1986.
- [83] 沢辺知子, 松井実, 萩原将文, 中川正雄, “可変ステップ適応デジタルフィルタの高速化,” 信学論 (A), vol.J70-A, no.12, pp.1858-11860, Dec. 1987.

- [84] 田口亮, 浜田望, “可変ステップ型学習同定法の一提案,” 信学論 (A), vol.J71-A, no.8, pp.1663-1665, Aug. 1988.
- [85] 加藤昭偉, 久保田一, “ファジー制御を用いた可変ステップアルゴリズムの提案,” 信学論 (A), vol.J73-A, no.11, pp.1844-1850, Nov. 1990.
- [86] 古川利博, 久保田一, 柴田岩夫, “観測雑音を考慮したブロック直交射影アルゴリズムの特性改善に関する一考察” 電学論 (C) ,vol.111-C, no.1, pp.8-16, Jan. 1991.
- [87] Carlos E. Davila, “A stochastic newton algorithm with data-adaptive step size,” IEEE Trans. ASSP, vol.ASSP-38, no.10, pp.1796-1798, Oct. 1990.

目 次

2.1 システム同定のモデル	12
2.2 J に関する等高線	14
2.3 学習同定法の幾何学的関係	16
2.4 信号の相関が強い場合	18
2.5 信号の相関が弱い場合	19
2.6 アフィン射影アルゴリズムの幾何学的図	21
2.7 拡張された学習同定法の幾何学的図	22
2.8 直交射影行列による部分空間への直交射影	23
2.9 BOP アルゴリズムと AP アルゴリズムのデータの運用法	25
2.10 BOP アルゴリズムのデータの運用法	26
2.11 BOP アルゴリズムの幾何学的図	27
2.12 共役こう配法の幾何学的図	29
2.13 適応ノイズキャンセラ	32
2.14 適応ノイズキャンセラにおけるパラメータ推定	33
2.15 デジタル伝送系	34
2.16 自動等化器におけるパラメータ推定	34
2.17 適応エコーキャンセラの例 1	35
2.18 適応エコーキャンセラの例 2	36
2.19 適応エコーキャンセラにおけるパラメータ推定	36
2.20 アクティブノイズコントローラ	37
2.21 アクティブノイズコントローラにおけるパラメータ推定	38

3.1	演算量の比較	54
3.2	適応フィルタによるパラメータ推定モデル	55
3.3	本方式の収束特性 (条件 (a) の場合)	59
3.4	本方式の収束特性 (条件 (b) の場合)	60
3.5	本方式と OPM-BOPA の収束特性比較 (条件 (b) の場合)	61
3.6	MSE 特性比較 (条件 (b) の場合)	62
4.1	パラメータ推定のモデル	68
4.2	収束特性の比較 (2) の場合)	77
4.3	収束特性の比較 (1) の場合)	78
4.4	収束特性の比較 (2) の場合)	79
4.5	収束特性の比較 (1) の場合)	80
4.6	収束特性の比較 (2) の場合)	81
4.7	収束特性の比較 (2) の場合)	82
4.8	データの取り方 ($m = 3$ の時)	85
5.1	パラメータ推定のモデル	88
5.2	GM-AP アルゴリズムと共役こう配法の幾何学的関係	91
5.3	等高線と固有値の関係	92
5.4	時刻 k における修正方向ベクトルの計算 ($m = 3$ の時)	97
5.5	演算量の比較	98
5.6	収束特性の比較 ($N = 20, r = 9, m = 9$)	100
5.7	収束特性の比較 ($N = 20, r = 12, m = 12$)	101
5.8	収束特性の比較 ($N = 20, r = 15, m = 15$)	102
6.1	収束特性の比較 (条件 a の場合)	110
6.2	収束特性の比較 (条件 b の場合)	111
6.3	収束特性の比較 (条件 c の場合)	112

表 目 次

3.1	本章で提案するアルゴリズム	52
3.2	1 サンプル当たりの演算量の比較	53
5.1	GM-AP アルゴリズム	89
5.2	提案する手法	96
5.3	演算量の比較	97