基数2の FFT に基づく任意項数の離散型 Fourier 変換†

鳥 居 達 生†† 杉 浦 洋††

任意項数の多項式 $f(z)=c_0+c_1z+\cdots+c_{N-1}z^{N-1}$ の係数を与えて，$N=2^{n_1}+2^{n_2}+\cdots+2^{n_m}$ 個の標本 $f(\exp(2\pi i(j+1/2)/2^{n_l}))$, $0\le j<2^l$, $1\le l\le m$ を求める．この際，N の2進表示に対応して項数が2のべき 2^{n_l} の中点公式による FFT が使われる．この逆変換は，$f(z)$ に対する一つの補間法であるが，計算量（複素数乗算回数）は，いずれも $(N\log_2 N)/2$ である．また，この補間法の収束性および数値的安定性を解析し，本算法が任意項数の（広義の）離散型 Fourier 変換の高速算法の名に値することを示す．

1. まえがき

離散型 Fourier 変換の高速算法を古い順に挙げるならば，Lanczos[1),2)], Good[3)], Cooley-Tukey[4)] および Winograd[5)] の方法であろう．いずれの方法においても変換の項数Nの因数分解の形式に算法は依存する．一般にNの因数分解の可能性と算法の高速性は相反する関係にある．すなわち項数 N が $r_1r_2\cdots r_n$ と因数分解されるとき，計算量は $N(r_1+r_2+\cdots+r_n)$ に比例する．Lanczos 法は，元来 sine, cosine 変換であるがNは2のべきに限られる．さらに算法の数値的安定性が，高速 sine 変換，cosine 変換[6),8)]に比べ若干わるい．

Good の方法は，Cooley-Tukey 法すなわち FFT の前身であるが，Nの因数 $r_1, r_2, \cdots, r_n$ が互いに素でなければならない．この方法においては，FFT における位相回転の操作が不要なので，その分だけ FFT に比べ計算量を少なくでき，特別な項数Nの場合，現在でも有効な算法である[9)]．

Winograd の方法は，FFT に比べ乗算回数が約 1/5 になるということであるが，加減算の回数はかえって増えること，標本数の制約が厳しいこと，プログラムが複雑なこと，さらに中間結果のための作業領域が大きいことから計算の手間は期待されるほど少なくならない[9)]．

このような事情から，離散型 Fourier 変換の算法として項数が2のべきの FFT が最もよく用いられている．FFT の高速性を犠牲にせずに項数が2のべきという制約を撤廃できるならばきわめて好都合である．そこで，われわれは離散型 Fourier 変換を次に述べるように拡大解釈して，その変換あるいは逆変換を高速に行う．

周知のように周期 2π の複素数値関数 $X(t)$ のN項離散型 Fourier 変換は

$$C_k=\frac{1}{N}\sum_{0\le j<N}X\left(\frac{2\pi}{N}j\right)e^{-2\pi ikj/N},\quad 0\le k<N \tag{1.1}$$

で定義される．これに伴い，N項多項式

$$p(z)=\sum_{0\le k<N}C_kz^k \tag{1.2}$$

は，複素平面上の単位円周上に等間隔に補間点が分布する補間式となる．いうまでもなく被近似関数は

$$f(z)=X(t),\quad z=e^{it}$$

である．

N 次元線形変換(1.1)の条件数は2乗ノルムの下で1である．

われわれは離散型 Fourier 変換の定義を次のように拡張する．

［定義］ $N-1$ 次の多項式 $f(z)$ の単位円周上のN個の標本から $f(z)$ のべき級数表示のN個の係数を求める線形変換が，補間法であって，かつその条件数が $O(N^p)$, $0\le p\le 1$ ならば，この変換を $f(z)$ の離散型 Fourier 変換という．

丸め誤差の伝播が $O(N)$ となる算法は，通常準安定とされるから上のように離散型 Fourier 変換を拡張された意味で用いることは許されるであろう．

実際，Lanczos 法もその一例であるが，複素 FFT を用いて sine 変換，cosine 変換を行う高速算法においては，$O(N)$ までの不安定性は許されている[7)]．ここで述べる算法の不安定性を条件数で測るならば $O(\sqrt{N})$ である．したがって離散型たたみ込み演算への応用は，有効である．

---

† An Algorithm for Discrete Fourier Transform of Arbitrary Length Based on the FFT with Radix 2 by TATSUO TORII and HIROSHI SUGIURA (Department of Information Science, Faculty of Engineering, Nagoya University).

†† 名古屋大学工学部情報工学科

2. 算　法

2.1 逆 変 換

複素係数のN項多項式 $f(z)$ をベクトル表示して $\tilde{\boldsymbol{f}}$ とおく.

$$\left.\begin{aligned}&f(z)=c_0+c_1z+\cdots+c_{N-1}z^{N-1}\\&\tilde{\boldsymbol{f}}=(c_0, c_1, \cdots, c_{N-1})^{\mathrm{T}}\end{aligned}\right\}\tag{2.1}$$

項数Nはnビットとする. これを2進法で展開し

$$\begin{aligned}&N=2^{n_1}+2^{n_2}+\cdots+2^{n_m}\\&\quad 0\leq n_1<n_2<\cdots<n_m=n-1\end{aligned}\tag{2.2}$$

と表す. 指数の集合

$$\Lambda(N)=\{n_1, n_2, \cdots, n_m\}$$

は, Nの2進法による一つの圧縮表現となる. 以下 $\Lambda(N)$ を Λ と略記する.

われわれがいう逆変換の高速算法とは, $f(z)$ の係数ベクトル $\tilde{\boldsymbol{f}}$ を与えて, $f(z)$ の単位円周上等間隔分布標本の族

$$\begin{aligned}&\boldsymbol{f}=\{f_l\,;\ l\in\Lambda\},\\&\boldsymbol{f}_l=\{f(z)\,;\ z=e^{2\pi i(j+1/2)/2^l}, 0\leq j<2^l\}\end{aligned}\tag{2.3}$$

を高速に求める算法のことである.

N項多項式 $f(z)$ をいったん

$$f(z),\quad \mathrm{mod}(z^{2^l}+1),\quad l\in\Lambda$$

により, 2^l 項の多項式に圧縮し, これに対し 2^l 項の中点公式による逆 FFT を適用すれば, l ブロックの標本 $\boldsymbol{f}_l$ が得られる. 同様な操作を, すべての $l\in\Lambda$ について繰り返せば, 求める標本 $\boldsymbol{f}$ が得られる.

円周等分多項式 $z^{2^l}+1$ を法として $f(z)$ と合同をとる上の演算は, それと双対的な $z^{2^l}-1$ を用いる演算を介入させることにより計算量を節減できる.

さて, $z^{2^l}\pm1$ を法として合同をとる演算を行列表示して

$$\left.\begin{aligned}&\tilde{M}_l\tilde{\boldsymbol{f}}=f(z), \mathrm{mod}(z^{2^l}+1)\\&\tilde{T}_l\tilde{\boldsymbol{f}}=f(z), \mathrm{mod}(z^{2^l}-1)\end{aligned}\right\}\tag{2.4}$$

とおけば, $\tilde{M}_l, \tilde{T}_l$ は長方形行列

$$\tilde{M}_l=[I_{2^l}, -I_{2^l}, \cdots]$$
$$\tilde{T}_l=[I_{2^l}, I_{2^l}, \cdots]$$

I_{2^l} ; 2^l 次の単位行列

で表される. これらの行列の列数は行列とベクトルの積が整合するよう適当にとる. いまの場合 $(2^l, N)$ 型行列と解釈する.

円周等分多項式 $z^{2^l}+1$ の零点上における $f(z)$ の標本に対する中点公式に基づく 2^l 項離散型 Fourier 変換を M_lf, $z^{2^l}-1$ の零点上における $f(z)$ の標本に対する台形公式に基づくその変換を T_lf とすれば

$$\left.\begin{aligned}&M_lf=\tilde{M}_l\tilde{\boldsymbol{f}}\\&T_lf=\tilde{T}_l\tilde{\boldsymbol{f}}\end{aligned}\right\}\tag{2.5}$$

が成り立つ. いうまでもなく, M_l, T_l は正規化すればユニタリ行列となる.

算法を記述する上で必要な作用素は M_l, M_l^{-1}, $\tilde{M}_l$ および $\tilde{T}_l$ である. $\tilde{M}_l, \tilde{T}_l$ の性質を示す.

[補題 1]　線形変換 $\tilde{M}_l, \tilde{T}_l$ は以下の性質をもつ.

1)　$f(z)$ の次数が 2^l より小ならば, $\tilde{M}_l, \tilde{T}_l$ は恒等変換となる.

$$\tilde{T}_l\tilde{\boldsymbol{f}}=\tilde{M}_l\tilde{\boldsymbol{f}}=\tilde{\boldsymbol{f}}$$

2)　任意の多項式 $f(z)$ に対し

$$\tilde{T}_l\tilde{\boldsymbol{f}}=\tilde{T}_l\tilde{T}_{l+1}\tilde{\boldsymbol{f}}$$
$$\tilde{M}_l\tilde{\boldsymbol{f}}=\tilde{M}_l\tilde{T}_{l+1}\tilde{\boldsymbol{f}}$$

3)　$\tilde{T}_{l+1}\tilde{\boldsymbol{f}}=\dfrac{1}{2}(\tilde{T}_l+\tilde{M}_l)\tilde{\boldsymbol{f}}+\dfrac{1}{2}z^{2^l}(\tilde{T}_l-\tilde{M}_l)\tilde{\boldsymbol{f}}$

4)　任意の二つの多項式 $f(z), g(z)$ の積 $\tilde{\boldsymbol{f}}\cdot\tilde{\boldsymbol{g}}$ に対し

$$\tilde{T}_l(\tilde{\boldsymbol{f}}\cdot\tilde{\boldsymbol{g}})=\tilde{T}_l(\tilde{T}_l\tilde{\boldsymbol{f}}\cdot\tilde{T}_l\tilde{\boldsymbol{g}})$$
$$\tilde{M}_l(\tilde{\boldsymbol{f}}\cdot\tilde{\boldsymbol{g}})=\tilde{M}_l(\tilde{M}_l\tilde{\boldsymbol{f}}\cdot\tilde{M}_l\tilde{\boldsymbol{g}})$$

簡単だから証明は略す.

以上の記法を用いると, 逆変換は

$$\begin{aligned}&\tilde{\boldsymbol{f}}_l=\tilde{M}_l\tilde{\boldsymbol{f}}\\&\boldsymbol{f}_l=M_l^{-1}\tilde{\boldsymbol{f}}_l,\quad l\in\Lambda\end{aligned}\tag{2.6}$$

と書ける. ここで補題の性質2) より

$$\tilde{\boldsymbol{f}}_l=\tilde{M}_l(\tilde{T}_{l+1}\tilde{\boldsymbol{f}}),\quad \tilde{T}_l\tilde{\boldsymbol{f}}=\tilde{T}_l(\tilde{T}_{l+1}\tilde{\boldsymbol{f}})$$

であるから, 補助的に

$$g_l(z)=f(z),\quad \mathrm{mod}(z^{2^l}-1)$$

を用いるならば能率的である.

[逆変換の算法]

初期値,　$\tilde{\boldsymbol{g}}_n=\tilde{\boldsymbol{f}}$,　$\tilde{\boldsymbol{f}}_{n-1}=\tilde{M}_{n-1}\tilde{\boldsymbol{f}}$

以下の手順1), 2) を $l=n-2, n-3, \cdots, \mu$ について繰り返す. ただし $\mu=\min l$, $l\in\Lambda$.

1)　$\tilde{\boldsymbol{g}}_{l+1}=\tilde{T}_{l+1}\tilde{\boldsymbol{g}}_{l+2}$

2)　$l\notin\Lambda$ ならばただちに 1) へ戻る.

$l\in\Lambda$ ならば

$$\tilde{\boldsymbol{f}}_l=\tilde{M}_l\tilde{\boldsymbol{g}}_{l+1},\quad \boldsymbol{f}_l=M_l^{-1}\tilde{\boldsymbol{f}}_l$$

以上によって, $f(z)$ の単位円周上の N 個の標本 $\boldsymbol{f}$ が求められる.

計算量をしらべる. 初期値の設定および手順2) における 2^l 項の逆 FFT, M_l^{-1} が計算量の主要部である. したがって, そのために要する複素数乗算回数は

$$\sum_{l\in\Lambda} l2^{l-1}<\frac{N}{2}\log_2 N$$

となり，加減算の回数は，この約2倍である．残りの演算は加減算だけであってたかだか $2N$ 回である．

2.2 順変換

前節では多項式 $f(z)$ の係数ベクトル $\tilde{\boldsymbol{f}}$ を与えて，単位円周上の標本 $\boldsymbol{f}$ を求めた．本節では，この逆を考える．すなわち $\tilde{\boldsymbol{f}}$ に関する N 元連立一次方程式

$$\tilde{M}_l\tilde{\boldsymbol{f}}=M_l\boldsymbol{f}_l,\quad l\in\Lambda \tag{2.7}$$

を解けばよい．ここで，標本の族 $\{\boldsymbol{f}_l, l\in\Lambda\}$ は(2.3)で与えられたものである．上式を解くことは補間多項式を求めることにほかならない．この補間式を，われわれは広義 Newton 補間式として表す．

広義 Newton 補間のため新しい記号を定義する．$f(z)$ の項数 N の2進表示(2.2)に対応して

$$\Lambda_l=\Lambda\cap\{k;\ k\geq l\},\ l\geq 0$$

とおく．集合 Λ_l の濃度を $|\Lambda_l|$，2進数 N のノルム $\|N\|$ を，その展開係数和で定義する．明らかに $\|N\|=|\Lambda|$ である．

また

$$\omega(\Lambda_l\,;\,z)=\prod_{k\in\Lambda_l}(z^{2^k}+1)$$

とおく．とくに多項式系 $\{\omega(\Lambda_l\,;\,z)\,;\,l\in\Lambda\}$ を広義 Newton 補間の基底とよぶ．

多項式 $\omega(\Lambda_l\,;\,z)$ の零点上における $f(z)$ の補間式を

$$f(\Lambda_l\,;\,z)=f(z),\ \bmod\,\omega(\Lambda_l\,;\,z)$$

とおく．これを簡単に $f(z)$ の指数化された標本点集合族 Λ_l 上の補間式という．

Λ_l 上の $f(z)$ の差分商を

$$f[\Lambda_l\,;\,z]=\frac{f(z)-f(\Lambda_l\,;\,z)}{\omega(\Lambda_l\,;\,z)}$$

で定義する．$f(z)$ が多項式ならば，差分商も多項式であるので，これを 2^k 項に圧縮する．

$$f[\Lambda_l\,;\,z]_k=f[\Lambda_l\,;\,z],\ \bmod\,(z^{2^k}+1)$$

これを $f(z)$ の 2^k 階差分商とよぶ．

表記を簡単にするため誤解のおそれがなければ，変数 z を省略する．たとえば $f(\Lambda_l\,;\,z)$ を $f(\Lambda_l)$ と書く．

補間式 $f(\Lambda_l\,;\,z)$ と差分商の定義により次の漸化式が成り立つ．

$$f(\Lambda_l)=\begin{cases}f(\Lambda_{l+1})+\omega(\Lambda_{l+1})f[\Lambda_{l+1}]_l, & l\in\Lambda\\ f(\Lambda_{l+1}), & l\notin\Lambda\end{cases} \tag{2.8}$$

ここで用いる差分商は，$l\in\Lambda$ のとき

$$f[\Lambda_{l+1}]_l=2^{-|\Lambda_{l+1}|}(f_l(z)-g_l(z)) \tag{2.9}$$

ただし

$$f_l(z)=M_l f$$
$$g_l(z)=h_{l+1}(z),\ \bmod\,(z^{2^l}+1)$$
$$h_{l+1}(z)=f(\Lambda_{l+1}),\ \bmod\,(z^{2^{l+1}}-1)$$

で求められる．これは，恒等式

$$f(z)=f(\Lambda_l\,;\,z)+\omega(\Lambda_l\,;\,z)f[\Lambda_l\,;\,z] \tag{2.10}$$

の両辺に離散型 Fourier 変換 M_l を作用させることにより導かれる．

いま，Λ_n は ϕ だから，便宜上

$$f(\Lambda_n)=0,\ \omega(\Lambda_n)=1$$

とおけば，漸化式(2.8)より $f(z)$ の広義 Newton 補間式の展開

$$f(z)=f(\Lambda_0)$$
$$=\sum_{l\in\Lambda}\omega(\Lambda_{l+1})f[\Lambda_{l+1}]_l \tag{2.11}$$

が得られる．通常の Newton 補間法においては，標本点を1点ずつ追加しいくらでも高次の補間式を形成できるが，本方法では補間点の個数 N の2進表示に対応して展開の項数は $\|N\|$ と一意に定まる．

計算量を節減する立場から差分商の計算において，補助的に用いられる多項式 $h_l(z)$ の計算法について説明を追加する．

補間式(2.8)の両辺に $\tilde{T}_l$ を作用させると

$$\tilde{\boldsymbol{h}}_l=\begin{cases}\tilde{\boldsymbol{f}}_l+(\tilde{T}_l-\tilde{M}_l)h_{l+1}, & l\in\Lambda\\ \tilde{T}_l\tilde{h}_{l+1}, & l\notin\Lambda\end{cases} \tag{2.12}$$

が成り立つ．

これは，補題1，2)，4)と(2.9)から導かれる．

なお，ここで $\tilde{h}_{l+1}$ の係数行列は

$$\tilde{T}_l-\tilde{M}_l=2[0,\ I_{2^l}]$$

と簡単な形をしていることを注意しておく．

残る問題は，$\|N\|$ 項の広義 Newton 補間式を N 項のべき級数に再編成することである．Newton 補間式は入れ子形式で書くことができるので，これを括弧の内側からべき級数に展開してゆけばよい．換言すれば，差分商に関する漸化式

$$f[\Lambda_{l+1}]=\begin{cases}f[\Lambda_{l+1}]_l+(z^{2^l}+1)f[\Lambda_l], & l\in\Lambda\\ f[\Lambda_l], & l\notin\Lambda\end{cases} \tag{2.13}$$

を上に辿り $f[\Lambda_n\,;\,z]=f(z)$ を求めることである．

上式の成立は，(2.8)，(2.10)を用いて示される．

［順変換の算法］

広義 Newton 補間のための初期値，$h_n(z)=f(z)$

以下の手順1)，2)を $l=n-1, n-2, \cdots, \mu\ (\mu=\min l,\ l\in\Lambda)$ の順に繰り返す．

1) $l\in\Lambda$ ならば

$f_l(z)=M_l f$

$g_l(z)=\tilde{M}_l \boldsymbol{h}_{l+1}$

$h_l(z)=\tilde{\boldsymbol{f}}_l+(\tilde{T}_l-\tilde{M}_l)\boldsymbol{h}_{l+1}$

$f[\Lambda_{l+1}]_l=2^{-|\Lambda_{l+1}|}(f_l(z)-g_l(z))$

2) $l \notin \Lambda$ ならば

$h_l(z)=\tilde{T}_l \boldsymbol{h}_{l+1}$

以上で，差分商 $f[\Lambda_{l+1}]_l$, $l\in\Lambda$ が生成された.

べき級数展開のための初期値, $f[\Lambda_{\mu+1}]=f[\Lambda_{\mu+1}]_\mu$.

以下の手順3), 4) を $l=\mu+1, \mu+2, \cdots, n-1$ の順に繰り返す.

3) $l\in\Lambda$ ならば

$f[\Lambda_{l+1}]=f[\Lambda_{l+1}]_l+(z^{2^l}+1)f[\Lambda_l]$

4) $l\notin\Lambda$ ならば

$f[\Lambda_{l+1}]=f[\Lambda_l]$

求める結果は $f[\Lambda_n]$ として与えられる.

プログラム上の注意を記す. 手順1) において $h_l(z)$ だけは，最終段の計算を省略できる. 項数が2のべきの FFT, $M_l f$ において，順変換は逆変換に比べ正規化のために実数の掛け算が項数だけ余分に必要となる.

そこで，差分商の計算において $2^{-|\Lambda_{l+1}|}$ と $f_l(z)$ の掛け算と $f_l(z)$ のための正規化を同時に行えば，計算量は若干少なくなる. そのためには補助として用いる多項式 $h_l(z)$ を $h_l(z)/2^{|\Lambda_l|}$ とすればよい.

標準的な FFT のための計算量を除外して本方法特有の計算量は，広義 Newton 補間のために実数掛け算 N 回（正規化の演算は含めない），加減算が $2N+2^{n-1}$ 回，べき級数展開のために加減算がたかだか N 回である.

加減算は両者合わせてたかだか $4N$ 回である.

以上述べた方法より若干計算量は多くなるが，中国の剰余定理に基づく高速算法もある[10].

3. 打切り誤差

複素平面上の単位閉円板上で解析的な関数 $f(z)$ を被近似関数とする前述の補間法の収束性について述べる.

標本数 N の2進表示から生成される補間点集合の族 $\Lambda=\Lambda(N)$ 上の補間式 $f(\Lambda ; z)$ の打切り誤差は，(2.10) より

$$f(z)-f(\Lambda ; z)=\omega(\Lambda ; z)f[\Lambda ; z] \tag{3.1}$$

と表される.

いま，関数 $f(z)$ のノルムを

$$\|f\|=\max_{|z|=1}|f(z)| \tag{3.2}$$

で定義する.

差分商に対する複素平面上の平均値の定理より

$$|f[\Lambda ; z]|\leq\frac{1}{N!}\|f^{(N)}\| \tag{3.3}$$

となる評価が知られている.

さらに

$$\max_{|z|=1}|\omega(\Lambda ; z)|=2^{\|N\|} \tag{3.4}$$

に注意すれば，考えている補間式の打切り誤差は

$$\|f(z)-f(\Lambda(N) ; z)\|\leq\frac{2^{\|N\|}}{N!}\|f^{(N)}\| \tag{3.5}$$

で評価される.

とくに，$f(z)=z^N$ ならば，上の不等式は等号で成り立つから最良である.

次に，$f(z)$ の Taylor 展開

$$f(z)=\sum_{k=0}^{\infty}c_k z^k \tag{3.6}$$

の係数 c_k と，その補間式 $f(\Lambda ; z)$ の補間係数の関係について述べる. これは，Fourier 係数と離散型 Fourier 係数の関係の一般化である.

補間法は，被近似関数に関する線形演算であるから z^k の打切り誤差

$$z^k-(z^k, \bmod\ \omega(\Lambda ; z)),\ k\geq 0 \tag{3.7}$$

について考えればよい.

次数 k の動く範囲を三つに分ける. まず $k<N$ ならば，補間法は恒等変換となり誤差は0である.

標本数 N が n ビットであるのに対し，次数 k のビット数が $n+1$ より大ならば，k の2進表示の $n+1$ 位以上は無視して n ビットとすればよい. なぜならば

$$\begin{aligned}&z^k, \bmod\ \omega(\Lambda ; z)\\&\quad=(z^k, \bmod\ (z^{2^n}-1)), \bmod\ \omega(\Lambda ; z)\end{aligned} \tag{3.8}$$

が成り立つからである. したがって $N\leq k\leq 2^n$ なる z^k だけ考えればよい.

さて，k の2進表示

$k=k_0 2^0+k_1 2^1+\cdots+k_{n-1}2^{n-1}$

の上位を打切り下位 l ビットの数を

$k|_l=k_0 2^0+k_1 2^1+\cdots+k_{l-1}2^{l-1}$

で表す. また

$\bar{\Lambda}_l=\Lambda-\Lambda_l$

とおく.

単項式 z^k を円周等分多項式 $z^{2^{n-1}}+1$ で割ったとき，商も余りも単項式で

$$\begin{aligned}&z^k, \bmod\ \omega(\bar{\Lambda}_n ; z)\\&\quad=z^{k'}+(z^{2^{n-1}}+1)\cdot(-z^{k'}, \bmod\ \omega(\bar{\Lambda}_{n-1} ; z))\end{aligned} \tag{3.9}$$

ただし　$k'=k|_{n-1}$

が成り立つ．ここで k の n ビットは0でないことが効いている．たかだか $n-1$ ビットの数 k' をあらためて k と考え同様な操作を続ける．すなわち $k|_{l+1}$ を，あらためて k とおき，k の $l+1$ ビットが0でなければ

$$z^k,\ \mathrm{mod}\,\omega(\bar{\Lambda}_{l+1};z)$$
$$=z^{k'}+(z^{2^l}+1)\cdot(-z^{k'},\mathrm{mod}\,\omega(\bar{\Lambda}_l;z)) \quad (3.10)$$

となる．ここで $k'=k|_l$，かつ $l\in\Lambda$．

k の $l+1$ ビットが0ならば

$$z^k,\ \mathrm{mod}\,\omega(\bar{\Lambda}_{l+1};z)=z^{k'} \quad (3.11)$$

となり終了である．

このようにして z^k の標本点集合の族 Λ 上での補間式を導くことが可能となる．これより，(3.8)左辺の多項式の単位円周上での最大値は $2\|N\|-1$ となる．

この評価は，任意の k に対して成り立つので，いま考えている補間式の打切り誤差について次の定理を得る．

［定理 1］　単位閉円板 $|z|\leq 1$ 上で，解析的な関数のべき級数展開を

$$f(z)=c_0+c_1z+c_2z^2+\cdots$$

とするとき，補間点数 N の2進表示

$$N=\sum_{l\in\Lambda}2^l$$

からつくられる補間点集合族 Λ 上の補間式 $f(\Lambda;z)$ の打切り誤差は，単位閉円板上で

$$|f(z)-f(\Lambda;z)|\leq 2\|N\|\sum_{k=N}^{\infty}|c_k|$$

で評価される．

この不等式は最良である．実際，$f(z)=z^N$ のとき，上式は等号で成立する．

解析関数 $f(z)$ のべき級数を N 項で打ち切ったとき，単位閉円板内で，その誤差は $|c_N|+|c_{N+1}|+\cdots$ となるが，$f(z)$ を前述の N 点補間で近似したときの誤差は，前者の $2\|N\|$ 倍にすぎない．とくに標本数 N が実質1ビットの数 2^{n-1} ならば，2倍であり，N が実質2ビットの数 $2^{n-1}+2^{n-2}$ ならば4倍となる．とくに前者の場合，Fourier 級数と離散型 Fourier 級数の相互の関係としてよく知られた結果である．

4. 数値的安定性

ここでは算法の安定性を用いる線形変換の条件数で測る．N 個の標本から N 個の補間係数を求める線形変換 $V_N^{-1}\boldsymbol{f}=\hat{\boldsymbol{f}}$ の数値的安定性について述べる[12]．

項数が 2^l の FFT，M_l あるいは T_l は正規化すればユニタリ変換であるから，順変換(2.7)の条件数は，$(2^l,N)$ 型行列を $\|N\|$ 個並べて得られる N 次行列

$$\tilde{V}_N=(2^{l/2}\tilde{M}_l),\ l\in\Lambda \quad (4.1)$$

の条件数に等しい．ただし行列のノルムはベクトルの2乗ノルムに従属する．

［例 1］　項数 N が $6=2^2+2^1$ の場合，行列 $\tilde{V}_N$ は，次のような形式となる．

$$\tilde{V}_6=\begin{bmatrix} 2 & 0 & 0 & 0 & -2 & 0 \\ 0 & 2 & 0 & 0 & 0 & -2 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ \sqrt{2} & 0 & -\sqrt{2} & 0 & \sqrt{2} & 0 \\ 0 & \sqrt{2} & 0 & -\sqrt{2} & 0 & \sqrt{2} \end{bmatrix}$$

さて，行列 $\tilde{V}_N=(v_{ij})$ の番号 i,j を，便宜上0から始めて，$0\leq i,j<N$ とする．

N が偶数ならば

$$v_{ij}=v_{i+1,j+1}$$
$$v_{i+1,j}=v_{i,j+1}=0 \quad (4.2)$$
$$i=0,\ j=0,\ \mathrm{mod}\,2$$

が成り立つ．

これは，$\tilde{V}_N$ の要素である行列 M_l の特殊性(2.4)から導かれる．

そこで，$\tilde{V}_N$ の要素の互換

$$\binom{2i}{i},\ \binom{2i+1}{\dfrac{N}{2}+i} \quad (4.3)$$

を行う．列番号 j についても同様，再び $\tilde{V}_N$ の定義から，互換(4.3)に対応する置換行列を P とすれば

$$P\tilde{V}_NP^{\mathrm{T}}=\sqrt{2}\begin{bmatrix}\tilde{V}_{N/2} & 0\\ 0 & \tilde{V}_{N/2}\end{bmatrix} \quad (4.4)$$

と表され，$\tilde{V}_N$ は，ブロック対角行列(4.4)と相似となる．これより次の定理を得る．

［定理 2］　N を任意の正の整数とする．項数 N の離散型 Fourier 変換(2.7)において，その条件数は項数を2倍にしても変わらない．すなわち

$$\|V_N\|\|V_N^{-1}\|=\|V_{2N}\|\|V_{2N}^{-1}\|$$

つぎに，変換行列 V_N の特異値を用いて，その条件数を評価しよう．

［補題 2］　任意の正規直交行列 A を対角ブロックが正方行列となるように分割して

$$A=\begin{bmatrix}A_1 & B\\ C & A_2\end{bmatrix}$$

とする．A_2 の特異値が $\sigma_1,\sigma_2,\cdots,\sigma_m$ ならば，A_1 の特異値は $\sigma_1,\sigma_2,\cdots,\sigma_m,1,\cdots,1$ でつくされる．ただし

A_2 の次数 m は，A_1 のそれより小とする．

（証明） Aの次数をnとする．m次行列 A_2 の特異値分解を $U_2\Sigma_2V_2^T$ とするとき

$$\begin{bmatrix} I_1 & O \\ O & U_2 \end{bmatrix} A \begin{bmatrix} I_1 & O \\ O & V_2^T \end{bmatrix} = \begin{bmatrix} A_1 & B' \\ C' & \Sigma_2 \end{bmatrix} \qquad (4.5)$$

も，直交行列となる．したがって B' および C' の特異値は $\sqrt{1-\sigma_k^2}$，$1\le k\le m$ となる．$[A_1 \ \ B']$の行ベクトルは正規直交系だから A_1 は，特異値 $\sigma_k, 1\le k\le m$ をもち，残り $n-m$ 個の特異値は1となる．

（証明終）

［定理 3］ Nをnビットの任意の奇数とするとき，項数Nの変換(2.7)の条件数は $2^{n/2}$ である．すなわち

$$\|V_N\|\|V_N^{-1}\|=2^{n/2}$$

（証明） N次行列 $\tilde{V}_N$ を 2^n 次の直交行列に次のように埋め込む．

$(2^l, 2^n)$ 型行列

$$2^{l/2}\tilde{M}_l=2^{l/2}[I_{2^k}, -I_{2^l}, \cdots]$$

$$l=n-1, n-2, \cdots, 1, 0$$

および $(1, 2^n)$ 行列 $(1, 1, \cdots, 1)$ を順次並べて

$$\begin{bmatrix} & 2^{(n-1)/2}\tilde{M}_{n-1} & \\ & \vdots & \\ & 2^{0/2}\tilde{M}_0 & \\ 1 & 1\cdots\cdots\cdots\cdots & 1 \end{bmatrix} \qquad (4.6)$$

とおけば，これは 2^n 次の直交行列で，各行ベクトルの長さは等しい．この左上隅に $\tilde{V}_N$ が現れるよう適当に行交換を行い

$$\begin{bmatrix} \tilde{V}_N & X \\ Y & Z \end{bmatrix}$$

とおく，N行から 2^n-2 行までの長方形行列の要素 $2^{l/2}\tilde{M}_l$，$l\in\Lambda$ は，番号 l の大きさの順に並べられ最後の行 $(2^n-1$ 行$)$ は $(1, 1, \cdots, 1)$ とする．

行列Zの列ベクトルを適当に並べ換え，最後の行を除き $\tilde{V}_{2^n-N}$ に一致させることができる．Zと同じ次数の対角行列

$$D=\begin{vmatrix} 1 & & & & \\ & -1 & & O & \\ & & 1 & & \\ & O & & -1 & \\ & & & & \ddots \\ & & & & & 1 \end{vmatrix}=D^{-1}$$

によってZを変換すれば

$$\tilde{V}_{2^n-N}=DZD^{-1}$$

が成り立つ．したがって行列(4.6)に対し，左右から適当な置換行列をかけ，さらに対角要素が ±1 である対角行列で変換すれば，(4.6)は

表 1 任意項数 FFT（順変換）の計算時間
Table 1 Computing time for FFT of arbitrary length.

項数 N	時間 t	Nの2進表示
32	19	1 00000
64	42	10 00000
128	92	100 00000
256	214	1000 00000
512	443	10000 00000
1024	991	1 00000 00000
2048	2108	10 00000 00000
4091	4644	100 00000 00000
8192	9786	1000 00000 00000
24	15	11000
48	31	1 10000
96	67	11 00000
192	146	110 00000
384	325	1100 00000
768	703	11000 00000
1536	1543	1 10000 00000
3072	3318	11 00000 00000
6144	7198	110 00000 00000
40	26	1 01000
80	57	10 10000
160	120	101 00000
320	271	1010 00000
640	577	10100 00000
1280	1285	1 01000 00000
2560	2724	10 10000 00000
5120	5991	101 00000 00000
10240	12599	1010 00000 00000
28	20	11100
56	40	1 11000
112	86	11 10000
224	184	111 00000
448	406	1110 00000
896	872	11100 00000
1792	1905	1 11000 00000
3584	4073	11 10000 00000
7168	8811	111 00000 00000
8191	10367	111 11111 11111
8193	10188	1000 00000 00001
6143	7497	101 11111 11111
6145	7528	110 00000 00001
5461	6583	101 01010 10101
4681	5549	100 10010 01001
5003	5946	100 11100 01011
8009	10090	111 11010 01001
8209	10257	1000 00000 10001

単位：msec, FCOM 230-38

$$\begin{bmatrix} \tilde{V}_N & B \\ C & \tilde{V}_{2^n-N} \end{bmatrix}$$

に帰着される．$2^{-n/2}$ をかければ，これは正規直交行列となる．$N=1$ のとき $\tilde{V}_N$ の特異値は1である．したがって，補題2と n に関する帰納法から $\tilde{V}_{2^n-N}$ の最小特異値が1ならば，$\tilde{V}_N$ の最小，最大特異値はそれぞれ1および $2^{n/2}$ である．ゆえに，$\tilde{V}_N$ の条件数は $2^{n/2}$ である．(証明終)

本補間法のルベック定数の評価は文献11) にある．

5. たたみ込み演算

たたみ込み演算を高速に行うために FFT は，よく使われる．通常の FFT は項数が2のべきであるため，たたみ込み演算のデータ長が，ちょうど2のべきならば無駄なくその演算を実行できるが，適当に0を補い2のべきに寸法を揃えて実行する．そのため計算量および記憶場所ともに無駄が生じる．しかしながら任意項数の FFT があるならば，このような不自然なことは避けられる．

二つの N 項の多項式

$$f(z)=a_0+a_1z+\cdots+a_{N-1}z^{N-1}$$
$$g(z)=b_0+b_1z+\cdots+b_{N-1}z^{N-1}$$

の循環型たたみ込み演算は，$\{a_k\}$, $\{b_k\}$ に対し任意項数の逆 FFT を適用し，それぞれの対応する標本の積に対して順 FFT を働かせればよい．

非循環型たたみ込み演算の場合は，$\{a_k\}$, $\{b_k\}$ の後に0を N 個補い見掛け上 $2N$ 項の多項式として，循環型たたみ込み演算と同様の手順で非循環型たたみ込み演算を実行できる．

6. 計 算 例

項数が2のべきの通常の FFT を補助ルーチンとして任意項数の順変換，逆変換のプログラムを作成し計算時間と精度についてしらべた．算法は準安定であり，実験結果もその事実を示している．ここでは紙数の節約上順変換の計算時間だけを示す (**表 1**)．

項数 N が2のべきの場合が最も速く，次に速いのは項数が $3\cdot2^k$ の場合である．しかしながら N が素数であっても2のべきの場合と比べると若干劣るが，算法の高速性は発揮されている．

7. おわりに

離散型 Fourier 変換の定義を拡張し，任意項数の広義離散型 Fourier 変換の高速算法を示した．この離散型 Fourier 変換を補間法とみて，補間式の打切り誤差を評価し算法の安定性について述べた．ここで提示した算法をたたみ込み演算に応用し従来の無駄を省いた．

本算法を実数値関数の離散型 Fourier 変換，関数の Chebyshev 級数展開に応用するのは次の課題である．

謝辞 おわりに，日頃討論していただく二宮市三教授にお礼申し上げる．

参 考 文 献

1) Danielson, G. C. and Lanczos, C.: Some Improvements in Practical Fourier Analysis and Their Application to X-Ray Scattering from Liquids, *J. Franklin Inst.*, Vol. 233, No. 4, pp. 365-380; No. 5, pp. 435-452 (1942).
2) Rudnick, P.: Note on the Calculation of Fourier Series, *Math. Comput.*, Vol. 20, No. 95, pp. 429-430 (1966).
3) Good, I. J.: The Interaction Algorithm and Practical Fourier Analysis, *J. R. Stat. Soc. Ser. B.*, Vol. 20, No. 2, pp. 361-372 (1958); Vol. 22, No. 2, pp. 372-375 (1960).
4) Cooley, J. W. and Tukey, J. W.: An Algorithm for the Machine Calculation of Complex Fourier Series, *Math. Comput.*, Vol. 19, No. 90, pp. 297-301 (1965).
5) Winograd, S.: On Computing the Discrete Fourier Transform, *Math. Comput.*, Vol. 32, No. 141, pp. 175-199 (1978).
6) Gentleman, W. M.: Implementing Clenshaw-Curtis Quadrature, *CACM*, Vol. 15, No. 5, pp. 337-342, 343-346 (1972).
7) Cooley, J. W. et al.: The Fast Fourier Transform Algorithm: Programming Considerations in the Calculation of sine, cosine and Laplace Transforms, *J. Sound Vib.*, Vol. 12, No. 3, pp. 315-337 (1970).
8) 鳥居達生：高速 sine 変換，cosine 変換とその数値積分への応用，情報処理，Vol. 15, No. 9, pp. 670-679 (1974).
9) 別宮俊夫他：離散型フーリエ変換に対する Good, Cooley-Tukey, Winograd 法の比較，情報処理第23回全国大会講演論文集，pp. 901-902 (1981).
10) 杉浦 洋，鳥居達生，二宮市三：中国の剰余定理と2巾 FFT による任意項数のフーリエ変換，情報処理学会第23回全国大会講演論文集，pp. 903-904 (1981).
11) 鳥居達生：Van der Corput 列に基づく補間法の数値的安定性，数理解析研究所講究録，463, pp. 56-71 (1982).
12) 杉浦 洋，鳥居達生：Van der Corput 列に基づく補間法の数値的安定性，情報処理学会第25回全国大会講演論文集，pp. 1207-1208 (1982).

(昭和58年3月28日受付)
(昭和58年6月20日採録)