

DFTを用いたプロセスデータ解析に関する考察—続報—

近藤 勝也

A study for analyzing process data -the second-

Katsuya KONDO

It is important in analyzing by DFT (Discrete Fourier Transform) to compare the frequency spectrum with an amplitude value of continuous wave before DFT. From this point of view, the equations of Fourier series and numerical calculating method of DFT in mathematical program are rearranged. Time windows are well used for improving fundamental problem on DFT that data cutting-off period affects the frequency spectrum. Two characteristic window functions, Blackman-Harris window and flat-top window, are examined by concrete numerical calculation. As a result, it is known that any time window is not perfect to solve the problem on DFT.

1. はじめに

ここでは、産業界の製造プロセスや研究室の実験プロセスで計測されたデータを、計算機を用いて解析する場合の注意事項について検討している。

これらのプロセスデータのほとんどはデジタル信号である。すなわち、時間的に連続なものの波形が、サンプリングによって離散数値系列のデータに変換された後に解析される。データ解析では周波数解析すなわちフーリエ変換とその逆変換が一番よく使われる^{1), 2)}が、実際にデジタル信号を計算するのは離散フーリエ変換(DFT: Discrete Fourier Transform)である。周波数解析の目的はもとの波形に含まれる三角関数の3つのパラメータ(周波数, 振幅および位相)をはっきりさせることであるが、周波数と振幅が特に大切である。

ところで、DFTを用いるにあたって、周波数の解析精度には問題が少ない。すなわち、もとの連続波形のサンプリング周波数を高くしたり、DFTを行うデータの切り取り区間を長くすることにより、メモリー容量や計算時間を無視すると、周波数の解析精度を上げることができる。一方、振幅の解析精度は本質的な課題が残されている。すなわち、DFTが有限な切り取り区間のデータを取り扱うことから、区間の長さに関係なく、もとの波形の振幅とDFT結果の振幅すなわち周波数スペクトルが

必ずしも一致しないという問題がある。

前報³⁾ではこの問題を改善する方法としてよく知られている窓関数の効果について、具体的な数値計算を行って検討した。すなわち、窓関数なしと等価である方形窓を用いた場合と、ハニング窓、ハミング窓、三角窓の3種類の窓関数を用いた場合とを、比較検討した。その結果これら3つの窓関数の間には大きな差が認められなかった。そこで、本報ではさらに特性が異なると思われるブラックマン-ハリス窓と折り返し窓の2つについて前報と同様の方法で検討を加えている。

なお、窓関数の使い方に先立つ問題として、DFTの計算式についてはその元になっているフーリエ級数展開の計算式を、もとの波形に含まれる三角関数の振幅の解析という観点に立って整理しておくことが必要である。従来はフーリエ変換とその逆変換の2つからなるフーリエ変換対として整理されている文献²⁾がほとんどであり、もとの三角関数の振幅とDFT後の周波数スペクトルの対応という観点で整理した文献は少ないと思われる。さらに、市販の数式処理プログラムに付属しているDFTの計算式についても、この関係をつきりさせておくことが必要である。

以上のことから本報では、まず第2章において、フーリエ級数展開やDFTなどで使用される周波数スペクトルの計算式をまとめている。次に第3章において、前報の窓関数に比べて特長のある前記2つの窓関数を、具体

的的な数値計算によって検討する。

また、本報の計算で使用した数式処理プログラムは、Mathsoft社のMathcadであり、計算結果の作図も同じプログラムを用いた。

2. 三角関数の振幅と周波数スペクトル

DFTは通常複素数を用いて計算される^{2), 4)}が、基本はフーリエ級数展開である。従ってこの章では、フーリエ級数、複素フーリエ級数、複素フーリエ級数の離散形すなわちDFTの順に説明し、最後にもとの三角関数の振幅とDFTの周波数スペクトルの関係を整理する。なお、FFT (Fast Fourier Transform) はDFTを高速で計算する方法と考えればよい。

2.1 フーリエ級数

周期T[s]を持つ時間t[s]の関数x(t)は次のフーリエ級数で展開される。

$$x(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos(k2\pi t/T) + \sum_{k=1}^{\infty} b_k \sin(k2\pi t/T) \quad (1)$$

(1)式において $\omega_k = k2\pi/T$ [rad/s]とおくと次のようになる。

$$x(t) = a_0 + \sum_{k=1}^{\infty} (a_k \cos\omega_k t + b_k \sin\omega_k t) \quad (2)$$

周波数を解析するという事は、もとの波形x(t)に含まれる a_k, b_k (角周波数 ω_k と、そのcosとsinの振幅)を求めることである。一方、(2)式の三角関数を展開して次のように書くと、周波数解析は角周波数 ω_k とそれに対する振幅 c_k と位相 θ_k を求めることになる。

$$x(t) = a_0 + \sum_{k=1}^{\infty} c_k \sin(\omega_k t + \theta_k) \quad (3)$$

ただし、 $c_k = \sqrt{a_k^2 + b_k^2}$, $\tan \theta_k = b_k / a_k$

次に(1)式の係数は、よく知られているように、次式で表される。

$$\begin{aligned} a_0 &= \frac{1}{T} \int_0^T x(t) dt \\ a_k &= \frac{2}{T} \int_0^T x(t) \cos\omega_k t dt \\ b_k &= \frac{2}{T} \int_0^T x(t) \sin\omega_k t dt \end{aligned} \quad (4)$$

以上の関係式の数学的な呼び方²⁾をまとめると、(1)、(2)式はx(t)のフーリエ級数といい、 a_k と b_k をx(t)のフーリエ係数という。次に(3)式のように表現した c_k と θ_k は周波数スペクトルと呼ばれ、前者を振幅スペクトル後者を位相スペクトルという。また一般に、周波数解析で求める三角関数の振幅は c_k であり、これを周波数スペクトルということが多い。

2.2 複素フーリエ級数

複素数を用いるとフーリエ級数の展開式が簡潔に表現できる。すなわち、次式で定義される複素数 D_k (D_k と D_{-k} は共役)を用いる。

$$D_k = \frac{1}{2} (a_k - j b_k), D_{-k} = \frac{1}{2} (a_k + j b_k) \quad (5)$$

$$D_0 = a_0, \quad k = 1 \sim \infty$$

さらに、オイラーの公式から導かれる次の式を用いる。

$$\cos\omega_k t = \frac{1}{2} \{ \exp(j\omega_k t) + \exp(-j\omega_k t) \} \quad (6)$$

$$\sin\omega_k t = \frac{1}{2j} \{ \exp(j\omega_k t) - \exp(-j\omega_k t) \}$$

(5)、(6)式を用いて整理すると、(2)式と(4)式はそれぞれ次の(7)式と(8)式で表される。

$$x(t) = \sum_{k=-\infty}^{\infty} D_k \exp(j\omega_k t) \quad (7)$$

$$D_k = \frac{1}{T} \int_0^T x(t) \exp(-j\omega_k t) dt \quad (8)$$

ただし、 $D_0 = a_0 = \frac{1}{T} \int_0^T x(t) dt$ ((8)式で $k=0$)

(7)式と(8)式は複素フーリエ級数の計算において対(ペア)となる式である。

次に(5)式から、

$$D_k D_{-k} = |D_k|^2 = \frac{1}{4} (a_k^2 + b_k^2)$$

となるので、(3)式のただし書きと合わせると、もとの波形x(t)に含まれる三角関数の振幅 c_k と複素フーリエ係数 D_k の関係は次のようになる。

$$c_k = \sqrt{a_k^2 + b_k^2} = 2|D_k| \quad (9)$$

すなわち、もとの波形に含まれる三角関数の振幅は D_k の絶対値の2倍である。また同様にして、(4)式の a_k

と b_k は、それぞれ複素数 D_k の実数部の2倍と虚数部の2倍である。すなわち、(8)式は(4)式の a_k と b_k を同時に計算している。

2.3 DFTの計算式

前述の(8)式がDFT、(7)式が逆DFTのもとになる式である。以下に離散データの計算式を説明する。

連続的な時間波形 $x(t)$ を τ [s]ごとに1周期 T [s]の間サンプリングした離散データを x_n とする。すると、サンプリング番号 $n=0\sim N-1$ 、全サンプリング数 $N=T/\tau$ 、時間 $t=n\tau$ となる。この離散データを(8)式に当てはめ、さらに $\omega_k=k2\pi/T$ を用いて積分を積和になおすと、次のようになる。

$$\begin{aligned} D_k &= \frac{1}{T} \int_0^T x(n\tau) \exp(-j\omega_k n\tau) dt \\ &= \frac{\tau}{T} \sum_{n=0}^{N-1} x_n \exp(-j2\pi kn\tau/T) \\ &= \frac{\tau}{T} \sum_{n=0}^{N-1} x_n \exp(-j2\pi kn/N) \end{aligned} \quad (10)$$

(10)式がDFTの基本式である。添字 k は $f_0=1/T$ [Hz]を基本周波数とした k 倍の高調波を意味する。 $k=0\sim N-1$ について D_k が計算されるが、その値はDFTの性質として知られているように、 $k=N/2$ に関して対象になる。

次に、(7)式に基づいて逆DFTの式を導くと次のようになる。離散データの条件は先と同じであり、 n の範囲は $0\sim N-1$ である。

$$\begin{aligned} x_n = x(n\tau) &= \sum_{k=0}^{N-1} D_k \exp(j\omega_k n\tau) \\ &= \sum_{k=0}^{N-1} D_k \exp(j2\pi kn/N) \end{aligned} \quad (11)$$

(10)式と(11)式がDFTの変換対である。 x_n をDFTして逆DFT(IDFTとも書く)した時にもとの波形に戻ればよいと考え、(10)式の右辺の係数である $1/N$ を(11)式に移しても同じ結果が得られる。したがって、 $1/N$ の位置をどちらにするかは参考書によっていろいろである。また、 $1/N$ を $1/\sqrt{N} \times 1/\sqrt{N}$ に分けて(10)、(11)式ともに、右辺に同じ係数 $1/\sqrt{N}$ をかけるという書き方もある。

数値計算ソフトウェアに付属しているDFT計算プログラムも、この $1/N$ の付け方がいろいろであり、プログラムの説明書をよく読んで使用することが必要である。ちなみに、本報の計算で使用したMathSoft社のMathcadは両式に $1/\sqrt{N}$ をかけるという式を用いている。したが

って、MathcadではDFTの計算結果(D_k の絶対値)に $2/\sqrt{N}$ をかけると(3)式の c_k になる。

なお、もとの波形の振幅とDFTの周波数スペクトルが、離散データの切り取り区間によっては必ずしも一致しない理由は次のように説明される。すなわち、(10)式の x_n が周期 T で無限に繰り返されると仮定している(そうしないと計算できない)ために、切り取り区間の始めと終わりの値が連続しない場合に前記2つの値が一致しなくなる。前報に詳しい計算結果が説明されている。

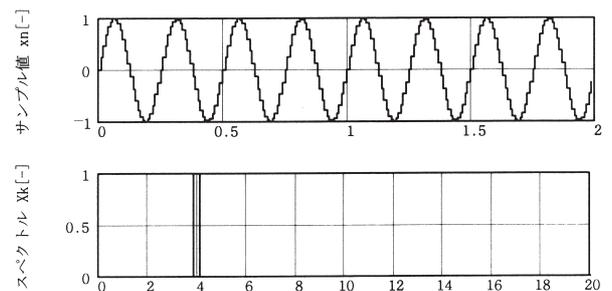
3. DFTの周波数スペクトルと時間窓

3.1 計算条件と前報のまとめ

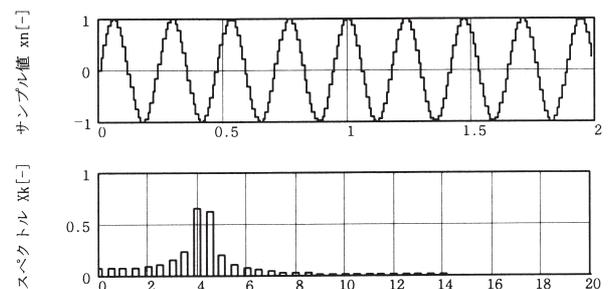
連続的な時間波形 $x(t)$ を τ [s]ごとにサンプリングして離散化した数値系列 x_n から、 T [s](前報では T の代わりに記号 L を用いた)間のデータを切り取ってDFTし、DFT結果の振幅スペクトル((9)式の c_k)を X_k とする。以下 X_k を単にスペクトルと呼ぶ。 k は2.3と同様である。以下の計算では、 $\tau=0.01$ s、 $T=2$ sとした。したがって、 $f_0=0.5$ Hz、 $N=200$ である。

ここで前報の計算結果を整理しておく。一番簡単な三角関数である次式において、 $f=4.0$ Hzと 4.25 Hzの場合について、時間波形 x_n とそのスペクトル X_k を図1に示す。

$$x(t) = \sin(2\pi ft) \quad (12)$$



(1) $f=4.0$ [Hz]



(2) $f=4.25$ [Hz]

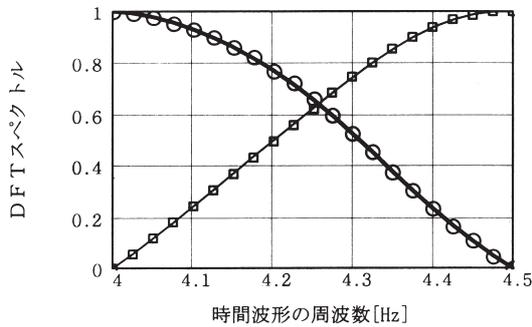
図1 データ切り取り区間長DFTへの影響

(横軸は各図上段；時間 $n\tau$ [s]，下段；周波数 kf_0 [Hz])

図1(1)は $f=4.0\text{Hz}$ で、スペクトルは該当する単一の周波数のみに現れ、その大きさは(12)式の振幅と同じ1.0である。しかし図1(2)の $f=4.25\text{Hz}$ では、スペクトルが複数の周波数に分散し、かつスペクトルの最大値が0.62になっている。この原因は前述したが、図からも分かるように、切り取り区間における始めの値と終わりの値が連続していないためである。

また別の見方をすると、スペクトルは $f_0=0.5\text{Hz}$ の整数倍周波数(これをスペクトル周波数と呼ぶ)のところに現れるので、 4.25Hz では前後の周波数スペクトルにスペクトルが分散して現れるともいえる。図1(2)においてスペクトルの最大値がもとの三角関数の振幅より小さくなるのが問題なので、(12)式の周波数を 4.0Hz から 4.5Hz まで 0.025Hz 刻みに変化させてDFTしたときのスペクトルを読み取った結果を図2に示す。図の縦軸はスペクトル周波数 4.0Hz と 4.5Hz のスペクトル、横軸はもとの三角関数の周波数である。スペクトルはもとの周波数 4.25Hz で最小になるが、それ以外でも無視できないほど小さくなる。

以上の問題を改善する方法として時間窓という重み関数が提案²⁾されている。これは切り取り区間の始めと終



スペクトル周波数：○-4.0[Hz], □-4.5[Hz]

図2 時間波形の周波数とDFTスペクトル

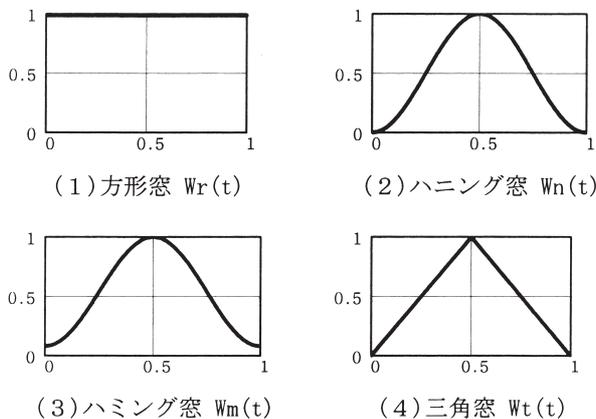


図3 窓関数の波形 (横軸は $T=1$ で表した時間)

わりのサンプリング値を徐々に小さくすることによって、周期的繰り返し時の不連続を少なくしようという考え方である。前報で検討した代表的な窓関数4種類の式を次に示す。

(1) 方形窓: $W_r(t) = 1$ (13)

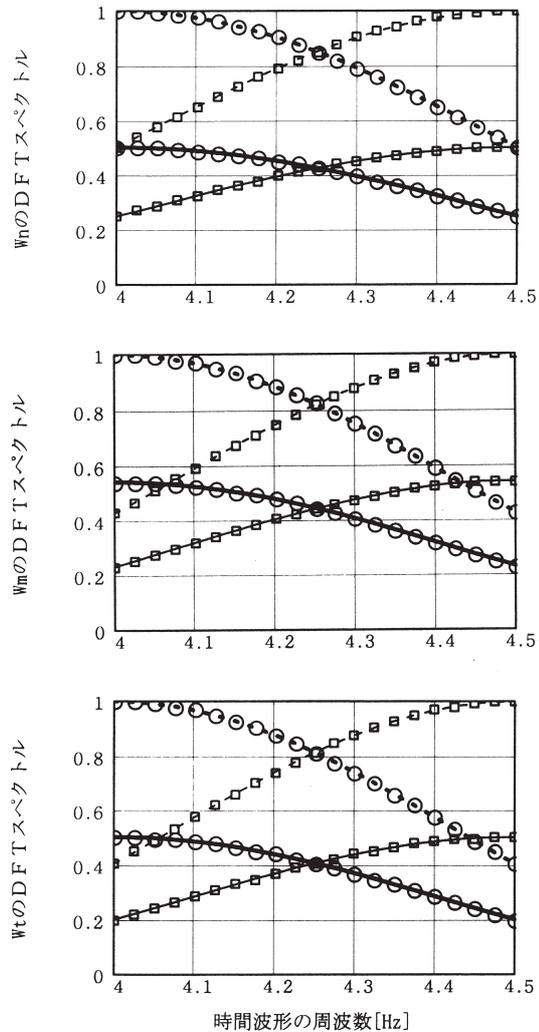
(2) ハニング窓: $W_n(t) = 0.5 - 0.5\cos(2\pi t/T)$ (14)

(3) ハミング窓: $W_m(t) = 0.54 - 0.46\cos(2\pi t/T)$ (15)

(4) 三角窓: $W_t(t) = 1 - (2/T)|t - (T/2)|$ (16)

すべて $0 \leq t \leq T$ で定義され、もとの時間波形 $x(t)$ に窓関数をかけてからDFTする。参考のため窓関数のグラフを図3に示す。方形窓は時間窓をかけないことと同じであり、その結果が図2である。

(14)~(16)式の窓関数を用いて図2と同様の計算をした結果を図4に示す。窓関数により、分散したスペクトル



スペクトル周波数：○-4.0[Hz], □-4.5[Hz]

各図の下段は計算値のまま、上段は正規化後

図4 時間窓を用いたDFTスペクトル

の最大値が小さくなることに対しては効果があるが、元々単一スペクトルが得られるような波形に対しては逆にスペクトルが分散し、すべてが解決されるわけではない。また窓関数の間に大きな差はなかった。

3.2 窓関数の追加

前報で検討した(14)~(16)式の窓関数の効果は、時間波形の周波数が4.25Hzの時のスペクトルの大きさが正規化後で0.81~0.85であり、各窓関数の間に大きな差はなかった。もとの波形の振幅を精度良く分析するためには、正規化後のスペクトルが1.0に近いほど良いという立場から、今までとは特性が異なる2つの窓関数²⁾について検討した。いずれも図3に示した窓関数の波形がより急峻な山形になる時間窓である。なお、窓の面積は小さくなっている。これにより、図4のグラフがより平坦な曲線になることが狙いである。

2つの窓関数はブラックマン-ハリス (Blackman-Harris) 窓と折り返し窓であり、それぞれ(17)式と(18)式に示す。折り返し窓はフラットトップ (flat-top) 窓ともいわれる。

$$Wb(t) = 0.423 - 0.498\cos(2\pi t/T) + 0.0792\cos(4\pi t/T) \quad (17)$$

$$Wf(t) = [0.54 - 0.46\cos(2\pi t/T)] \frac{\sin[(4\pi t/T) - 2\pi]}{(4\pi t/T) - 2\pi} \quad (18)$$

いずれも先の窓関数と同様に $0 \leq t \leq T$ で定義され、使い方も同様である。

2つの窓関数のグラフを図5に示す。図3の窓関数と比べると、山の高さがより急峻になり、特に折り返し窓の両側の関数値が負になっているのが特徴的である。

次(12)式の $f = 4.0\text{Hz}$ と 4.25Hz の時間波形に窓関数をかけた波形とそのDFT結果を図6と図7に示す。ブラックマン-ハリス窓の図6を見ると、窓関数をかけることにより時間波形が大きく変化し、その分スペクトルの大きさが小さくなっている。窓関数をかける前には振幅が1.0であったにもかかわらず、(1) $f = 4.0\text{Hz}$ の場合でもスペクトルは0.423になる。またその分、窓関数をかける前の単一スペクトルが、窓関数により大きく分散している。しかし、(2) $f = 4.25\text{Hz}$ の場合のスペクトルの大きさは(1)に比べそんなに小さくなっていない。折り返し窓の図7についても同様のことがいえるが、図6の傾向がより強くなっている。すなわち窓関数により波形はさらに大きく変化し、スペクトルがより小さくなっている。

時間波形の周波数とDFTスペクトルとの関係を、図2および図4と同じ方法で計算して図8に示す。上図が

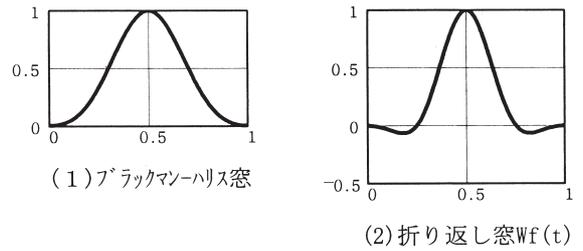
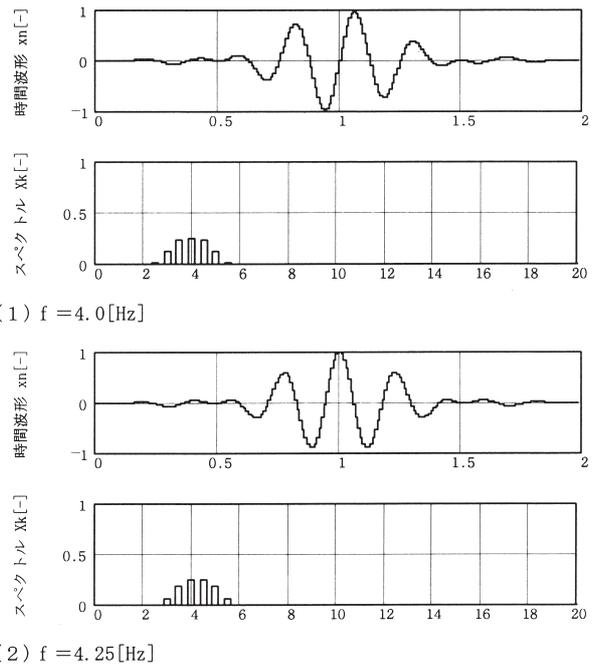
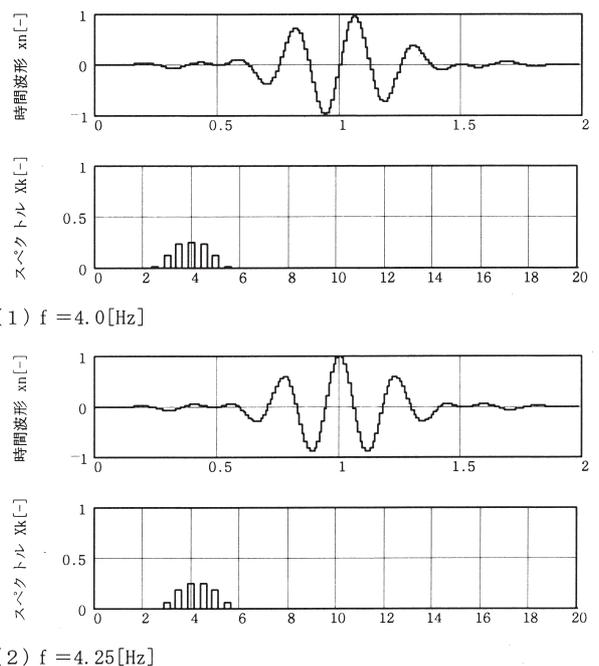


図5 窓関数の波形 (横軸は $T = 1$ で表した時間)



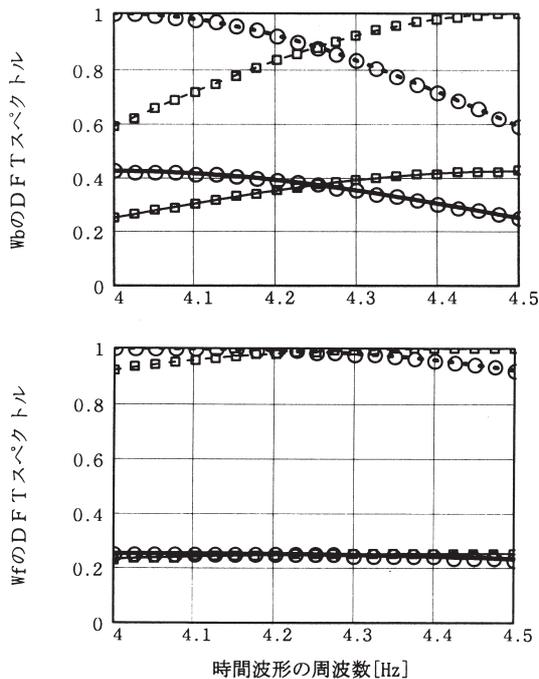
横軸；上段-時間[s], 下段-スペクトル周波数[Hz]

図6 ブラックマン-ハリス窓による波形とDFT



横軸は図6と同じ

図7 折り返し窓による波形とDFT



スペクトル周波数：○-4.0[Hz], □-4.5[Hz]
各図の下段は計算値のまま，上段は正規化後

図8 時間窓を用いたDFTスペクトル

ブラックマン-ハリス窓，下図が折り返し窓である．各図の下段は図6および図7のようなスペクトルをそのままプロットしたグラフである．下段のグラフでは時間波形の周波数4.0Hzのスペクトルが0.5以下になっているが，窓関数のない時にはスペクトル周波数4.0Hzでスペクトルが1.0になるので，窓関数をかけた時にも1.0になるように正規化してプロットしたのが上段のグラフである．

時間波形の周波数4.25Hzの時のスペクトルに注目して，正規化後のグラフを見る．ブラックマン-ハリス窓ではスペクトル値が0.88と図4の場合よりさらに1.0に近くなり，折り返し窓では0.99になって，ほぼ平坦なグラフになっている．これはスペクトルをもとの波形の振幅に一致させるという点では理想に近い時間窓であることを意味している．しかし一方では図7で説明したように，窓関数により時間波形が極端に小さくなったり，かえってスペクトルの分散が大きくなるという欠点が目立つようになる．したがって，切り取り区間長に起因したDFTの問題を，時間窓をかけることで完全に解決することは難しいといえる．言いかえると，時間窓を有効に使うにはその長所と短所を十分に理解することが大切といえる．

4. おわりに

計測された時間波形をDFTで分析する場合には，もとの波形の振幅とDFTの周波数スペクトルの対応が大切であるという立場から，DFTに関わる問題を整理した．

第2章ではこの観点でDFTの計算式を整理し，数式処理プログラムなどで使われるDFTパッケージによる計算結果の取扱方法を分かりやすくした．

第3章では切り取り区間長によっては時間波形の振幅とDFTのスペクトルが必ずしも一致しないというDFTの本質的な問題について，これを改善するための時間窓について検討した．前報の時間窓と特性が異なると思われるブラックマン-ハリス窓と折り返し窓について，具体的な計算によって時間窓の効果を検討した．その結果折り返し窓が特徴的で，DFTのスペクトルを時間波形の振幅と一致させるという点ではほぼ理想的な時間窓であることが分かった．しかし，折り返し窓をかけることにより時間波形が小さくなったり，スペクトルの分散が大きくなるという短所が無視できないことも分かった．結論として次のことがいえる．すなわち，DFTの問題を時間窓で完全に解決することは不可能なので，いろいろの時間窓の長所と短所をよく理解して使うことが大切である．

最後に，本研究の一部を本校情報工学科の平成11年度卒業研究として取り組んだ，井立充洋君に謝意を表します．

参考文献

- 1) 南茂夫：科学計測のための波形データ処理，CQ出版(1986-4)
- 2) 城戸健一：デジタル信号処理入門，丸善(1985-7)
- 3) 近藤勝也：“DFTを用いたプロセスデータ解析に関する考察”，奈良高専研究紀要，第35号，pp.81～86(1999)
- 4) 安居院猛，中嶋正之：FFTの使い方，産報出版(1981-3)