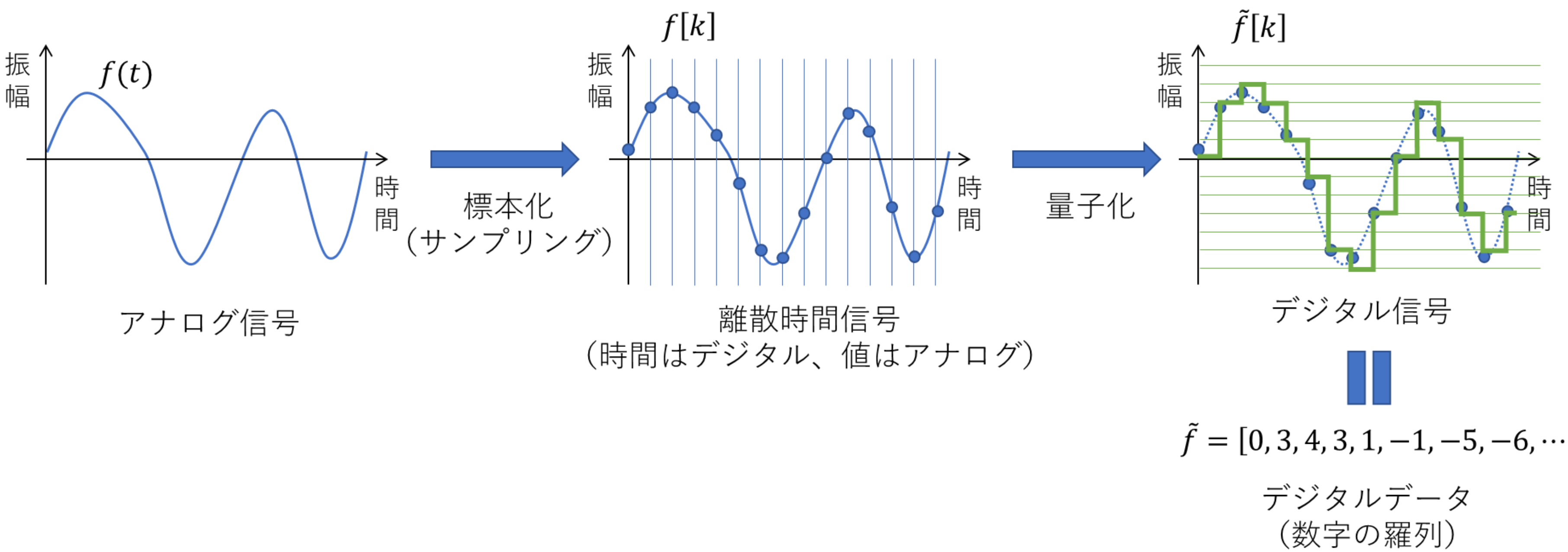
デジタル信号処理
第９回 離散フーリエ変換

2023年6月6日

立命館大学 情報理工学部

前回までのおさらい

- 時間領域信号をデジタルで表現する方法を明らかにした。
  - 標本化・量子化、
  - 標本化定理・ナイキストレート
    - **信号がもつ最大周波数の2倍の周波数でサンプリングすればOK！**



今回の概要

- 第6回講義で，帯域制限信号 $f(t)$（アナログ信号）のフーリエ変換は，無限個の標本値 $\{f(kT)\}_{k=-\infty}^{\infty}$ を用いた離散時間フーリエ変換により計算できることを説明しました．
- 実際には，有限個の標本値しか得られないため，離散時間フーリエ変換の無限和を有限で打ち切る必要があります．今回は，この観点から導出される離散フーリエ変換について説明します．

第６回

(復習) 帯域制限信号と離散時間フーリエ変換

- 信号 $f(t)$ は，そのフーリエ変換 $F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt$ が

$$F(\omega) = 0 \quad (|\omega| \geq W)$$

（$W > 0$ は定数）

を満たすとき，帯域制限信号と呼ばれる.

- 帯域制限信号 $f(t)$ のフーリエ変換は, $T \leq \dfrac{\pi}{W}$ を満たす間隔 T の標本値 $\{f(kT)\}_{k=-\infty}^{\infty}$ から計算できる．すなわち,

$$F(\omega) = \begin{cases} F_d(\omega) & (|\omega| < W) \\ 0 & (|\omega| \geq W) \end{cases}$$

$$F_d(\omega) = T \sum_{k=-\infty}^{\infty} f(kT)\, e^{-ikT\omega}$$

この計算式を離散時間フーリエ変換と呼ぶ

離散フーリエ変換の導出 1/4

- 離散時間フーリエ変換 $F_d(\omega) = T \sum_{k=-\infty}^{\infty} f(kT)\, e^{-ikT\omega}$

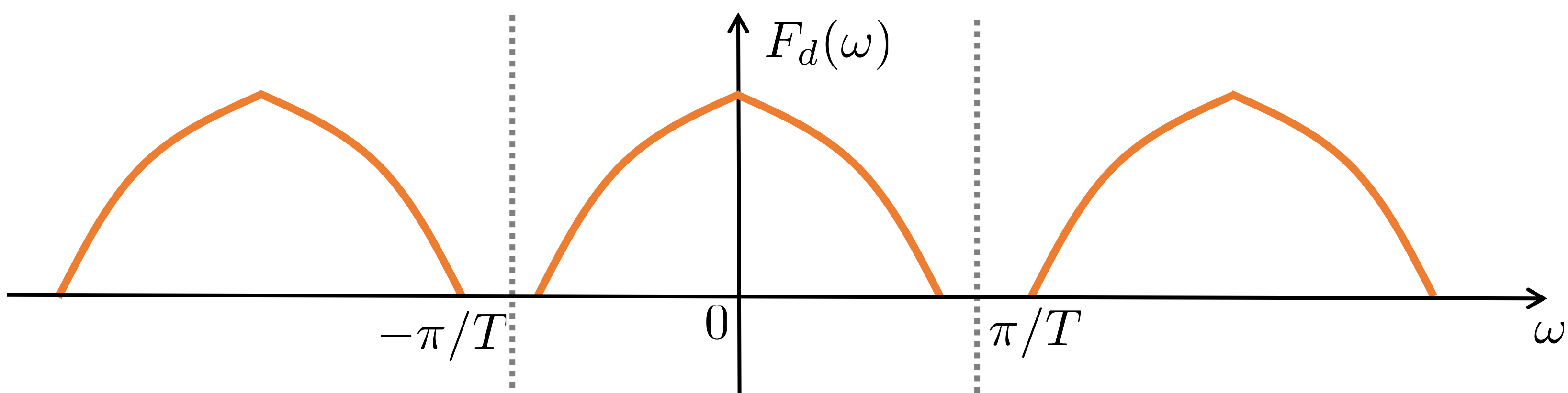
  は無限個の標本値 $\{f(kT)\}_{k=-\infty}^{\infty}$ を使っているが，
  現実的には有限個の標本値しか得られない．

- 無限和を打ち切って，N個の標本値 $\{f(kT)\}_{k=0}^{N-1}$ のみを
  使った計算式で離散時間フーリエ変換を近似する．
  つまり，

$$F_d(\omega) \sim \tilde{F}_d(\omega) = T \sum_{\boxed{k=0}}^{\boxed{N-1}} f(kT)\, e^{-ikT\omega}$$

離散フーリエ変換の導出 2/4

- 今, N 個の標本値のみを使うので, $\tilde{F}_d(\omega) = T\sum_{k=0}^{N-1} f(kT)\, e^{-ikT\omega}$ を任意の周波数 ω に対して計算する必要はなく，(適切に選んだ) N 個の周波数成分に対して計算すれば十分である.（時間領域の N 個の標本値は周波数領域でも N 個になる）
- ここで $F_d(\omega)$ の周期は $\frac{2\pi}{T}$ なので，1周期分の長さの区間をN等分した点に対応する周波数の値を計算する.

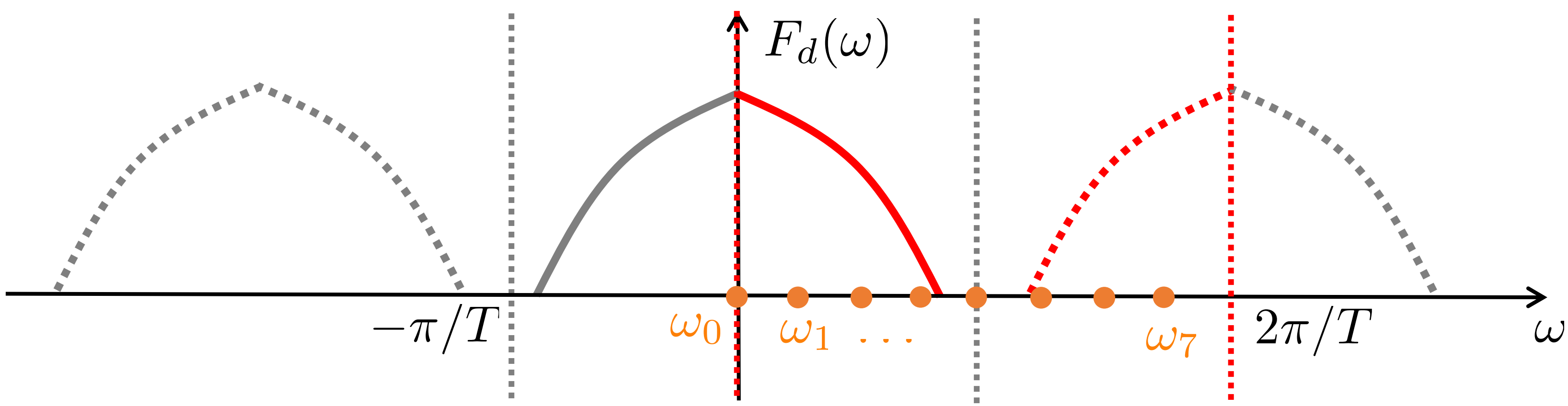


離散フーリエ変換の導出 3/4

- N 等分する点が簡潔に書けるので，1周期分の区間として $[0, 2\pi/T)$ を選ぶ．この区間を N 等分する点 $\omega_0, \ldots, \omega_{N-1}$ は

$$\omega_n = \frac{2\pi}{T} \cdot \frac{n}{N} = \frac{2\pi n}{NT} \qquad (n = 0, 1, \ldots, N-1)$$

で与えられる．



($N = 8$ の場合の分割点)

離散フーリエ変換の導出 4/4

- $\tilde{F}_d(\omega) = T\sum_{k=0}^{N-1} f(kT)\, e^{-ikT\omega}$ に $\omega = \omega_n = \dfrac{2\pi n}{NT}$ を代入すると，

$$\tilde{F}_d(\omega_n) = T\sum_{k=0}^{N-1} f(kT)\, e^{-ikT\frac{2\pi n}{NT}} = T\sum_{k=0}^{N-1} f(kT)\, e^{-i\frac{2\pi nk}{N}}$$ を得る．

- 標本値を $f(kT) = f[k]$ と置き，T で割った式

$$F[n] = \sum_{k=0}^{N-1} f[k] e^{-i\frac{2\pi nk}{N}} \quad (n = 0, 1, \ldots, N-1)$$

を離散フーリエ変換(Discrete Fourier Transform; DFT)
と呼ぶ．なお，$\tilde{F}_d(\omega_n) = TF[n]$ である．

(補足) 標本値が実数の場合の離散フーリエ変換

- 実関数 $f(t)$ のフーリエ変換 $F(\omega)$ は, $F(\omega)$ と $F(-\omega)$ が複素共役の関係 $\overline{F(\omega)} = F(-\omega)$ を満たす.

$$\left(\overline{F(\omega)} = \overline{\int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt} = \int_{-\infty}^{\infty} \overline{f(t)}\ \overline{e^{-i\omega t}} dt = \int_{-\infty}^{\infty} f(t) e^{i\omega t} dt = F(-\omega) \right)$$

- 離散時間フーリエ変換 $F_d(\omega)$ も同様に複素共役の関係を満たす.

- 離散フーリエ変換は，有限項で打ち切った離散時間フーリエ変換の $[0, 2\pi/T)$ を N 等分した点の値に対応するため, 実関数の標本値から計算される離散フーリエ変換では

$$\overline{F[n]} = F[N-n] \quad (n = N/2+1, \ldots, N-1)$$

の複素共役の関係が成り立つ．(前々ページ図参照)なお，簡単のため, N は偶数とした．この関係から，標本値が実数の場合は $n = 0, 1, \ldots, N/2$ の $F[n]$ だけを計算し，その他が必要な場合は複素共役の関係から求めることが多い.

離散フーリエ変換の例1 1/2

https://colab.research.google.com/drive/11bBSEOdr0CmO0wVjBwNtlOKYrD5XLcXl?hl=ja#scrollTo=RxJz4YL4TyG3

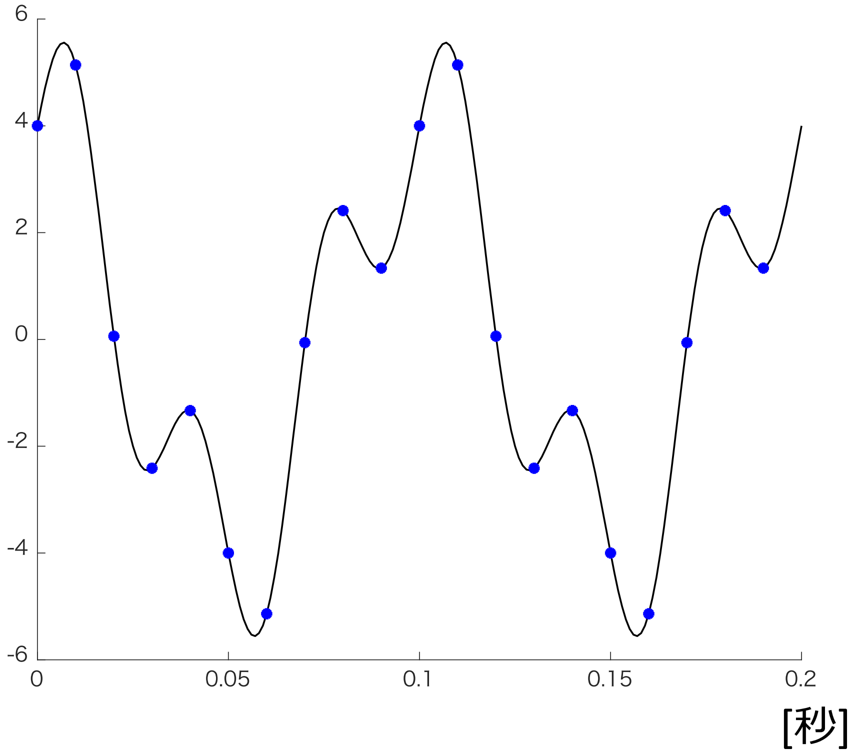
- $g(t) = 4\cos(2\pi f_1 t) + 2\sin(2\pi f_2 t)$ を考える．
  $(f_1 = 10\mathrm{Hz}, f_2 = 30\mathrm{Hz})$
- 標本化周波数を100Hzに設定する．これは $g(t)$ の最大周波数(30Hz)の2倍以上になっている．
- このとき，標本化間隔は $T = 1/100 = 0.01\mathrm{s}$ となる．この間隔で $N = 20$ 個の標本値を取得する．より詳細には，$t = 0, 0.01, \ldots, 0.19$ 秒の標本値を $\{g[k]\}_{k=0}^{19}$ とする．
- これらの標本値から計算した離散フーリエ変換(DFT)

$$G[n] = \sum_{k=0}^{N-1} g[k] e^{-i\frac{2\pi nk}{N}} \quad (n = 0, 1, \ldots, N-1)$$

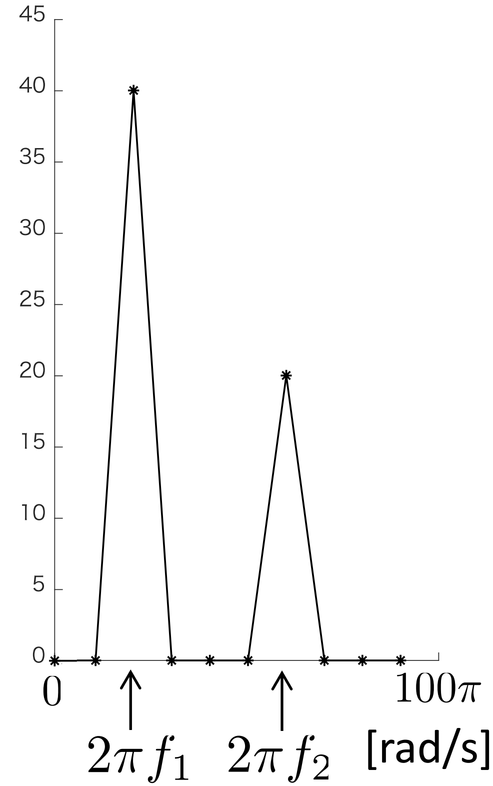
の絶対値を次ページに示す．なお，標本値が実数なので，$n = 0, 1, \ldots, N/2$ のみを表示する．

離散フーリエ変換の例1 2/2

$g(t)$ の連続時間波形と標本値 $\{g[k]\}_{k=0}^{19}$



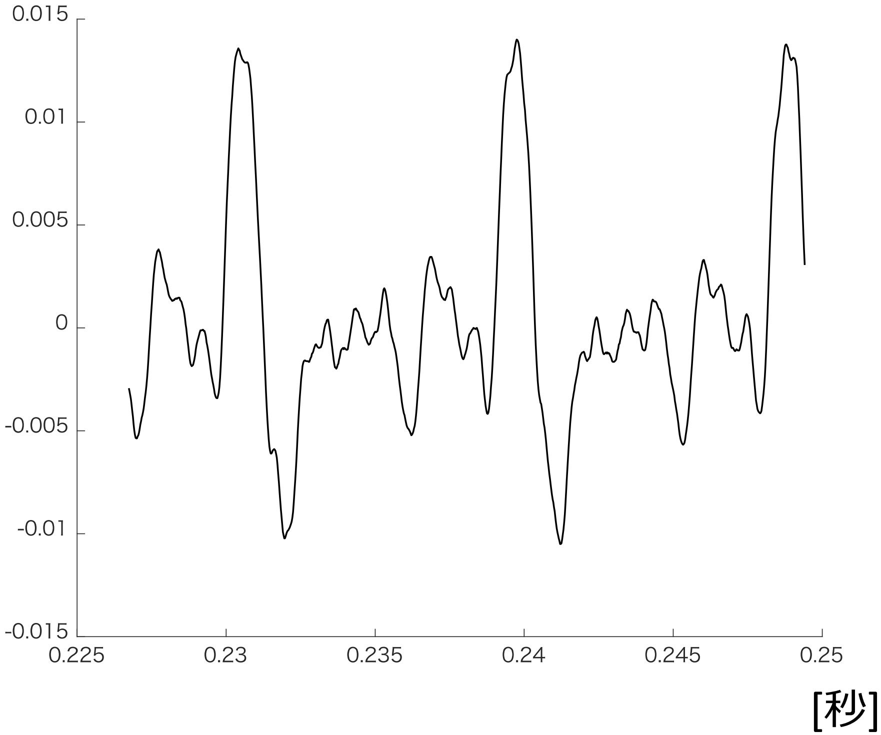
DFT $\{G[n]\}_{n=0}^{10}$ の絶対値



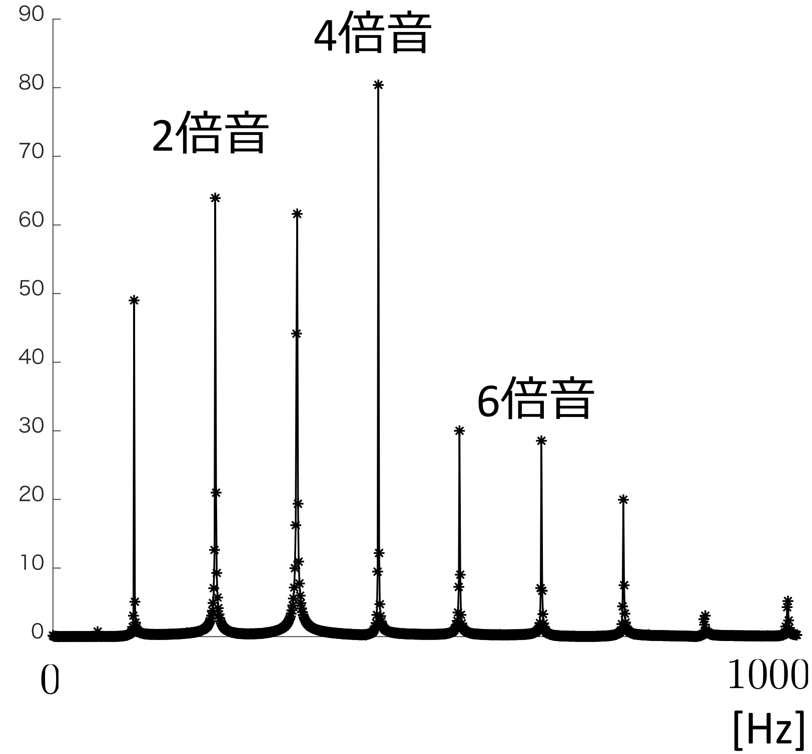
(注: DFTのグラフの横軸の単位は角周波数にしている．より正確には，$[0, \pi/T)$ を10等分した点の角周波数に対応するようにDFTの結果を当てはめている)

離散フーリエ変換の例2

ギターの五弦音(110Hz)の時間波形



DFTの絶対値



DFTの絶対値は倍音の周波数で
ピークを持つことが見てとれる

逆離散フーリエ変換

- 離散周波数成分 $\{F[n]\}_{n=0}^{N-1}$ から標本値 $\{f[k]\}_{k=0}^{N-1}$ を求める変換を逆離散フーリエ変換(Inverse DFT; IDFT)と呼ぶ.
- DFTの計算式は

$$
\begin{aligned}
F[0] &= f[0] + f[1] + \cdots + f[N-1] \\
F[1] &= f[0] + e^{-i\frac{2\pi}{N}} f[1] + \cdots + e^{-i\frac{2\pi(N-1)}{N}} f[N-1] \\
&\vdots \\
F[N-1] &= f[0] + e^{-i\frac{2\pi(N-1)}{N}} f[1] + \cdots + e^{-i\frac{2\pi(N-1)(N-1)}{N}} f[N-1]
\end{aligned}
$$

と書けるが，これを $f[0], \ldots, f[N-1]$ に関する連立方程式として見て解くと，IDFTの計算式

$$
f[k] = \frac{1}{N} \sum_{n=0}^{N-1} F[n] e^{i\frac{2\pi nk}{N}} \quad (k = 0, 1, \ldots, N-1)
$$

が導かれる. （導出は省略）

逆離散フーリエ変換の検算

- $f[k] = \dfrac{1}{N}\displaystyle\sum_{n=0}^{N-1} F[n] e^{i\frac{2\pi nk}{N}}$ となることを確かめておく.

- 上式に $F[n] = \displaystyle\sum_{\ell=0}^{N-1} f[\ell] e^{-i\frac{2\pi n\ell}{N}}$ を代入すると,

$$
\begin{aligned}
\frac{1}{N}\sum_{n=0}^{N-1}\left\{\sum_{\ell=0}^{N-1} f[\ell] e^{-i\frac{2\pi n\ell}{N}}\right\} e^{i\frac{2\pi nk}{N}} &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{\ell=0}^{N-1} f[\ell] e^{i\frac{2\pi n(k-\ell)}{N}} \\
&= \frac{1}{N}\sum_{\ell=0}^{N-1} f[\ell] \sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}} \\
&= f[k]
\end{aligned}
$$

シグマ和の
順序交換

となる. ここで, 最後の変形には $\displaystyle\sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}} = \begin{cases} N & (\ell = k) \\ 0 & (\ell \neq k) \end{cases}$

の関係式(次ページで導出)を用いた.

関係式 $\displaystyle\sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}} = \begin{cases} N & (\ell = k) \\ 0 & (\ell \neq k) \end{cases}$ の導出

$\displaystyle\sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}}$ は $r = e^{i\frac{2\pi(k-\ell)}{N}}$ と置くと $\displaystyle\sum_{n=0}^{N-1} r^n$ と書けるので，

公比 r の等比数列の和になっている．したがって，$r \neq 1$ ならば

$\displaystyle\sum_{n=0}^{N-1} r^n = \frac{1-r^N}{1-r}$ となる．

ここで，$r^N = e^{i2\pi(k-\ell)} = \cos(2\pi(k-\ell)) + i\sin s(2\pi(k-\ell))$ であり，

$k-\ell$ は整数だから，$r^N = 1$．なお，$-N+1 \leq k-\ell \leq N-1$ だから，

$\ell \neq k$ の場合，$r = e^{i\frac{2\pi(k-\ell)}{N}} = \cos(2\pi(k-\ell)/N) + i\sin(2\pi(k-\ell)/N) \neq 1$.

したがって，$\ell \neq k$ の場合，

$$\sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}} = \frac{1-r^N}{1-r} = 0$$

$\ell = k$ の場合は，$e^{i\frac{2\pi n(k-\ell)}{N}} = 1$ だから，$\displaystyle\sum_{n=0}^{N-1} e^{i\frac{2\pi n(k-\ell)}{N}} = \sum_{n=0}^{N-1} 1 = N$.

DFT行列

$$F[n] = \sum_{k=0}^{N-1} f[k] e^{-i\frac{2\pi nk}{N}} \qquad (n = 0, 1, \ldots, N-1)$$

- DFTは行列形式でも書くことができる．表記の簡潔化のため，$w = e^{-i\frac{2\pi}{N}}$と置く．

$$\boxed{※\, w^{nk} = \left(e^{-i\frac{2\pi}{N}}\right)^{nk} = e^{-i\frac{2\pi nk}{N}}}$$

$$\begin{pmatrix} F[0] \\ F[1] \\ \vdots \\ F[N-1] \end{pmatrix} = \begin{pmatrix} w^0 & w^0 & \cdots & w^0 \\ w^0 & w^1 & \cdots & w^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ w^0 & w^{N-1} & \cdots & w^{(N-1)(N-1)} \end{pmatrix} \begin{pmatrix} f[0] \\ f[1] \\ \vdots \\ f[N-1] \end{pmatrix}$$

この行列をDFT行列と呼ぶ．

- DFT行列を$\boldsymbol{W}$とすると，$\boldsymbol{W}/\sqrt{N}$ は線形代数で勉強したユニタリ行列になっている．つまり，複素共役転置を$*$で表すと, $\boldsymbol{W}^*\boldsymbol{W}/N$は単位行列になる(前ページの関係式から確かめることができる)．これを利用し，IDFTを導出することもできる．($\boldsymbol{W}^*/N$がIDFT行列になっている)

高速フーリエ変換 (FFT)

- DFTを定義通りに

$$\begin{pmatrix} F[0] \\ F[1] \\ \vdots \\ F[N-1] \end{pmatrix} = \begin{pmatrix} w^0 & w^0 & \cdots & w^0 \\ w^0 & w^1 & \cdots & w^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ w^0 & w^{N-1} & \cdots & w^{(N-1)(N-1)} \end{pmatrix} \begin{pmatrix} f[0] \\ f[1] \\ \vdots \\ f[N-1] \end{pmatrix}$$

を用いて計算した場合，N^2 回の乗算が必要．例えば，$N=1024$ の場合，約104万回の掛け算が必要となる．しかし，工夫したアルゴリズムを用いれば，およそ $N\log_2 N$ 回の乗算で計算できる．すなわち，$N=1024$ の場合，$\log_2 1024=10$ だから，およそ1万回の乗算で済む．このアルゴリズムを高速フーリエ変換 (Fast Fourier Transform; FFT)と呼ぶ．要するに，FFTはDFTを高速計算するアルゴリズムである．

（なお，FFTはNが2のべき乗であることを前提としているので，
そうでない場合は値が0の標本値を追加してNを2のべき乗にする）

FFTの基本アイディア (N=4の場合)

- $N=4$ の場合のDFTの計算は

$$\begin{pmatrix} F[0] \\ F[1] \\ F[2] \\ F[3] \end{pmatrix} = \begin{pmatrix} w^0 & w^0 & w^0 & w^0 \\ w^0 & w^1 & w^2 & w^3 \\ w^0 & w^2 & w^4 & w^6 \\ w^0 & w^3 & w^6 & w^9 \end{pmatrix} \begin{pmatrix} f[0] \\ f[1] \\ f[2] \\ f[3] \end{pmatrix}$$

※ $w^{nk} = \left(e^{-i\frac{2\pi}{N}}\right)^{nk} = e^{-i\frac{2\pi nk}{N}}$

である．ここで，$w^0 = 1$, $w^2 = e^{-i\pi} = \cos(-\pi) + i\sin(-\pi) = -1$, $w^4 = e^{-i2\pi} = \cos(-2\pi) + i\sin(-2\pi) = 1$ を用いて整理すると，

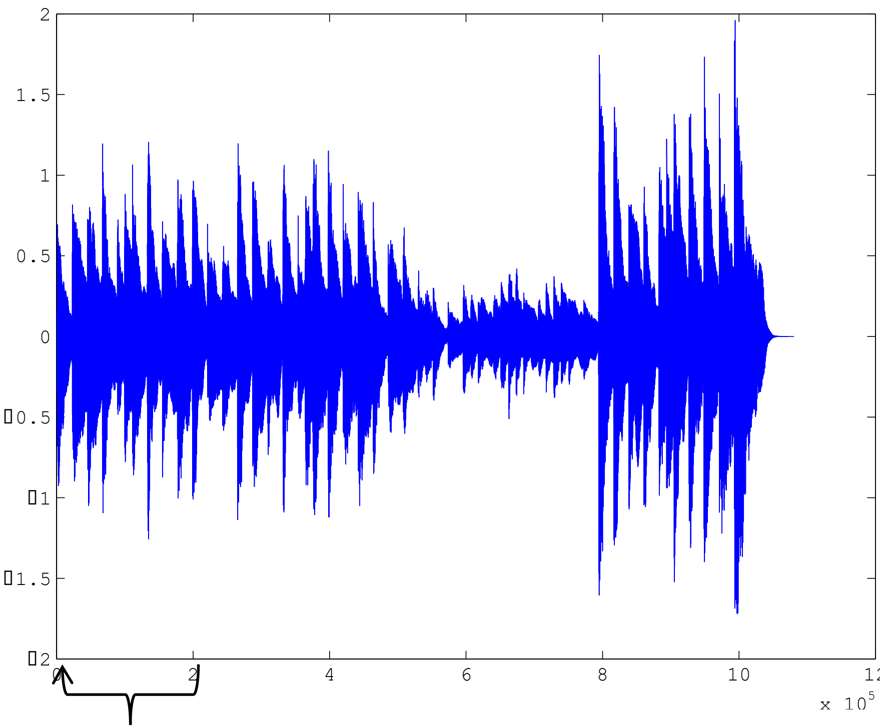
$$\begin{pmatrix} F[0] \\ F[1] \\ F[2] \\ F[3] \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & w^1 & -1 & -w^1 \\ 1 & -1 & 1 & -1 \\ 1 & -w^1 & -1 & w^1 \end{pmatrix} \begin{pmatrix} f[0] \\ f[1] \\ f[2] \\ f[3] \end{pmatrix}$$

$$= \begin{pmatrix} (f[0] + f[2]) + (f[1] + f[3]) \\ (f[0] - f[2]) + w^1(f[1] - f[3]) \\ (f[0] + f[2]) - (f[1] + f[3]) \\ (f[0] - f[2]) - w^1(f[1] - f[3]) \end{pmatrix}$$

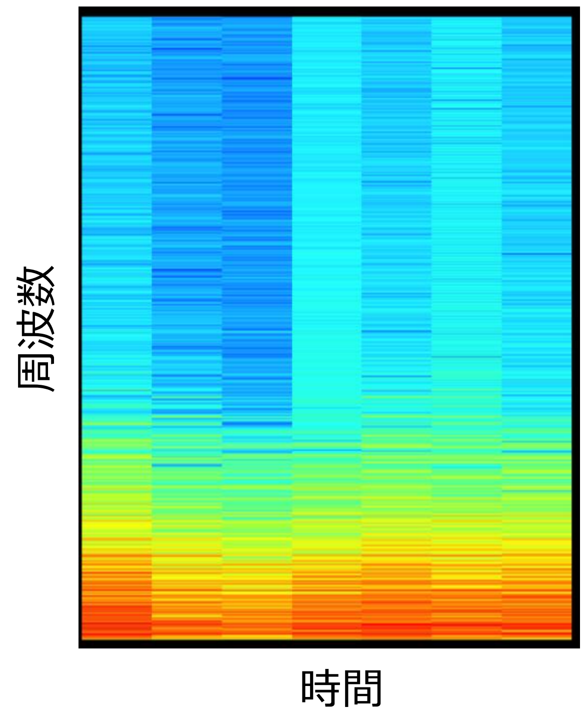
同じ計算式(同じ色の式)は一度計算した値を保存して使いまわせるので，乗算回数のオーダーを $\mathcal{O}(N \log_2 N)$ まで削減できる．

スペクトログラム (短時間ごとのDFTの絶対値)

離散時間信号（標本値）



スペクトログラム

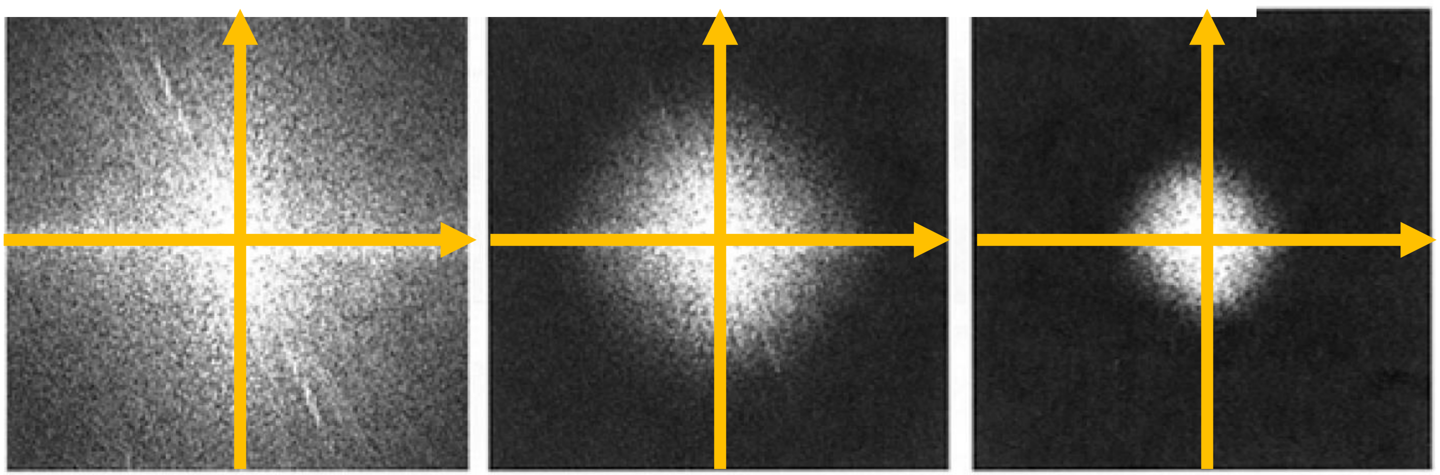


この区間でDFTを計算して，周波数成分の絶対値を表示，
この処理をずらして繰り返した図がスペクトログラム

画像のフーリエ変換（2次元FFT）

入力画像（時間領域）

振幅スペクトル（周波数領域）



高周波成分が多い＝くっきりしている or ノイズが多い

今回のまとめ

- 有限個の標本値から離散周波数成分を計算する離散フーリエ変換(DFT)について説明した.
- 離散フーリエ変換の逆変換(IDFT)を導出した.
- DFTの高速計算アルゴリズムである高速フーリエ変換(FFT)の基本アイディアを説明した.

理解度確認 小テスト

manaba +R にログインして，第9回小テストを行います。
制限時間は 10分間 です。

スライドを見返しながら，解いてよいです.
今回の問題は全て選択式の問題です.

宿題

- 次の問題の結果を記載したpdfファイルを提出してください
- Manaba+R レポート機能で提出してください

デジタル信号 $f[k]$は以下の式で定義される：

$$f[k] = \begin{cases} \sin(\omega_1 kT) + \sin(\omega_2 kT), & k = 0,1,2,\dots,31 \\ 0, & \text{otherwise} \end{cases}$$

$$\boxed{\begin{aligned} T &= 1 \\ \omega_1 &= 0.125\pi/T \\ \omega_2 &= 0.5\pi/T \end{aligned}}$$

(1) 信号$f[k]$ の値を計算し、グラフを描画してください。

(2) 信号$f[k]$ に対して $N = 32$ の離散フーリエ変換を行い、
振幅スペクトル $|F[n]|$を計算し、描画してください。

- プログラミング言語は問いません(C, python, matlabなど)
- サンプルとしてpythonとmatlabのコードを配布します(次ページを参照)

サンプルコード

以下の問題のサンプルコード(python, matlab)を配布します
サンプルコードを参考にして，宿題に取り組んでください

デジタル信号 $f[k]$は以下の式で定義される：

$$f[k] = \begin{cases} \sin(\omega_1 kT), & k = 0,1,2,\ldots,31 \\ 0, & \text{otherwise} \end{cases}$$

$$\boxed{\begin{aligned} T &= 1 \\ \omega_1 &= 0.25\pi/T \end{aligned}}$$

(1) 信号$f[k]$ の値を計算し、グラフを描画してください。

(2) 信号$f[k]$ に対して $N = 32$ の離散フーリエ変換を行い、
振幅スペクトル $|F[n]|$を計算し、描画してください。