

パイプライン形平面回転アルゴリズムの誤差解析に基づく設計

正 員 菊池 久和[†] 正 員 中 静 真[†] 正 員 渡 辺 弘 道[†]

Design of a Pipelined Plane-Rotation Algorithm Based on the Error Analysis

Hisakazu KIKUCHI[†], Makoto NAKASHIZUKA[†],
and Hiromichi WATANABE[†], Member

あらまし 本論文では、離散時間無損失散乱システムを構成するため、パイプライン処理が可能な平面回転アルゴリズムを提案している。CORDIC アルゴリズムを単位演算操作に分割し、それぞれの単位演算操作を担う操作セルを縦続接続することによって、パイプライン CORDIC アルゴリズムに再構成する。この縦続接続による構成は、パイプライン処理が可能なばかりでなく、基本セルのリニアアレーとなるためモジュール性も高い。次に、提案アルゴリズムの演算誤差を有限操作誤差と有限語長誤差に分類し、解析結果よりアルゴリズムの設計法を提案している。この設計法では、指定した有限操作誤差と回転角の範囲から、必要な演算操作だけを選択できること、そしてアルゴリズムの有限語長誤差を最小化し得る点に特徴がある。最後にウェーブデジタルフィルタと正弦波対発振器への適用例を示している。

1. ま え が き

デジタル信号処理システムを、無損失散乱行列で表現し、効果的な回路構造を得る研究が広く行われている^{(1)~(5)}。離散時間無損失散乱行列は、最も基本的な無損失散乱行列である平面回転行列と遅延によって構成される⁽⁵⁾。無損失散乱行列によるデジタル信号処理システムで高速な処理を実現するためには、平面回転アルゴリズムのパイプライン処理が有効である。また、信号処理システムの VLSI 化を容易にするためには、回路の規則性とモジュール性についても配慮する必要がある。

一方、平面回転アルゴリズムとしては、CORDIC アルゴリズム⁽⁶⁾がある。CORDIC は原理的にビットシフトと加減算操作によって構成され、簡単な制御で動作させることができる。Haviland と Tuszynski⁽⁷⁾は、LSI チップとして実現しておりデジタルフィードバックと称する再帰的アルゴリズムを採用している。このアルゴリズムでは、一つのハードウェアを繰り返して使用するためパイプライン処理が困難である。

そこで、本論文では、離散時間無損失散乱システム of VLSI 実現を目標として、パイプライン構成による

CORDIC アルゴリズムを提案する。本アルゴリズムでは、1 回の演算操作を一つの独立した操作セルに担わせ、操作セルの縦続接続によってパイプライン CORDIC アルゴリズムを構成する。すべての操作セルは同一の回路構造で実現できるため規則性が高く、各操作セルは独立に動作できるためモジュール性が高い。更に操作セル間の通信は隣接する操作セル間だけで行われる。これらの特徴により提案するアルゴリズムは VLSI 向きのアルゴリズムになっている。

2.では、Haviland, Tuszynski の CORDIC アルゴリズムの演算順序と演算操作に対する制約を取り払い、パイプライン CORDIC アルゴリズムに再構成する。次に 3.ではパイプライン CORDIC の誤差モデルについて述べる^{(8)~(10)}。4.では、有限語長誤差がセルの接続順序に依存することに着目し、これを最小化するアルゴリズムの設計法を述べる。これにより Haviland, Tuszynski の CORDIC アルゴリズムよりも演算操作数が少なく、等価出力雑音が最小となるアルゴリズムが得られる。5.では、無損失散乱行列によって得られるパワーウェーブデジタルフィルタ^{(4),(5)}のためのアルゴリズムとデジタル正弦波対発振器⁽¹¹⁾のためのアルゴリズムを設計し、コンピュータシミュレーションにより本設計法の有効性を確認する。

[†] 新潟大学工学部電気電子工学科, 新潟市
Faculty of Engineering, Niigata University, Niigata-shi, 950-21
Japan

2. パイプライン CORDIC アルゴリズム

CORDIC は初等関数を統一的に計算する演算法として知られているが、本論文では無損失散乱行列による信号処理へ応用するため平面回転機能だけを取り扱う。

CORDIC による平面回転アルゴリズムとして、Haviland, Tuszyński の CORDIC アルゴリズムがある。Haviland らのアルゴリズムは、数値の右方向ビットシフトと加減算に基づく基本演算操作から構成されている。基本演算操作は、スケーリングファクタサイクル⁽⁷⁾と呼ばれる規格化操作を伴う回転操作と、規格化操作を伴わない回転操作に分類できる。2種類の基本演算操作を選択しながら、繰り返し実行することで入力ベクトルに対する平面回転を実現している。

Haviland らのアルゴリズムでは、1回の基本演算におけるビットシフトの回数を基本演算操作の繰返し回数と一致させている。また規格化操作を伴う回転操作においては、常に回転操作の次に規格化操作が実行される。

しかし、平面回転演算のためには、演算の繰返し回数とビットシフトの回数を一致させる必要はない。また、回転操作と規格化操作の実行順序も前述の実行順序に従う必要はない。そこで、Haviland らのアルゴリズムを1回の基本演算ごとにビットシフトの回数を自由に決められるように修正し、更に1回の基本演算の繰返しごとに回転操作と規格化操作を自由に選択できるように修正する。これを修正 HT 法と呼ぶことにする。

[修正 HT 法]

初期値としてベクトル $[X_0, Y_0]^T$ と回転角 Z_0 を得た後、以下の計算を $i=0$ から $n-1$ まで繰り返す。

$$\begin{bmatrix} X_{i+1} \\ Y_{i+1} \end{bmatrix} = (1 - e(i)2^{-s(i)}) \begin{bmatrix} 1 & -u_i 2^{-s(i)} \\ u_i 2^{-s(i)} & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (1)$$

$$Z_{i+1} = Z_i - u_i \tan^{-1} 2^{-s(i)} \quad (2)$$

ここで、 $s(i)$ は 0 以上の整数で、シフトビットの個数を示す。 $e(i)$ は 0 か 1 の値をとり $u(i)$ は

$$u_i = 0 \text{ for } e(i) = 1 \quad (3a)$$

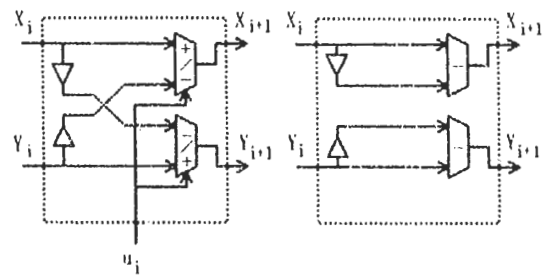
$$u_i = \text{sign}(Z_i) \text{ for } e(i) = 0 \quad (3b)$$

で示される値をとる。 $\text{sign}(Z_i)$ は Z_i の符号を表す。

式(1)、(2)は、 i 番目の基本演算を表す。 $e(i)$ は i 番目の基本演算の機能を定める定数であり、機能数列

と呼ぶ。 $e(i)=0$ は i 番目に行う基本演算が回転操作であることを表し、 $e(i)=1$ は規格化操作であることを意味する。回転操作における回転方向は、 u_i によって指定し、これを方向数列と呼ぶ。また、2のべき乗倍の乗算は、 X_i, Y_i の右方向ビットシフトで実現され、シフトビット数はシフト数列 $s(i)$ によって与えられている。つまり、修正 HT 法は機能数列、シフト数列、方向数列 $\{e(i), s(i), u_i\}$ によって記述される。そして、サイクル数 i と基本演算の機能ならびにシフトビット数は、各基本演算ごとに分離されており、基本演算の機能とシフトビット数を独立に指定できる。

このアルゴリズムを実現する CORDIC プロセッサとして2種類の構成が考えられる。一つは、式(1)の演算を実行するハードウェアのみを用意し、あらかじめ式(2)の計算を行い Z_0 を u_i 列にコーディングしておき、 u_i 列と X_0, Y_0 を入力し、式(1)の演算を1サイクルごとに実行する構成である。式(1)で示される単位演算操作を担う回転操作セルと規格化操作セルの回路構造を図1に示す。また、この単位演算操作の回路を一つの操作セルとして、その縦続接続により図2に示す CORDIC プロセッサを構成する。もう一つの構成は、式(1)と式(2)の演算を実行する CORDIC プロセッサを用意し、 X_0, Y_0, Z_0 の入力を与えられた後、式



$e(i)=0$: Rotation cell $e(i)=1$: Normalization cell

図1 操作セルの構成
Fig. 1 Two configurations of a cell.

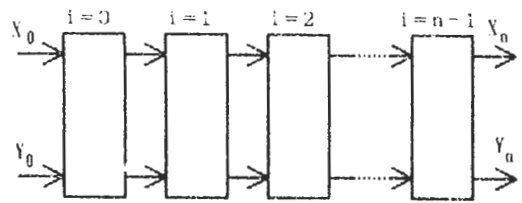


図2 パイプライン CORDIC プロセッサのブロックダイヤグラム

Fig. 2 Block diagram of a pipelined CORDIC processor.

(1)と式(2)の演算を1サイクルごとに同時に実行する構成である。この構成での回転操作セルは、図1の回転操作セルの回路に、 $\tan^{-1} 2^{-s(i)}$ を記憶するROMと加減算器を加えることで実現できる。

いずれの構成でも、規格化操作セルと回転操作セルの両者を同一の回路で実現し、マルチプレクサによって回転操作と規格化操作の機能を切り換えることが可能である。従って同一の回路パターンを繰り返し配置することで回路を実現することができ、VLSI化が容易である。また、各々のセルの処理時間は1回のシフトと加減算器の動作時間と同一であるから、すべてのセルの処理時間を等しくすることができる。従って、同期パイプライン処理を行うことができる。更に、すべてのセル間の信号伝送は一方に統一されており、隣接セルの間で行われるためにVLSI化の際のレイアウトが容易である。

ところで、式(1)の演算は

$$\begin{bmatrix} X_{i+1} \\ Y_{i+1} \end{bmatrix} = k_i \begin{bmatrix} \cos \phi_i & -u_i \sin \phi_i \\ u_i \sin \phi_i & \cos \phi_i \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \quad (4)$$

と等価であり、これに対応して式(2)は

$$Z_{i+1} = Z_i - u_i \phi_i \quad (5)$$

と等価である。ここで、ベクトルのノルムを記号 $\|\cdot\|$ により表すことにし、ノルム $\|X, Y\|$ を

$$\|X, Y\| = \sqrt{X^2 + Y^2} \quad (6)$$

と定義する。この定義により、操作セルへの入力ベクトル $[X_i, Y_i]^T$ のノルムと、操作セルの出力ベクトル $[X_{i+1}, Y_{i+1}]^T$ のノルムの関係は

$$k_i = \frac{\|X_{i+1}, Y_{i+1}\|}{\|X_i, Y_i\|} \quad (7)$$

と表され、 k_i はシフト数列 $s(i)$ により

$$k_i = 1 - 2^{-s(i)} \quad \text{for } e(i)=1 \quad (8a)$$

$$k_i = \sqrt{1 + 2^{-2s(i)}} \quad \text{for } e(i)=0 \quad (8b)$$

で与えられる。 k_i を i 番目の基本演算操作におけるノルム増加率と呼ぶ。

角度 ϕ_i はシフト数列によって定まり

$$\phi_i = \tan^{-1} 2^{-s(i)} \quad (9)$$

と表される。これにより、式(1)と式(2)の演算を n 回繰り返すことにより出力ベクトル

$$\begin{bmatrix} X_n \\ Y_n \end{bmatrix} = K \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} \quad (10)$$

が得られ、出力角度 Z_n は

$$Z_n = Z_0 - \theta \quad (11)$$

となる。ここで、回転角 θ には

$$\theta = \sum_{i=0}^{n-1} u_i \phi_i \quad (12)$$

の関係があり、その最大値は

$$|\theta|_{\max} = \sum_{i=0}^{n-1} (1 - e(i)) \phi_i \quad (13)$$

である。最大値より回転角 θ の範囲は

$$-|\theta|_{\max} \leq \theta \leq |\theta|_{\max} \quad (14)$$

となる。また、最終的なノルム増加率 K は、

$$K = \prod_{i=0}^{n-1} k_i \quad (15)$$

で与えられる。回転操作の場合に k_i は1以上の値をとり、規格化操作の場合に k_i は1以下の値となる。従って、有限回の回転操作に対して有限回の規格化操作を加えることにより、ノルム増加率 K を1に近づけることができる。

3. パイプラインCORDICアルゴリズムの演算誤差

3.1 有限操作誤差

有限操作誤差は、CORDICアルゴリズムが有限回の操作から構成されるために生じる誤差であり、出力ベクトルのノルムと回転角に現れる。

ノルムに現れる有限操作誤差は式(15)の K で与えられ、入力ベクトルのノルムを r_0 、出力ベクトルのノルムを r_n と書けば

$$K = \frac{r_n}{r_0} \quad (16)$$

の関係があり、理想的には $K=1$ となる。このノルム増加率 K は、機能数列 $e(i)$ とシフト数列 $s(i)$ から定まる。回転角に生じる有限操作誤差 $\Delta\theta$ は、CORDICへの入力角度 Z_0 と表現される回転角 θ により

$$\Delta\theta = Z_0 - \theta \quad (17)$$

と定義され、理想的には $Z_0 = \theta$ となり、 $\Delta\theta = 0$ となる。また、回転角に生じる有限操作誤差 $\Delta\theta$ は式(11)の演算結果である出力角度 Z_n と一致する。

この $\Delta\theta$ の値は機能数列 $e(i)$ 、シフト数列 $s(i)$ と入力角度 Z_0 により計算することができる。

3.2 有限語長誤差

式(1)で示したCORDICアルゴリズムの単位演算では、2のべき乗が係数となっている。そのため、数値が2進数固定小数点によって表現されている場合、セルインデックスが i のセルにおける乗算は、被乗数 X_i, Y_i の $s(i)$ 回の右方向ビットシフトで行われる。従って、図1のシフト器の入力語長と出力語長を等しくすれば、LSBから $s(i)$ ビットが失われる。

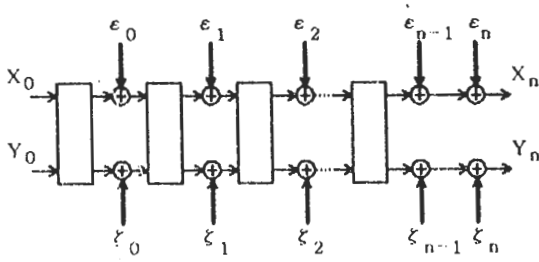


図3 有限語長誤差のモデル
Fig. 3 A model of finite wordlength error.

また、修正 HT 法では CORDIC の演算精度を向上させるために、CORDIC 内部の語長は出力語長に比べて下位に M ビットだけ拡張したものとす。従って、内部語長と出力語長を一致させる際に、内部データの低位ビットが失われ、誤差が生じる。

図2の各操作セルを理想的で誤差が生じないものと仮定し、シフト器による誤差 ϵ_i, ζ_i は、各操作セルの出力である X_{i+1}, Y_{i+1} に接続される雑音源から発生すると仮定する。また、内部語長と出力語長の不一致による誤差を ϵ_n, ζ_n と表すと、図2に対応する雑音モデルが図3のように得られる。 ϵ_i, ζ_i ならびに ϵ_n, ζ_n を有限語長誤差と呼び、 $[\epsilon_i, \zeta_i]^T$ と $[\epsilon_n, \zeta_n]^T$ を有限語長誤差ベクトルと呼ぶ。

本節では、有限語長誤差ベクトルのノルムの値の範囲を求めることで、修正 HT 法の出力端に現れる等価出力雑音の分布範囲を明らかにする。

セルインデックスが i のセルにおける有限語長誤差 ϵ_i, ζ_i の絶対値は、数値表現の方式によらず、内部の数値表現の LSB が表す値 q から、切捨ての場合は

$$|\epsilon_i| < q \quad (18 a)$$

$$|\zeta_i| < q \quad (18 b)$$

で示される範囲の値となる。また、内部の数値表現の LSB より一つ下位のけたの情報によりまるめを行った場合

$$|\epsilon_i| \leq \frac{1}{2} q \quad (18 c)$$

$$|\zeta_i| \leq \frac{1}{2} q \quad (18 d)$$

で示される範囲の値となる。従ってセルインデックスが i のセルでの有限語長誤差ベクトルのノルムは、切捨ての場合

$$\|[\epsilon_i, \zeta_i]^T\| < \sqrt{2} q \quad (19 a)$$

であり、まるめの場合

$$\|[\epsilon_i, \zeta_i]^T\| \leq \frac{\sqrt{2}}{2} q \quad (19 b)$$

で示される範囲の値となる。

セルインデックスが $i=0$ から $n-2$ までのセルで生じた有限語長誤差ベクトルは、後段のセルを伝搬し、出力端に

$$\begin{bmatrix} E_i \\ Z_i \end{bmatrix} = \prod_{j=i+1}^{n-1} k_j \begin{bmatrix} \cos \phi_j & -\sin \phi_j \\ \sin \phi_j & \cos \phi_j \end{bmatrix} \begin{bmatrix} \epsilon_i \\ \zeta_i \end{bmatrix} \quad (20)$$

となって現れる。ここで、誤差伝搬率として a_i を

$$a_i = \prod_{j=i+1}^{n-1} k_j \quad (21)$$

と定義する。 a_i により、出力端での有限語長誤差ベクトルのノルムは

$$\|[\epsilon_i, \zeta_i]^T\| = a_i \|[\epsilon_i, \zeta_i]^T\| \quad (22)$$

と表すことができる。セルインデックスが $n-1$ のセルでは、 ϵ_{n-1} と ζ_{n-1} はそのまま出力に現れ

$$\begin{bmatrix} E_{n-1} \\ Z_{n-1} \end{bmatrix} = \begin{bmatrix} \epsilon_{n-1} \\ \zeta_{n-1} \end{bmatrix} \quad (23)$$

となる。当然、誤差伝搬率 a_i は 1 である。

また、出力語長と内部語長の不一致による有限語長誤差は、 $M > 0$ で内部のデータの低位 M ビットを切り捨てた場合、 ϵ_n と ζ_n の絶対値はそれぞれ

$$|\epsilon_n| < 2^M q \quad (24 a)$$

$$|\zeta_n| < 2^M q \quad (24 b)$$

で示される範囲の値となる。まるめの場合は

$$|\epsilon_n| \leq 2^{M-1} q \quad (24 c)$$

$$|\zeta_n| \leq 2^{M-1} q \quad (24 d)$$

で示される範囲の値となる。従って有限語長誤差ベクトル $[\epsilon_n, \zeta_n]^T$ のノルムは、

$$\|[\epsilon_n, \zeta_n]^T\| < \sqrt{2} 2^M q \quad (25 a)$$

で示される範囲の値となり、まるめの場合

$$\|[\epsilon_n, \zeta_n]^T\| \leq \sqrt{2} 2^{M-1} q \quad (25 b)$$

で示される範囲の値となる。どちらの場合も、 $M=0$ ではノルムが 0 になることは言うまでもない。

出力端には、各セルで生じた有限語長誤差が伝搬するほか、語長不一致による誤差が発生するために、有限語長誤差ベクトルは全体として

$$\begin{bmatrix} E_X \\ E_Y \end{bmatrix} = \sum_{i=0}^n \begin{bmatrix} E_i \\ Z_i \end{bmatrix} \quad (26)$$

となる。 $[E_X, E_Y]^T$ による CORDIC の等価出力雑音モデルを図4に示す。またノルムは切捨ての場合

$$\|[\epsilon_n, \zeta_n]^T\| < \sum_{i=0}^n a_i \|[\epsilon_i, \zeta_i]^T\| \quad (27 a)$$

の範囲の値となり、まるめの場合

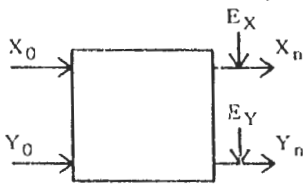


図4 有限語長誤差による等価出力雑音モデル
Fig. 4 Equivalent output error model for finite wordlength error.

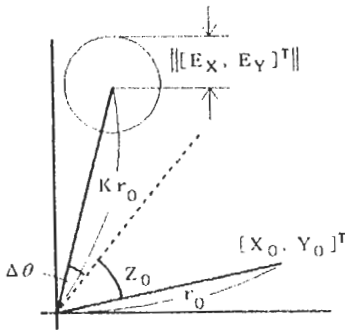


図5 有限操作誤差と有限語長誤差による出力誤差分布モデル

Fig. 5 Total output error distribution caused by finite cell error and finite wordlength error.

$$\| [E_X, E_Y]^T \| \leq \sum_{i=0}^n a_i \| [\varepsilon_i, \zeta_i]^T \| \quad (27 \text{ b})$$

の範囲の値となる。以下、出力端に現れる有限語長誤差ベクトル $[E_X, E_Y]^T$ のノルムの上限を $\| [E_X, E_Y]^T \|_{\max}$ として、各場合について簡条書きする。

・切捨て、 $M > 0$ の場合

$$\| [E_X, E_Y]^T \|_{\max} = \sqrt{2 \left(\left(\sum_{i=0}^{n-2} \left(\prod_{m=i+1}^{n-1} k_m \right) q \right) + q + 2^M q \right)} \quad (28 \text{ a})$$

・切捨て、 $M = 0$ の場合

$$\| [E_X, E_Y]^T \|_{\max} = \sqrt{2 \left(\left(\sum_{i=0}^{n-2} \left(\prod_{m=i+1}^{n-1} k_m \right) q \right) + q \right)} \quad (28 \text{ b})$$

・まるめ、 $M > 0$ の場合

$$\| [E_X, E_Y]^T \|_{\max} = \sqrt{2 \left(\left(\sum_{i=0}^{n-2} \left(\prod_{m=i+1}^{n-1} k_m \right) 2^{-1} q \right) + 2^{-1} q + 2^M \right)} \quad (28 \text{ c})$$

・まるめ、 $M = 0$ の場合

$$\| [E_X, E_Y]^T \|_{\max} = \sqrt{2 \left(\left(\sum_{i=0}^{n-2} \left(\prod_{m=i+1}^{n-1} k_m \right) 2^{-1} q \right) + 2^{-1} q \right)} \quad (28 \text{ d})$$

結局、有限操作誤差と有限語長誤差の両者を考察すると、図5のような修正 HT 法の誤差分布モデルが得

られる。出力ベクトル $[X_{n+1}, Y_{n+1}]^T$ は、式(28)で与えられる $\| [E_X, E_Y]^T \|_{\max}$ を半径とする円の中に入り、 $\| [E_X, E_Y]^T \|_{\max}$ が有限語長誤差の分布範囲を与える。また、以後 $[E_X, E_Y]^T$ のノルム $\| [E_X, E_Y]^T \|$ を有限語長誤差ノルムと呼ぶことにする。

4. パイプラインCORDIC アルゴリズムの設計

前章までの議論により、回転操作だけによって決められるCORDIC アルゴリズムの特性は、角度に対する有限操作誤差 $\Delta\theta$ と回転角の最大値 $|\theta|_{\max}$ の二つであることがわかる。また、回転操作と規格化操作によって決められるCORDIC アルゴリズムの特性は、ノルムに対する有限操作誤差であるノルム増加率 K と有限語長誤差ノルムの上限値 $\| [E_X, E_Y]^T \|_{\max}$ である。そこで、はじめに、許容できる $\Delta\theta$ の分布範囲と所望する $|\theta|_{\max}$ から、回転操作のシフト数列 $\{s(i)|e(i)=0\}$ を決定し、その後、許容できるノルム増加率 K の範囲から規格化操作のシフト数列 $\{s(i)|e(i)=1\}$ を決定する。最後に、有限語長誤差ノルムが最小となるように並べ換える。以上の決定を順に説明する。

4.1 回転操作に対するシフト数列の決定法

回転操作セルに対する、シフト数の集合は、

$$R = \{s(i)|e(j)=0\} \quad (29)$$

と表される。集合 R の要素の総数を L とし、大きいものから順に $r(0)$ から $r(L-1)$ とおく。本決定方法では、所望する回転角の範囲に入力角度 Z_0 が入っている場合に、 $|\Delta\theta|$ が $r(0)$ で定められる最も小さい回転角よりも等しいか小さくなるように R の要素を決めることにする。すなわち、アルゴリズムの実行終了後、 Z_n が

$$-\tan^{-1} 2^{-r(0)} \leq Z_n \leq \tan^{-1} 2^{-r(0)} \quad (30)$$

となるように R の要素を決める。以下、これを実現するための制約について説明する。

最後に実行される回転操作のセルインデックスが i であると仮定する。そのセルに $Z_i=0$ が入力された場合、式(30)を満たすためには、 $s(i)=r(0)$ でなくてはならない。そこで、最後に実行される回転操作のシフト数は $r(0)$ とする。また、 $s(i)=r(0)$ として、式(30)を満たす Z_i は

$$-2 \tan^{-1} 2^{-r(0)} \leq Z_i \leq 2 \tan^{-1} 2^{-r(0)}$$

の範囲となる。次に、最後から2番目に実行される回転操作のセルインデックスが k であると仮定する。そのセルに $Z_k=0$ が入力された場合に式(30)を満たす

には、シフト数 $s(k)$ が

$$\tan^{-1} 2^{-s(k)} \leq 2 \tan^{-1} 2^{-r(0)}$$

を満たしてはならない、また、 $r(0)$ が R の中で最大という条件より、

$$\tan^{-1} 2^{-r(0)} \leq \tan^{-1} 2^{-s(k)}$$

である、故に $s(k)$ は

$$r(0) - 1 \leq s(k) \leq r(0)$$

で示される範囲の値であり、 $s(k)$ が $r(1)$ であるのは明らかである。また、 $s(k) = r(1)$ として、式(30)を満たす Z_k は、

$$\begin{aligned} -\tan^{-1} 2^{-r(k)} - 2 \tan^{-1} 2^{-r(0)} \\ \leq Z_k \leq \tan^{-1} 2^{-r(k)} + 2 \tan^{-1} 2^{-r(0)} \end{aligned}$$

の範囲となる。

以上と同様に、シフト数に対する制約と Z の範囲を最後から $L-1$ 番目の回転操作、つまり、最初に実行される回転操作まで求めれば、式(30)を満たすためにはシフト数の集合 R の要素に

$$r(j) - 1 \leq r(j+1) \leq r(j) \tag{31}$$

で示される制約を課し、 $r(L)$ に対応する回転操作から j について降順に実行させればよいことがわかる。

また、式(30)を満たす入力角度 Z_0 は

$$-|\theta|_{\max} - \tan^{-1} 2^{-r(0)} \leq Z_0 \leq |\theta|_{\max} + \tan^{-1} 2^{-r(0)} \tag{32}$$

で示される範囲の値となる。

以上をもとに、回転角の範囲を決める $|\theta|_{\max}$ と、 $r(0)$ の大きさを決める許容できる有限操作誤差の最大値 $|\Delta\theta|_{\max}$ より、シフト数の集合 R の要素を決定する。

所望回転角の最大値 $|\theta|_{\max}$ は、 $\pi/2$ 以下とする。これ以上の回転角の範囲を所望する場合には、付録に述べる前処理をCORDICアルゴリズムに付け加える。また、式(29)では、同一の角度による2回以上の回転を許している。しかし、同一の角度による2回以上の回転操作によって、精度が向上しないことは明らかである。また、回転角の範囲の拡大のためには、同一の角度より大きな角度を選択する方が妥当である。そこで本決定法では

$$r(j+1) = r(j) - 1 \tag{33}$$

として2回以上の同一の角度による回転を禁止する。以下に決定法を述べる。

[回転シフト列 $r(j)$ の設計手順]

(R1) 角度に対する有限操作誤差の許容値 $|\Delta\theta|_{\max}$ より

$$r(0) = \min\{r \mid \tan^{-1} 2^{-r} < |\Delta\theta|_{\max}\} \tag{34}$$

を計算する。

(R2) 所要回転角の最大値 $|\theta|_{\max}$ が、 $\pi/2$ を超えるとき $L = r(0) - 1$ 、 $|\theta|_{\max} = \pi/2$ として、CORDICアルゴリズムに付録の前処理を加える。そうでないときは

$$L = \max\left\{l \mid \sum_{j=r(0)-l+1}^{r(0)} \tan^{-1} 2^{-j} > |\theta|_{\max}\right\} \tag{35}$$

を計算し、回転操作数 L を計算する。

(R3) 式(33)により、 $r(1)$ から、 $r(L-1)$ までを決定する。

この決定方法では、 $|\theta|_{\max}$ が大きいほど、また、 $|\Delta\theta|_{\max}$ が小さいほど全回転操作数 L は増加する。また、回転操作によるノルム伸張率 d_0 は回転に対するシフト数列によって決まり

$$d_0 = \prod_{j=0}^{L-1} \sqrt{1 + (2^{-r(j)})^2} \tag{36}$$

となる。

4.2 規格化操作に対するシフト数の決定法

規格化セルに対するシフト数の集合は、

$$C = \{s(j) \mid e(j) = 1\} \tag{37}$$

と表される。集合 C の要素の総数を N とし、要素の小さいものから順に、 $c(0)$ から $c(N-1)$ とおく。

さて、回転操作によりノルムが d_0 倍に引き伸ばされるため、規格化操作でノルムを $1/d_0$ 倍しなければならぬ。しかし、実際に $K=1$ を満たすためには、無限回の規格化操作が必要である。そこで、許容ノルム変動として $|\Delta r|$ を与え、CORDICに入力されるベクトルのノルムの最大値を r_{\max} とおいて、有限操作誤差 K が

$$|\Delta r| > r_{\max} |K - 1| \tag{38}$$

を満足したところで、規格化操作を打ち切ることにする。

[規格化シフト列 $c(j)$ の設計手順]

(c1) $j=0$ とする。

$$(c2) \quad |\Delta r| > r_{\max} |d_j - 1| \tag{39}$$

が成立するとき、 $N = j + 1$ として操作を終了する。そうでないときは(c3)からの操作を行う。

(c3) $c(j)$ の候補として、次の二つの数値

$$c(j, 1) = \min\{c \mid d_c(1 - 2^{-c}) \geq 1\} \tag{40 a}$$

$$c(j, 2) = \max\{c \mid d_c(1 - 2^{-c}) < 1\} \tag{40 b}$$

を計算する。

(c4) $c = c(j, 1)$ および $c = c(j, 2)$ とおいて不等式

$$|\Delta r| > r_{\max} |d_c(1 - 2^{-c}) - 1| \tag{41}$$

の成立の可否を調べる。すなわち、

(1) いずれの候補も式(41)を満足しない場合、 $c(j) = c(j, 1)$ とおいて(c5)へ進む。

(2) 両者とも式(41)を満足する場合、式(41)の右辺をより小さくする方を $c(j)$ として(c5)へ進む。

(3) どちらか一方だけが式(41)を満足する場合、満足させる候補を $c(j)$ として(c5)へ進む。

$$(c5) \quad d_{j+1} = d_j(1 - 2^{-c(j)}) \quad (42)$$

とし、 $j \leftarrow j+1$ として(c2)に戻る。

以上の手続きを終了すれば、

$$d_N = K \quad (43)$$

となり、式(38)が成立する。

4.3 操作順序の決定

操作セルの実行順序を、式(28)で示される有限語長誤差ベクトルの分布範囲を最小にする操作順序に決定する。

式(18)と式(27)より有限語長誤差ノルムの範囲は、切捨ての場合

$$\| [E_x, E_y]^T \| < \left(\sqrt{2q} \sum_{i=0}^{n-1} a_i \right) + \| [\varepsilon_n, \zeta_n]^T \| \quad (44a)$$

と表すことができる。まるめの場合

$$\| [E_x, E_y]^T \| \leq \left(\sqrt{2q} \sum_{i=0}^{n-1} a_i \right) + \| [\varepsilon_n, \zeta_n]^T \| \quad (44b)$$

と表すことができる。

式(44)から、操作セルの実行順序を、誤差伝搬率 a_i の $i=0$ から $n-1$ までの和を最小とする実行順序にすれば、有限語長誤差ベクトルの分布範囲は最小となることがわかる。分布範囲を最小とする実行順序を明らかにするために、任意の操作セル i に接続された雑音源からの誤差伝搬率 a_i を最小とする条件を導く。

雑音源出力 $[\varepsilon_i, \zeta_i]^T$ は $i+1$ 番目の操作セルから、出力端まで、 m 個の操作セルを通過するものとする。雑音源出力 $[\varepsilon_i, \zeta_i]^T$ に対応する誤差伝搬率 a_i が最小であるには、 $i+1$ 番目から出力端までの m 個の操作セルが、すべての操作セル n 個の中でノルム増加率 k_i が小さいものから m 個までのセルであればよい。

この条件を $i=0$ から $i=n-1$ の雑音源に対応する誤差伝搬率 a_i で同時に満足させるには、各操作セルのノルム増加率 k_i の関係が

$$k_i \geq k_{i+1} \quad (45)$$

但し、 $0 \leq i \leq n-2$

となるように実行順序を決めればよい。すなわちノルム増加率の大きい操作セルから順に実行する実行順序とすればよい。

式(45)を満たすように操作順序を決定すれば、 $i=0$ から $n-1$ までの誤差伝搬率 a_i を最小とすることができ、 $i=0$ から $n-1$ までの誤差伝搬率 a_i の和は最小

となる。

ノルム増加率を示す式(8)より、式(45)を満たすためには、シフトビット数の小さい回転操作セルから大きい回転操作セルへと順に実行し、すべての回転操作セルの実行が終了した後、シフトビット数の大きい規格化操作セルから小さい規格化操作セルへと順に実行すればよいことがわかる。また、この実行順序は、4.1で述べた回転操作セルの実行順序に対する制約を満たしている。

以上の議論より、セルインデックス i に対応する $e(i)$ を 1 と 0 から選び、シフトビット数 $s(i)$ を R と C の要素から選ぶと、式(1)の $e(i)$ と $s(i)$ は、 $i=0$ から $L-1$ までは、

$$e(i) = 0 \quad (46)$$

となり、シフトビット数 $s(i)$ は

$$s(i) = r(L-1-i) \quad (47)$$

となる。 $i=L$ から $i=L+N-1$ までは

$$e(i) = 1 \quad (48)$$

$$s(i) = c(L+N-1-i) \quad (49)$$

となる。この操作順序に従う修正 HT アルゴリズムを、特に PN (Post Normalization) アルゴリズムと呼ぶことにする。

この PN アルゴリズムでは回転操作がすべて終了した段階で、入力ベクトルのノルムが d_0 倍されることになる。その最大値 d_{\max} は

$$d_{\max} = \prod_{j=0}^{\infty} \sqrt{1 + (2^{-j})^2} \quad (50)$$

であり、2未満であることが知られている⁽⁶⁾。従って、PN アルゴリズム内部の数値表現を入出力の数値表現の MSB より、上位のけたを 1 ビット拡張すれば、オーバーフローは発生しない。

5. 設計例とそのシミュレーション

5.1 アルゴリズムの用途別設計

平面回転アルゴリズムの用途に応じて、下記 3 種類の PN アルゴリズムを設計した。入出力データは固定小数で、語長は 16 ビットである。

(1) 汎用 CORDIC アルゴリズム

このアルゴリズムでは、あらゆる回転角が指定でき、かつ、有限操作誤差による出力への影響が出力の LSB より小さくなるようにした。

(2) パワーウェーブデジタルフィルタのためのアルゴリズム

無損失散乱行列の分解から得られるパワーウェーブ

デジタルフィルタ^{(4),(5)}のためのアルゴリズムを設計した。パワーウェーブデジタルフィルタは低係数感度であるため、精密な回転角を必要としない場合があるが、回転角の範囲は $-\pi \sim \pi$ にわたる。そこで、回転操作数の削減を目的として、汎用アルゴリズムの仕様よりも大きな $|D|_{\max}$ を与えた。また、 $|D|_{\max}$ は汎用アルゴリズムの場合と同じ値を与えた。

(3) デジタル正弦波対発振器のためのアルゴリズム

低周波数発振を目的としたデジタル正弦波対発振器⁽¹¹⁾のためのアルゴリズムを設計した。最大発振周波数を $0.004\pi(\text{rad/sec})$ とし、回転角の許容誤差は汎用アルゴリズムと同一とした。

上記3種のアルゴリズムに対する仕様を表1に示し、前章の設計法によって得たアルゴリズムの回転角の範囲と、有限操作誤差の最大値を表2に示す。また、表3は、まるめと切捨ての場合の有限語長誤差ノルムの上限值を示している。表4~6は得られたアルゴリズムを記述している。いずれもPNアルゴリズムであり、内部データは、上位のけたを1ビットだけ拡張してある。また、表3に示した有限語長誤差ノルムの値は $M=0$ として計算した。

表1と表2を比較すると、本設計法によって得たアルゴリズムは、いずれの場合も表1の仕様を満足して

いることがわかる。また、汎用アルゴリズムと比較して、正弦波対発振器のアルゴリズムでは、有限操作誤差の最大値を大きく設定したため、セルの総数が削減されている。同様に、パワーウェーブフィルタのアルゴリズムでも、回転角の範囲を狭く設定したためにセル数が減少している。また、操作セル数の減少に伴い、雑音源の個数も減るため、まるめと切捨てのいずれの場合も有限語長誤差ベクトルの分布範囲が縮小している。これは、所要回転角の範囲と許容有限操作誤差を与えて必要な操作のみを選択する本設計法の有効性を示している。

5.2 シミュレーション

以下の条件により、前項(1)の汎用アルゴリズムによるベクトルの回転のシミュレーションを行い、出力に含まれる有限語長誤差 $[E_x, E_y]^T$ の分布を観察し

表2 アルゴリズムの特性

	$ D _{\max}$	$ D _{\max}$	$ D _{\max}$
(1)	0.5549π	$0.4857 \times 10^{-5}\pi$	0.1857×10^{-1}
(2)	0.5350π	$0.1987 \times 10^{-1}\pi$	0.9032×10^{-5}
(3)	$0.4969 \times 10^{-2}\pi$	$0.4857 \times 10^{-5}\pi$	0.1017×10^{-1}

- (1) 汎用アルゴリズム
- (2) パワーウェーブデジタルフィルタのためのアルゴリズム
- (3) デジタル正弦波対発振器のためのアルゴリズム

表1 アルゴリズムの仕様

	$ D _{\max}$	$ D _{\max}$	$ D _{\max}$
(1)	0.5π	$0.9714 \times 10^{-5}\pi$	0.3052×10^{-1}
(2)	0.5π	$0.2 \times 10^{-1}\pi$	0.3052×10^{-1}
(3)	$0.4 \times 10^{-2}\pi$	$0.9714 \times 10^{-5}\pi$	0.3052×10^{-1}

- (1) 汎用アルゴリズム
- (2) パワーウェーブデジタルフィルタのためのアルゴリズム
- (3) デジタル正弦波対発振器のためのアルゴリズム

表3 各アルゴリズムの有限語長誤差ノルムの最大値

	切捨て	まるめ
(1)	0.6124×10^{-3}	0.2428×10^{-3}
(2)	0.2996×10^{-3}	0.1861×10^{-3}
(3)	0.4741×10^{-3}	0.2371×10^{-3}

- (1) 汎用アルゴリズム
- (2) パワーウェーブデジタルフィルタのためのアルゴリズム
- (3) デジタル正弦波対発振器のためのアルゴリズム

表4 汎用アルゴリズム

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$e(i)$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
$s(i)$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	14	12	10	8	7	4	3	2

表5 パワーウェーブデジタルフィルタのためのアルゴリズム

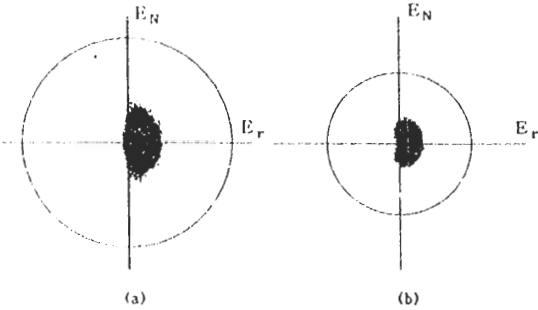
i	0	1	2	3	4	5	6	7	8	9	10	11	12
$e(i)$	0	0	0	0	0	1	1	1	1	1	1	1	1
$s(i)$	0	1	2	3	4	15	13	11	8	7	4	3	2

表6 デジタル正弦波対発振器のためのアルゴリズム

i	0	1	2	3	4	5	6	7	8	9	10
$e(i)$	0	0	0	0	0	0	0	0	0	0	1
$s(i)$	7	8	9	10	11	12	13	14	15	16	15

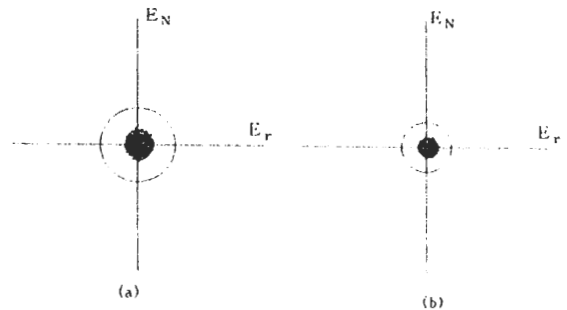
表7 Haviland, Tuszynskiのアルゴリズム

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$e(i)$	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0
$s(i)$	0	1	2	2	3	3	4	4	5	6	7	7	8	8	9	10	10	11	12	12	13	14	14	15	16



(a) Haviland and Tuszynski's algorithm
(b) PN algorithm

図6 切捨てによる有限語長誤差の分布
Fig. 6 Finite wordlength error distributions by truncation.



(a) Haviland and Tuszynski's algorithm
(b) PN algorithm

図7 まるめによる有限語長誤差の分布
Fig. 7 Finite wordlength error distributions by rounding.

た。
出力語長：16ビット，固定小数点
 小数点以下15ビット
内部語長：17ビット，固定小数点
 小数点以下15ビット
いずれも符号・絶対値表現とした。
入力ベクトル：入力ベクトルの要素 X_0, Y_0 に、それぞれ対応する乱数を1回のシミュレーションごとに発生し、ベクトルを構成する。その後、ベクトルのノルムを入力語長の許す最大値に規格化して入力する。

比較対象として、汎用アルゴリズムと同じ規格化操作と同転操作をもつ Haviland⁽⁷⁾らのアルゴリズムを選んだ。このアルゴリズムを修正 HT アルゴリズムで表現したものを表7に示す。

シミュレーションによって得られた有限語長誤差分布を、図6(切捨ての場合)と図7(まるめの場合)に示す。図8は分布図の座標系を示す。この座標系では、ベクトルのノルムに比べて有限操作誤差が十分小さいため、有限操作誤差による出力ベクトルのノルムの変動を E_r が示し、回転角の変動を E_N が示すと考えてよい。また、比較のために $\| [E_x, E_y]^T \|_{\max}$ を半径とする円を描いている。

シミュレーション結果より、いずれのアルゴリズムにおいても、有限語長誤差ノルムは $\| [E_x, E_y]^T \|_{\max}$ のほぼ1/2の範囲に入っていることがわかる。また、有限語長誤差の分布は $\| [E_x, E_y]^T \|_{\max}$ の大きさに比例し

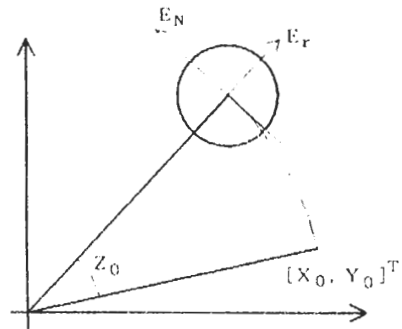


図8 有限語長誤差分布図の座標系
Fig. 8 Coordinate system for finite wordlength error distribution.

ており、式(28)は誤差分布範囲の推定に有効な近似値を与えると考えられる。Haviland らのアルゴリズムと比較して、PN アルゴリズムの誤差分布は2/3程度に縮小されており、有限語長誤差を最小とする操作順序に並べ換える本構成法の有効性を示している。

まるめの場合、誤差分布は真値を中心にして均等に分布している。これに対して切捨ての場合は、出力ベクトルのノルムを増加させる方向に偏って分布している。また、まるめの場合、誤差分布範囲が切捨ての場合のほぼ1/3となっている。

6. むすび

基本操作セルの縦続接続によって構成される、ハイ

ライン平面回転アルゴリズムを提案した。本アルゴリズムは、パイプライン処理が可能であり、データフローが隣接セル間で一方向系であること、構造が規則的なことを考慮すると VLSI 回路実現に好適と考えられる。また、本アルゴリズムの演算誤差を有限操作誤差と有限語長誤差の二つに分類して解析した。誤差発生 の検討をもとに、有限操作誤差の許容範囲と回転角の所要範囲から、必要な演算操作だけを選択し、等価出力雑音を最小とする操作順序を明らかにした。

最後に、CORDIC アルゴリズムの設計例により、本設計法の有効性を示し、コンピュータシミュレーションを通じて誤差分布を観察し、誤差解析による誤差分布モデルの有効性を確認した。今後の課題として、切捨ての場合の誤差分布に生じるバイアスの解決が挙げられる。また、パワーウェーブデジタルフィルタ、デジタル正弦波対発振器等への CORDIC アルゴリズムの応用の詳細に関しては別途報告する。

文 献

- (1) E. F. Deprettere and P. Dewilde: "Orthogonal cascade realization of real multiport digital filters", Int. J. Circuit Theory & Appl., 8, pp. 245-272 (1980).
- (2) A. Fettweis: "Wave digital filters: theory and practice", Proc. IEEE, 74, 2, pp. 270-327 (Feb. 1986).
- (3) P. P. Vaidyanathan and S. K. Mitra: "Low pass-band sensitivity digital filters: A generalized viewpoint and synthesis procedures", Proc. IEEE, 72, pp. 403-423 (April 1984).
- (4) H. Kikuchi, H. Watanabe, A. Nishihara and T. Yanagisawa: "Pipelined low-sensitivity digital filters based on localization of transmission zeros", Trans. IEICE, E71, 5, pp. 493-504 (May 1988).
- (5) H. Kikuchi, H. Watanabe, A. Nishihara and T. Yanagisawa: "Systematic synthesis of power-wave digital filters", Trans. IEICE, E72, 4, pp. 363-374 (April 1989).
- (6) J. E. Volder: "The CORDIC trigonometric computing technique", IRE Trans. Electr. Comp., EC-8, pp. 330-334 (Sept. 1959).
- (7) G. L. Haviland and A. A. Tuszyński: "A CORDIC arithmetic processor chip", IEEE Trans. Comput., C-29, 2, pp. 68-79 (Feb. 1980).
- (8) 中静 真, 菊池久和, 渡辺弘道: "演算誤差の検討と CORDIC アルゴリズムの構成", 信学技報, CAS88-140 (1989-03).
- (9) H. Kikuchi, M. Nakashizuka and H. Watanabe: "Computationally-efficient pipelined plane-rotation algorithms", Proc. Joint Tech. Conf. Circuits/Syst., Comput. & Commun., pp. 425-430, Sapporo (June 1989).
- (10) H. Kikuchi, M. Nakashizuka and H. Watanabe: "Error analysis and its application to the design of

CORDIC algorithms", Digest 8th Kobe Int. Symp. Electron. & Inf. Sci., pp. 7.1-7.9, Kobe (July 1989).

- (11) 中静 真, 菊池久和, 渡辺弘道: "CORDIC に基づくデジタル正弦波対発振器", 信学技報, CAS88-32 (1988-07).

付 録

CORDIC への入力角度 $|Z_0|$ が $\pi/2$ を超える場合,

$$X_0 \leftarrow uY_0$$

$$Y_0 \leftarrow uX_0$$

$$Z_0 \leftarrow Z_0 + \pi/2$$

$$u = -\text{sign}(Z_0)$$

による回転角が $-\pi/2$, 若しくは $\pi/2$ の回転を, CORDIC アルゴリズムの前処理として行う。この前処理を加えることで, CORDIC アルゴリズムの $|0|_{\max}$ を $\pi/2$ とすることで, すべての回転角を実現できる。また, この式を実現する回路は, 図 1 に示した回転操作セルのシグナルフローを一部変更するだけで実現できる。

(平成 2 年 2 月 13 日受付, 6 月 11 日再受付)

菊池 久和



昭 49 新潟大・工・電子卒, 昭 51 同大学院修士課程了, 同年富士通(株)入社, 昭 54 新潟大・工助手, 同講師を経て平元同助教授(電気電子), 工博, デジタル信号処理, 電子回路, 光波フィルタの研究に従事, IEEE 会員。

中静 真



昭 63 新潟大・工・電子卒, 現在, 同大学院修士課程在学中, デジタル信号処理, 特に平面回転アルゴリズムとその応用に関する研究に従事。

渡辺 弘道



昭 42 東工大・工・電子卒, 昭 49 同大学院博士課程了, 工博, 同年新潟大・工・電子助手, 昭 53 助教授, 現在, 同電気電子助教授, 能動 RC 伝送回路網, スイッチトキヤパシクフィルタ等, 電子回路, 回路網理論の研究に従事。