THE HONG KONG POLYTECHNIC UNIVERSITY
Department of Electronic and Information Engineering
EIE413

DIGITAL SIGNAL PROCESSING

TOPIC:

18. Digital Transform Fourier Analysis

STUDENT NAME: *WONG TANG PAAI*
STUDENT NO: *03906165D*
COURSE NO.: *42070- YR 2*

Content

---

1. TOPICS

18. *Digital Transform Fourier Analysis*
    Write a simple program, for which a superposition of several simple harmonic functions can be inputted as an input signal. And plot the corresponding Fourier transform spectrums. Choose and print some typical input signals and the corresponding output spectrums. Finally, draw some concluding remarks of the analysis.

2. INTRODUCTION

Truly Periodic signals are rarely encountered in practical DSP. Aperiodic signals and data with a finite number of nonzero sample values are far more common. For example, the dollar price of gold or the midday temperature record. The Discrete Fourier Transform (DFT) of such a signal x[n], defined over the range 0≦n ≦N-1, is given by:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2pkn/N}$$, where X[K] is spectral coefficients

The inverse DFT, or IDFT, which allows us to recover the signal from its spectrum, is given by:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2pkn/N}$$

The DFT may be regarded as a third Fourier representation, applicable to aperiodic digital signal of finite length. It can be viewed in two ways. If the signal x[n] is truly periodic, the equation gives a form of discrete Fourier Series; but if x[n] is basically aperiodic, and only being treated as periodic for the purposes of computation, the equation represents the DFT.

3. DERIVATION OF DISCRETE FOURIER TRANSFORM

The analysis equation of the Discrete Fourier Series, already defined by

$$a_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-j2pkn/N}$$

This applies, of course, to a strictly periodic signal with period N. Each spectral coefficient a_k is found by multiplying the signal x[n] by an exponential of the relevant frequency, and summing over one period.

Suppose we stretch all the adjacent repetitions of the signal apart, filling the gaps between them with zeros. Then N approach to infinite and the signal become aperiodic. We now consider Na_k rather than a_k as N approach to infinite will causes a_k vanish, but Na_k still remain finite.

$$Na_k = \sum_{n=0}^{N-1} x[n]e^{-j2pkn/N}$$

The limits of summation should also be changed, to take account of the fact that x[n] is now aperiodic. In general x[n] will exist for both positive and negative values of n, so for generality we sum between n=±∞.

$$X[k] = \sum_{n=-\infty}^{\infty} x[n]e^{-j2pkn/N}$$

Using similar arguments and substitutions, we can develop the inverse transform form the synthesis equation of the Discrete Fourier Series. The inverse transform tells us how to derive the signal x[n] from its spectrum X[k]. The synthesis equation is:

$$x[n] = \sum_{k=0}^{N-1} a_k e^{j2pkn/N}$$

and for DFT $$x[n] = \sum_{k=0}^{N-1} \frac{X[k]}{N} e^{j2pkn/N}$$

The analogy signals and Discrete Fourier Transform in fact have many similarities. However, one major difference is the spectrum of digital signal is always repetitive, unlike that of an analogy signal. This is an inevitable consequence of sampling, and reflects the ambiguity of digital signals.

4. SPECTRA OF PERIODIC DIGITAL SIGNAL

A periodic digital can be represented by a Fourier Series. The coefficients of its line spectrum indicate the amount of various frequencies contained in the signal. The may be found using the equation:

$$a_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-j2pkn/N}$$

a_k represents the kth spectral component, or harmonic, and N is the number of samples values in each period of the signal. The equation above to find the spectral component is call analysis equation of discrete Fourier Series. Conversely, using a_k to regenerate x[n] the equation is call synthesis equation.

$$x[n] = \sum_{k=0}^{N-1} a_k e^{j2pkn/N}$$

A periodic digital signal with N samples per period may therefore be completely specified in the frequency domain by a set of N consecutive harmonics. Half this number of harmonics is adequate if x[n] is real, because of the mirror-image pattern we have already noted.

5. PROPERTIES OF THE DFT

In this section we summarize a number of important properties of the DFT, including linearity, time-shifting, convolution and modulation. The properties are closely related to those of the discrete Fourier series. We also describe the effects on the DFT when the signal being transformed is real or complex, even or odd.

By definition we have $X[k]=\sum_{n=0}^{N-1} x[n]e^{-j2pkn/N}$

Circulant: x[n]=x[n+N]
X[k]=X[k+N]

Linearity:

If $x_1[n] \leftrightarrow X_1[k]$ and $x_2[n] \leftrightarrow X_2[k]$
Then $A\,x_1[n] + Bx_2[n] \leftrightarrow AX_1[k] + B\,X_2[k]$

Time-shifting:

If $x[n] \leftrightarrow X[k]$
Then $x[n-n_0] \leftrightarrow \sum_{n=0}^{N-1} x[n-n_0]W_N^{kn} = \sum_{s=0}^{N-1} x[s]W_N^{ks}W_N^{kn_0} = X[k]W_N^{kn_0}$

Convolution:

If $x_1[n] \leftrightarrow X_1[k]$ and $x_2[n] \leftrightarrow X_2[k]$
Then $x_1[n] * x_2[n] \leftrightarrow X_1[k] .X_2[k]$

Let $y[m] = \sum_{n=0}^{N-1} x_1[n]x_2[m-n]$

$$\sum_{m=0}^{N-1} y[m]W^{mk} = \sum_{m=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1[n]x_2[m-n]\right]W^{mk}$$
$$= \sum_{s=-n}^{N-1}\left[\sum_{n=0}^{N-1} x_1[n]x_2[s]\right]W^{s+nk}$$
$$= \left[\sum_{s=0}^{N-1} x_2[s]W^{sk}\right]\left[\sum_{n=0}^{N-1} x_1[n]W^{nk}\right]$$
$$= X_1[k].X_2[k]$$

Modulation:

If $x_1[n] \leftrightarrow X_1[k]$ and $x_2[n] \leftrightarrow X_2[k]$
Then $x_1[n] . x_2[n] \leftrightarrow X_1[k] * X_2[k]$

Let $Y[L] = \frac{1}{N}\sum_{k=0}^{N-1} X_1[k]X_2[L-k]$

$$\frac{1}{N}\sum_{L=0}^{N-1} Y[L]W^{-nL} = \frac{1}{N^2}\sum_{L=0}^{N-1}\left[\sum_{k=0}^{N-1} X_1[k]X_2[L-k]\right]W^{-nL}$$

$$= \frac{1}{N^2} \sum_{\Phi=-k}^{N-1} \left[\sum_{k=0}^{N-1} X_1[k] X_2[\Phi] \right] W^{-(\Phi+k)n}$$

$$= \left[\frac{1}{N} \sum_{\Phi=0}^{N-1} X_2[\Phi] W^{-\Phi n} \right] \left[\frac{1}{N} \sum_{k=0}^{N-1} X_1[k] W^{-nk} \right]$$

$$= x_1[n].x_2[n]$$

6. COMPUTING THE DFT

In this section, I will explain the calculation method of the program DFT V0.1, before explore the program code, let have a general view of DFT calculation first.

General View of computing DFT

The most obvious approach to computing the DFT and IDFT is to implement the below equation directly. The method is straightforward, but relatively low. Expressing the imaginary exponentials in term of sinusoidal and cosines, we have DFT.

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2pkn/N}$$

$$= \sum_{n=0}^{N-1} x[n] (\cos 2pkn/N - j \sin 2pkn/N)$$

$$R(X[k]) = \sum_{n=0}^{N-1} x[n] \cos 2pkn/N$$

$$I(X[k]) = -\sum_{n=0}^{N-1} x[n] \sin 2pkn/N$$

$$|X[k]| = \sqrt{(R(X[k]))^2 + (I(X[k]))^2}$$

Similarly, we have IDFT

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2pkn/N}$$

$$R(x[n]) = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cos 2pkn/N$$

$$I(x[n]) = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \sin 2pkn/N$$

$$|x[n]| = \sqrt{(R(x[n]))^2 + (I(x[n]))^2}$$

Direct implement the equation above is costly in terms of the number of multiplications involved. There require N^2 complex multiplication, and each complex multiplication require 4 real multiplication and 2 real addition as (a+jb)(c+jd)=(ac+ bd) + j(ad+bd).

Many of the multiplication turn out to be redundant, because of the exponentials specified in the DFT and IDFT equation are calculated many times as k and n vary. This is particularly true with lengthy transforms. High efficient FFT algorithms are available for reducing the redundancy.

<u>Program code</u>

In the program I use the straightforward way to implement the calculation because of the simplicity. The function below provides the DFT calculation. The input parameter is x[n] and number of sample. The program is divided into two parts, the first perform DTF and get the N samples F[k], the second part copy the N samples along the frequency axis. The characteristic is because of the sampling. When calculation is complete return the DFT values to the caller.

```
Public Function DFT(x() As Double, NoOfSample As Integer) As Variant

    Dim n As Integer
    Dim ans As Single                    'Temporary answer
    Dim T As Double
    Dim pi As Double                     '3.14..
    Dim i As Integer
    Dim k As Integer

    Dim Fr(1000) As Double              'real part of F
    Dim Fi(1000) As Double              'image part of F
    Dim F(1000) As Double               'magnitude of F
    Dim Φ(1000) As Double               'phase of F

    pi = 4 * Atn(1)

   '******Perform DTF transformation******
   For k = 0 To NoOfSample - 1

        'Real Part
        ans = 0
        For n = 0 To NoOfSample - 1
            ans = ans + x(n) * Math.Cos(2 * pi * k * n / NoOfSample)
        Next n
        Fr(k) = ans

        'Image Part
        ans = 0
        For n = 0 To NoOfSample - 1
            ans = ans + x(n) * Math.Sin(2 * pi * k * n / NoOfSample)
        Next n
        Fi(k) = ans

      'magnitude
      F(k) = (Fr(k) ^ 2 + Fi(k) ^ 2) ^ 0.5
      If Fi(k) = 0 Then
        Φ(k) = pi / 2
      Else
        Φ(k) = Math.Atn(Fr(k) / Fi(k))   'inverse trigonometric function of Tan
      End If

   Next k

      '******Fill the repetitive samples in Frequency Domain******
    'fill N samples signal in the time domain and form a peridic signal
    i = 0
    For k = NoOfSample To 1000
```

```
        F(k) = F(i)
        i = i + 1
        If i = NoOfSample Then i = 0
    Next k

    DFT = F()          '* return DFT calculation result to caller

End Function
```

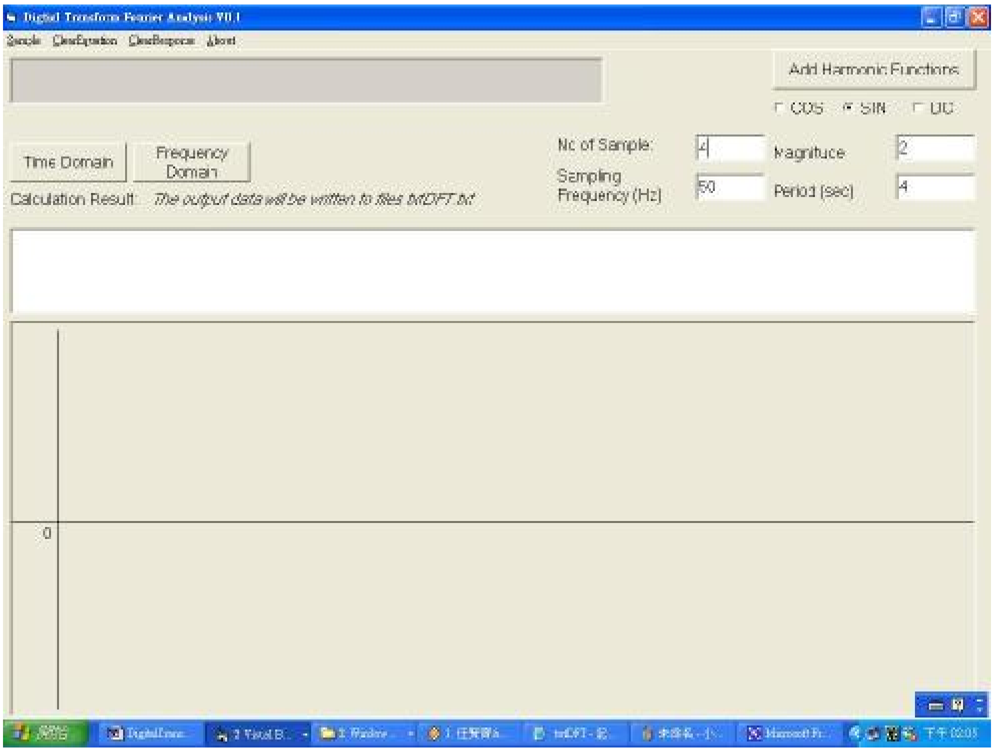
7. SOFTWARE DEMONSTRATION

In this section I will demonstrate the usage of the software DFT Analysis V0.1. That is software users can input some harmonic signal and then implement DFT transformation. By setting different sampling frequency, and the number of sample, the sampled data will be different, and therefore the DFT will also different.

Press the button "Time Domain" will draw the sampled signal in time domain. And press the button "Frequency Domain" will draw the DFT transformed signal in frequency domain.

User Interfacing

When the program starts the following interfacing will be appeared. Users can input some harmonic input signal for implementing the DFT analysis. Select the appropriate magnitude, and period for harmonic input, when input data is ok, press "Add Harmonic Functions", then the input signal will added and printed in text box at the top-right edge. When the input stream is completed, enter the number of sample and the sampling frequency for sampling the harmonic signal. Notice that by entering lower sampling frequency, aliasing effect will appear. After DFT calculation, time domain and frequency domain spectrum can be plotted. By pressing the buttons "Time Domain" and "Frequency Domain". All the input and output data will written to txt file txtDFT.txt that located in program directory, users can open it for reference or record.

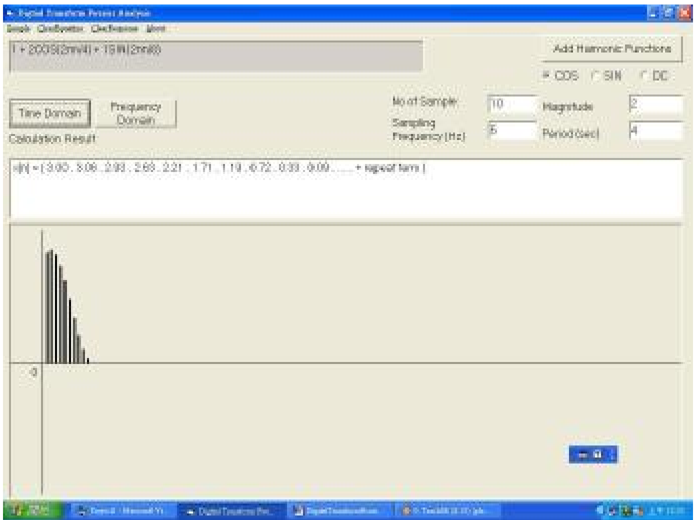


EXAMPLE 1: 1 + 2COS(2πn/4) + 1SIN(2πn/8)

That is a sample equation of the DFT program. There is some samples equation for demonstrating the DFT properties. In this example, the sampling frequency is 5Hz>>1/4 and >>1/8. Therefore there is no aliasing problem. Because the number of samples is ten in this case, the repetition cycle in frequency domain is also ten.

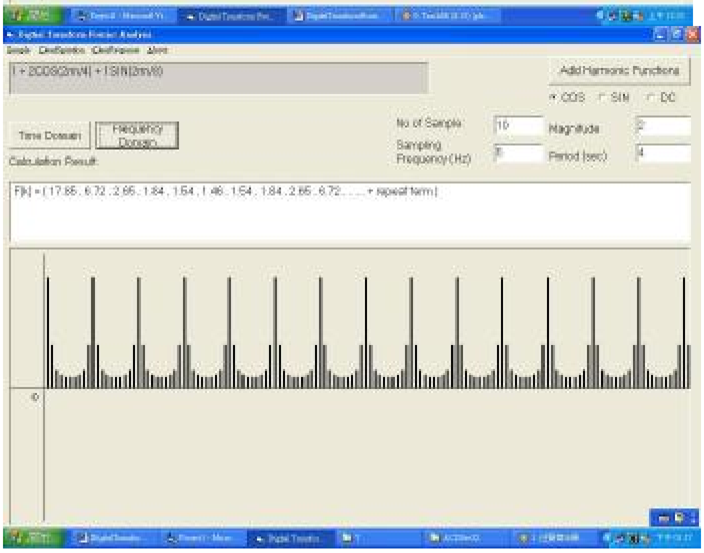
The Time domain spectrum. Press Time domain to plot the sampled signal in time domain. There are 10 samples.

x[n] = { 3.00 , 3.06 , 2.93 , 2.63 , 2.21 , 1.71 , 1.19 , 0.72 , 0.33 , 0.09 , ... + repeat term }

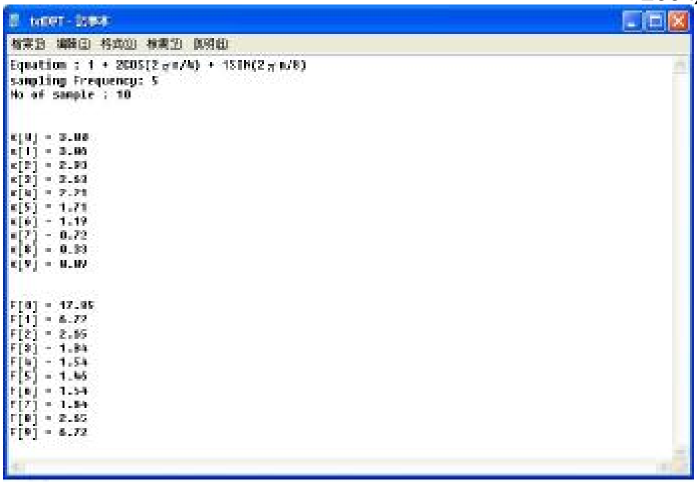


The frequency domain spectrum. Because the number of samples is ten in this case, the repetition cycle in frequency domain is also ten.

F[k] = { 17.85 , 6.72 , 2.65 , 1.84 , 1.54 , 1.46 , 1.54 , 1.84 , 2.65 , 6.72 , ... + repeat term }



When implement the DFT, the sampled input signal and the corresponding DFT transformed signal will be written to the txtDFT.txt file that located in the program directory. Therefore user can record or have a view of the output data.

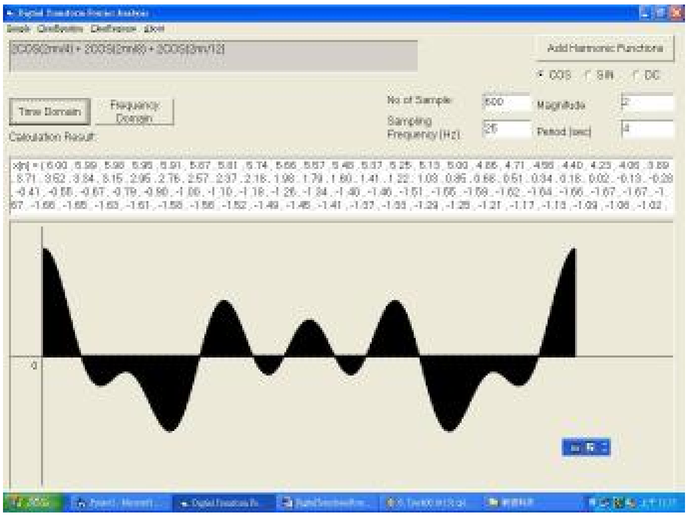


EXAMPLE 2: 2COS(2πn/4) + 2COS(2πn/8) + 2COS(2πn/12)

That is a sample equation of the DFT program. There is some samples equation for demonstrating the DFT properties. In this example, the sampling frequency is 25Hz>>1/8 and >>1/12. Therefore there is no aliasing problem. Because the number of samples is 600 in this case, the repetition cycle in frequency domain is also 600.

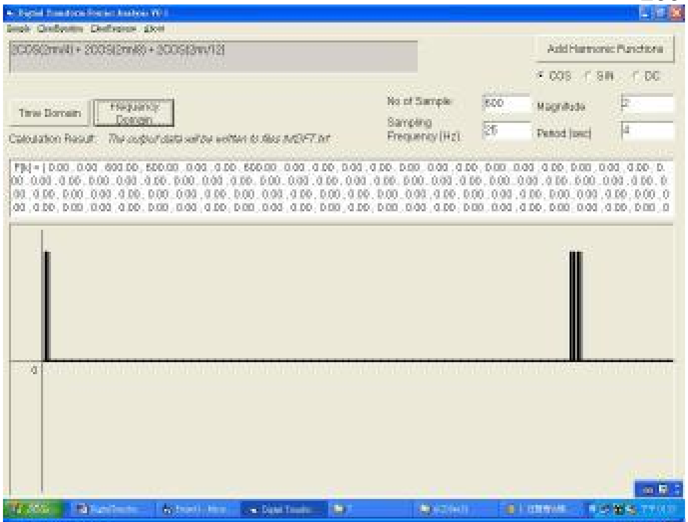
The Time domain spectrum. Press Time domain to plot the sampled signal in time domain. There are 600 samples.

x[n] = { 6.00 , 5.99 , 5.98 , 5.95 , 5.91 , 5.87 , 5.81 , 5.74 , 5.66 , 5.57 , 5.48 , 5.37 , 5.25 , 5.13 ……………………



The frequency domain spectrum. Because the number of samples is 600 in this case, the repetition cycle in frequency domain is also 600.

F[k] = { 0.00 , 0.00 , 600.00 , 600.00 , 0.00 , 0.00 , 600.00 , 0.00 , 0.00 , 0.00 , 0.00 …………………



Similarly. The output data can be found in file txtDFT.txt.

8. CONCLUSION

In this essay writing and presentation, I need to write a simple program, for demonstrate the superposition of several simple harmonic functions. And plot the corresponding Fourier transform spectrums. Actually, the DFT calculation in this program is straightforward, although the calculation time is costly, for simplicity a short length of DFT is affordable for general-purpose computer. When the program starts, user can input some harmonic functions. By setting difference sampling frequency and the number of sample, the sampled data is different and hence it DFT is also different. The program mainly divided into two parts, the first is sampling and the second is DFT calculation.

If we look at the derivation of DFT, we find the DFT can be regard as a third Fourier representation, applicable to aperiodic digital signals of finite length. The discrete fourier series is applicable for harmonic signal, however for those signal that are not periodic, we still can use a track to tactile it. We can consider the signal adjacent with zero magnitude and take the period infinite, then we use fourier series to calculate the fourier transformation and this transformation call DFT.

Because the discrete fourier series has implement the sampling, repetition in frequency domain is inevitable. The sampled signal with bandwidth equal to the original analog signal bandwidth, however, we only consider the non-zero term of the sampled input x[n], the length of DFT is fixed. Because the existing of exponential term in DFT equation, when the length in frequency domain exist the length of non-zero input term x[n], the signal just repeat itself. We can also consider the repetition due to convolute of the pulse train used for sampling and original analog signal in frequency domain. The convolute is due to the sampling in time domain as it is using multiplication of analog signal with pulse train.