006 TR-AC-0001 マルチビーム形成部にファンフィルタを用いた 広帯域ビームスペース形アダプティブアレー 関口 高志

1997. 2.25

ATR環境適応通信研究所

マルチビーム形成部にファンフィルタを用いた 広帯域ビームスペース形アダプティブアレー

(株) エイ・ティ・アール環境適応通信研究所

第三研究室

関口 高志

あらまし

伝搬波信号の比帯域幅がかなり広い(50%超)場合を対象として,マルチビームフォーミン グ回路に2次元 FIR ファンフィルタを適用したビームスペース形アダプティブアレーを提 案する.

まず,この2次元 FIR ファンフィルタの設計法について述べる.特性近似は,1次元 FIR ディジタルフィルタに1次元→2次元周波数変換を施した後にインパルス応答を打ち切る 方法によっている.周波数変換は,ファンフィルタの指向特性が近似的に周波数に依存し ないように決めている.

次いで、本方法で設計したファンフィルタを用いてビームスペース形アダプティブアレ ーを構成し、比帯域幅の非常に広い干渉信号を抑圧できること、従来の Griffiths-Jim ビーム フォーマより収束が速いことを計算機シミュレーションにより示す. 目次

1. まえがき	1
2. 広帯域信号に対応したビームスペース形アダプティブアレー	2
3. スペクトル変換	2
4. ファンフィルタの設計法	4
5. 例題	5
5.1 ファンフィルタの設計	5
5.2 ビームスペース形アダプティブアレーの性能評価	6
5.2.1 シミュレーション条件	6
5.2.2 収束特性と干渉信号抑圧性能	8
5.2.3 指向特性·周波数特性	1 0
6. むすび	1 0
参考文献	1 2
付録A(例題のファンフィルタの係数)	13
付録B(ファンフィルタの設計プログラムリスト)	14

<u>1. まえがき</u>

アダプティブアレーの構造は、アンテナ素子出力信号を制御対象とするエレメントスペース構成と、マル チビーム形成器の出力信号を制御対象とするビームスペース構成に大別される.

後者は、あらかじめ複数の異なる方向にビームを形成しておいて、それらを制御することにより干渉波を 抑圧する.そして、少ない自由度で効率よく所望波補足や干渉波抑圧が可能であることが知られている[2][3]. 筆者らは比帯域幅が約 40%以内の伝搬波信号を対象としたビームスペース形アダプティブアレーを提案した [4] (図1).図1(a)において、センサアレー配置は間隔 d の等間隔リニアアレーである.このアダプティブ アレーにおけるマルチビームフォーミング回路は比帯域幅が約 40%以内の広帯域信号に対応しており、マルチ ビームフォーミング回路を構成する各ビームフォーミング回路の指向特性が周波数に依存しないように図 1 (b)の係数を決めておく[5].

ファンフィルタはビームフォーマの一種である[1]. この場合,対象とする伝搬波信号は,音響信号や地震 波信号のようなベースバンド伝搬波信号であるが,当然のことながら,比帯域幅が50%を超える帯域通過形伝 搬波信号に対しても適用可能である.本報告は,[4]よりさらに広帯域の,比帯域幅が50~60%を超える広帯域 信号に対応する図1(a)のビームスペース形アダプティブアレーを構成することを目的とする.まず,図1(a) のマルチビームフォーミング回路を構成する各ビームフォーミング回路(図1(b))に適用することを目的とし た線形位相FIRファンフィルタの設計法を提案する.ファンフィルタの指向特性が阻止域を含めて近似的に周 波数に依存しないような特性近似を行う必要があるが,それには周波数(スペクトル)変換法と窓関数法を組 合せている.スペクトル変換によるファンフィルタ設計法は数多く提案されているが[6]-[8],指向特性が周波





図1(a) 広帯域信号に対応したビームスペース形 アダプティブアレー

図1(b) ビームフォーミング回路(BFN)の内部

数に依存しないようにしたものは見あたらない.また、ふつう窓関数法では、所望周波数応答値として、通過 域1,阻止域0という与え方をする[10]が、これでは阻止域特性をコントロールするのは困難である.つまり、 本方法は、このような点を克服すべく、窓関数法で与える所望周波数応答値をスペクトル変換で与えている.

次いで,提案法を用いて設計したファンフィルタを図1のビームスペース形アダプティブアレーに適用して,比帯域幅の広い広帯域干渉信号の抑圧が可能となること,従来のGriffiths-Jim ビームフォーマ[11]に比べて 収束が速いことを示す.

2. 広帯域信号に対応したビームスペース形アダプティブアレー

図1のアダプティブアレーについて簡単に説明する.

文献[4][5]と異なり,受信信号を周波数変換しないまま処理を行う.従って,マルチビームフォーミング回路入力信号は実数値をとる.マルチビームフォーミング回路は広帯域信号を通過させるようにする.マルチビ ームフォーミング回路における各広帯域ビームフォーミング回路は図1(b)の構成をとる.係数は実数である. 係数の計算法については後で説明する.マルチビームフォーミング回路のL個の出力信号のうち,電力の大き いb個のビームを選択して,それらの線形結合により干渉信号を抑圧する.線形結合係数w_i(*i=1,2,...,b*)は実数 で,適当な適応アルゴリズムで制御する.このとき,マルチビームフォーミング回路を構成する各ビームフォ ーミング回路が以下の条件を満たす必要がある[4][5].

a. 各ビームフォーミング回路は同一の位相特性を持つこと。

b. 各ビームフォーミング回路の指向特性が近似的に周波数に依存しないこと

条件bは、ビームフォーマの任意のサイドローブ方向から入射した干渉波の受信後のスペクトルが乱れな いようにするためのものである.そのスペクトルが乱れてしまうと干渉信号のレプリカが生成できなくなる.

正規化していない周波数fと入射角 θ に関する図1(b)の回路の応答 $H_i(f,\theta)$ は式(1)のようになる.

$$H_i(f,\theta) = \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} a_{i,n,k} e^{-j2\pi f k T_s} e^{j2\pi f \frac{d \sin \theta}{c} n} \qquad (i=1,2,...,L)$$
(1)

ここで,Nは素子数,Kはタップ数,T,はサンプリング間隔,cは伝搬波の速さである.

3. スペクトル変換

図1(b)のビームフォーミング回路は時間と空間の変数を持つ2次元(2D)ディジタルフィルタとみなすこと ができる.これらの係数はスペクトル変換法と窓関数法を組合せて決める.一般的には,振幅特性として低域 通過形の1D正規化周波数Fの実関数に対して,これから説明する変数変換を行い,それを窓関数法の所望特 性として用いる.このFの実関数は,必ずしもディジタルフィルタの振幅特性のような cos 2πF, sin 2πF の関数 である必要はない.しかし,後で述べるように,ディジ タルフィルタの振幅特性(正確にいえば零位相部分)と するのが便利である.所望特性としての阻止域の値は, 0ではなく,小さな値を持たせることが必要である.

伝搬波信号受信後に周波数変換を行わない場合,図 1(a)に示す等間隔リニアアレーに角度 θ で信号が入射 すると,ビームフォーミング回路入力信号の時間-空間ス ペクトルは,式(2)の直線上に現れる[1].



 $F_2 = (df_s/c)\sin\theta F_1 \tag{2}$

ここで, $f_s(=1/T_s)$ はサンプリング周波数, F_1 は正規化時間周波数, F_2 は正規化空間周波数で, $F_1=ff_s, F_2=(df\sin\theta)/c$ である.従って,メインビーム方向を θ_0 としたとき,設計すべきビームフォーミング回路の理想特性は次式のようになる.これは知られているように,ファンフィルタそのものである.

$$D(F_1, F_2) = \begin{cases} 1, & \frac{df_s}{c} \sin(\theta_0 - \varepsilon) \cdot F_1 \le F_2 \le \frac{df_s}{c} \sin(\theta_0 + \varepsilon) \cdot F_1 \\ 0, & otherwise \end{cases}$$
(3)

 ϵ は小さな正数である.式(3)は $F_1 \ge 0$ に対するもので、 $F_1 < 0$ に対しては式(3)の原点に関して対称な特性である. ただし、式(3)を窓関数法の所望特性として用いない.

スペクトル変換は、ファンフィルタの指向特性が周波数に依存しないように決める.これは、ファンフィルタの 2D 周波数平面上での振幅等高線が阻止域を含めて式(2)に沿うようにすることと等価である.

低域通過形の 1D 正規化周波数 Fの実関数として, 狭帯域低域通過 FIR ディジタルフィルタを用いるものと する. このときの 1D ディジタルフィルタを 1D 原形フィルタと呼ぶことにする. 通過域の中心である 1D 周波 数 F=0 が 2D 周波数平面上で直線 $F_2=(df/c)\sin\theta_0F_1$ に写像されるように, また, その他の 1D 周波数も式(2)のど こかの直線上に写像されるように 1D 周波数 Fと 2D 周波数 F_1 , F_2 の写像関係を定めると, 次式のようになる.

$$F = \frac{1}{2} \left\{ cF_2 / (df_s F_1) - \sin \theta_0 \right\}$$
(4)

図2に $\theta_{n=10^{\circ}, d=0.5\{cl(0.5f_{s})\}=cf_{s}$ としたときの式(4)の写像関係を示す.

式(4)より、1D原形フィルタとして、インパルス応答長が奇数の実係数1D零位相FIR ディジタルフィルタ を用いた場合の変換は次式のようになる.

3

$$\cos 2\pi F \to \cos \left[\pi \left\{ cF_2 / (df_s F_1) - \sin \theta_0 \right\} \right]$$

このような写像関係により,スペクトル変換法により指向特性が周波数に依存しないようなファンフィル タを設計できる.

4. ファンフィルタの設計法

素子数Nとタップ数Kは奇数とする.

式(1)より,図1(b)のビームフォーミング回路の 2D ディジタルフィルタとしての周波数応答 G_i(F₁, F₂) (*i*=1,2,...,*L*)は以下のようになる.

$$G_i(F_1, F_2) = \sum_{n=0}^{N-1K-1} \sum_{k=0}^{k-1} a_{i,n,k} e^{-j2\pi F_1 k} e^{j2\pi F_2 n}$$
(6)

ここでは、メインビーム方向が θ_0 であるビームフォーミング回路(ファンフィルタ)#*i* を設計するものとし、その係数 $a_{i,n,k}$ の計算法を説明する.

低域通過形の 1D 正規化周波数 Fの実関数として,狭帯域低域通過 FIR ディジタルフィルタを用いると,サ イドローブレベルを容易にコントロールできるので便利である.そこで,本節ではそのような場合について説 明する.

Step 1 インパルス応答長 *M* (<*N*)の 1D 零位相狭帯域低域通過ディジタルフィルタを設計する. *p*(*m*)をその インパルス応答とするとき,周波数応答 *P*(*F*)は以下のようになる.*M*を奇数とすると,

$$P(F) = p(0) + 2 \sum_{m=1}^{(M-1)/2} p(m) T_m(\cos 2\pi F)$$
(7)

T_m(cos 2πF)は cos 2πF に関する m 次の Chebyshev 多項式である.

Mの値の目安は,経験上,素子数Nの1/3程度である.また,Mは偶数でも構わないが,これから先は奇数であるとして説明する.

Step 2 式(7)に式(5)の周波数変換を施して,次式の 2D 周波数応答 Q₁(F₁, F₂)を得る.

$$Q_i(F_1, F_2) = p(0) + 2 \sum_{m=1}^{(M-1)/2} p(m) \operatorname{T}_m \left(\cos \left[\pi \left(\frac{cF_2}{df_1 F_1} - \sin \theta_0 \right) \right] \right)$$
(8)

そして、適当な 2D 周波数サンプル点上で式(8)の周波数応答値を計算する.

Step 3 $Q_i(F_1, F_2)$ のインパルス応答 $q_i(n_1, n_2)$ を求める. これは step 2 で計算した周波数応答値を逆離散フー リエ変換すればよい. $Q_i(F_1, F_2)$ は零位相で, $Q_i(F_1, F_2)=Q_i(-F_1, -F_2)$ であることから, $q_i(n_1, n_2)$ は実数である. Step 4 Step 3 で求めたインパルス応答 $q_i(n_1, n_2)$ を素子数Nとタップ数Kに応じて打ち切って, $a_{i,n,k}$ を得る.

 $a_{i,n,k} = q_i \left(k - \frac{K-1}{2}, -n + \frac{N-1}{2} \right) \qquad (n=0,1,\dots,N-1; \ k=0,1,\dots,K-1)$ (9)

タップ数Kの目安は経験上素子数Nと同程度である.

マルチビームフォーミング回路を構成する各ビームフォーミング回路のタップ数を等しくすれば, 2. で 述べた条件 a を満足できる.

付録Bにainkを計算するためのプログラムを示す.

5. 例題

5.1 ファンフィルタの設計

メインビーム方向 θ_θ=24°のファンフィルタを設計する.ビーム方向が中途な値なのは,後でこのファンフ ィルタをマルチビームフォーミング回路に用いる都合による.設計パラメータは,

素子間隔 d=0.5{c/(0.5f,)}=c/f,

素子数 N=15

タップ数*K*=15

である. 1D 原形フィルタのインパルス応答長は M=5, インパルス応答 p(m)=1/M (m=-2, ..., 2)とした.

本設計法ではスペクトル変換関数の近似は行っていないが、インパルス応答長5の原形フィルタから、イン パルス応答サイズ 15×15 の 2D フィルタを得るので、スペクトル変換関数のインパルス応答サイズを考えるな ら 7×7 相当ということになる.

図3に得られたファンフィルタの特性を示す.図3(b)は濃度が薄いほど振幅値が大きい.図3(b)より,多 少誤差はあるが、 $|F_1|\ge 0.2$ に対しては振幅等高線(影の境界線)が放射状になっていることがわかる.これは 図3(c)の指向特性からも確かめられる.従って、図1(a)のマルチビームフォーミング回路を構成するビーム フォーミング回路としては、帯域が $|F_1|\ge 0.2$ である広帯域帯域通過信号に対して有効である.現状ではその周 波数範囲をさらに低域に広げることは難しい.係数を付録Aに示す.

なお,図3(a)において, $F_1 \ge 0$ かつ $F_2 \ge F_1$ かつ $F_2 \ge F_1$,およびその原点に関して反対側の領域で,振幅値が0.3 弱となっているが,この領域には入射波のスペクトルは現れず,物理的に意味を持たない領域であり,指向特 性になんら影響を与えない.

5

5.2 ビームスペース形 アダプティブアレーの 性能評価

上で設計したファンフィルタを図 1(a)のマルチビームフォーミング回 路を構成するビームフォーミング回 路の1つとして用いて、ビームスペー ス形アダプティブアレーをサイドロ ーブキャンセラとして動作させる計 算機シミュレーションを行う.比較対象 として、Griffiths-Jim ビームフォーマを用 いた.

 5.2.1 シミュレーション条件 各センサ素子の特性は理想的なもの とする.すなわち,周波数特性はフラッ トで指向性は等方性である.素子配置は 等間隔リニアアレーである.

表1にシミュレーション条件を示す. -0 干渉波は2波で,所望信号入射方向は既 知であるものとする.信号は瞬時周波数が不規 則な FM 信号,雑音は白色ガウス性である.表 2に改めて素子数,タップ数を示す.併せて, マルチビームの数と主ビーム方向を示す.マル チビームは $|F_1|\ge 0.2$ において近似的に直交ビー ムを形成するように主ビーム方向とビーム数 *L* を決めた.主ビーム方向が 0°以上のビームに対 して, $F_1=0.3$ におけるマルチビームパターンを 図4に示す.あるビームの主ビーム方向が,近 似的に他のビームのヌル方向になっていること



(a) 振幅特性







がわかる.

図1(a)のビームスペース形アダプティブアレーをサ イドローブキャンセラとして用いたときの構成図を図 5に示す.所望波の入射方向と同じ方向のビーム(0°方 向のビーム)は常に選択し,その出力につながる荷重係 数値を1に固定する[2].そして他の4個のビームの中か ら出力電力の大きい順に3ビーム選択し,それらの線形 結合によって0°方向のビームで受信した信号に含まれ る干渉信号のレプリカを生成する.選択された3つのビ ームは,信号帯域において0°近傍にヌルを形成している



ため(図4),干渉信号レプリカ生成部への所望信号の漏れ込みは小さい.

ここでは線形結合係数である荷重係数 w_i(i=1,2,3)を LMS アルゴリズムで制御する.図5の記号を使うと, 荷重係数更新式は以下のようになる.初期値はすべての荷重値が0である.

$$W^{(m+1)} = W^{(m)} + \mu Y^{*}(m) z(m)$$

(10)

 $W^{(m)} = [w_1, w_2, w_3]^T \text{ at time} = m, \ Y(m) = [y_{j_1}(m), y_{j_2}(m), y_{j_3}(m)]^T$ $z(m) = y_0(m) - W^{(m)T}Y(m)$



7

図5 マルチビームフォーミング回路に ファンフィルタを用いたビームスペース形 サイドローブキャンセラ



{*j*₁, *j*₂, *j*₃}は選択されたビームの番号の組で,この例ではそれぞれ24°,-24°,53°のビームに対応する.ただし, 荷重更新回数が100回に達するまでは,ビーム出力の平均電力に応じて選択ビームの組を変えた.ビームの組 が変わったときは荷重値を初期値(全て0)にリセットした.荷重更新回数が100回以降はビームの組を固定 した.24°,-24°,53°のビームはそのときのビームである.mは信号の時間変数である.

比較対象とした Griffiths-Jim ビームフォーマは図6のような構成をとる[11].0°方向を向くビームで所望信 号を受信する.この中にはサイドローブで受信した干渉信号を含む.一方,干渉信号レプリカを生成する補助 ビーム形成側は,まず所望波方向である0°方向からの入射信号を遮断する前処理の後,3タップの適応ディジ タルフィルタ(タップ付遅延線回路)が続く.その荷重係数はLMS アルゴリズムにより制御する.初期値は すべての荷重値が0である.制御すべき荷重係数の数は,(15-1)×3=42 である.タップ数は,Wiener 解におい てビームスペース形と同等の干渉信号抑圧度が得られるように設定した.

表1 シミュレーション条件

	入射方向	雑音に対する電力	比帯域幅,中心周波数
所望波(S)	0°	20 dB	94%, 0.31 <i>f</i> s
干涉波 1(I ₁)	-18°	37 dB	99%, 0.32fs
干涉波 2(I2)	35°	40 dB	95%, 0.30fs

表2 素子数,タップ数,ビーム数

素子数N	15
タップ数K	15
マルチビーム数Lとビーム方向	5 (0°, ±24°, ±53°)

5.2.2 収束特性と干渉信号抑圧性能

図5のビームスペース形と図6のGriffiths-Jim ビームフォーマのステップサイズは,理論上同等の誤調整と なるように設定した.つまり,相関行列固有値和とステップサイズの積が同じになるように設定した[9].その 設定は表3に示す3通りである.参考のため,相関行列の条件数,固有値最大値と最小値,固有値和,固有値 平均値を表4に示す.表4は一例であり, 乱数の種を変えれば固有値も若干変化する.

図7に係数更新回数に対する出力における信号対干渉信号電力比(SIR)を示す.明らかにビームスペース形 の方が収束が速いことがわかる.これらのSIR曲線に直接関係する学習曲線の時定数は,ステップサイズと相 関行列固有値平均値との積に反比例する[9].この時定数が小さいほど収束が速い.ステップサイズと相関行 列固有値平均値との積は,ビームスペース形はGriffiths-Jim ビームフォーマの10倍強であり,そのため収束が 速くなっている. 荷重係数1回の更新に要する信号処理演算 量は,図1(a)・図5のビームスペース形の場合, マルチビーム形成処理にかかる乗・加算回数が かなり多いが,この点を克服できれば収束の速 い本構成は魅力的なものとなる.

なお,図7(a)~(c)のビームスペース形の収束 後のSIR はステップサイズが小さいほど高い.

干渉信号抑圧性能は信号対干渉信号電力の 改善度(インプルーブメントファクタ,IMF)に より評価する. IMFの定義は式(11)の通りであ る.

$$IMF = SIR_{out} / SIR_{in} \tag{11}$$

SIR_{in}はアダプティブアレー入力における1 素子あたりの信号対干渉信号電力比, SIR_{out}はア ダプティブアレー出力における信号対干渉信号 電力比である.

4000 回荷重係数更新後の表 3 の Case 2 に対 する IMF を表 5 に, Wiener 解における IMF を表 6 に示す. ビームスペース形サイドローブキャ ンセラでは, 4000 回荷重係数更新後に, 入力信 号の SIR が-21.8 dB だったのが, 出力では SIR が -21.8+29.2=7.4 dB に改善されたということにな る.

一方, Griffiths-Jim ビームフォーマは,4000 回荷重係数更新しても収束していないため,4000 回荷重係数更新しても IMF は Wiener 解における IMF よりかなり低いものとなっている.







(c) Case 3図7 収束特性(荷重係数更新回数対出力 SIR)

表3 LMS アルゴリズムにおけるステップサイズ

	Case 1	Case 2	Case 3
ビームスペース形	1.0×10 ⁻⁵	3.0×10 ⁻⁶	1.0×10 ⁻⁶
Griffiths-Jim ビームフォーマ	2.5×10 ⁻⁷	8.0×10 ⁻⁸	2.5×10 ⁻⁸

表4 相関行列の条件数,固有値最大値と最小値,固有値和,固有値平均値

	条件数	最大固有值	最小固有值	固有値和	固有值平均值
ビームスペース形	5.3×10 ²	9.1×10 ³	17	1.3×10 ⁴	4.3×10 ³
Griffiths-Jim ビーム	9.2×10 ⁶	1.5×10 ⁵	1.6×10 ⁻²	5.3×10 ⁵	1.3×10⁴
フォーマ(3タップ)					

表5 4000回荷重係数更新後のインプルーブメントファクタ(case 2)

	干渉波1に対して	干渉波2に対して	総合
ビームスペース形	29.3 dB	29.1 dB	29.2 dB
Griffiths-Jim ビームフォーマ(3 タップ)	24.4 dB	26.8 dB	25.9 dB

表6 Wiener 解におけるインプルーブメントファクタ

	干渉波1に対して	干渉波2に対して	総合
ビームスペース形	30.2 dB	30.0 dB	30.1 dB
Griffiths-Jim ビームフォーマ(3タップ)	28.1 dB	29.8 dB	29.2 dB

5. 2. 3 指向特性·周波数特性

図8にビームスペース形サイドローブキャンセラ,図9にGriffiths-Jim ビームフォーマの指向特性と周波数 特性(振幅特性)を示す.いずれも表3のCase 2で,4000回荷重係数更新後の特性である.ビームスペース 形サイドローブキャンセラは,信号帯域において干渉波方向にヌルが形成されていることがわかる.一方, Griffiths-Jim ビームフォーマでは,まだ収束していないために干渉波方向にビームスペース形ほどの深いヌル が形成されていない.ただし,ここでは示していないが,Wiener 解ではヌルを形成していた.

6. むすび

比帯域幅が 50~60%を超える広帯域信号に対応するビームスペース形アダプティブアレーを構成し、そのマ ルチビームフォーミング回路を構成するのに適した 2 次元 FIR ファンフィルタの設計法を提案した.特性近似 は周波数変換法と窓関数法を組合せた周波数変換は、ファンフィルタの指向特性が近似的に周波数に依存しな いように決めている.

本方法で設計したファンフィルタを用いてビームスペース形アダプティブアレーを構成し、比帯域幅の非常に広い干渉信号を抑圧でき、また、従来の Griffiths-Jim ビームフォーマより収束が速いことを計算機シミュ

10







レーションにより示した.

<u>謝辞</u>

本研究を進めるにあたり御指導・御助言いただいた ATR環境適応通信研究所小宮山牧兒社長,第三研究 室唐沢好男室長,第三研究室の諸氏に感謝します. 参考文献

- K. Nishikawa, T. Yamamoto, K. Oto, and T. Kanamori, "Wideband beamforming using fan filter," Proc. IEEE Int. Symp. Circuits & Syst., pp. 533-536, 1992.
- [2] K. Takao and K. Uchida, "Beamspace partially adaptive antenna," IEE Proc. vol. 136, Pt. H, no. 6, pp. 439-444, Dec. 1989
- [3] 千葉, 中條, 藤瀬, "ビームスペース CMA アダプティブアレーアンテナ," 信学論 (B-II), vol. J77-B-II, No. 3, pp. 130-138, 1994.
- [4] 関口, 三浦, 唐沢, "広帯域信号に対応したビームスペース形アダプティブアレーアンテナ,"信学論(B-II), vol. J80-B-II, No. 2, Feb. 1997.
- [5] 関口, 三浦, A. Klouche-Djedid, 唐沢, "スペクトル変換と窓関数法の組合せによる広帯域ディジタルビーム フォーミングアンテナのための2次元複素係数FIR ディジタルフィルタの設計法". 信学論(A), vol. J80-A, No. 1, pp. 61-73, Jan. 1997.
- [6] J. H. McClellen, "The design of two-dimensional digital filters by transformations," Proc. 7th Annu. Princeton Conf. Inform. Sci. and Syst., pp. 247-251, 1973.
- [7] E. Z. Psarakis, V. G. Mertzios, and G. P. Alexiou, "Design of two-dimensional zero phase FIR fan filters via the McClellan transform," IEEE Trans. Circuits & Syst., vol. CAS-37, No. 1, pp. 10-16, Jan. 1990.
- [8] S. S. Kidambi, "Design of two-dimensional non-recursive filters based on frequency transformations," IEEE Trans. Signal Process., vol. 43, No. 12, pp. 3025-3029, Dec. 1995.
- [9] S. Heykin, 武部 幹訳, 適応フィルタ入門, 現代工学社, 1987.
- [10] 西川, 唐, "フーリエ級数法によるファンフィルタの設計," 信学会第6回ディジタル信号処理シンポジウム, B-3-2, pp. 287-292, Nov. 1991.
- [11] L. J. Griffiths and C. W. Jim, "An alternative approach to linear constrained adaptive beamforming," IEEE Trans. Antennas & Propag., vol. AP-30, No. 1, pp. 27-34, Jan. 1982.

<u>付録A</u> 例題のファンフィルタの係数

横方向に素子番号 n (n=0,1,...,14),縦方向にタップの位置 k (k=0,1,...,14)を示す.

タップ番号乀素子番号		#0	#1	#2	#3	#4
	0	-4.4145E-03	-2.1030E-03	-2.0703E-03	-1.1057E-03	-1.9751E-03
	1	-5.6418E-03	-6.0587E-03	-2.5682E-03	-2.8267E-03	-8.1041E-04
	2	7.5262E-04	-6.8268E-03	-8.8252E-03	-3.0847E-03	-4.5394E-03
	3	8.6404E-03	4.8406E-03	-7.5566E-03	-1.4032E-02	-3.0505E-03
	4	1.3144E-02	1.5729E-02	1.4568E-02	-4.9263E-03	-2.5752E-02
	5	1.0493E-02	1.6685E-02	2.5464E-02	3.6559E-02	1.5811E-02
	6	2.9521E-03	5.7683E-03	1.2612E-02	2.6770E-02	6.8937E-02
	7	-4.8372E-03	-6.3639E-03	-9.6004E-03	-1.3862E-02	-2.8605E-02
	8	-7.4467E-03	-1.0116E-02	-1.3071E-02	-1.7578E-02	-9.3216E-03
	9	-4.3159E-03	-2.9885E-03	7.8303E-04	1.3450E-02	2.2157E-02
	10	1.7895E-03	5.3785E-03	1.1053E-02	6.8692E-03	-1.3761E-02
	11	5.2049E-03	6.8047E-03	1.1995E-03	-8.6884E-03	-2.3810E-03
	12	3.8904E-03	-7.4314E-04	-5.8308E-03	-1.9711E-03	-2.9022E-03
	13	-1.3544E-03	-4.1644E-03	-1.6138E-03	-1.8712E-03	-6.1729E-04
	14	-3.1207E-03	-1.3286E-03	-1.3595E-03	-7.1595E-04	-1.4314E-03
			1			
		1 # C	#6	#7	#0	#0
タップ番号へ素十番号		#5	#0	π /		# 5
	0	3.8604E-03	8.684/E-03	-1.8/99E-03	-8.62/8E-03	-6.0900E-03
	1	-7.3722E-03	-1.2923E-02	-1.93/6E-05	7.5915E-03	4.88/2E-03
	2	4.5938E-03	1.2162E-02	-3.6618E-03	-1.2025E-02	-9.2301E-03
	3	-1.2813E-02	-2.1733E-02	-1.9498E-05	9.5226E-03	6.7274E-03
	4	2.5106E-03	2.0082E-02	-1.0129E-02	-1.9756E-02	-1.9640E-02
	5	-5.5145E-02	-6.1812E-02	-1.9571E-05	6.1387E-03	3.9843E-03
	6	1.1159E-01	2.9065E-02	-9.0969E-02	-5.8925E-02	1.8717E-03
	7	-1.7829E-02	1.1524E-01	-2.4417E-02	1.1524E-01	-1.7829E-02
	8	1.8717E-03	-5.8925E-02	-9.0969E-02	2.9065E-02	1.1159E-01
	9	3.9843E-03	6.1387E-03	-1.9572E-05	-6.1812E-02	-5.5145E-02
	10	-1.9640E-02	-1.9756E-02	-1.0129E-02	2.0082E-02	2.5107E-03
	11	6.7274E-03	9.5226E-03	-1.9498E-05	-2.1733E-02	-1.2813E-02
	12	-9.2301E-03	-1.2025E-02	-3.6618E-03	1.2162E-02	4.5938E-03
	13	4.8872E-03	7.5915E-03	-1.9376E-05	-1.2923E-02	-7.3722E-03
	14	-6.0900E-03	-8.6278E-03	-1.8799E-03	8.6847E-03	3.8604E-03
タップ番号乀素子番号		#10	#11	#12	#13	#14
	0	-1.4314E-03	-7.1595E-04	-1.3595E-03	-1.3286E-03	-3.1207E-03
	1	-6.1729E-04	-1.8712E-03	-1.6138E-03	-4.1644E-03	-1.3544E-03
	2	-2.9022E-03	-1.9711E-03	-5.8308E-03	-7.4314E-04	3.8904E-03
	3	-2.3810E-03	-8.6884E-03	1.1995E-03	6.8047E-03	5.2049E-03
	4	-1.3761E-02	6.8692E-03	1.1053E-02	5.3785E-03	1.7895E-03
	5	2.2157E-02	1.3450E-02	7.8303E-04	-2.9885E-03	-4.3159E-03
	6	-9.3216E-03	-1.7578E-02	-1.3071E-02	-1.0116E-02	-7.4467E-03
	7	-2.8605E-02	-1.3862E-02	-9.6004E-03	-6.3639E-03	-4.8372E-03
	Ŕ	6.8937E-02	2.6770E-02	1.2612E-02	5.7683E-03	2.9521E-03
	9	1.5811E-02	3.6559E-02	2.5464E-02	1.6685E-02	1.0493E-02
	10	-2.5752E-02	-4.9263E-03	1.4568E-02	1.5729E-02	1.3144E-02
	11	-3.0505E-03	-1.4032E-02	-7.5566E-03	4.8406E-03	8.6404E-03
	12	-4.5394E-03	-3.0847E-03	-8.8252E-03	-6.8268E-03	7.5262E - 04
	13	-8 1041E-04	-2.8267E-03	-2.5682E-03	-6.0587E-03	-5.6418E-03
	11	_1 9751E_03	-1 1057E-03	-2.0703E-03	-2.1030E-03	-4 4145E-03
	7.4	1.07,011 00	1.1.100/11 00		1 1.10301 03	1

<u>付録B</u> ファンフィルタの設計プログラムリスト

4. で説明したファンフィルタの設計プログラムリスト (fortran) を示す.

ユーザーは以下のプログラムとファイルを自分で用意する必要がある.

・2次元逆FFTのサブルーチン(リスト内ではfft2d_rc)

・1次元原形フィルタ係数を記録したファイル(係数の書式に応じて読み込み部分の書式を改める必要あ

り)

本プログラムでは1次元原形フィルタ係数が実数でも複素数でも使用できるが、インパルス応答長は奇数 としている.

リスト

```
C٠
С
      Design of element-space baseband DBFN (fan filter)
с
с
      by windowing impulse response of ideally spectrally transformed
с
      2D DF
с
с
      (C)ATR Adaptive Commun. Res. Lab., 1995-1996.
с
      ver. 1.0, Apr. 23, 1996.
с
            1.1, Sept. 3, 1996.
С
с
С
с
с
С
      Output logical unit no.=81
С
      program design_base_2d_trans
      parameter (narymax=100, nfiltmax=100, nfftpoint=256)
С
      complex*8 weight(narymax, 0:nfiltmax), z(256, 256),
      h2d(-256:256, -256:256), hz(0:nfiltmax)
dimension h2(-256:256), hzr(0:nfiltmax), win1(101), win2(101)
     0
      character*50 flname
С
      common /param/d
С
с
      pi=3.1415927
      rad=pi/180
с
С
      Input parameters
С
c٠
С
      d=0.5
     write(*, *) ' Main beam direction (deg)='
      read(*, *) th0
     write(*, *) ' #array='
read(*, *) n2
     write(*, *) ' #taps ='
read(*, *) n1
     write(*, *) ' Element distance d='
     write(', ') ' ([*wavelength of 0.5fs], usually d=0.5)'
```

```
read(*, *) d
write(*, *) ' Prototype filter file name='
read(*, *) flname
      th0=th0*pi/180
С
С
      iflform=1
      write(*, *) ' Prototype 1D filter file name=', flname
      write(*, *) ' Prototype 1D filter file format:'
write(*, *) ' Real(>=0)/Complex(<0)='</pre>
      read(*, *) iflform
с
с
      Read the prototype filter's coefficient file
с
       (Please change according to your coefficient-file format)
С
С
      ndv=30
      if (iflform .ge. 0) then ! Real filter file format
        open(ndv, file=flname, status='old', form='formatted')
        do 10 i=1, 8
        read(ndv, 1000)
read(ndv, '(47x, i3)') nfilt
read(ndv, 1000)
 10
        read(ndv, 1000)
        read(ndv, 1000)
read(ndv, '(33x, e15.8)') (hzr(i), i=(nfilt-1)/2, 0, -1)
        close(ndv)
        do i=0, (nfilt-1)/2
          hz(i) = hzr(i)
        enddo
      else
                                   ! Complex filter file format
        open(ndv, file=flname, status='old', form='formatted')
        read(ndv, '(1h)')
read(ndv, '(1h)')
read(ndv, '(15x, i3)') nfilt
read(ndv, '(1h)')
        do i=1, (nfilt-1)/2
        read(ndv, '(lh )')
        enddo
        read(ndv, '(8x, e20.7, 4x, e20.7)') (hz(i), i=0, (nfilt-1)/2)
        close(ndv)
      endif
с
 1000 format(1h )
с
С
      if (iflform .ge. 0) then
с
c-
      Frequency response of 2D filter by transformation
с
      (Prototype filter = real)
с
C--
с
      do 30 j=-nfftpoint/2, nfftpoint/2
      do 30 i=-nfftpoint/2, nfftpoint/2
        F1=float(i)/nfftpoint
        F2=float(j)/nfftpoint
с
        TRc=T_ideal(F1, F2, th0)
с
        s=hz(0)+2*real(hz(1))*TRc
        T0=1
        T1=TRc
        do 35 m=2, (nfilt-1)/2
```

15

```
T2=2*TRc*T1-T0
         s=s+2*real(hz(m))*T2
         T0=T1
         T1=T2
 35
       continue
       h2d(i, j)=s
 30
     continue
с
     else
с
c-
     Frequency response of 2D filter by transformation
с
      (Prototype filter = complex)
с
c
С
     do 70 j=-nfftpoint/2, nfftpoint/2
     do 70 i=-nfftpoint/2, nfftpoint/2
       F1=float(i)/nfftpoint
       F2=float(j)/nfftpoint
с
       TRc=Tc_ideal(F1, F2)
       TRs=Ts_ideal(F1, F2)
С
       s=hz(0)+2*real(hz(1))*TRc+2*aimag(hz(1))*TRs
       T0=1
       T1=TRc
       U0=0
       U1=TRs
       do 75 m=2, (nfilt-1)/2
        T2=2*TRc*T1-T0
        U2=2*TRc*U1-U0
        s=s+2*real(hz(m))*T2+2*aimag(hz(m))*U2
        T0=T1
        T1=T2
        U0=U1
        U1=U2
 75
       continue
       h2d(i, j)=s
 70
     continue
с
     endi f
с
c-
     Impulse response from IFFT of 2D frequency response data
с
c--
    C,
     do j=0, nfftpoint/2
     do i=0, nfftpoint/2
      il=i+1
       j1=j+1
      z(i1, j1)=h2d(i, j)
     enddo
     enddo
с
     do j=0, nfftpoint/2
     do i=nfftpoint/2+1, nfftpoint-1
       il=i+1
       i1=i+1
      z(i1, j1)=h2d(i-nfftpoint, j)
     enddo
     enddo
с
     do j=nfftpoint/2+1, nfftpoint-1
     do i=0, nfftpoint/2
```

```
16
```

```
i1=i+1
       j1=j+1
       z(i1, j1)=h2d(i, j-nfftpoint)
     enddo
     enddo
с
     do j=nfftpoint/2+1, nfftpoint-1
     do i=nfftpoint/2+1, nfftpoint-1
       il=i+1
       j1=j+1
       z(i1, j1)=h2d(i-nfftpoint, j-nfftpoint)
     enddo
     enddo
С
     call fft2d_rc(z, nfftpoint, nfftpoint, 1, 1) ! 2D IFFT
с
c-
     Conversion: impulse response -> weight coefficients
с
C--
С
      (Change of variable: zero-phase filter impulse response)
С
с
     do j=0, nfftpoint/2
     do i=0, nfftpoint/2
       h2d(i, j)=z(i+1, j+1)
     enddo
     enddo
с
     do j=0, nfftpoint/2
     do i=-nfftpoint/2, -1
       h2d(i, j)=z(i+nfftpoint+1, j+1)
     enddo
     enddo
с
     do j=-nfftpoint/2, -1
     do i=0, nfftpoint/2
      h2d(i, j)=z(i+1, j+nfftpoint+1)
     enddo
     enddo
с
     do j=-nfftpoint/2, -1
     do i=-nfftpoint/2, -1
       h2d(i, j)=z(i+nfftpoint+1, j+nfftpoint+1)
     enddo
     enddo
с
     do k=0, n1-1
     do n=1, n2
       we=real (h2d(k-(n1-1)/2, -n+(n2-1)/2+1))
       weight(n, k)=we
     enddo
     enddo
с
C--
      _____
     Output
С
C-----
                     _____
С
     ndv=81
     write(ndv, 1200)
     write(ndv, 1210) n2, n1, npc, nps
     do i=1, n2
       write(ndv, '(i5,2e20.7)') i, weight(i, 0)
write(ndv, '(5x,2e20.7)') (weight(i, j), j=1, n1-1)
     enddo
```

```
17
```

```
с
1200 format ('Element-space_TDL_wide-band-DBFN (2D_freq_domain',
       '_by_transformation)')
1210 format('#arrays =', i5, ' filter length=', i5,
* ' trans. order(cos, sin)=', 2i5)
с
    end
С
с
    Function : COS/SIN ideal transform
с
С
с
    function T_ideal(F1, F2, th0)
с
    common /param/d
с
    pi =3.1415927
    pi2=6.2831853
С
    a=2*d
    fm = (1 - sin(th0))/2
    if (F1.eq. 0.and. F2. eq. 0) then
     T_ideal=0
    else if (F1.eq. 0) then
     T_ideal=cos(pi2*fm)
    else if ((Fl.gt.0 .and. F2.le.a*F1 .and. F2.ge.-a*F1) .or.
           (F1.1t.0 .and. F2.ge.a*F1 .and. F2.le.-a*F1)) then
   @
      T_ideal=cos(pi*(1/a*F2/F1-sin(th0)))
    else
     T_ideal=cos(pi2*fm)
    endif
с
    return
    end
С
с
    Function : COS ideal transform
с
с
с
    function Tc_ideal(F1, F2)
с
    common /param/d
с
    pi =3.1415927
с
    a=2*d
    if (F1.eq. 0.and. F2. eq. 0) then
     Tc_ideal=0
    else if (Fl.eq. 0) then
     Tc_ideal=-1
    else if ((F1.gt.0 .and. F2.le.a*F1 .and. F2.ge.-a*F1) .or.
           (F1.lt.0 .and. F2.ge.a*F1 .and. F2.le.-a*F1)) then
   6
     Tc_ideal=cos(pi*(1/a*F2/F1))
    else
     Tc_ideal=-1
    endif
с
    return
    end
С
```

```
18
```

]

C, Function : SIN ideal transform с с с function Ts_ideal(F1, F2) с common /param/d , c pi =3.1415927 с a=2*d if (F1 .eq. 0 .and. F2. eq. 0) then Ts_ideal=1 else if (F1.eq. 0) then Ts_ideal=0 else if ((F1.gt.0 .and. F2.le.a*F1 .and. F2.ge.-a*F1) .or. (F1.lt.0 .and. F2.ge.a*F1 .and. F2.le.-a*F1)) then @ Ts_ideal=sin(pi*(1/a*F2/F1)) else Ts_ideal=0 endi f с return

end