

FPGA を用いたアフィン射影型適応デジタルフィルタの 高性能アーキテクチャ

High-Performance Architecture for Affine Projection Adaptive Filter Using FPGA

○石井勇也, 恒川佳隆

○ Yuya Ishii, Yoshitaka Tsunekawa

岩手大学

Iwate University

キーワード : FPGA(Field Programmable Gate Array), アフィン射影アルゴリズム (Affine Projection Algorithm), デイレードアップデート (Delayed Update), 高性能アーキテクチャ (High-Performance Architecture), 桁上げ伝播加算器 (Ripple Carry Adder)

連絡先 : 〒 020-8551 盛岡市上田 4-3-5 岩手大学大学院総合科学研究科理工学専攻電気電子通信コース
恒川・佐藤研究室 石井勇也, Tel:(019)621-6468, Fax:(019)621-6468, E-mail: tsune@iwate-u.ac.jp

1. はじめに

適応デジタルフィルタはデジタル信号処理技術の 1 つであり, 通信システムや, 計測制御, オーディオシステムなど幅広い分野で広く利用されている. 近年浸透が著しい移動通信システムにおいて, エコーキャンセラやノイズキャンセラ, 適応等化器などに用いられており, これらは移動通信システムの中でも, 重要な役割を担っている. 求められる性能としては, 良好な収束特性や低消費電力, 高速性, 有色性など様々な性能が求められる. しかし, それらを同時に満たすのは困難であるため, できる限り良好な性能を同時に実現できる高性能アーキテクチャが必要とされている.

代表的な適応フィルタで知られる, NLMS 適応フィルタ (NLMS-ADF) は比較的簡単に演算

が出来るため広く利用されている. しかし, 有色信号入力時, 収束速度が劣化する問題点を持つ. それに対し, 有色性に強く, 収束速度が早い適応フィルタとして, RLS 適応デジタルフィルタ (RLS-ADF) が挙げられる. しかし, 演算量が多くハードウェア量が膨大になる問題点をもつ. そこで, 本報告では NLMS-ADF と RLS-ADF の中間の性質を持つアフィン射影型適応フィルタ (APA-ADF) を用いる.

一方, デジタルシステムの実現手段として, 近年, FPGA (Field Programmable gate array) が注目されている. FPGA は書き換え可能なロジック・デバイスであり, 設定の変更・開発が容易であるなどの利点がある. そのため, 通信基地局, 航空宇宙分野など高度な処理を行う機器から, ディスプレイ, プロジェクタ, 携帯端末など生活に身近な製品にまで幅広く利用され

ている。

本報告では，FPGA 実装を前提とし高性能化を行っていく．まず，APA-ADF の演算に用いる素子を共有化させ全体の素子数を削減する．次に，ディレードアップデートを用いることでサンプリングレートを向上させた高性能アーキテクチャを提案する．そして，収束特性評価と FPGA 評価により，本提案型の有効性を明らかにする．

2. アフィン射影型適応デジタルフィルタ (APA-ADF)

2.1 アフィン射影アルゴリズム

アフィン射影アルゴリズム (APA: Affine Projection Algorithm) は演算量が少ない NLMS アルゴリズムと，非常に高速に収束する RLS アルゴリズムの中間の性質を持ったアルゴリズムとして知られている．NLMS アルゴリズムが 1 時刻の誤差を最小化する手法，RLS アルゴリズムが過去の観測データすべての誤差を最小化する手法であるのに対し，このアルゴリズムでは射影次数 p を導入することで $p-1$ 時刻前までの誤差を最小化するようにフィルタ係数を更新するアルゴリズムである．

アフィン射影アルゴリズムを用いた適応デジタルフィルタ (APA-ADF) の計算式を示していく． N タップでの時刻 k における入力信号ベクトル，入力信号行列，フィルタ係数ベクトル，出力信号ベクトルを式 (1) ~ (4) に，所望信号ベクトル，誤差信号ベクトルを式 (5)~(6) に示す．

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

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

$$\mathbf{h}(k) = [h_0(k), h_1(k), \dots, h_{N-1}(k)]^T \quad (3)$$

$$\mathbf{y}(k) = [y_0(k), y_1(k), \dots, y_{N-1}(k)]^T \quad (4)$$

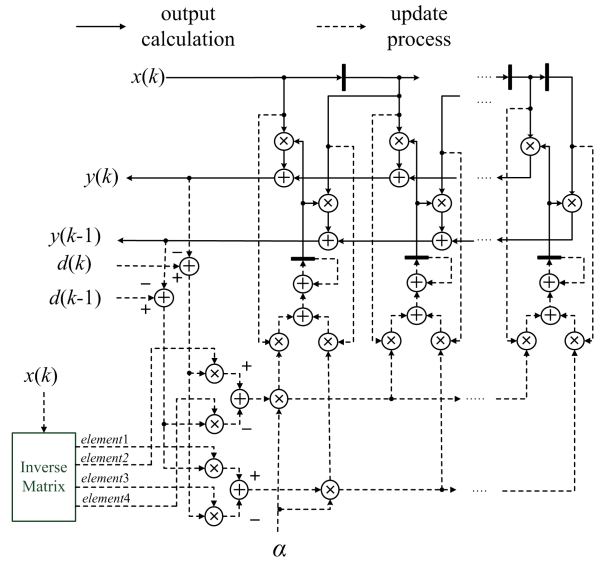


Fig. 1 Structure of APA-ADF.

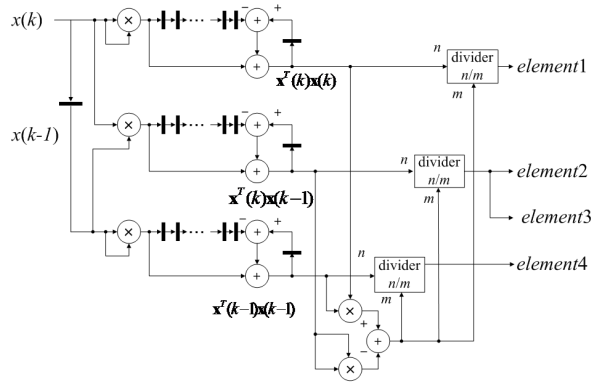


Fig. 2 Structure of Inverse Matrix.

$$\mathbf{d}(k) = [d(k), d(k-1), \dots, d(k-p+1)]^T \quad (5)$$

$$\begin{aligned} \mathbf{e}(k) &= \mathbf{d}(k) - \mathbf{X}^T(k) \mathbf{h}(k) \\ &= [e(k), e(k-1), \dots, e(k-p+1)]^T \quad (6) \end{aligned}$$

なお，収束速度と演算量の兼ね合いにより射影次数 $p=2$ の適応デジタルフィルタを用いる．

このときの APA-ADF を Fig.1 に示す．フィルタ係数の更新式を式 (7) に示す．また，式 (7) 中の $\mathbf{R}^{-1}(k)$ は式 (8) に示す入力信号行列の自己相関行列の逆行列であり，これにより入力信号電力による正規化を行っている．Fig.1 中の，逆行列演算部を Fig.2 に示す． $element1 \sim element4$ は $\mathbf{R}^{-1}(k)$ の要素とする． α はステップゲイン

と呼ばれる定数で 0 から 2 の値を取る.

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \alpha \mathbf{X}(k) \mathbf{R}^{-1}(k) \mathbf{e}(k) \quad (7)$$

$$\mathbf{R}(k) = \begin{bmatrix} \mathbf{x}^T(k) \mathbf{x}(k) & \mathbf{x}^T(k) \mathbf{x}(k-1) \\ \mathbf{x}^T(k-1) \mathbf{x}(k) & \mathbf{x}^T(k-1) \mathbf{x}(k-1) \end{bmatrix} \quad (8)$$

過去分の誤差信号を用いてフィルタ係数の更新を一括に行う為, 有色信号における収束速度の影響が抑制できる. しかし, NLMS-ADF と比べ多くの加算器と乗算器が必要となる. 従って, 高次になる程に APA-ADF はハードウェア量が膨大になる問題点がある.

2.2 積和演算分割化

演算に用いる素子を共有化し, 全体の素子数を削減する構成を提案する¹⁾. フィルタ出力演算式をそれぞれ項数 r の積和演算に分割し, 同一の素子を共有することでハードウェア量を削減する. APA-ADF の出力信号ベクトルを式 (9) に示す.

$$\mathbf{y}(k) = [y_0(k), y_1(k)]^T = \begin{bmatrix} \mathbf{x}(k) \mathbf{h}(k) \\ \mathbf{x}(k-1) \mathbf{h}(k-1) \end{bmatrix} \quad (9)$$

余りを 0 とし, タップ数 N を項数 r で割ったときの商を q とする. 出力信号ベクトルの要素 $y_0(k)$ を q 個の積和演算を足し合わせる形に変形する. なお $y_1(k)$ も同様に変形する.

$$\begin{aligned} y_0(k) &= \sum_{j=0}^{N-1} x(k-j)h_j(k) \\ &= \sum_{j=0}^{r-1} x(k-j)h_j(k) + \sum_{j=r}^{2r-1} x(k-j)h_j(k) + \\ &\quad \dots + \sum_{j=q(r-1)}^{qr-1} x(k-j)h_j(k) \quad (10) \end{aligned}$$

各項の積和演算において計算に用いる素子を共有化し, r クロックかけて全体の計算を行う. 共有化することにより, 素子数を大幅に削減す

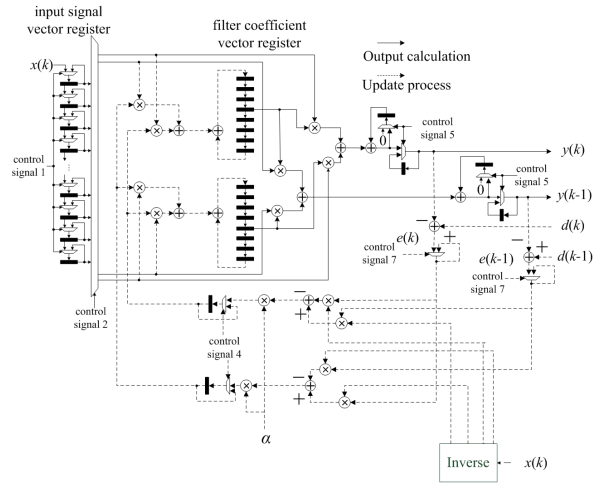


Fig. 3 Structure of APA-ADF sharing elements. することが出来る. また, 項数 r を大きく取るほどに素子が共有化されハードウェア量が削減できる. その反面, 項数 r を大きく取るほど演算に必要なクロック数が増加する. 提案型ではタップ数に関わらず項数 $r=8$ とする.

タップ数 $N = 16$ タップ, 各積和演算の項数 $r = 8$ としたときの $y_0(k)$ の式を式 (11) に示す. $y_1(k)$ も同様に処理する.

$$\begin{aligned} y_0(k) &= \sum_{j=0}^{15} x(k-j)h_j(k) \\ &= \sum_{j=0}^7 x(k-j)h_j(k) + \sum_{j=8}^{15} x(k-j)h_j(k) \quad (11) \end{aligned}$$

素子の共有化を行った APA-ADF の構成を Fig.3 に示す. この構成を取り入れることでフィルタ出力演算における加算器, 乗算器を大幅に減らすことができ全体の素子数を削減することができる. しかし, 通常 1 回のクロックで演算を行うのに対して, この構成では r クロックをかけて出力を行う為, 処理速度が遅くなりサンプリングレートが劣化する. 従って, サンプリングレートの向上が求められる.

2.3 ディレードアップデート

サンプリングレートを向上するために, ディレードアップデートを用いる. なお, ディレー

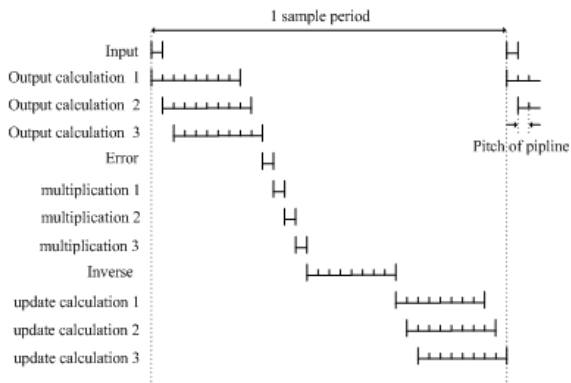


Fig. 4 Timing chart of 1 sample period.

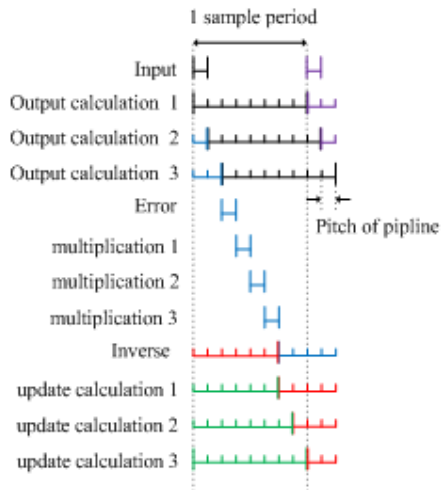


Fig. 5 Timing chart of 1 sample period (Using delayed update).

ドアップデートを用いる前処理として加算器1個、乗算器1個につきレジスタを挟むことでパイプライン化を行っている。

ディレードアップデートとはフィルタ系数更新に m 時刻前の誤差信号と入力信号ベクトルを用いることで出力計算と更新動作の並列化を可能とする手法である。並列化を行うことで演算に必要なクロック数を削減できる。ディレードアップデートを用いる問題点として、過去の信号を用いる為、収束特性の劣化が挙げられる。このときのフィルタ系数更新式を式 (12) に示す。

$$\begin{aligned} \mathbf{h}(k+1) = \\ \mathbf{h}(k) + \alpha \mathbf{X}(k-m) \mathbf{R}^{-1}(k-m) \mathbf{e}(k-m) \end{aligned} \quad (12)$$

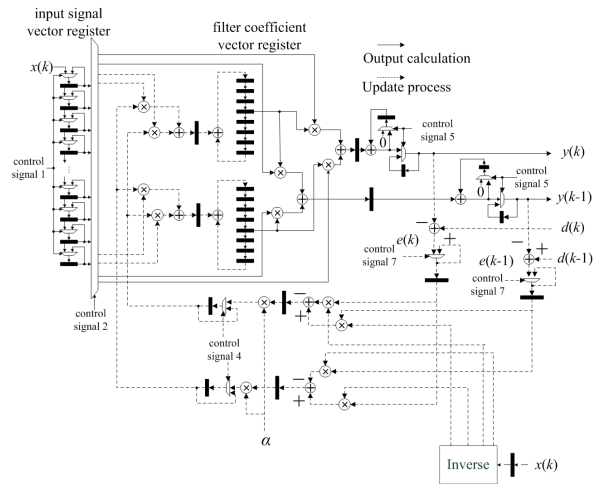


Fig. 6 Structure of proposed architecture.

提案型の構成において、 r クロックで1回の適応が完了するよう各素子間にレジスタを挿入することで信号が送られるタイミングを調整する。

$N = 16$ タップで $r = 8$ において、ディレードアップデートを用いる前のタイミングチャートを Fig.4 に、ディレードアップデートを用いたタイミングチャートを Fig.5 に示す。 $N = 16$ タップで $r = 8$ の場合の遅延は $m = 1$ 時刻となる。なお、タップ数 $N = 2^{11}$ までは $m = 1$ 、以降タップ数が2の $3r$ 乗増化する毎に m を1ずつ増加させる。ディレードアップデートを用いたアフィン射影型適応デジタルフィルタ (DAPA-ADF) を Fig.6 に示す。これによりクロック数を削減しサンプリングレートが向上される。

3. 収束特性評価

システム同定問題を用いた収束特性評価を行う。 Fig.7 にシステム同定問題のシュミレーションモデルを示す。 Fig.7 は未知システムのフィルタ系数を推定するシステムであり、未知システムと適応フィルタの誤差を用いてフィルタ系数の更新を行っている。

NLMS-ADF と DAPA-ADF に対して、収束特性評価を行っていく。両者ともステップゲイン

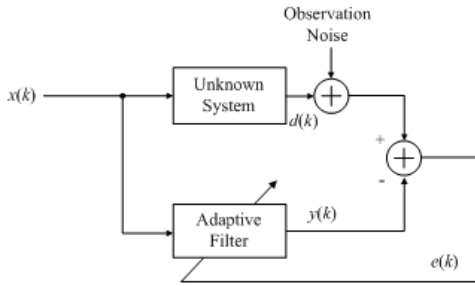


Fig. 7 Model of identification system.

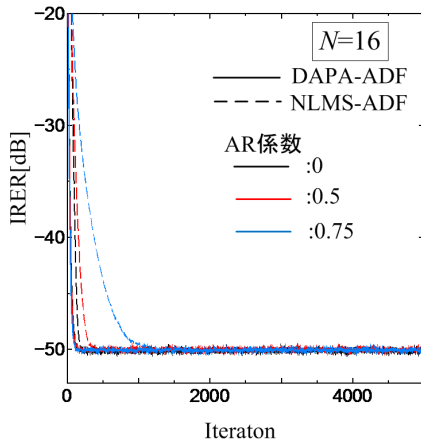


Fig. 8 Computation of the convergence(N=16).

により収束の速さと推定の精度が変化する．そこで、両者を確認する方法として IRER(インパルス応答誤差比:Impulse Response Error Ratio)を用いる．IRERは未知システムと適応フィルタにおけるフィルタ係数を用いた評価方法となっている． k を時刻， N をタップ数， $w_i(k)$ を未知システムのフィルタ係数， $h_i(k)$ を適応フィルタのフィルタ係数としたときの IRER の計算を式 (13) に示す．

$$IRER(k) = E \left[10 \log_{10} \sum_{i=0}^{N-1} \frac{(w_i(k) - h_i(k))^2}{w_i^2(k)} \right] \quad (13)$$

条件として、未知システムはタップ数16,64の FIR フィルタ、入力信号は平均0.00、分散0.05の白色信号、係数0.50と0.75の一次AR過程による有色信号とし、観測雑音は-50dBの入力信号に無相関の白色ガウス雑音を加えた。また、ステップゲインはIRERが-50dBを示す値を選択し、評価を行う。タップ数毎AR係数別でNLMS-ADF

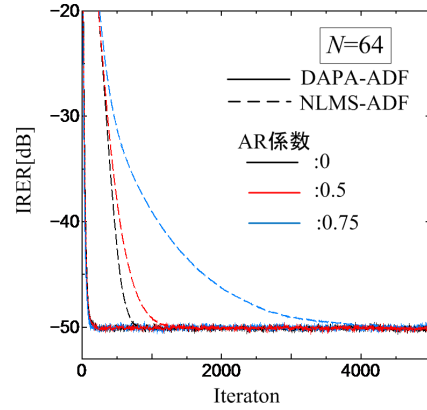


Fig. 9 Computation of the convergence(N=64).

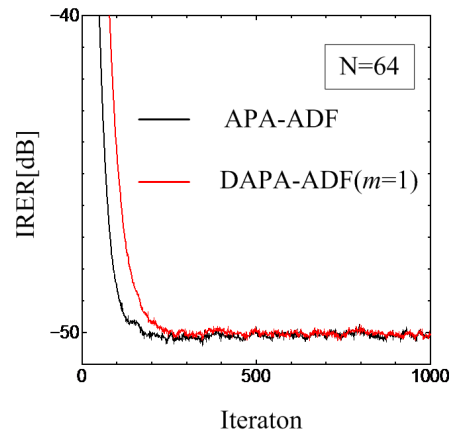


Fig. 10 Computation of the convergence (N=64,APA-ADF,DAPA-ADF).

とDAPA-ADFを比較する． $N = 16$ の収束特性評価結果を Fig.8 に、 $N = 64$ の収束特性評価結果を Fig.9 に示す．

Fig.8, Fig.9よりDAPA-ADFはNLMS-ADFよりも、収束速度が速く、有色性に強いことが確認できる．また、高次に成る程、収束速度の優位性が増していることが確認できる．

Fig.10は64タップ、白色信号が入力した際のDAPA-ADF, APA-ADFの比較である．このことからディレードによる収束特性の影響が少ないことが確認できる．

4. FPGA 実装

今回は実装に Intel 社製の FPGA デバイスである Cyclone III LS を用いる．

加算器に桁上げ伝播加算器 (RCA) を用いる．

Table 1 Compton of the adder.

	RCA	CLAA
Number of LE	18	60
Propagation delay[ns]	11.531	15.918

Table 2 Circuit evaluation(N=16).

	DAPA-ADF	NLMS-ADF
Number of LE	11,713	18,453
Sampling rate[MHz]	3.14	3.14

Table 3 Circuit evaluation(N=64).

	DAPA-ADF	NLMS-ADF
Number of LE	27,593	69,190
Sampling rate[MHz]	3.14	3.14

一般的に、高速な加算器として、桁上げ先見加算器が知られている (CLAA). その一方、桁上げ伝搬加算器 (RCA) はゲート数が少ないものの出力時間が長いことで知られている。しかし、Cyclone III LS では 高速伝播する Carry Chain 配線が施されている²⁾。これにより RCA は高速に動作できる。

Table 1 に 16bit 精度における RCA と CLAA の比較を示す。RCA の LE 数は CLAA の半分以下に抑えられ、伝搬遅延時間から RCA の方が高速であることが確認できる。

乗算器には部分席の加算に RCA を用いた Booth 型乗算器を用いる。また、除算器には引き放し方に基づいたセルアレイ除算器を用いる。

セルアレイ除算器は Fig.11 に示す Controll Add Subtract(CAS) セルをアレイ状に並べた構成で Fig.12 に 4bit 精度出力での構成を示す。なお、提案型は出力 16bit 精度のパイプライン化したセルアレイ除算器を用いる。

今回提案する DAPA-ADF と NLMS-ADF についてタップ数 16,64 について回路評価を行った。その結果を Table 2 と Table 3 に示す。

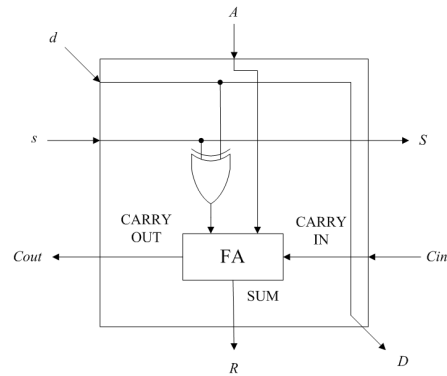


Fig. 11 Structure of CAS cell.

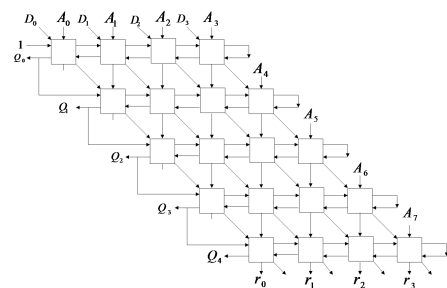


Fig. 12 Structure of CAS module.

2つの表から、DAPA-ADF が NLMS-ADF と同等のサンプリングレートであることが確認できる。また、ハードウェア量も少なく、高次に成る程、優位性が増し、タップ数 64 では NLMS-ADF の約 40% の LE 数で構成可能である。

5. まとめ

本報告では、APA-ADF の演算に用いる素子を共有化することにより全体の素子数を削減し、ディレードアップデートを用いることで NLMS-ADF と同等のサンプリングレート、かつ有色性に強く小規模な高性能アーキテクチャを提案した。

今後は、射影次数 3 以上のアフィン射影型適応フィルタの高性能化を行い、更に有色性に強いアーキテクチャを検討していく。

参考文献

- 1) 工藤寛士, 恒川佳隆, “高速性及び面積効率に優れた NLMS 適応デジタルフィルタの構成法”, 情報処理学会東北支部研究会, Mar., 2016.
- 2) 細田晃史, 佐々木拓朗, 内田勝也, 高橋 強, 恒川佳隆, “分散演算型 LMS 適応フィルタの FPGA 実現”, 計測自動制御学会東北支部第 257 回研究集会, 257-10, May 31, 2010.