

4.2 離散フーリエ積分

この節では、前節の離散フーリエ変換にならって、離散フーリエ積分 (離散逆フーリエ変換) を導きましょう。

周波数分解能を Δf [Hz]、サンプリング周波数を f_s [Hz] とし、周波数領域のスペクトル $X(f)$ の N 個の数値列 (離散化されたスペクトル) を

$$\begin{aligned} & \{ X(0), X(\Delta f), X(2\Delta f), X(3\Delta f), \dots, X((N/2)\Delta f), \\ & \quad X((N/2+1)\Delta f), \dots, X((N-2)\Delta f), X((N-1)\Delta f) \} \\ &= \{ X(0), X(\Delta f), X(2\Delta f), X(3\Delta f), \dots, X((N/2)\Delta f), \\ & \quad X((-N/2+1)\Delta f), \dots, X(-2\Delta f), X(-\Delta f) \} \end{aligned}$$

としましょう (前節の考察より、離散化されたスペクトルの後半の項は負の周波数のスペクトルに対応していることに注意しましょう)。離散フーリエ積分を得るために、積分区間が $[-f_s/2, f_s/2]$ となることに注意し、区分求積法を使って、近似されたフーリエ積分 $x(t)$ を求めると、

$$\begin{aligned} x(t) &\sim \int_{-\infty}^{\infty} X(f) e^{i2\pi f t} df = \int_{-f_s/2}^{f_s/2} X(f) e^{i2\pi f t} df \\ &\doteq \sum_{k=-N/2+1}^{N/2} X(k\Delta f) e^{i2\pi(k\Delta f)t} \Delta f \quad (\because \text{区分求積法}) \\ &= \Delta f \sum_{k=0}^{N-1} X(k\Delta f) e^{i2\pi(k\Delta f)t} \quad (\because \text{負の周波数の書き換え}) \end{aligned}$$

となります。さらに、サンプリング間隔を Δt [秒] とおくと、各時間 $n\Delta t$ [秒] ($n = 0, 1, 2, \dots, N-1$) における近似されたフーリエ積分の値 $x(n\Delta t)$ は、

$$x(n\Delta t) = \Delta f \sum_{k=0}^{N-1} X(k\Delta f) e^{i2\pi(k\Delta f)(n\Delta t)}$$

となります。なお、前節の③式より、関係式

$$T_0 = \frac{1}{\Delta f} \text{ [秒]}$$

が成り立ち、観測区間 $[0, T_0]$ は $[0, 1/\Delta f]$ となることも注意しておきましょう。続けて、近似されたフーリエ積分の値 $x(n\Delta t)$ に前節の④式を代入すると、

$$x(n\Delta t) = \Delta f \sum_{k=0}^{N-1} X(k\Delta f) e^{i\frac{2\pi n k}{N}}$$

となります。ここで、簡潔に記述するために、記号

$$W'_N = e^{i\frac{2\pi}{N}}$$

および前節の⑤式の記号を導入すると、離散化されたフーリエ積分は、

$$x_n = \Delta f \sum_{k=0}^{N-1} X_k W_N'^{nk} \quad (n = 0, 1, 2, \dots, N-1)$$

となります。離散フーリエ変換との対応を考慮してさらに書き直すと、離散化されたフーリエ積分は、

$$x_n = \frac{1}{T_0} \sum_{k=0}^{N-1} X_k W_N'^{nk} \quad (n = 0, 1, 2, \dots, N-1)$$

となります ($\because$ 前節の③式)。この式を **離散フーリエ積分 (離散逆フーリエ変換)** と呼びます。

理解を深めるために、離散フーリエ変換と同じように、 $N = 4$ の場合の離散フーリエ積分を書き下してみると、

$$\begin{cases} x_0 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^0 + X_2 W_4'^0 + X_3 W_4'^0), \\ x_1 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^1 + X_2 W_4'^2 + X_3 W_4'^3), \\ x_2 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^2 + X_2 W_4'^4 + X_3 W_4'^6), \\ x_3 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^3 + X_2 W_4'^6 + X_3 W_4'^9) \end{cases}$$

となり、関係式 $W_4'^4 = 1 (= W_4'^0)$ を用いると、

$$\begin{cases} x_0 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^0 + X_2 W_4'^0 + X_3 W_4'^0), \\ x_1 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^1 + X_2 W_4'^2 + X_3 W_4'^3), \\ x_2 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^2 + X_2 W_4'^0 + X_3 W_4'^2), \\ x_3 = \frac{1}{T_0} (X_0 W_4'^0 + X_1 W_4'^3 + X_2 W_4'^2 + X_3 W_4'^1) \end{cases}$$

となります。また、行列の形式

$$\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} = \frac{1}{T_0} \begin{pmatrix} W_4'^0 & W_4'^0 & W_4'^0 & W_4'^0 \\ W_4'^0 & W_4'^1 & W_4'^2 & W_4'^3 \\ W_4'^0 & W_4'^2 & W_4'^0 & W_4'^2 \\ W_4'^0 & W_4'^3 & W_4'^2 & W_4'^1 \end{pmatrix} \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix}$$

で表現することもできます。さらに、

$$W_4'^0 = 1, \quad W_4'^1 = i, \quad W_4'^2 = -1, \quad W_4'^3 = -i$$

を代入すれば、具体的に離散フーリエ積分

$$\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} = \frac{1}{T_0} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix}$$

を求めることができます。

ここで、離散フーリエ変換と離散フーリエ積分の関係について述べておきます。そもそも、離散フーリエ変換は離散化された波を離散化されたスペクトルに変換し、離散フーリエ積分は離散化されたスペクトルを離散化された波に変換します。すなわち、離散化された波を離散フーリエ変換し、さらに離散フーリエ積分すると、元の離散化された波に戻ります。したがって、離散フーリエ変換を表す行列を A_N とし、離散フーリエ積分を表す行列を B_N とすると、

$$A_N B_N = I_N \quad (I_N \text{ は } N \times N \text{ 単位行列})$$

の関係を満たします。 $N = 4$ の場合、

$$A_4 = \frac{T_0}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix}, \quad B_4 = \frac{1}{T_0} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

より、

$$A_4 B_4 = \frac{T_0}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix} \cdot \frac{1}{T_0} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

となります。また、行列 A_N および B_N は、それぞれ、行列自身の転置行列となっていて、

$$A_N^T = A_N \quad \text{および} \quad B_N^T = B_N$$

が成り立ちます。さらに、ノルム (スカラー) を調整し、

$$A'_N = \frac{\sqrt{N}}{T_0} A_N \quad \text{および} \quad B'_N = \frac{T_0}{\sqrt{N}} B_N$$

とおくと、行列 A'_N と行列 B'_N は互いに共役転置⁴の関係にあり、

$$A'_N{}^* = B'_N \quad (B'_N{}^* = A'_N)$$

が成り立ちます。 $N = 4$ の場合、

$$A'_4 = \frac{\sqrt{4}}{T_0} \cdot \frac{T_0}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix} = \frac{1}{\sqrt{4}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix},$$

$$B'_4 = \frac{T_0}{\sqrt{4}} \cdot \frac{1}{T_0} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} = \frac{1}{\sqrt{4}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

⁴行列 C の共役転置行列 C^* は、行列 C の各成分の複素共役をとり、さらに転置をほどこした行列です (行列 C を転置し、さらに各成分の複素共役をとってもよい)。i.e. $C^* = [\overline{c_{ij}}]^T = [\overline{c_{ji}}]$ 。

より、

$$\begin{aligned} A_4'^* &= \frac{1}{\sqrt{4}} \begin{pmatrix} \overline{1} & \overline{1} & \overline{1} & \overline{1} \\ \overline{1} & \overline{-i} & \overline{-1} & \overline{i} \\ \overline{1} & \overline{-1} & \overline{1} & \overline{-1} \\ \overline{1} & \overline{i} & \overline{-1} & \overline{-i} \end{pmatrix}^T = \frac{1}{\sqrt{4}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}^T \\ &= \frac{1}{\sqrt{4}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} = B_4' \end{aligned}$$

となります。このような、

$$A_N' B_N' = I_N \quad \text{かつ} \quad A_N'^* = B_N' \quad (B_N'^* = A_N')$$

を満たす行列は、数学的にも非常に重要です (特に線形写像として)。また、応用面においても、システム解析やシステム設計の分野でこのような行列が様々な形で利用されています⁵。

4.3 高速フーリエ変換

第4.1節では、コンピュータを使って離散的な波 (デジタルデータ) を処理 (フーリエ変換) できるように、連続的な波のフーリエ変換から離散的な波のフーリエ変換である離散フーリエ変換

$$X_k = \frac{T_0}{N} \sum_{n=0}^{N-1} x_n W_N^{nk} \quad (k = 0, 1, 2, \dots, N-1)$$

を導入しました。この離散フーリエ変換の数式をそのまま計算すると、複素数の乗算の回数だけでも N^2 回の演算が必要となります。言い換えれば、離散フーリエ変換の計算量 (Order) は $O(N^2)$ となります。そのため、 N が大きくなるにしたがって計算量が爆発的に増加し、コンピュータを用いても、 N がある程度大きくなると計算不可能となってしまいます。この計算量の問題を解決した離散フーリエ変換のアルゴリズム (algorithm) に、西暦 1965 年にジェイムズ・クーリー (J. W. Cooley) とジョン・テューキー (J. W. Tukey) によって発見された、高速フーリエ変換 (Fast Fourier Transform; **FFT**) と呼ばれる方法があります⁶。この方法は、離散フーリエ変換 (DFT) の周期性および対称性に着目して計算量を減らしたアルゴリズムで、 $N = 2^l$ (l は自然数) のときオーダーは $O(N \log_2 N)$ となります⁷。また、この方法は、再帰的に分割統治法 (divide and conquer) を繰り返すアルゴリズムとなっており、演算構造を図式化すると蝶のはねのようになることから、**バタフライ演算**とも呼ばれます。

⁵工学系では、行列を計算したつもりが、実は離散フーリエ変換を計算していたということがよくあります。

⁶Cooley-Tukey アルゴリズムとも呼ばれ、後に高速フーリエ変換と呼ばれるようになりました。一般に、高速フーリエ変換といえば、Cooley-Tukey アルゴリズムによる離散フーリエ変換を指します。また、西暦 1805 年ごろにガウス (Gauss) が同様のアルゴリズムを独立に発見していたといわれています。

⁷ $N \neq 2^l$ の場合についても、高速フーリエ変換を改良したアルゴリズムが開発されています。また、精度は落ちますが、データ数が $N = 2^l$ となるようにデータに 0 を補充してから、高速フーリエ変換することもできます。

例として、 $N = 4 = 2^2$ ($l = 2$) の場合の高速フーリエ変換について見て行きましょう。まず、 W_4^n の対称性に着目して図 4.10 のように x_0 と x_2 および x_1 と x_3 をそれぞれペアにします。

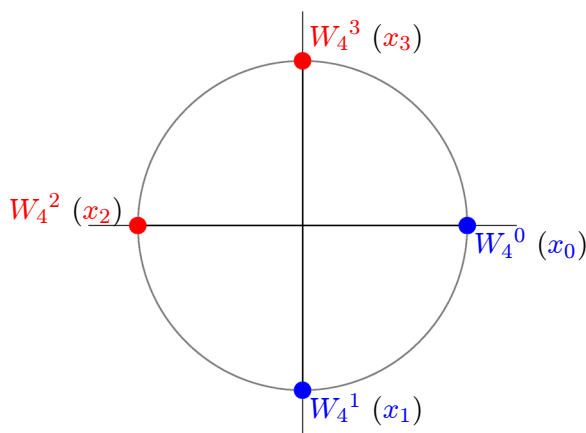


図 4.10: 4 等分された単位円周上の点

言い換えると、以下のように行列の形式で表した離散フーリエ変換の第 2 列と第 3 列を入れ替えます。

$$\begin{aligned}
 \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix} &= \frac{T_0}{4} \begin{pmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^0 & W_4^2 \\ W_4^0 & W_4^3 & W_4^2 & W_4^1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} \\
 &= \frac{T_0}{4} \begin{pmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^2 & W_4^1 & W_4^3 \\ W_4^0 & W_4^0 & W_4^2 & W_4^2 \\ W_4^0 & W_4^2 & W_4^3 & W_4^1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_2 \\ x_1 \\ x_3 \end{pmatrix}
 \end{aligned}$$

次に、以下のように分割し、小行列で表します (分割)。

$$\begin{aligned}
 (\text{与式}) &= \frac{T_0}{4} \left(\begin{array}{cc|cc} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^2 & W_4^1 & W_4^3 \\ \hline W_4^0 & W_4^0 & W_4^2 & W_4^2 \\ W_4^0 & W_4^2 & W_4^3 & W_4^1 \end{array} \right) \begin{pmatrix} x_0 \\ x_2 \\ x_1 \\ x_3 \end{pmatrix} \\
 &= \frac{T_0}{4} \left(\begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^1 & W_4^3 \end{bmatrix} \right) \begin{pmatrix} x_0 \\ x_2 \\ x_1 \\ x_3 \end{pmatrix} \\
 &= \frac{T_0}{4} \left(\begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_2 \end{bmatrix} + \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^1 & W_4^3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \right) \\
 &= \frac{T_0}{4} \left(\begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_2 \end{bmatrix} + \begin{bmatrix} W_4^2 & W_4^2 \\ W_4^3 & W_4^1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \right)
 \end{aligned}$$

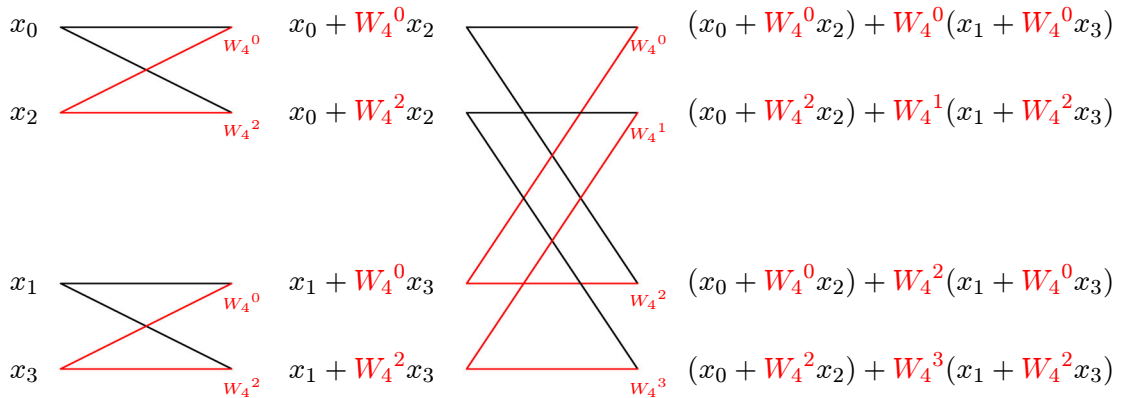
このとき、各行の第2項をそれぞれ W_4^0 と W_4^2 でくくると同じ小行列が現れることがわかります。

$$(\text{与式}) = \frac{T_0}{4} \left(\begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_2 \end{bmatrix} + W_4^0 \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^1 & W_4^3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \right. \\ \left. \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_2 \end{bmatrix} + W_4^2 \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^1 & W_4^3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \right)$$

さらに、演算順序を変えないで小行列を展開します (統治)。

$$(\text{与式}) = \frac{T_0}{4} \begin{pmatrix} (W_4^0 x_0 + W_4^0 x_2) + W_4^0 (W_4^0 x_1 + W_4^0 x_3) \\ (W_4^0 x_0 + W_4^2 x_2) + W_4^0 (W_4^1 x_1 + W_4^3 x_3) \\ (W_4^0 x_0 + W_4^0 x_2) + W_4^2 (W_4^0 x_1 + W_4^0 x_3) \\ (W_4^0 x_0 + W_4^2 x_2) + W_4^2 (W_4^1 x_1 + W_4^3 x_3) \end{pmatrix} \\ = \frac{T_0}{4} \begin{pmatrix} (W_4^0 x_0 + W_4^0 x_2) + W_4^0 (W_4^0 x_1 + W_4^0 x_3) \\ (W_4^0 x_0 + W_4^2 x_2) + W_4^1 (W_4^0 x_1 + W_4^2 x_3) \\ (W_4^0 x_0 + W_4^0 x_2) + W_4^2 (W_4^0 x_1 + W_4^0 x_3) \\ (W_4^0 x_0 + W_4^2 x_2) + W_4^3 (W_4^0 x_1 + W_4^2 x_3) \end{pmatrix} \\ = \frac{T_0}{4} \begin{pmatrix} (x_0 + W_4^0 x_2) + W_4^0 (x_1 + W_4^0 x_3) \\ (x_0 + W_4^2 x_2) + W_4^1 (x_1 + W_4^2 x_3) \\ (x_0 + W_4^0 x_2) + W_4^2 (x_1 + W_4^0 x_3) \\ (x_0 + W_4^2 x_2) + W_4^3 (x_1 + W_4^2 x_3) \end{pmatrix} \quad (\because W_4^0 = 1)$$

このようにして得られた数式が高速フーリエ変換です。なお、同じ数式が繰り返し現れるため、全体の乗算の回数 (計算量) を $4^2 (= 16)$ から $4 \log_2 4 (= 8)$ に減らすことができます。また、この数式の演算構造を図式化すると図 4.11 のようになります。



* 黒線はそのまま加算し、赤線は線の下に W_4^n を掛けてから加算します。

図 4.11: バタフライ演算 ($N=4$)

理解を深めるために、 $N = 8 = 2^3$ ($l = 3$) の場合の高速フーリエ変換についても見ておきましょう。まず、 $N = 4$ の場合と同様に、 x_0 と x_4 , x_2 と x_6 , x_1 と x_5 , x_3 と x_7 をそれぞれペアにします (図 4.12 参照)。

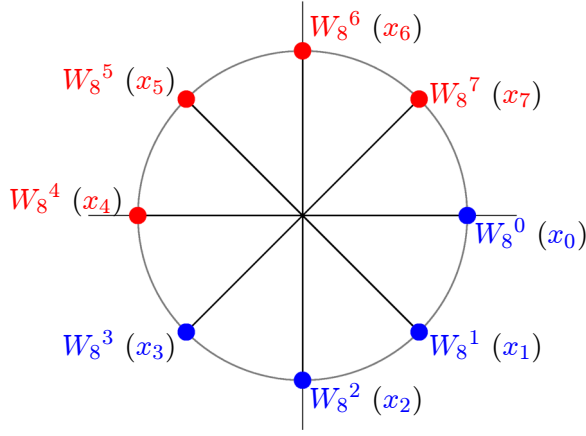


図 4.12: 8 等分された単位円周上の点

すなわち、行列の形式で表した離散フーリエ変換の列を交換して以下のように変形します。

$$\begin{aligned}
 \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \end{pmatrix} &= \frac{T_0}{8} \begin{pmatrix} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 \\ W_8^0 & W_8^1 & W_8^2 & W_8^3 & W_8^4 & W_8^5 & W_8^6 & W_8^7 \\ W_8^0 & W_8^2 & W_8^4 & W_8^6 & W_8^0 & W_8^2 & W_8^4 & W_8^6 \\ W_8^0 & W_8^3 & W_8^6 & W_8^1 & W_8^4 & W_8^7 & W_8^2 & W_8^5 \\ W_8^0 & W_8^4 & W_8^0 & W_8^4 & W_8^0 & W_8^4 & W_8^0 & W_8^4 \\ W_8^0 & W_8^5 & W_8^2 & W_8^7 & W_8^4 & W_8^1 & W_8^6 & W_8^3 \\ W_8^0 & W_8^6 & W_8^4 & W_8^2 & W_8^0 & W_8^6 & W_8^4 & W_8^2 \\ W_8^0 & W_8^7 & W_8^6 & W_8^5 & W_8^4 & W_8^3 & W_8^2 & W_8^1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \\
 &= \frac{T_0}{8} \begin{pmatrix} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 \\ W_8^0 & W_8^4 & W_8^2 & W_8^6 & W_8^1 & W_8^5 & W_8^3 & W_8^7 \\ W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^2 & W_8^2 & W_8^6 & W_8^6 \\ W_8^0 & W_8^4 & W_8^6 & W_8^2 & W_8^3 & W_8^7 & W_8^1 & W_8^5 \\ W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^4 & W_8^4 \\ W_8^0 & W_8^4 & W_8^2 & W_8^6 & W_8^5 & W_8^1 & W_8^7 & W_8^3 \\ W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^6 & W_8^6 & W_8^2 & W_8^2 \\ W_8^0 & W_8^4 & W_8^6 & W_8^2 & W_8^7 & W_8^3 & W_8^5 & W_8^1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_4 \\ x_2 \\ x_6 \\ x_1 \\ x_5 \\ x_3 \\ x_7 \end{pmatrix}
 \end{aligned}$$

次に、以下のように分割し、小行列で表します (分割)。

$$\begin{aligned}
 \text{(与式)} &= \frac{T_0}{8} \left(\begin{array}{cccc|cccc} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 \\ W_8^0 & W_8^4 & W_8^2 & W_8^6 & W_8^1 & W_8^5 & W_8^3 & W_8^7 \\ W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^2 & W_8^2 & W_8^6 & W_8^6 \\ W_8^0 & W_8^4 & W_8^6 & W_8^2 & W_8^3 & W_8^7 & W_8^1 & W_8^5 \\ \hline W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^4 & W_8^4 \\ W_8^0 & W_8^4 & W_8^2 & W_8^6 & W_8^5 & W_8^1 & W_8^7 & W_8^3 \\ W_8^0 & W_8^0 & W_8^4 & W_8^4 & W_8^6 & W_8^6 & W_8^2 & W_8^2 \\ W_8^0 & W_8^4 & W_8^6 & W_8^2 & W_8^7 & W_8^3 & W_8^5 & W_8^1 \end{array} \right) \begin{pmatrix} x_0 \\ x_4 \\ x_2 \\ x_6 \\ \hline x_1 \\ x_5 \\ x_3 \\ x_7 \end{pmatrix}
 \end{aligned}$$

[illegible]

$$(\text{与式}) = \frac{T_0}{8} \left(\begin{pmatrix} \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^0 & W_8^4 \end{bmatrix} \begin{bmatrix} x_0 \\ x_4 \end{bmatrix} + W_8^0 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_2 \\ x_6 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^0 & W_8^4 \end{bmatrix} \begin{bmatrix} x_0 \\ x_4 \end{bmatrix} + W_8^4 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_2 \\ x_6 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^0 & W_8^4 \end{bmatrix} \begin{bmatrix} x_0 \\ x_4 \end{bmatrix} + W_8^0 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_2 \\ x_6 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^0 & W_8^4 \end{bmatrix} \begin{bmatrix} x_0 \\ x_4 \end{bmatrix} + W_8^4 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_2 \\ x_6 \end{bmatrix} \end{pmatrix} + W_8^0 \begin{pmatrix} \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^1 & W_8^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_5 \end{bmatrix} + W_8^0 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^3 & W_8^7 \end{bmatrix} \begin{bmatrix} x_3 \\ x_7 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^1 & W_8^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_5 \end{bmatrix} + W_8^4 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_3 \\ x_7 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^1 & W_8^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_5 \end{bmatrix} + W_8^0 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^3 & W_8^7 \end{bmatrix} \begin{bmatrix} x_3 \\ x_7 \end{bmatrix} \\ \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^1 & W_8^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_5 \end{bmatrix} + W_8^4 \begin{bmatrix} W_8^0 & W_8^0 \\ W_8^2 & W_8^6 \end{bmatrix} \begin{bmatrix} x_3 \\ x_7 \end{bmatrix} \end{pmatrix} \right)$$

さらに、演算順序を変えないで小行列を展開します (統治)。

$$\begin{aligned} (\text{与式}) &= \frac{T_0}{8} \begin{pmatrix} ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^0 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^0 ((W_8^1 x_1 + W_8^5 x_5) + W_8^0 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^2 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^2 ((W_8^1 x_1 + W_8^5 x_5) + W_8^4 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^4 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^4 ((W_8^1 x_1 + W_8^5 x_5) + W_8^0 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^6 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^6 ((W_8^1 x_1 + W_8^5 x_5) + W_8^4 (W_8^3 x_3 + W_8^7 x_7)) \end{pmatrix} \\ &= \frac{T_0}{8} \begin{pmatrix} ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^0 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^0 ((W_8^1 x_1 + W_8^5 x_5) + W_8^0 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^2 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^2 ((W_8^1 x_1 + W_8^5 x_5) + W_8^4 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^4 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^4 ((W_8^1 x_1 + W_8^5 x_5) + W_8^0 (W_8^3 x_3 + W_8^7 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^6 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^6 ((W_8^1 x_1 + W_8^5 x_5) + W_8^4 (W_8^3 x_3 + W_8^7 x_7)) \end{pmatrix} \\ &= \frac{T_0}{8} \begin{pmatrix} ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^0 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^1 ((W_8^0 x_1 + W_8^4 x_5) + W_8^0 (W_8^2 x_3 + W_8^6 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^2 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^3 ((W_8^0 x_1 + W_8^4 x_5) + W_8^4 (W_8^2 x_3 + W_8^6 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^4 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^0 (W_8^2 x_2 + W_8^6 x_6)) + W_8^5 ((W_8^0 x_1 + W_8^4 x_5) + W_8^0 (W_8^2 x_3 + W_8^6 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^6 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^4 (W_8^2 x_2 + W_8^6 x_6)) + W_8^7 ((W_8^0 x_1 + W_8^4 x_5) + W_8^4 (W_8^2 x_3 + W_8^6 x_7)) \end{pmatrix} \\ &= \frac{T_0}{8} \begin{pmatrix} ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^0 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^2 (W_8^0 x_2 + W_8^4 x_6)) + W_8^1 ((W_8^0 x_1 + W_8^4 x_5) + W_8^2 (W_8^0 x_3 + W_8^4 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^2 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^6 (W_8^0 x_2 + W_8^4 x_6)) + W_8^3 ((W_8^0 x_1 + W_8^4 x_5) + W_8^6 (W_8^0 x_3 + W_8^4 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^0 (W_8^0 x_2 + W_8^0 x_6)) + W_8^4 ((W_8^0 x_1 + W_8^0 x_5) + W_8^0 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^2 (W_8^0 x_2 + W_8^4 x_6)) + W_8^5 ((W_8^0 x_1 + W_8^4 x_5) + W_8^2 (W_8^0 x_3 + W_8^4 x_7)) \\ ((W_8^0 x_0 + W_8^0 x_4) + W_8^4 (W_8^0 x_2 + W_8^0 x_6)) + W_8^6 ((W_8^0 x_1 + W_8^0 x_5) + W_8^4 (W_8^0 x_3 + W_8^0 x_7)) \\ ((W_8^0 x_0 + W_8^4 x_4) + W_8^6 (W_8^0 x_2 + W_8^4 x_6)) + W_8^7 ((W_8^0 x_1 + W_8^4 x_5) + W_8^6 (W_8^0 x_3 + W_8^4 x_7)) \end{pmatrix} \end{aligned}$$

以上より、 $W_8^0 = 1$ に注意すれば、 $N = 8 = 2^3$ ($l = 3$) の場合の高速フーリエ変換

$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \end{pmatrix} = \frac{T_0}{8} \begin{pmatrix} ((x_0 + W_8^0 x_4) + W_8^0 (x_2 + W_8^0 x_6)) + W_8^0 ((x_1 + W_8^0 x_5) + W_8^0 (x_3 + W_8^0 x_7)) \\ ((x_0 + W_8^4 x_4) + W_8^2 (x_2 + W_8^4 x_6)) + W_8^1 ((x_1 + W_8^4 x_5) + W_8^2 (x_3 + W_8^4 x_7)) \\ ((x_0 + W_8^0 x_4) + W_8^4 (x_2 + W_8^0 x_6)) + W_8^2 ((x_1 + W_8^0 x_5) + W_8^4 (x_3 + W_8^0 x_7)) \\ ((x_0 + W_8^4 x_4) + W_8^6 (x_2 + W_8^4 x_6)) + W_8^3 ((x_1 + W_8^4 x_5) + W_8^6 (x_3 + W_8^4 x_7)) \\ ((x_0 + W_8^0 x_4) + W_8^0 (x_2 + W_8^0 x_6)) + W_8^4 ((x_1 + W_8^0 x_5) + W_8^0 (x_3 + W_8^0 x_7)) \\ ((x_0 + W_8^4 x_4) + W_8^2 (x_2 + W_8^4 x_6)) + W_8^5 ((x_1 + W_8^4 x_5) + W_8^2 (x_3 + W_8^4 x_7)) \\ ((x_0 + W_8^0 x_4) + W_8^4 (x_2 + W_8^0 x_6)) + W_8^6 ((x_1 + W_8^0 x_5) + W_8^4 (x_3 + W_8^0 x_7)) \\ ((x_0 + W_8^4 x_4) + W_8^6 (x_2 + W_8^4 x_6)) + W_8^7 ((x_1 + W_8^4 x_5) + W_8^6 (x_3 + W_8^4 x_7)) \end{pmatrix}$$

が得られます。なお、 $N = 4$ の場合と同様に、同じ数式が繰り返し現れるため、全体の乗算の回数 (計算量) を $8^2 (= 64)$ から $8 \log_2 8 (= 24)$ に減らすことができます。また、この数式の演算構

造を図式化すると図 4.11 のようなバタフライ演算を描くことができますが、これは読者への宿題としておきます⁸。

この章の最後に、離散フーリエ積分と離散フーリエ変換についてまとめておくと、下表のようになります。

	離散フーリエ積分	離散フーリエ変換
数値列	$x_n = \frac{1}{T_0} \sum_{k=0}^{N-1} X_k W_N'^{nk}$ $(n = 0, 1, 2, \dots, N-1)$	$X_k = \frac{T_0}{N} \sum_{n=0}^{N-1} x_n W_N^{nk}$ $(k = 0, 1, 2, \dots, N-1)$

表 4.2: 離散フーリエ積分 (離散逆フーリエ変換) と離散フーリエ変換

【注意】 一般的な離散フーリエ変換の定義は、本テキストの周波数スペクトル密度の定義と一致し、 T_0 が消えてすっきりした数式となっています。ただし、 T_0 は Δf を決める重要な値であることに注意してください。

$$\begin{cases}
 x_n = \frac{1}{T_0} \sum_{k=0}^{N-1} X_k W_N'^{nk}, \\
 X_k = \frac{T_0}{N} \sum_{n=0}^{N-1} x_n W_N^{nk}
 \end{cases}
 \quad (\text{本テキストの定義}),$$

$$\begin{cases}
 x_n = \sum_{k=0}^{N-1} X_k W_N'^{nk}, \\
 X_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n W_N^{nk}
 \end{cases}
 \quad (\text{一般的な定義}),$$

$$\begin{cases}
 x_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k W_N'^{nk}, \\
 X_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n W_N^{nk}
 \end{cases}
 \quad (\text{数式処理ソフト Mathematica のデフォルトの定義})$$

⁸一般的な、 $N = 2^l$ (l は自然数) の高速フーリエ変換の数式を求めてみましょう。