高速フーリエ変換

高校１年３組４２番　平野正浩

1　はじめに

本日は数学研究部にわざわざお越し頂きありがとうございます。初めてTEXというものをつかって部誌を書きますので、みにくい部分は多々あるかと思いますがご了承ください。

1.1　Introduction

まず、フーリエ変換の対象になっている「波」がどのような概念のものかを説明します。波のイメージとしてはなにかが揺れ動いていることと、それが空間的あるいは時間的に広がっていることだと認識できます。単に物が動いているだけでは空間的や時間的な広がりはありません。要するに空間的、時間的に揺れ動きながら広がっていることと認識できます。厳密に言うと波は、規則的に繰り返す周期的な現象、とかけます。さらに波には「時間とともにどのように振動するか」という要素と「１秒間に何回振動する波がどのように混ざり合っているか」という要素があります。これら二つの要素は、実は数学的な変換によって相互に結びついています。これが**フーリエ変換**です。具体的には時間領域から周波数領域への変換を「フーリエ変換」、その逆を「逆フーリエ変換」といいます。いったい、それはどんなものなのかを調べるのがこの記事の目的です。

1.2　波の周波数

波の特徴を決めている要素の１つに、「ある一点で観測した物理量が１秒間に何回振動するか」があります。この１秒間に何回振動するかということを「**周波数**」とよび、その単位は [Hz]（ヘルツ）で、Hz の次元は 1/s です。

1.3　スペクトル

時間領域の波をフーリエ変換して作り出した周波数領域のグラフ、あるいは、周波数と逆数の関係にある波長を横軸にしたグラフを「**スペクトル**」と呼びます。

2　基本事項

ここでは、フーリエ解析を学ぶ上において必要な知識を整理します。

2.1　関数の直行性

直行とはある直線 AB と直線 CD が点 P で交差していて、$\angle APD$ と $\angle BPD$ が等しいとき、これら二本の直線は直角に交差していることです。
いま、2 つのベクトルの内積を考えます。$\vec{a}$ と $\vec{b}$ が角度 θ で交差しているとき、その内積 C は、$C = |\vec{a}| \cdot |\vec{b}| \cdot \cos\theta$ で与えられます。このとき、$C = 0$ ならば $\theta = \pi/2[rad]$ といえます。つまり二つのベクトル $\vec{a}$ と $\vec{b}$ は直行しているということになります。
このことを使って直行の概念を関数で捉えてみましょう。ここで「ベクトルの内積」に相当する「関数の内積」は、2 つの関数が定義される領域内で、関数の積の積分をすることです。つまり、

$$C = \int_a^b f(x) \cdot g(x) dx$$

この積分値 C が 0 となるときに、関数 $f(x)$ と $g(x)$ は互いに直行しているといえます。これは**フーリエ級数**の基礎となる重要な概念なのでしっかり押さえてくださいね。
ここで実際に例をひとつあげてみます。二つの関数として $\sin x$ と $\cos x$ を考えます。

$$\begin{aligned} & \int_0^{2\pi} \sin x \cdot \cos x dx \\ = \quad & \frac{1}{2} \int_0^{2\pi} \sin(2x) dx = \frac{1}{2} \times \left(-\frac{1}{2}\right) \Big[\cos(2x)\Big]_0^{2\pi} \\ = \quad & 0 \end{aligned}$$

このようにして定義域内で、2 つの関数の積の定積分が 0 となるので、$\sin x$ と $\cos x$ が直行していることが確かめられました。

同様にして $\sin mx$ と $\cos nx$ $m \neq n$ についても調べておきましょう。

$$\begin{aligned} & \int_0^{2\pi} \sin(mx) \cdot \cos(nx) dx \\ = \quad & \frac{1}{2}\Big[\cos(m-n)x - \cos(m+n)x\Big]_0^{2\pi} \end{aligned}$$

これも定積分値は 0 となり、周期が異なる正弦関数も互いに直行していることが確かめられました。これってすごくないですかw？同じようにして周期の違う余弦関数も互いに直行しています。

2.2 周期関数

ある、T が存在し、$t \in \mathbb{R}$ であるすべての t に対して、

$$f(t) = f(t+T)$$

を満たすとき、この関数 $f(x)$ を**周期関数**といいます。この条件を満足する最小の T を、この関数の周期といいます。

2.3 区分的に滑らかということ

ある関数のグラフを書いたときにグラフがつながっているものが「連続な関数」で、途中で跳んでいるようなものが「不連続な関数」です。こういうと何かあやふやなものを感じますのでもう少し数学的に説明します。ある関数 $f(x)$ の a 点で

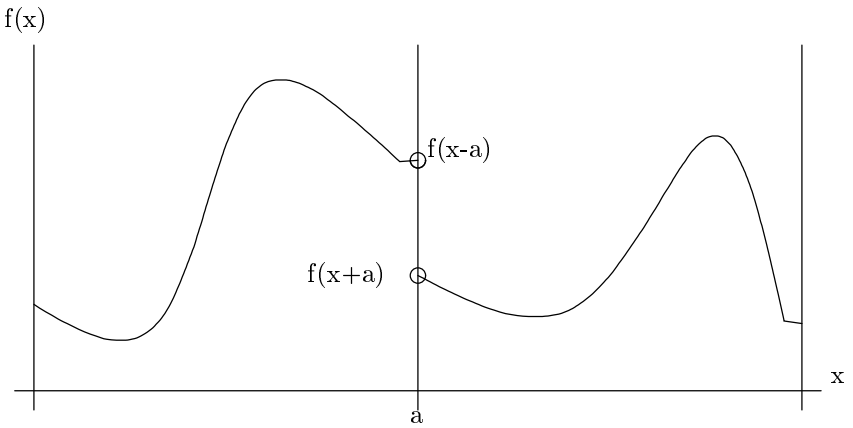
右からの極限値

$$\lim_{x \to a+0} f(x)$$

と, 左からの極限値

$$\lim_{x \to a-0} f(x)$$

を考えたとき、この２つの極限値がともに有限な値として存在し、かつ、これらの２つの有限な値が一致したときを**連続**、そうでないときを**不連続**といいます。



今見ている区間の中で、有限個の点（上で示した不連続点）を除いて連続な関数があるとき、この関数を区分的に連続と呼びます。さらに、区分的に連続な関数 $f(x)$ の導関数 $f'(x)$ がさらに連続であるとき、関数 $f(x)$ は区分的に滑らかであるといえます。すなわち、有界閉区間 $[a,b]$ で定義された関数 $f(x)$ に対して、$f'(x)$ が $[a,b]$ で区分的に連続であるとき、$f(x)$ は区分的に滑らかである。$f(x)$ が区分的に滑らかであったら、区分的に連続であることが容易にわかります。

3　フーリエ級数

互いに周期の異なる正弦関数同士や余弦関数同士、あるいは同じ周期であっても正弦関数と余弦関数は互いに直行していることをすでに確かめましたね。ここでは直行した関数を組み合わせることによって、新たな関数を作り出すイメージを示したいと思います。なかなか本題に近くなってきましたね。気合も入ってきたところです。

3.1　その背景

昔々、フーリエというフランス・ブルゴーニュ地方に生まれた数学者がいました。彼が始めてフーリエ級数展開なるものを開発したころ、周りの数学者からその証明の杜撰さを非難されました。当時のフーリエの手法は任意の周期関数を三角関数の組み合わせであらわすというものでした。フーリエはすべての周期関数に対して彼の手法が適用できると主張していたのに対し、それに異を唱える数学者も多かったのです。実は、彼の方法は稚拙といわれ

ても仕方のないものでした。ですが、彼の手法は現代でも画期的なものであることはだれも否定しなかったのです。確かにフーリエの手法で解析できないものもありますがそれは関数といっても特殊なもので特に役に立たないものばかりなのです。いまからこの解析の手法とその用途について述べようと思います。

3.2 フーリエ級数

いま $[-\pi, \pi]$ で定義された関数 $f(x)$ が、

$$F(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$$

という風に展開できたと仮定する。このとき、$a_0, a_1, a_2 \cdots$ や $b_1, b_2 \cdots$ をフーリエ係数といいます。

3.3 フーリエ係数

フーリエ級数のそれぞれの係数、$a_0, a_1, a_2 \cdots$ や $b_1, b_2 \cdots$ が、原始関数 $F(x)$ の中にどれだけの大きさで含まれているかをみていきます。
いま、関数 $F(x)$ の周期を 2π とします。$F(x) = \sin nx$ とすると、

$$\begin{aligned} & \int_0^{2\pi} F(x) \sin nx dx \\ = & \int_0^{2\pi} \sin nx \cdot \sin nx dx = \int_0^{2\pi} \frac{1}{2} (1 - \cos 2nx)\, dx \\ = & \frac{1}{2} \Big[x - \frac{1}{2n} \sin 2nx \Big]_0^{2\pi} = \pi \end{aligned}$$

となります。従って、$\sin nx$ の成分の大きさを 1 に規格化 [1]するために π で割ります。よって、$\sin nx$ の成分の大きさを求めるには、

$$\frac{1}{\pi} \int_0^{2\pi} F(x) \sin nx dx$$

を計算すればいいということになります。これを一般的に表現すれば、各周波数成分について、

$$b_n = \frac{1}{\pi} \int_0^{2\pi} F(x) \sin nx dx$$

[1]規格化 合成された関数の振幅をいつも一定にするために、合成された関数をその値で割ること。

となり、$\cos nx$ の係数も同様に、

$$a_n = \frac{1}{\pi}\int_0^{2\pi} F(x)\cos nx dx$$

となります。また、$\frac{a_0}{2}$ は $F(x)$ の平均値に対応しているので、

$$a_0 = \frac{1}{2\pi}\int_0^{2\pi} F(x) dx$$

となります。

3.4 周波数成分と位相

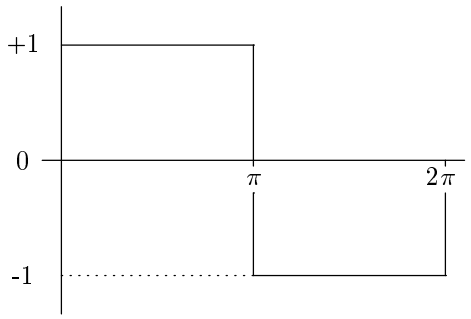
同一周期の三角関数 $\sin x$ と $\cos x$、$\sin 2x$ と $\cos 2x \cdots$、$\sin nx$ と $\cos nx$ は、同じ周期で位相が 90° 違う成分ですね。したがって、n 番目の周期で振動する成分の大きさ（絶対値）は、

$$r_n = \sqrt{a_n^2 + b_n^2}$$

となりますね。この r_n を n 周期成分の振幅とよび、周期を横軸に、r_n を横軸にプロットするとスペクトルの振幅のグラフ得られます。また、a_n と b_n の正接角は、その周期成分の初期位相に相当し、

$$\theta_n = \tan^{-1}\frac{b_n}{a_n}$$

で計算されます。
ここで、注意しなければならないことは、スペクトルの振幅だけを表示すると初期位相の項 (θ_n) が見えなくなってしまうということです。つまり周波数の分布はわかってもそれぞれの周波数成分の初期位相がわからなくなってしまうということです。つまり、スペクトルの振幅だけからは元の信号波形を再現することはできないということです。ここで、実際に方形波と呼ばれる波形のフーリエ係数を求めてみましょう。



これが方形波と呼ばれるものです。
この方形波は 0 から π までは+1、π から 2π までは-1 という関数です。すると sin の第一周期は、

$$\begin{aligned} & \frac{1}{\pi}\Big(\int_0^{\pi} 1\cdot\sin x dx + \int_{\pi}^{2\pi}(-1)\cdot\sin x dx\Big) \\ = \quad & \frac{1}{\pi}\Big(\int_0^{\pi}\sin x dx - \int_0^{\pi}(-\sin)\,dx\Big) \\ = \quad & \frac{2}{\pi}\int_0^{\pi}\sin x dx = \frac{2}{\pi}\big[-\cos x\big]_0^{\pi} = \frac{2}{\pi}(1+1) \\ = \quad & \frac{4}{\pi} \simeq 1.27324\cdots \end{aligned}$$

となります。sin の偶数周期の係数は 0 になります。なぜなら、方形波の+1 と-1 との領域で、どちらにも同じだけの sin 波の波の数があるわけですから、その積分は 0 から π までと、π から 2π までとで打ち消しあうためです。 n が正の奇数の時には、

$$\begin{aligned} & \frac{2}{\pi}\int_0^{\pi}\sin nx dx = \frac{2}{\pi}\cdot\frac{1}{n}\big[-\cos nx\big]_0^{\pi} \\ = \quad & \frac{4}{\pi}\cdot\frac{1}{n} \end{aligned}$$

となります。この結果は第 1 周期の大きさを 1 に規格化すると、第 3 周期、第 5 周期 $\cdots$、と奇数時の周期の大きさが $\frac{1}{3}$、$\frac{1}{5}\cdots$ となっていくことを示しています。一方、cos のほうは、0 から π と π から 2π との両方の区間でいかなる周期であっても対称なので方形波の+1 の側と-1 の側とで打ち消しあい、すべての周期の係数で 0 になります。

このことから、この方形波は、奇数次の sin 成分だけで構成されていることがわかります。つまり、方形波のフーリエ級数展開は、

$$f(x) = \frac{4}{\pi}\left(\sin x + \frac{1}{3}\sin 3x + \frac{1}{5}\sin 5x + \cdots\right)$$

となります。このようにして、方形波のフーリエ係数には $\frac{4}{\pi}$ という定数がかかってくることがわかりました。これがさっき説明した規格化の定数です。フーリエ変換とは、まさにこのフーリエ係数を求めることなのです！ただし、先ほど説明したように任意の関数や実際に測定された波形などを解析的な計算によって、いつでも係数を求められるというわけではありません。フーリエ変換を音声波形などに適用するには数値解析をしなければなりません。この詳しい手法は**4 数値解析**で説明します。ここが一番面白いところですね。はい。

3.5 フィルター

いよいよ数値解析の概念に入っていきます。なんか、数学研究部の部誌なのに物理的なことになってますね。父の影響のような気がしてきますw実際に波形データをサンプリングするときに、ナイキスト周波数より高い周波数を、あらかじめ取り除かなければなりません。このことは後々説明しますのでご安心を。この高い周波数を取り除くためのフィルターを「ハイカットフィルター」、低い周波数を取り除くフィルターを「ローカットフィルター」といいます。ナイキスト周波数を境目にしてぴったりと高周波数をカットするような理想的なフィルターを作るのは事実上無理といわれています。

4 数値解析

いままで考えてきたフーリエ変換では解析される関数があらかじめわかっている物ばかりでしたね。しかし一般的な波形、たとえば音などは簡単な解析関数で表現できるようなものではありません。実際にわたしの身の回りにある波形のスペクトル解析をするためには、波形を測定し、計算可能な形の数値列に変換し、パソコンなどを使ってフーリエ解析をします。この様にさまざまな現象を調べることを「**数値解析**」といいます。ここでフーリエ変換を実際に利用するにはいろいろ補足があるので最初にそのあたりをまとめます。

4.1　サンプリング

たとえば、音波を解析するときに数値化して扱わなくてはいけません。この際に時間軸を数値化して、一定の時間間隔で音波の強さに相当する値を「切り取る」ことをします。このことを**サンプリング**といいます。波の大きさの瞬時値も、数値計算に適したように数値化しなければなりません。この、大きさをデジタル計算に適用させるための手法を**デジタイズ**といいます。サンプリングしたデータ列が、いかに元の波形を忠実に表現しているかは数値解析で非常に重要な問題です。

4.2　スペクトルの振幅

実際に観測された波形を、数値解析によるフーリエ変換を利用して周波数分析するとき、求めるものはそれぞれの周波数成分の強さですね。ふつう、スペクトルといえば、「その強さの周波数分布」、つまりスペクトルの振幅をさします。

4.3　DFT によるスペクトル

先ほど 3.2 でフーリエ係数を求めた関数 $F(x)$ は、x 軸に関して区分的に滑らかでした。しかしこれから数値解析しようとする対象は先ほどサンプリングのくだりで述べたようにサンプリングによって時間軸に関して不連続な数値が並んだ形になります。いかにサンプリングの周期を短くしても不連続な状態であることには変わりありません。このように、時間軸に関して不連続に連なる数値を、1 つの関数と見立てて、積分の変わりに総和で数値計算する手法を $DFT\,(DiscreteFourierTransform$ ＝離散フーリエ変換) といいます。いま、サンプリングされた m ポイントからなるデータ列を $F(k)$ で表すことにします。ここで、k はデータ列の k 番目を表すものとします。すると、ＤＦＴの各係数は、

$$a_n = \frac{1}{\pi}\sum_{k=0}^{m-1} F(k)\cos n\Big(\frac{2\pi k}{m}\Big)$$

$$b_n = \frac{1}{\pi}\sum_{k=0}^{m-1} F(k)\sin n\Big(\frac{2\pi k}{m}\Big)$$

という風に表すことができます。なお、スペクトルの振幅の成分はこれまでと同様に、

$$r_n = \sqrt{a_n^2 + b_n^2}$$

です。

4.4 窓関数

ここでは、非周期成分がサンプリングデータの中に含まれている場合に、どのような現象が現れるかを調べます。サンプリングしたデータの中に非整数周期成分がふくまれていると、ゴーストが生じることがしられています。このサンプリングデータを拡張したとき、滑らかにつながるようにするにはサンプリングデータからその両端に向かって緩やかに 0 に収束していけばうまくいきそうです。つまりサンプリングデータの中央付近に対し、両側かに向かって小さくなっていくような関数を用意し、その関数をサンプリングデータを掛け合わせた関数のスペクトルを計算します。この目的のために作られた関数を「**窓関数**」といいます。窓関数の代表的なもの人は次のような種類があります。

ハニング (Hanning) 窓　　$0.5 - 0.5\cos\left(\frac{2\pi n}{N}\right)$

ハミング (Hamming) 窓　　$0.54 - 0.4\cos\left(\frac{2\pi n}{N}\right)$

ブラックマン (Blackman) 窓　　$0.423 - 0.498\cos\left(\frac{2\pi n}{N}\right) + 0.0792\cos\left(\frac{2\pi n}{N}\right)$

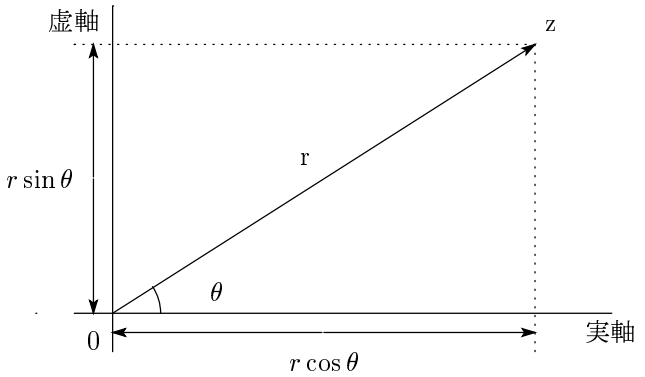
これらの窓関数は、計測方法や観測結果の特性に応じて選ぶことが大事です。できればこのようなものを使わずにデータそのものを解析してみることをお勧めします。

5　高速フーリエ変換 (FFT)

5.1　準備

ここでは高速フーリエ変換を理解するのに必要な知識を整理します。

5.1.1　複素数の極座標表示



ここで、複素数 z で閉めさせる点と原点を結ぶ直線が実軸となす角を θ とします。すると、z の実軸成分は $r\cos\theta$、虚軸部分は $r\sin\theta$ となり、

$z = r\left(\cos\theta + i\sin\theta\right)$

と表せます。この表し方を複素数の**極座標表示**あるいは**極形式**といいます。ここでひとつ面白いお話を。三角関数を級数展開すると、

$$\begin{aligned}\cos\theta &= 1 - \frac{1}{2!}\theta^2 + \frac{1}{4!}\theta^4 - \frac{1}{6!}\theta^6 + \frac{1}{8!}\theta^8 - \cdots \\ &= \sum_{n=0}^{\infty}(-1)^n\frac{\theta^{2n}}{(2n)!}\end{aligned}$$

$$\begin{aligned}\sin\theta &= \theta - \frac{1}{3!}\theta^3 + \frac{1}{5!}\theta^5 - \frac{1}{7!}\theta^7 + \frac{1}{9!}\theta^9 - \cdots \\ &= \sum_{n=0}^{\infty}(-1)^n\frac{\theta^{2n+1}}{(2n+1)!}\end{aligned}$$

となります。

これを極形式の三角関数部分に当てはめて整理すると、

$$\begin{aligned}\cos\theta + i\sin\theta &= \left(1 - \frac{1}{2!}\theta^2 + \frac{1}{4!}\theta^4 - \frac{1}{6!}\theta^6 + \cdots\right) + i\left(\theta - \frac{1}{3!}\theta^3 + \frac{1}{5!}\theta^5 - \frac{1}{7!}\theta^7 + \cdots\right) \\ &= 1 + i\theta - \frac{1}{2!}\theta^2 - \frac{1}{3!}i\theta^3 + \frac{1}{4!}\theta^4 + \frac{1}{5!}i\theta^5 - \frac{1}{6!}\theta^6 - \frac{1}{7!}i\theta^7 + \cdots\end{aligned}$$

となります。
ここで、複素数の冪乗の性質を考えてみると、上の式は

$$1 + i\theta + \frac{1}{2!}(i\theta)^2 + \frac{1}{3!}(i\theta)^3 + \frac{1}{4!}(i\theta)^4 + \frac{1}{5!}(i\theta)^5 + \frac{1}{6!}(i\theta)^6 + \frac{1}{7!}(i\theta)^7 + \cdots$$

$$= \sum_{n=0}^{\infty} \frac{1}{n!}(i\theta)^n$$

となります。
これは指数関数の展開式である、

$$e^x = \sum_{n=0}^{\infty} \frac{1}{n!}x^n = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \frac{1}{4!}x^4 + \cdots$$

と同じ形になっていますね。

5.1.2 オイラーの公式

前項の 2 つの式を見比べて、指数関数の展開式で $x \to i\theta$ と書き換えることで、

$$e^{i\theta} = \cos\theta + i\sin\theta$$

と表現できることがわかります。
さらに θ を $-\theta$ 置き換え、三角関数の偶関数 · 奇関数の性質を利用すると、

$$\begin{aligned} e^{-i\theta} &= \cos(-\theta) + i\sin(-\theta) \\ &= \cos\theta - i\sin\theta \end{aligned}$$

となります。
この二つをまとめて、

$$e^{\pm i\theta} = \cos\theta \pm i\sin\theta$$

という、オイラーの公式が導き出されました。（この方法は厳密な証明ではありません。）

5.1.3 ド・モアブルの定理と 1 の n 乗根

ここで、指数法則 $\left(e^{\pm i\theta}\right)^n = e^{\pm i(n\theta)}$ をオイラーの公式に適用することで、

$$(\cos\theta \pm i\sin\theta)^n = \cos n\theta \pm i\sin n\theta$$

が導き出されます。この公式は**ド・モアブルの定理**とよばれます。これらの式を改めて書き直すと、

$$e^{in\theta} = \cos(n\theta) + i\sin(n\theta)$$
$$e^{-in\theta} = \cos(n\theta) - i\sin(n\theta)$$

となります。この二つの式を加えて 2 で割ることで cos 関数が、引いて $2i$ でわることで sin 関数が以下のようにもとまります。

$$\cos(n\theta) = \frac{e^{in\theta} + e^{-in\theta}}{2}$$
$$\sin(n\theta) = \frac{e^{in\theta} - e^{-in\theta}}{2i}$$

ここで、オイラーの公式に $\theta = \pi$ を代入すると、

$$\begin{aligned} e^{i\pi} &= \cos\pi + i\sin\pi = -1 + 0 \\ &= -1 \end{aligned}$$

が得られます。この式は非常にきれいなので映画の題材にもなったほどです。複素数の極座標表示において $r = 1$ とすると、

$$z = \cos\theta + i\sin\theta$$

となり、z は単位円 (半径=1 の円) の任意の 1 点を表していますね。また、三角関数の 2π ごとの周期に注目してオイラーの公式を当てはめると、k が整数の場合、

$$e^{i2k\pi} = \cos(2k\pi) + i\sin(2k\pi) = 1$$

が成り立つことがわかります。そこで、n を自然数として $x^n = 1$ の根を求めてみましょう。上述の 2 つの関係の、

$$x^n = 1 \quad , e^{i2k\pi} = 1$$

を見比べると、

$$x^n = e^{i2k\pi}$$

とおけることがわかります。ここで指数法則によって両辺の n 乗根を求めると、

$$x = e^{i\frac{2k\pi}{n}}$$

となります。この右辺に再びオイラーの公式を当てはめると、

$$x = \cos\Big(\frac{2k\pi}{n}\Big) + i\sin\Big(\frac{2k\pi}{n}\Big)$$

と表すことができます。このようにして x の n 乗根は複素平面煮の単位円上を n 等分した点に相当しています。

5.1.4 複素フーリエ級数

複素フーリエ級数の概念は、FFT(高速フーリエ変換) の基礎となるものです。まず、フーリエ級数の式、

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty}(a_n\cos nx + b_n\sin nx)$$

にオイラーの公式を代入します。すると右辺は、

$$\begin{aligned} & \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\Big\{a_n\frac{e^{in\theta}+e^{-in\theta}}{2} + b_n\frac{e^{inx}-e^{-inx}}{2i}\Big\} \\ = \quad & \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\Big\{\frac{1}{2}\Big(a_n + \frac{b_n}{i}\Big)e^{inx} + \frac{1}{2}\Big(a_n - \frac{b_n}{i}\Big)e^{-inx}\Big\} \end{aligned}$$

i をかけることは、先ほど示したように $\frac{\pi}{2}$ ずつ逆時計回りに複素平面内を回転することなので、i で割ることは $\frac{\pi}{2}$ ずつ時計回りに複素平面内を回転することに対応します。すなわち、$\frac{1}{i} = -i$ になります。つまり、上の式は

$$= \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\Big\{\frac{1}{2}(a_n - ib_n)e^{inx} + \frac{1}{2}(a_n + ib_n)e^{-inx}\Big\}$$

となります。ここで新たに複素フーリエ係数を、

$$c_0 = \frac{a_0}{2}, c_n = \frac{a_n - ib_n}{2}, c_{-n} = \frac{a_n + ib_n}{2}(n = 1, 2, 3, \cdots)$$

と定義すると、フーリエ級数は次のようになります。

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}$$

また、さっき取り上げたフーリエ係数

$$b_n = \frac{1}{\pi}\int_0^{2\pi} F(x)\sin nxdxa_n = \frac{1}{\pi}\int_0^{2\pi} F(x)\cos nxdx$$

を定義式に代入すると、

$$
\begin{aligned}
c_n &= \frac{1}{2}(a_n - ib_n) \\
&= \frac{1}{2\pi}\int_0^{2\pi} F(x)(\cos nx - i\sin nx)dx \\
&= \frac{1}{2\pi}\int_0^{2\pi} F(x)e^{-inx}dx
\end{aligned}
$$

とかけます。これを、**複素フーリエ係数**といいます。いま、c_{-n} を考えると、

$$
c_{-n} = \frac{1}{2\pi}\int_0^{2\pi} F(x)e^{+inx}dx
$$

とかけます。さらに ${c_n}^*$ を c_n の複素共役とすると、

$$
\begin{aligned}
{c_n}^* &= \Big(\frac{1}{2\pi}\int_0^{2\pi} F(x)e^{-inx}dx\Big)^* \\
&= \frac{1}{2\pi}\int_0^{2\pi} F^*(x)e^{+inx}dx \\
&= \frac{1}{2\pi}\int_0^{2\pi} F(x)e^{inx}dx
\end{aligned}
$$

となります。ここで、$F(x)$ が実関数であることから $F(x) = F^*(x)$ という関係を使っています。すなわち

$$
c_{-n} = {c_n}^*
$$

となることがわかりました。これから、c_{-n} を計算しなくても c_n を計算し、その複素共役をつくればいいということがわかります。

5.2 FFT とは？

FFT とは DFT の計算回数、特に掛け算の回数を劇的に減少させるために考案された演算手法です。コンピューターが現在のように高速ではなかった時代には、特に掛け算の演算処理には時間がかかり、掛け算回数を減らすことは重要な意味がありました。FFT アルゴリズムは 1995 年に、クーレイとターキーによって学会誌に発表されました。この論文はわずか 5 ページほどの内容でしたがその発想はさまざまな分野で爆発的に利用されるようになりました。

5.2.1 FFT アルゴリズムの概要

c_n を DFT（離散フーリエ変換）の形に書き直します。ここで、m をサンプリングされたポイント数とすると、

$$\begin{aligned} c_n &= \frac{1}{2\pi}\int_0^{2\pi} F(x)e^{-inx}dx \\ &\Rightarrow \frac{1}{m}\sum_{k=0}^{m-1} F(k)e^{-in\frac{2\pi}{m}k} \end{aligned}$$

と書き直せます。
式だけ眺めていては、こう、実感がわいてこないので実際に簡単な例で試してみましょう。まず、データのポイント数を 8 にします。なぜなら、FFT アルゴリズムではポイント数が 2^n(n は正の整数) になっていることが重要なのです。そこで、DFT の c_n の計算式を用い、c_0〜c_7 を求めてみます。ナイキスト周波数の概念からは、8 ポイントのデータに対して意味があるデータは半分の第 4 周期までですが、ここでは、FFT の計算と関連ずけさせるために第 8 周期までの計算結果を島します。ここで $\omega = e^{-i\frac{2\pi}{m}}$ とおき、ポイント数による規格化前のフーリエ係数に注目します。すると、mc_n は次のようになります。

$$\begin{aligned}
c_0 &= \omega^0F(0)+\omega^0F(1)+\omega^0F(2)+\omega^0F(3)+\omega^0F(4)+\omega^0F(5)+\omega^0F(6)+\omega^0F(7) \\
c_1 &= \omega^0F(0)+\omega^1F(1)+\omega^2F(2)+\omega^3F(3)+\omega^4F(4)+\omega^5F(5)+\omega^6F(6)+\omega^7F(7) \\
c_2 &= \omega^0F(0)+\omega^2F(1)+\omega^4F(2)+\omega^6F(3)+\omega^8F(4)+\omega^{10}F(5)+\omega^{12}F(6)+\omega^{14}F(7) \\
c_3 &= \omega^0F(0)+\omega^3F(1)+\omega^6F(2)+\omega^9F(3)+\omega^{12}F(4)+\omega^{15}F(5)+\omega^{18}F(6)+\omega^{21}F(7) \\
c_4 &= \omega^0F(0)+\omega^4F(1)+\omega^8F(2)+\omega^{12}F(3)+\omega^{16}F(4)+\omega^{20}F(5)+\omega^{24}F(6)+\omega^{28}F(7) \\
c_5 &= \omega^0F(0)+\omega^5F(1)+\omega^{10}F(2)+\omega^{15}F(3)+\omega^{20}F(4)+\omega^{25}F(5)+\omega^{30}F(6)+\omega^{35}F(7) \\
c_6 &= \omega^0F(0)+\omega^6F(1)+\omega^{12}F(2)+\omega^{18}F(3)+\omega^{24}F(4)+\omega^{30}F(5)+\omega^{36}F(6)+\omega^{42}F(7) \\
c_7 &= \omega^0F(0)+\omega^7F(1)+\omega^{14}F(2)+\omega^{21}F(3)+\omega^{28}F(4)+\omega^{35}F(5)+\omega^{42}F(6)+\omega^{49}F(7)
\end{aligned}$$

この中の冪乗の部分だけをみてやると「九九の表」になっています。ここで、ω の冪乗は単位円周上にある 8 個の点で表される周期関数で、

$$\omega^{8p} = 1 \quad (p = 1, 2, \cdots)$$

であることを考えると先ほどの冪の部分だけを取り出すと次のように書けます。

c_k \ $F(k)$	$F(0)$	$F(1)$	$F(2)$	$F(3)$	$F(4)$	$F(5)$	$F(6)$	$F(7)$
c_0	ω^0	ω^0	ω^0	ω^0	ω^0	ω^0	ω^0	ω^0
c_1	ω^0	ω^1	ω^2	ω^3	ω^4	ω^5	ω^6	ω^7
c_2	ω^0	ω^2	ω^4	ω^6	ω^0	ω^2	ω^4	ω^6
c_3	ω^0	ω^3	ω^6	ω^1	ω^4	ω^7	ω^2	ω^5
c_4	ω^0	ω^4	ω^0	ω^4	ω^0	ω^4	ω^0	ω^4
c_5	ω^0	ω^5	ω^2	ω^7	ω^4	ω^1	ω^6	ω^3
c_6	ω^0	ω^6	ω^4	ω^2	ω^0	ω^6	ω^4	ω^2
c_7	ω^0	ω^7	ω^6	ω^5	ω^4	ω^3	ω^2	ω^1

ここで、n と k を 2 進数で表して (p,q,r,s,t,u はそれぞれ 1 または 0 の値をとるものとする)、

$$n = 4p + 2q + r \quad , k = 4s + 2t + u$$

と置くとき、先ほどと同じ定義で ω を扱うと、

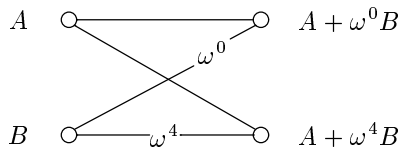
$$\begin{aligned} c_n \equiv c_{pqr} &= \sum_s \sum_t \sum_u \omega^{(4p+2q+r)(4s+2t+u)} F(stu) \\ &= \sum_s \sum_t \sum_u \omega^{\{(4p+2q+r)4s+(4p+2q+r)2t+(4p+2q+r)u\}} F(stu) \\ &= \sum_s \sum_t \sum_u \omega^{(16ps+8qs+4rs+8pt+4qt+2rt+4pu+2qu+rn)} F(stu) \end{aligned}$$

ここでも、$\omega^{8h} = 1 (h = 0, 1, 2, \cdots)$ という関係を使うと、

$$\begin{aligned} c_{pqr} &= \sum_s \sum_t \sum_u \omega^{(4rs+4qt+2rt+4pu+2qu+ru)} F(stu) \\ &= \sum_s \sum_t \sum_u \omega^{\{4rs+(4q+2r)t+(4p+2q+r)u\}} F(stu) \\ &= \sum_u \omega^{(4p+2q+r)u} \left\{ \sum_t \omega^{(4q+2r)t} \left(\sum_s \omega^{4rs} F(stu) \right) \right\} \end{aligned}$$

と書き直せます。この式が FFT アルゴリズムの根本になっている式です。この式の最も重要な点は、変換の対象となるデータ ($F(stu)$) に ω^{4rs} をかけた後は、その計算された値に次々と ω の冪乗を掛け合わせていくところです。すなわち。元になるデータはポイント数分だけ元に掛け合わされてしまうと、そのあとはデータそのものを使う必要がなくなる点です。では、8 ポイントデータで、実際の演算手順を示しましょう。まず、計算表の左端に上から下に向かって $F(0)$〜$F(7)$ を並べます。このデータ ($F(0)$〜$F(7)$ を元に下の図

でしめすような ω の冪乗を使った積と和を計算していきます。この三角形を 2 つ組み合わせた図式が、蝶の形を連想させるところから**バタフライ演算**というニックネームがついています。ちなみに筆者は水泳のバタフライはできません。



このようなことをして、FFT アルゴリズムで計算された係数では、最終結果の序列をビットリバースによって位置を入れ替えることで、周波数スペクトルの順序にすることができます。実際にビットリバースをする前の係数の番号の順序がビットリバースによって入れ替わる理由は、FFT の計算に際して ω の冪乗をかける順序が 2^n ごとに計算され、その順序が ω^n の回転数の大きいほうから計算されるためです。ここまでで、FFT アルゴリズムに関するイメージがつかめましたか？なかなか数学的なのかどうか疑問が残りますけど、面白いものにはなっていると思います。最後に DFT と FFT の結果の比較を見ておきます。

5.3 FFT と DFT の結果の比較

ここで、sin や cos の係数を計算するときの掛け算の回数を実際に確かめてみましょう。0 周期に対する sin の係数は常に 0 であることは明白ですが、その計算回数も含むことにすると、DFT の場合、

$$(\text{sin 係数を求める掛け算回数}) = \frac{(\text{ポイント数}) \times (\text{ポイント数})}{2}$$
$$(\text{cos 係数を求める掛け算回数}) = \frac{(\text{ポイント数}) \times (\text{ポイント数})}{2}$$

全体では（但し、スペクトルの振幅計算のための二乗や平方根の計算を考えずに）、

$$(DF \text{ の掛け算回数}) = (\text{ポイント数}) \times (\text{ポイント数}) = m^2$$

となります (但し、ポイント数を m とします)。一方 FFT の場合は、縦方向のポイント数はそのままですが、横方向の掛け算回数はポイント数の 2 を底とした対数だけですみます。すなわち 16 ポイントの時には 4 回、1024 ポイ

ントの時には 10 回という風にです。一般的にあらわすと、

$(FFT$ の掛け算回数$) =$ (ポイント数$) \times \log_2$(ポイント数$) = m \times \log_2 m$

となります。m=16 の場合ですから DFT の場合には 256 回、FFT では 64 回と FFT にした場合には掛け算回数は 1/4 になります。さらに、1024 ポイントになると DFT の場合には約 100 万なのに対して、FFT では約 1 万回で終わります。すなわち 1024 ポイントの FFT では掛け算回数がおよそ 1/100 になります。驚異的ですね。

6　あとがき

いかがだったでしょうか。あまり一貫性もない議論の繰り返しで論文としては悪い形になってしまったことは悔やまれます。ですが、これで数値解析がどのように行われているのかが数学的に理解されたことだろうと思います。実際に音などを市販のソフトウェアで採取して解析してみるのはどうでしょうか。結構面白い結果が得られると思いますよ。さて、今年の部誌はこれぐらいにさせていただきます。この記事を書き上げるのに苦労しました。初めて TEX を使う上に記事が長いし図もあるし。いままでの文化祭準備で一番大変だったことだろうと思います。また、この記事をかくのに使用させていただいた参考文献を羅列しておきます。

「なるほどフーリエ解析」(村上雅人 著　海鳴出版)
「LATEX 2ε 文典」(生田誠三 著　朝倉書店)

最後になりましたが、TEX を使う上においていろいろご教授いただいた河口氏には感謝です。もし、おかしな点やご質問ご感想等がございましたら
hiranomasahiro@msn.com
までメールをお願いします。それでは、また来年文化祭でお会いいたしましょう。

と、いってみたものの、このままでは奇数ページで終わってしまうので無理やり偶数ページにするためにネタをひとつ。先ほど $\log_2$ とかで回数が減っていくとかいう話がありましたが、プログラミングの世界にもソートと呼ばれる、ある数値列を大きさの順に並べ替えるものがあります。それのクイックソートというやつは個数を N とすると $N \log_2 N$ 回の計算で大小に並び替えてしまうという代物です。実際にコードを書いてみましょう。ここは数学からは逸脱した内容なので興味のない人は飛ばしてくださってもかまいません。

```
#include <stdio.h>
#define N 8

void quick_sort(int d[],int top,int end)
{
int key,wk,i,j;

key = d[(top+end)/2];
i = top-1;j=end+1;
while(true){
while(d[++i] < key);
while(d[--j] > key);
if(i >= j) break;
wk = d[i];d[i]=d[j];d[j]=wk;
}
if (top < i-1) quick_sort(d,top,i-1);
if (j+1 <end) quick_sort(d,j+1,end);
}

int main(void)
{
int i;
int d[N] = {60,20,10,50,30,40,80,70};

quick_sort(d,0,N-1);
for(i=0;i<N;i++){printf("%d",d[i];}printf("\n");

return 0;
}
```