


2022年度秋学期 画像情報処理 第5回
 離散フーリエ変換, フーリエ変換の実例

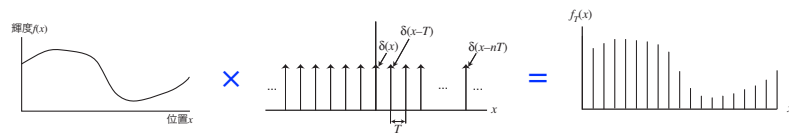
浅野 晃
 関西大学総合情報学部



離散フーリエ変換 🤔

サンプリングされた関数のフーリエ変換

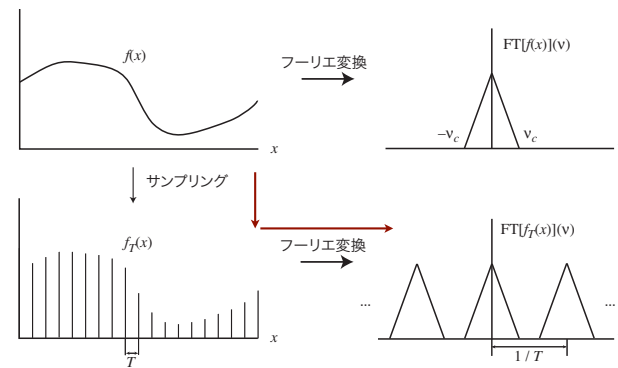
サンプリング $f_T(x) = f(x)\text{comb}_T(x)$



サンプリングされた関数のフーリエ変換

$$\begin{aligned}
 FT[f_T(x)](\nu) &= FT[f(x)\text{comb}_T(x)](\nu) \\
 &= \int_{-\infty}^{\infty} f(x)\text{comb}_T(x) \exp(-i2\pi\nu x) dx
 \end{aligned}$$

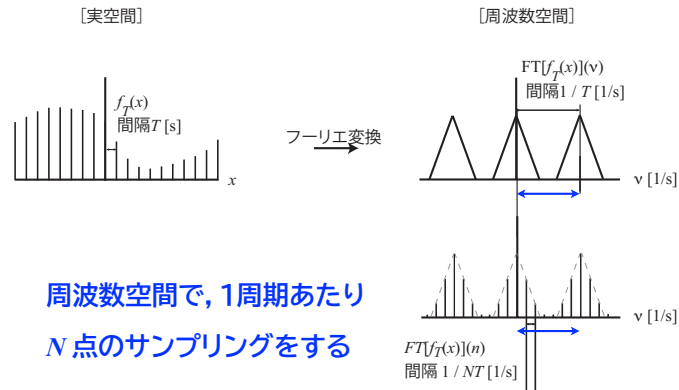
サンプリングされた関数のフーリエ変換



こちらは離散的だが

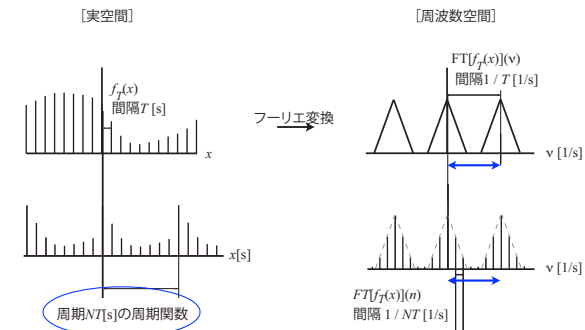
こちらは離散的でない→コンピュータで扱えない

周波数空間でもサンプリング

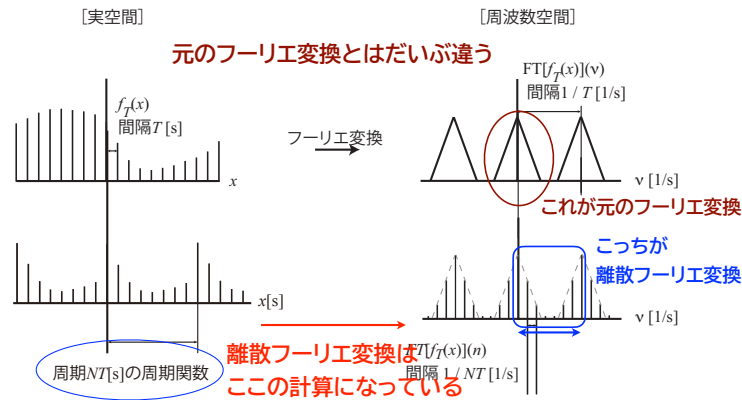


実空間ではどうなる？

実空間でサンプリング → 周波数空間で周期的に現れる
周波数空間でサンプリング → 実空間で周期的に現れる



離散フーリエ変換



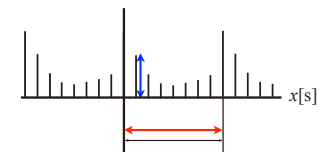
数列の計算にする

元の関数は忘れて、サンプリングされたものを数列とみなす

$$u(n) = f_T(nT)$$

デルタ関数の並びを積分

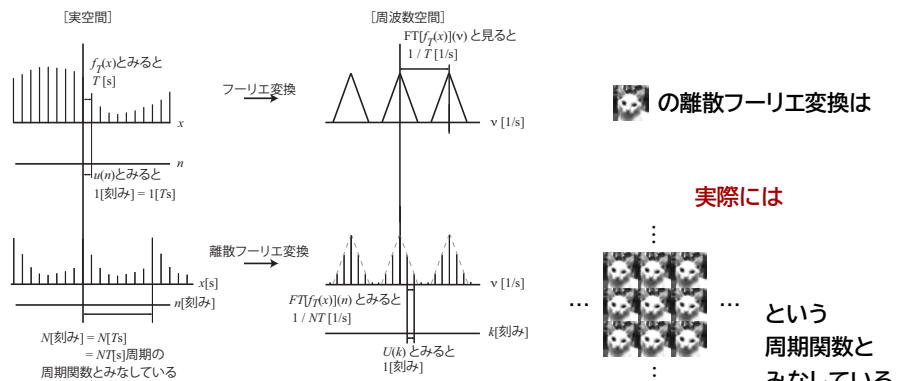
→ 離散の場合は、その値を合計するだけ



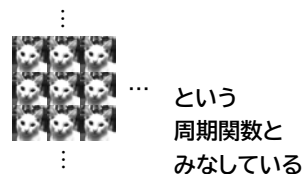
離散フーリエ変換 (DFT)

$$U(k) = \sum_{n=0}^{N-1} u(n) \exp\left(-i2\pi \frac{k}{N}n\right) \quad (k = 0, 1, \dots, N-1)$$

離散フーリエ変換



フーリエ変換の実例 (MATLABを使って説明します)



高速フーリエ変換について

「高速」フーリエ変換とは

高速フーリエ変換 (Fast Fourier Transformation, FFT)

離散フーリエ変換の計算に含まれる掛け算の回数を減らす工夫

コンピュータでは、掛け算は足し算に比べて時間がかかるので
掛け算を減らすと全体の計算にかかる時間を短くできる

例えば

$$5 \times 4 + 3 \times 5 \quad \text{掛け算は2回}$$

$$5 \times (4 + 3) \quad \text{掛け算は1回}$$

離散フーリエ変換と掛け算

4点だけの信号(N = 4)の離散フーリエ変換について考える

離散フーリエ変換の式は

$$U(k) = \sum_{n=0}^3 u(n) \exp\left(-i2\pi \frac{k}{4} n\right) \quad (k = 0, 1, \dots, 3)$$

ここでは $U(0), U(1), U(2), U(3)$ の4つの $U(\cdot)$ を計算する

1つの $U(\cdot)$ を計算するのに、

$u(n) \exp\left(-i2\pi \frac{k}{4} n\right)$ という掛け算を、 $n = 0, 1, 2, 3$ の4回行う

全部で
 $4 \times 4 = 16$ 回の
掛け算

掛け算を減らすには

$$U(k) = \sum_{n=0}^3 u(n) \exp\left(-i2\pi \frac{k}{4} n\right) \quad (k = 0, 1, \dots, 3)$$

この指数関数は、 $W \equiv \exp\left(-i\frac{2\pi}{4}\right)$ を使うと $W^{k \times n}$ と表せる

指数関数と三角関数の関係を用いると

$$W^0 = \exp\left(-i2\pi \frac{0}{4}\right) = \cos 0 + i \sin 0 = 1 + 0i = 1$$

$$W^1 = \exp\left(-i2\pi \frac{1}{4}\right) = \cos\left(-\frac{\pi}{2}\right) + i \sin\left(-\frac{\pi}{2}\right) = 0 - i = -i$$

$$W^2 = \exp\left(-i2\pi \frac{2}{4}\right) = \cos(-\pi) + i \sin(-\pi) = -1 + 0i = -1$$

$$W^3 = \exp\left(-i2\pi \frac{3}{4}\right) = \cos\left(-\frac{3\pi}{2}\right) + i \sin\left(-\frac{3\pi}{2}\right) = 0 + i = i$$

$$W^4 = \exp\left(-i2\pi \frac{4}{4}\right) = \cos(-2\pi) + i \sin(-2\pi) = 1 + 0i = 1$$

指数関数の、くりかえしの性質を使う

前ページの式は

$$W^0 = \exp\left(-i2\pi \frac{0}{4}\right) = \cos 0 + i \sin 0 = 1 + 0i = 1$$

$$W^1 = \exp\left(-i2\pi \frac{1}{4}\right) = \cos\left(-\frac{\pi}{2}\right) + i \sin\left(-\frac{\pi}{2}\right) = 0 - i = -i$$

$$W^2 = \exp\left(-i2\pi \frac{2}{4}\right) = \cos(-\pi) + i \sin(-\pi) = -1 + 0i = -1$$

$$W^3 = \exp\left(-i2\pi \frac{3}{4}\right) = \cos\left(-\frac{3\pi}{2}\right) + i \sin\left(-\frac{3\pi}{2}\right) = 0 + i = i$$

$$W^4 = \exp\left(-i2\pi \frac{4}{4}\right) = \cos(-2\pi) + i \sin(-2\pi) = 1 + 0i = 1$$

$$W^0 = 1$$

$$W^1 = -i$$

$$\rightarrow W^2 = -1 = -W^0$$

$$W^3 = i = -W^1$$

$$W^4 = 1 = W^0$$

W^0 と W^1 だけで表せる

さらに、指数関数の、くりかえしの性質を使う

前ページの式は

$$W^0 = 1$$

$$W^1 = -i$$

$$W^2 = -1 = -W^0$$

$$W^3 = i = -W^1$$

$$W^4 = 1 = W^0$$



$$W^5 = W^4 \times W^1 = 1 \times W^1 = W^1$$

$$W^6 = W^4 \times W^2 = 1 \times (-W^0) = -W^0$$

$$W^7 = W^4 \times W^3 = 1 \times (-W^1) = -W^1$$

$$W^8 = W^4 \times W^4 = 1 \times W^0 = W^0$$

$$W^9 = W^8 \times W^1 = W^0 \times W^1 = 1 \times W^1 = W^1$$

...

ずっと、 W^0 と W^1 だけで表せる

N = 4の離散フーリエ変換は

N = 4 の離散フーリエ変換は

$$\begin{aligned} U(0) &= u(0)W^{0 \times 0} + u(1)W^{0 \times 1} + u(2)W^{0 \times 2} + u(3)W^{0 \times 3} \\ U(1) &= u(0)W^{1 \times 0} + u(1)W^{1 \times 1} + u(2)W^{1 \times 2} + u(3)W^{1 \times 3} \\ U(2) &= u(0)W^{2 \times 0} + u(1)W^{2 \times 1} + u(2)W^{2 \times 2} + u(3)W^{2 \times 3} \\ U(3) &= u(0)W^{3 \times 0} + u(1)W^{3 \times 1} + u(2)W^{3 \times 2} + u(3)W^{3 \times 3} \end{aligned}$$

つまり

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 + u(2)W^2 + u(3)W^3 \\ U(2) &= u(0)W^0 + u(1)W^2 + u(2)W^4 + u(3)W^6 \\ U(3) &= u(0)W^0 + u(1)W^3 + u(2)W^6 + u(3)W^9 \end{aligned}$$

N = 4の離散フーリエ変換は

ここまで述べたくりかえしの性質

$W^0 \rightarrow W^1 \rightarrow -W^0 \rightarrow -W^1 \rightarrow W^0 \rightarrow W^1 \rightarrow \dots$ を使うと

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 + u(2)W^2 + u(3)W^3 \\ U(2) &= u(0)W^0 + u(1)W^2 + u(2)W^4 + u(3)W^6 \\ U(3) &= u(0)W^0 + u(1)W^3 + u(2)W^6 + u(3)W^9 \end{aligned}$$

は、すなわち

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 - u(2)W^0 - u(3)W^1 \\ U(2) &= u(0)W^0 - u(1)W^0 + u(2)W^0 - u(3)W^0 \\ U(3) &= u(0)W^0 - u(1)W^1 - u(2)W^0 + u(3)W^1 \end{aligned}$$

となる

行の順番を入れ替えると

行の順番を入れ替えてみる

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 - u(2)W^0 - u(3)W^1 \\ U(2) &= u(0)W^0 - u(1)W^0 + u(2)W^0 - u(3)W^0 \\ U(3) &= u(0)W^0 - u(1)W^1 - u(2)W^0 + u(3)W^1 \end{aligned}$$

を

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(2) &= u(0)W^0 - u(1)W^0 + u(2)W^0 - u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 - u(2)W^0 - u(3)W^1 \\ U(3) &= u(0)W^0 - u(1)W^1 - u(2)W^0 + u(3)W^1 \end{aligned}$$

とすると…

← W^0 と W^1 が
← 出てくる順番が同じ

行の順番を入れ替えると

行の順番を入れ替えると

$$\begin{aligned} U(0) &= u(0)W^0 + u(1)W^0 + u(2)W^0 + u(3)W^0 \\ U(2) &= u(0)W^0 - u(1)W^0 + u(2)W^0 - u(3)W^0 \\ U(1) &= u(0)W^0 + u(1)W^1 - u(2)W^0 - u(3)W^1 \\ U(3) &= u(0)W^0 - u(1)W^1 - u(2)W^0 + u(3)W^1 \end{aligned}$$

は、

$$\begin{aligned} U(0) &= (u(0) + u(2))W^0 + (u(1) + u(3))W^0 \\ U(2) &= (u(0) + u(2))W^0 - (u(1) + u(3))W^0 \\ U(1) &= (u(0) - u(2))W^0 + (u(1) - u(3))W^1 \\ U(3) &= (u(0) - u(2))W^0 - (u(1) - u(3))W^1 \end{aligned}$$

と、一定の手順で
まとめられる

掛け算の回数は
16回から8回に減った

この続きは

$N = 8, 16, 32, \dots$ と、2の累乗になっている場合は、

同じ手順を繰り返すことで、最終的に2行ずつの組にすることができる

そんな複雑な数式は書けません…😓

「行列」を使えば、もうすこしシンプルに書くことができます。
次回、行列を説明したあと、
高速フーリエ変換のもう少し細かいところを説明します。

テキスト付録2:
離散フーリエ変換すると
データが2倍に増えているのか? 🤔

離散フーリエ変換の結果は複素数

N 個の実数値は、 N 個の複素数値に変換される

複素数は $a + bi$ の形で、2つの実数の組になっている

離散フーリエ変換によって、
データの大きさが2倍になっているのか? 🤔

そんなことはないはずです。

離散フーリエ変換と対称性

数列 $u(n)$ を離散フーリエ変換したものが 数列 $U(k)$ であるとき

$U^*(N - k) = U(k)$ という対称性がある

*は複素共役

$U = a + bi$ のとき、 $U^* = a - bi$

なぜならば、👉

離散フーリエ変換と対称性

数列 $u(n)$ を離散フーリエ変換したものが 数列 $U(k)$ であるとき

$U^*(N - k) = U(k)$ なぜならば、

$u(n)$ が実数ならば、 $u^*(n) = u(n)$ なので

$$\begin{aligned} U^*(N - k) &= \sum_{n=0}^{N-1} u^*(n) \exp(i2\pi \frac{N - k}{N} n) \\ &= \sum_{n=0}^{N-1} u(n) \exp(i2\pi n) \exp(i2\pi \frac{-k}{N} n) \end{aligned}$$

離散フーリエ変換と対称性

$$\begin{aligned} U^*(N - k) &= \sum_{n=0}^{N-1} u^*(n) \exp(i2\pi \frac{N - k}{N} n) \\ &= \sum_{n=0}^{N-1} u(n) \exp(i2\pi n) \exp(i2\pi \frac{-k}{N} n) \end{aligned}$$

n が整数のとき 指数関数と三角関数の関係から

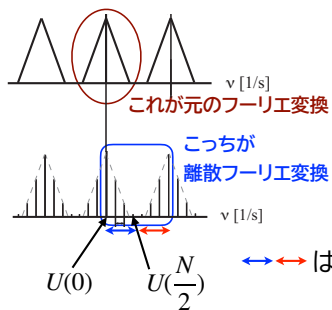
$\exp(i2\pi n) = \cos(2\pi n) + i \sin(2\pi n) = 1 + 0 = 1$, よって

$$U^*(N - k) = \sum_{n=0}^{N-1} u(n) \exp(i2\pi \frac{-k}{N} n) = U(k)$$

やっぱりデータは2倍にはなっていない

n が整数のとき 指数関数と三角関数の関係から

$\exp(i2\pi n) = \cos(2\pi n) + i \sin(2\pi n) = 1 + 0 = 1$, よって



$u(n)$ が実数ならば、 $U^*(k) = U(N - k)$ なので
離散フーリエ変換で表されている
最大の周波数は $N/2$

「指数関数2つの組でひとつの波」だから、
当然といえば当然ですね☺

第2部へ

第2部は画像データ圧縮

画像の細かいところを、見た目にはわからないようにごまかして、データ量を減らす

「細かいところ」はどのように表現されるか？

→周波数で表現される

そういうわけで、もう少しフーリエ変換とおつきあいください。

もっと一般的な原理から説明します。まずは数学の「行列」から。