

信号処理とフーリエ変換 第8回

～離散 Fourier 変換 (1)～

かつらだ まさし
桂田 祐史

2020年11月18日

- 1 本日の内容・連絡事項
- 2 離散 Fourier 変換
 - 離散 Fourier 係数
 - Fourier 係数のサンプリング定理

- これから3回、**離散 Fourier 変換**を説明する。今回は離散 Fourier 係数を説明する。周期関数をサンプリングしたデータから Fourier 係数を近似的に求めたものが、離散 Fourier 係数で、それを求めるのが離散 Fourier 変換とみなせる。離散フーリエ係数の基本的な性質と、Fourier 係数に関するサンプリング定理を紹介する。講義ノート [1] の §3.1 まで。

3 離散 Fourier 変換

これから説明する **離散 Fourier 変換** は、Fourier 級数の話の離散化とみなすことができる。実際にデータ処理する場合はサンプリングした離散データを扱わざるを得ず、離散 Fourier 変換の応用上の重要性はとても高い。

一方、離散 Fourier 変換は、周期数列についての Fourier 変換であり、Fourier 級数の近似理論にとどまらない意味を持っている。

§2 (普通の Fourier 変換) もそうであったが、複素指数関数のみで説明する (あまり時間に余裕がなく、式を短く書きたいので、三角関数バージョンの説明はサボる)。

3.1 離散 Fourier 係数 サンプリング

$f: \mathbb{R} \rightarrow \mathbb{C}$ は周期 T とする。 f がある程度滑らかならば¹、次式が成り立つ。

$$(1) \quad f(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}x} \quad (x \in \mathbb{R}),$$

$$(2) \quad c_n := \frac{1}{T} \int_0^T f(x) e^{-in\frac{2\pi}{T}x} dx \quad (n \in \mathbb{Z}).$$

次式で $\{x_j\}$, $\{f_j\}$ を定める。 x_0, x_1, \dots, x_N は $[0, T]$ の N 等分点となる。

$$(3) \quad h := \frac{T}{N}, \quad x_j = jh, \quad f_j := f(x_j) \quad (j \in \mathbb{Z}).$$

f が周期 T であることから ($x_{j+N} = x_j + T$ なので)

$$(4) \quad f_{j+N} = f_j \quad (j \in \mathbb{Z}).$$

すなわち $\{f_j\}_{j \in \mathbb{Z}}$ は周期 N の周期数列である。

信号処理では、 f を (連続) 信号、 x_j を**標本点**、 h を**サンプリング周期** (標本化周期, sampling period)、 $1/h$ を**サンプリング周波数** (標本化周波数, sample rate, sampling rate) と呼ぶ。また、信号を測定して $\{f_j\}$ を得ることを**サンプリング** (標本化) と呼ぶ。

¹ これまでは原点について対称な区間での積分 $c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(x) e^{-in\frac{2\pi}{T}x} dx$ で表していた。

3.1 離散 Fourier 係数 周期積分は台形公式で計算すべし

Fourier 係数 c_n を知りたいとき、 $\{f_j\}_{j=0}^{N-1}$ を用いて近似値を計算することを考える。

どのように計算するのが良いか。結論を天下一りに述べると

周期関数の 1 周期区間における積分の計算には台形則がベスト
(正しい意味でベスト。しばしば驚異的な高精度が達成される。)

(これは数値解析の常識であるが、説明は省略する。)

3.1 離散 Fourier 係数 数値積分の台形公式

$F: [a, b] \rightarrow \mathbb{C}$ の定積分

$$I = \int_a^b F(x) dx$$

に対して

$$(5) \quad I_N := h \sum_{j=1}^N \left(\frac{F(x_{j-1}) + F(x_j)}{2} \right) = h \left(\frac{F_0}{2} + \sum_{j=1}^{N-1} F_j + \frac{F_N}{2} \right)$$

をその近似値として採用するのが (複合) **台形公式** である。ただし

$$(6) \quad h := \frac{b-a}{N}, \quad x_j := a + jh, \quad F_j := F(x_j) \quad (j = 0, 1, \dots, N).$$

F が周期 $b-a$ であれば、 $F(a) = F(b)$ であるから $F_0 = F_N$. ゆえに次式が成り立つ:

$$(7) \quad I_N = h \sum_{j=0}^{N-1} F_j = h \sum_{j=1}^N F_j.$$

3.1 離散 Fourier 係数 離散 Fourier 係数の導入

$F(x) := \frac{1}{T} f(x) e^{-in\frac{2\pi}{T}x}$ の積分に台形則を適用して、 c_n を近似計算したものを C_n (大文字表記) とする:

$$(8) \quad C_n := \frac{1}{T} \cdot h \sum_{j=0}^{N-1} f_j e^{-in\frac{2\pi}{T}x_j}.$$

$$(9) \quad \omega := e^{i\frac{2\pi}{T}h} = e^{\frac{2\pi i}{N}} \quad (\because \frac{h}{T} = \frac{1}{T} \cdot \frac{T}{N} = \frac{1}{N})$$

とおくと

$$e^{-in\frac{2\pi}{T}x_j} = e^{-in\frac{2\pi}{T} \cdot jh} = e^{-inj\frac{2\pi}{N}} = \omega^{-nj}.$$

ゆえに

$$(10) \quad C_n = \frac{1}{N} \sum_{j=0}^{N-1} f_j \omega^{-nj}.$$

この C_n を f の **離散 Fourier 係数** と呼ぶ。

3.1 離散 Fourier 係数 準備: 1 の原始 N 乗根 ω の性質

補題 7.1 (1 の N 乗根の性質)

$N \in \mathbb{N}$ に対して $\omega := e^{2\pi i/N}$ とおくと、次の (1), (2) が成り立つ。

- ① $1 \leq m \leq N-1$ ならば $\omega^m \neq 1$, $\omega^N = 1$ (ω は 1 の原始 N 乗根).
(ゆえに $m \equiv 0 \pmod{N}$ ならば $\omega^m = 1$, そうでないならば $\omega^m \neq 1$.)
- ② 任意の $m \in \mathbb{Z}$ に対して

$$\sum_{j=0}^{N-1} \omega^{mj} = \begin{cases} N & (m \equiv 0 \pmod{N}) \\ 0 & (\text{それ以外}). \end{cases}$$

証明.

- ① (常識的だけれど一応) $\theta := \frac{2\pi}{N}$ とおくと、 $\omega = e^{i\theta}$, $\omega^m = e^{im\theta}$. $1 \leq m \leq N-1$ ならば $0 < m\theta < 2\pi$ であるから、 $\omega^m = e^{im\theta} \neq 1$. $\omega^N = e^{iN\theta} = e^{2\pi i} = 1$.
- ② $m \equiv 0 \pmod{N}$ であれば、 $\omega^m = 1$ であるから $\sum_{j=0}^{N-1} \omega^{mj} = \sum_{j=0}^{N-1} 1 = N$. (続く)

□

3.1 離散 Fourier 係数 準備: 1 の原始 N 乗根 ω の性質

証明 (続き).

$m \equiv 0 \pmod{N}$ でなければ、 $\omega^m \neq 1$. 初項 1, 公比 ω^m の等比級数の和であるから

$$\sum_{j=0}^{N-1} \omega^{mj} = 1 \cdot \frac{1 - (\omega^{mN})}{1 - \omega^m} = \frac{1 - (\omega^N)^m}{1 - \omega^m} = \frac{1 - 1}{1 - \omega^m} = 0.$$

□

3.1 離散 Fourier 係数 離散 Fourier 係数の性質

定理 7.2 (離散 Fourier 係数の性質)

周期 T の周期関数 $f: \mathbb{R} \rightarrow \mathbb{C}$ と $N \in \mathbb{N}$ に対して

$$h := \frac{T}{N}, \quad \omega := e^{2\pi i/N}, \quad x_j := jh, \quad f_j := f(x_j) \quad (j \in \mathbb{Z}),$$

$$C_n := \frac{1}{N} \sum_{j=0}^{N-1} f_j \omega^{-nj} \quad (n \in \mathbb{Z})$$

により $\{C_n\}_{n \in \mathbb{Z}}$ を定めるとき、次の (1), (2) が成り立つ。

① $\{C_n\}_n$ は周期 N の周期数列である: $C_{n+N} = C_n$ ($n \in \mathbb{Z}$).

② f の複素 Fourier 係数 c_n が $\sum_{n=-\infty}^{\infty} |c_n| < \infty$ を満たすならば、任意の $n \in \mathbb{Z}$ に対して

$$(11) \quad C_n = \sum_{m \equiv n} c_m \quad \left(= \sum_{p=-\infty}^{\infty} c_{n+pN} \right).$$

$\sum_{m \equiv n}$ は、 $m \equiv n \pmod{N}$ を満たすすべての m についての和を意味する。

3.1 離散 Fourier 係数 離散 Fourier 係数の性質

$\sum_{n=-\infty}^{\infty} |c_n| < \infty$ という条件は、例えば f が連続で区分的に C^1 級であれば満たされる。

証明.

① $\omega^{-(n+N)j} = \omega^{-nj}\omega^{-Nj} = \omega^{-nj}$ であるから $C_{n+N} = C_n$.

② $f(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}x}$ であるから

$$(12) \quad f_j = f(x_j) = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}x_j} = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}j\frac{T}{N}} = \sum_{n=-\infty}^{\infty} c_n \omega^{nj}.$$

ゆえに (絶対収束することに注意して)

$$\begin{aligned} C_n &= \frac{1}{N} \sum_{j=1}^{N-1} f_j \omega^{-nj} = \frac{1}{N} \sum_{j=0}^{N-1} \left(\omega^{-nj} \sum_{m=-\infty}^{\infty} c_m \omega^{mj} \right) \\ &= \frac{1}{N} \sum_{m=-\infty}^{\infty} c_m \sum_{j=0}^{N-1} \omega^{(m-n)j} = \frac{1}{N} \sum_{m \equiv n} c_m N = \sum_{m \equiv n} c_m. \end{aligned}$$

3.1 離散 Fourier 係数 離散 Fourier 係数の性質

定理 7.2 の (1) から、 $\{C_n\}_{n \in \mathbb{Z}}$ を求めよ、と要求されたとき、連続した N 項、例えば

$$\begin{pmatrix} C_0 \\ C_1 \\ \vdots \\ C_{N-1} \end{pmatrix}$$

を計算すれば十分である。

“入力” $\{f_j\}_{j \in \mathbb{Z}}$ についても同様に、例えば

$$\begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{pmatrix}$$

があれば十分である。

問題の舞台は \mathbb{C}^N ということになる。

3.2 Fourier 係数のサンプリング定理

C_n は c_n を近似するように定めたが、本当にそうか？ 答えは、 “In a sense, Yes. But, ...”

定理 7.3 (Fourier 係数 (周期関数に対する Fourier 変換) に関するサンプリング定理)

周期 T の関数 $u: \mathbb{R} \rightarrow \mathbb{C}$ が、有限 Fourier 級数

$$u(t) = \sum_{n=-M}^M c_n e^{in\frac{2\pi}{T}t} \quad (t \in \mathbb{R})$$

で表せるとき、すなわち u の Fourier 係数 $\{c_n\}$ について

$$|n| > M \quad \Rightarrow \quad c_n = 0$$

が成り立つとき、 $N > 2M$ を満たす N に対して、 $\{C_n\}_{n=0}^{N-1}$ は、

$$\begin{aligned} (\star) \quad & C_n = c_n \quad (0 \leq n \leq M), \\ & C_{N-n} = c_{-n} \quad (1 \leq n \leq M), \\ & C_n = 0 \quad (M < n < N - M) \end{aligned}$$

を満たす。(特に、全ての (0 でない) Fourier 係数 $\{c_n\}_{n=-M}^M$ は、離散 Fourier 係数 $\{C_n\}_{n=0}^{N-1}$ から求まる。ゆえに $u(t)$ も完全に再現できる。)

3.2 Fourier 係数のサンプリング定理 vs. 通常のサンプリング定理

(現段階では、このスライドに書いてあることは分かりにくいかも)

通常、サンプリング定理と呼ばれるのは、(普通の Fourier 変換に関する) 別の定理であるが、上の定理 7.3 はそれに近い内容を持っている。(個人的な意見になるが、定理 7.3 の方が現実の (音などの) 現象の説明に便利である。この辺は “通常のサンプリング定理” を紹介したときに再び取り上げよう。)

仮定の自然さについて: Riemann-Lebesgue の定理から、

$$\lim_{n \rightarrow \pm\infty} c_n = 0$$

であるから、 $|n|$ が大きいとき $|c_n|$ が小さいと期待するのは、それなりにもっともである。

しかし、上の定理のように、 $|n|$ が大きいとき $c_n = 0$ としてしまうと、 f は実解析的となり、非常になめらかな関数ということになる。これは極端かもしれない。不連続関数にも使えるのが Fourier 級数の良いところだったのでは？

(信号処理分野の人は、小さいことと 0 であることの差をおおらかに考えているのかもしれないが、無限がからむので、そんなに簡単ではないのでは…)

3.2 Fourier 係数のサンプリング定理 証明の前に

定理 7.3 の証明を書く前に、具体的な M, N に対して主張を確認すると、カラクリが見えてくる (と思う)。

$M = 1$ (つまり $|n| > 1 \Rightarrow c_n = 0$), $N = 10$ の場合、

$$C_0 = \sum_{m \equiv 0} c_m = c_0 + c_{10} + c_{-10} + c_{20} + c_{-20} + c_{30} + \cdots = c_0 + 0 + 0 + \cdots = c_0,$$

$$C_1 = \sum_{m \equiv 1} c_m = c_1 + c_{-9} + c_{11} + c_{-19} + c_{21} + \cdots = c_1 + 0 + 0 + \cdots = c_1,$$

$$C_9 = \sum_{m \equiv 9} c_m = c_9 + c_{-1} + c_{19} + c_{-11} + c_{29} + c_{-21} + \cdots = 0 + c_{-1} + 0 + 0 + \cdots = c_{-1},$$

$$C_2 = \sum_{m \equiv 2} c_m = c_2 + c_{-8} + c_{12} + c_{-18} + \cdots = 0 + 0 + \cdots = 0,$$

$$C_8 = \sum_{m \equiv 8} c_m = c_8 + c_{-2} + c_{18} + c_{-12} + \cdots = 0 + 0 + \cdots = 0,$$

同様にして $2 \leq n \leq 8$ に対して、 $C_n = 0$ が得られる。

0 でない c_n は、 $c_0 = C_0, c_1 = C_1, c_{-1} = C_9$ と求まる。

3.2 Fourier 係数のサンプリング定理 証明の前に

一方、 $M = 5$ (つまり $|n| > 5 \Rightarrow c_n = 0$), $N = 10$ の場合は

$$C_0 = \sum_{m \equiv 0} c_m = c_0 + c_{-10} + c_{20} + c_{-20} + c_{30} + \cdots = c_0 + 0 + 0 + \cdots = c_0,$$

$$C_1 = \sum_{m \equiv 1} c_m = c_1 + c_{-9} + c_{11} + c_{-19} + c_{21} + \cdots = c_1 + 0 + 0 + \cdots = c_1,$$

$$C_9 = \sum_{m \equiv 9} c_m = c_9 + c_{-1} + c_{19} + c_{-11} + c_{29} + c_{-21} + \cdots = 0 + c_{-1} + 0 + 0 + \cdots = c_{-1},$$

$$C_2 = \sum_{m \equiv 2} c_m = c_2 + c_{-8} + c_{12} + c_{-18} + \cdots = 0 + 0 + \cdots = c_2,$$

$$C_8 = \sum_{m \equiv 8} c_m = c_8 + c_{-2} + c_{18} + c_{-12} + \cdots = 0 + 0 + \cdots = c_{-2},$$

\vdots
 \vdots

$$C_4 = \sum_{m \equiv 4} c_m = c_4 + c_{-6} + c_{14} + c_{-16} + \cdots = c_4 + 0 + 0 + \cdots = c_4,$$

$$C_6 = \sum_{m \equiv 6} c_m = c_6 + c_{-4} + c_{16} + c_{-14} + \cdots = 0 + c_{-4} + 0 + \cdots = c_{-4},$$

ここまでは調子が良い。ところが

$$C_5 = \sum_{m \equiv 5} c_m = c_5 + c_{-5} + c_{15} + c_{-15} + \cdots = c_5 + c_{-5} + 0 + 0 + \cdots = c_5 + c_{-5}.$$

$C_5 = c_5$ も $C_5 = c_{-5}$ も成り立たない。 c_5, c_{-5} は簡単に求まりそうにない。

少し考えると、 $M = 5$ であっても、 $N > 10$ であれば、うまく行く ((★) が成り立つ) ことが分かる。
落ち着いて一般化すると、 $N > 2M$ であれば (★) が成り立つ。 □

定理 7.3 の証明.

すでに示したように

$$C_n = \sum_{m \equiv n} c_m = c_n + \sum_{p=1}^{\infty} (c_{n+pN} + c_{n-pN}).$$

$0 \leq n \leq M$ であれば、

- $n + pN \geq N > 2M > M$ であるから、 $c_{n+pN} = 0$.
- $n - pN \leq M - N < -M$ であるから、 $c_{n-pN} = 0$.

ゆえに $C_n = c_n$.

残りも同様にして証明できる。 □

次のことはぜひ頭に入れて欲しい。

$0 \leq n \ll N$ であるとき、

- c_n の近似は C_n
- c_{-n} の近似は C_{N-n} (C_{N-n} は c_{N-n} ではなく、 c_{-n} の近似である)

- [1] 桂田祐史：「信号処理とフーリエ変換」講義ノート, <http://nalab.mind.meiji.ac.jp/~mk/fourier/fourier-lecture-notes.pdf>, 以前は「画像処理とフーリエ変換」というタイトルだったのを直した。(2014～).