PS403 - Digital Signal processing

III. DSP - Digital Fourier Series and Transforms

Key Text:

Digital Signal Processing with Computer Applications (2nd Ed.)

Paul A Lynn and Wolfgang Fuerst, (Publisher: John Wiley & Sons, UK)

We will cover in this section

How to compute the Fourier series for a periodic digital waveform

How to compute the Fourier transform for an aperiodic digital waveform

Deconvolution in the frequency domain

Introduction - Digital Fourier Series and Transforms

Jean Baptiste Fourier - (1768 - 1830).

Reasons to work in the Fourier domain.

1. Sinusoidal waveforms occur frequently in nature

2. Given the frequency spectrum of an I/P signal I(f) and the frequency transfer function of of an LTI processor H(f) , we can compute the spectrum of the processed signal by simple multiplication:

$$O(f) = I(f) \times H(f)$$

3. Much of DSP design is concerned with frequency transmission

Introduction - Digital Fourier Series and Transforms

Properties of signals in the frequency domain:

1. Signals, symmetric (centred) about time t = 0 contain only cosines

2. Periodic and infinitely long signals (waveforms) may be synthesised from a superposition of harmonically related sinusoids. Hence they may be represented by Fourier series and exhibit line spectra

3. Aperiodic signals (such as single isolated pulses exponential waveforms, etc.) contain a continuum of frequencies (continuum spectrum) are so are represented by the Integral Fourier Transform

Digital Fourier Series

Infinitely long, periodic waves can be represented by a superposition of sinusoids of varying amplitude and relative phase at the fundamental frequency and its harmonics.

The amplitudes of each component sinusoid for a sampled data signal/waveform x[n], where x[n] contains N values, are given by:

$$a_k = \frac{1}{N}\sum_{n=0}^{N} x[n]\exp\left(\frac{-j2\pi kn}{N}\right)$$

Analysis Equation

Digital Fourier Series

x[n] may also be reconstructed from its 'Harmonic Amplitudes' (a_k) using the so-called 'Synthesis Equation'

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} a_k \exp\left(\frac{j2\pi kn}{N}\right)$$

Example:

See fig 3.1. Waveform with a period of 7 samples per cycle. Equation 1 is complex and has to be split into two parts for computation:

$$a_k = \frac{1}{N}\left\{\sum_{k=0}^{N-1} x[n]Cos\left(\frac{-j2\pi kn}{N}\right) + j\sum_{k=0}^{N-1} x[n]Sin\left(\frac{-j2\pi kn}{N}\right)\right\}$$

Digital Fourier Series

x[n]; -1, -1, 1, 2, 1, -2, 3, -1, -1, 1, 2, 1, -2,.......

One cycle (7 samples)

k	Real Part of a_k	Imaginary Part of a_k
0	0.4 285 715	0
1	0.3 018 007	-0.1 086 581
2	0.7 864 088	0.3 847 772
3	-0.3 024 935	-0.6 687 913
4	-0.3 024 928	0.6 687 927
5	0.7 864 058	-0.3 847 782
6	0.3 018 006	0.1 086 581

$$\mathrm{Re}(a_k) = \frac{1}{7}\sum_{n=0}^{6} x[n]Cos\left(\frac{-2\pi kn}{N}\right)$$

$$\mathrm{Im}(a_k) = \frac{1}{7}\sum_{n=0}^{6} x[n]Sin\left(\frac{-2\pi kn}{N}\right)$$

Digital Fourier Series

Input file
"Xn.dat"

2
1
-2
3
-1
-1
1

```
DIM X(100), AKR (100), AKI (100)
OPEN "Xn.dat" FOR INPUT AS #1
OPEN "Ak.dat" FOR OUTPUT AS #2
PRINT #2, "K", TAB(20);"Re (ak)";TAB(40); "Im(ak)"
FOR i = 0 TO 6
INPUT #1, X(i)
NEXT I
FOR k = 0 to 6
AKR(k) = 0.0, AKI(k) = 0.0
FOR j = 0 to 6
AKR(k) = AKR(k) + X(j)*COS((2*3.1.41.6*j*k)/7)
AKI(k) = AKI(k) + X(j)*Sin((2*3.1.41.6*j*k)/7)
NEXT j
AKR(k) = AKR(k)/7
AKI(k) = AKI(k)/7
PRINT #2, k, TAB(20); AKR(k);TAB(40); AKI(k)
NEXT k
```

Digital Fourier Series

Points to note:

1. A sampled periodic data signal with 'N' samples/period in the 'time' domain will yield 'N' real and 'N' imaginary harmonic amplitudes (or Fourier coefficients) in the 'k' or discete frequency domain.

2. The line spectrum will repeat itself every 'N' values, i.e., the spectrum itself is repetitive and periodic - but we need only the 'N' harmonic amplitudes to completely specify/ synthesise an 'N' valued' signal

3. Notice for a sampled data signal x[n] which is a real function of 'n', i.e., real valued signal, the real values of a_k display mirror image symmmetry; $a_1 = 1_6$, $a_2 = a_5$, etc. (True also for the imaginary coefficients but with a sign change - see Fig 3.1)

Note that as $N \rightarrow \infty$, we move towards a single, non repeating waveform, i.e., an aperiodic signal. The harmonic amplitudes get very small (1/N) and frequencies infinitely close - i.e., we go to a continuum of frequencies - Discrete Fourier Transform needed to analyse/synthesise such signals.

Digital Fourier Series - Amplitude and Phase Spectra

For a periodic signal the magnitude of each harmonic amplitude is given by:

$$|a_k| = \sqrt{\mathrm{Re}(a_k)^2 + \mathrm{Im}(a_k)^2}$$

The corresponding phase angle for each frequency (harmonic) is given by:

$$\Phi_k = Tan^{-1}\left\{\frac{\mathrm{Im}(a_k)}{\mathrm{Re}(a_k)}\right\}$$

A plot of $|a_k|$ vs k yields the 'Amplitude Spectrum'

A plot of Φ_k vs k yields the 'Phase Spectrum'

Digital Fourier Series

Example. Consider a signal with three frequency components;

$$x[n] = 1 + Sin\left(\frac{n\pi}{4}\right) + 2Cos\left(\frac{n\pi}{2}\right)$$

dc cmpt (offset)

8 samples per cycle

4 samples per cycle

It is clear that a full cycle of x[n] will require its evaluation over the 'n' range, $0 \le n \le 7$:

n	0	1	2	3	4	5	6	7
X[n]	3.000	1.707	0.000	1.707	3.000	0.293	-2.000	0.293

Digital Fourier Series

We can extract the harmonic amplitudes (a_k) directly from the Analysis Equation is we rewrite x[n] in Euler notation:

$$Sin\left(\frac{n\pi}{4}\right) = \frac{1}{2j}\left[\exp\left(\frac{jn\pi}{4}\right) - \exp\left(\frac{-jn\pi}{4}\right)\right]$$

$$Cos\left(\frac{n\pi}{2}\right) = \frac{1}{2}\left[\exp\left(\frac{jn\pi}{2}\right) + \exp\left(\frac{-jn\pi}{2}\right)\right]$$

So we can now write x[n] as:

$$x[n] = 1 + \frac{1}{2j}\left[\exp\left(\frac{jn\pi}{4}\right) - \exp\left(\frac{-jn\pi}{4}\right)\right] + {}^{1}\left[\exp\left(\frac{jn\pi}{2}\right) + \exp\left(\frac{-jn\pi}{2}\right)\right]$$

$$x[n] = 1 + Sin\left(\frac{n\pi}{4}\right) + 2Cos\left(\frac{n\pi}{2}\right)$$

Digital Fourier Series

Noting that: $1/2j = -j/2$ and $\exp(jn\pi/2) = \exp(2jn\pi/4)$

We can then write x[n] as:

$$x[n] = 1 + \frac{j}{2}\exp\left(\frac{-jn\pi}{4}\right) - \frac{j}{2}\exp\left(\frac{jn\pi}{4}\right) + \exp\left(\frac{-2jn\pi}{4}\right) + \exp\left(\frac{2jn\pi}{4}\right)$$

Comparing this with: $$x[n] = \sum_{k=0}^{N-1} a_k \exp\left(\frac{+jkn\pi}{4}\right)$$

We see that: $a_0 = 1$, $a_1 = -j/2$, $a_2 = 1$, $a_{-1} = j/2$ and $a_{-2} = 1$

x[n] has eight values but there are only five non-zero a_k values $\Rightarrow$ the other three values must be = 0

Also there are three real values of a_k and two imaginary values of a_k

Digital Fourier Series

The values can be tabulated below -

k	$Re(a_k)$	$Im(a_k)$	$\lvert a_k \rvert$	Φ_k
-2	1	0	1	0
-1	0	0.5	0.5	$\pi/2$
0	1	0	1	0
1	0	-0.5	0.5	π /2
2	1	0	1	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0

PLOT all x[n] and a_k values

Notes:

If x[n] = -x[-n], i.e., x[n] is an odd function of 'n', then all $Re(a_k) = 0$ or equivalently, odd periodic signals can be constructed from 'Sine' fns

Conversely, if x[-n] = x[n], i.e., x[n] is an even function of 'n', then all $Im(a_k) = 0$ and so even periodic signals can be constructed from 'Cosine' fns

Digital Fourier Series

Useful properties of Fourier Series

(a) Linearity

if $x_1[n] \rightarrow a_k$ and $x_2[n] \rightarrow b_k$, then $A.x_1[n] + B.x_2[n] \rightarrow A.a_k + B.b_k$

(b) Time-Shifting

if $x[n] \rightarrow a_k$, then if $x[n-n_0] \rightarrow a_k \exp[-j2\pi k n_0/N]$

Magnitude spectrum unchanged

But phase spectrum shifted by extra factor

Note: for $n_0 = N$ (1 Complete cycle of x[n]),

$\exp[-j2\pi k n_0/N] = \exp[-j2\pi k] = 1$,

Hence the spectrum is said to be 'circular' or 'cyclic'

Digital Fourier Series

(c) Differentiation

if $x[n] \rightarrow a_k$, then $x[n] - x[n-1] \rightarrow a_k \{1 - \exp[-j2\pi k/N]\}$

1st order difference of
$x[n] \rightarrow$ differentiation

The time-shifting property gives $x[n - 1] \rightarrow a_k \exp[-j2\pi k/N]$.
Then apply linearity property to get spectrum of the differentiated signal !

(d) Integration $\rightarrow$ running sum of $x[n]$

if $x[n] \rightarrow a_k$, then

$$\sum_{k=-\infty}^{k=n} x[k] \rightarrow a_k \left\{1 - \exp\left(\frac{-2j\pi k}{N}\right)\right\}^{-1}$$

Digital Fourier Series

(e) Convolution

if $x_1[n] \rightarrow a_k$ and $x_2[n] \rightarrow b_k$, then

$$R_{x1,x2} = \sum_{m=0}^{N-1} x_1[m]\,x_2[n-m] \rightarrow N a_k b_k$$

Convolution over a single cycle of 'N' data points

Circular Convolution

(f) Modulation Property [Inverse of convolution]

if $x_1[n] \rightarrow a_k$ and $x_2[n] \rightarrow b_k$, then

$$x_1 . x_2 \rightarrow \sum_{m=0}^{m=N-1} a_m b_{k-m}$$

Digital Fourier Series - Progamme no. 8

Investigaton of a multifrequency signal (Fig 3.3)

Effect of 'end-to-end' vs non integral number of cycles (Fig 3.4)

Spectrum of a unit impulse (Fig 3.5). [Actually it is an impulse train and periodic (as it has to be in order that Fourier Series can be used to represent the signal - period is set by the 64 samples]. Assume that the signal repeats every 64 samples. Note Φ_k = 0 and a_k = 1/N = 1/64.

Delayed unit impulse (Fig 3.6). a_k = 1/64 as before but is a Φ_k = 'Linear Phase Characteristic'

Equation 3.9. Parseval's Theorem applied to sampled data signals

$$\frac{1}{N}\sum_{n=0}^{n=N-1}\left(x[n]\right)^2 = \sum_{k=0}^{k=N-1}\left(a_k\right)^2$$

Average Energy/Cycle
in the Time Domain

Average Energy/Cycle
in the Frequency Domain

Digital Fourier Transform

To represent aperiodic signals and noise 'signals' we need to invoke **Fourier Transforms**

APERIODIC

- Not strictly repetitive
- Repetitive but not strictly periodic

Cf: Appendix 2 of Paul and Fuerst for a review of continuous time Fourier Transforms

In fact we are not studying the DFT. Rather it is the continuous FT of a discrete (and finite) sampled data signal that we will deal with here !

Digital Fourier Transform

In what follows we develop the digital FT from digital FS.

Consider again the analysis (equation) form of the Fourier Series:

$$a_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n].\exp\left[\frac{-j2\pi kn}{N}\right]$$

Where x[n] is cyclic or periodic with period 'N' samples, e.g., consider the signal in Fig 3.7(a) where N = 5

Now consider what happens when adjacent cycles are artificially separated or spaced in time - Fig 3.7 (b)

Signal is in effect 'stretched' from N=5 to N=12 samples/cycle -

Digital Fourier Transform

But what happens in the frequency domain ?

(a) Amplitudes will become smaller since $a_k \propto 1/N$

(b) Also the number of frequencies (k values) will increase, i.e., the frequency components labelled by 'k' will get closer to each other (bunch up)

In the limit as $N \rightarrow \infty$, frequency will become a continuous variable. i.e., we move from a line spectrum to a continuum spectrum

In the limit as $N \rightarrow \infty$, the summation of discrete frequency which characterises the FS will become an integral over continuous frequency

See this on next slides......................

Digital Fourier Transform

Development of Fourier Transform from Fourier Series

The discrete frequency $2\pi k/N$ becomes Ω - continuous frequency (radians)

$$2\pi k/N \rightarrow \Omega$$

$$\text{DFS} \rightarrow \text{'DFT'}$$

Firstly we rewrite the FS analysis equation with continuous frequency -

$$X(\Omega) = Na_k = \sum_{n=0}^{n=\infty} x[n]\exp(-jn\Omega)$$

We often make x[n] symmetric about n = 0 and so more generally:

$$X(\Omega) = Na_k = \sum_{n=-\infty}^{n=\infty} x[n]\exp(-jn\Omega)$$

Continuous Fourier Transform of a discrete sampled signal x[n]

Digital Fourier Transform

Similarly from the Digital Fourier Series:

$$x[n] = \sum_{k=0}^{N-1} a_k \exp\left[\frac{j2\pi kn}{N}\right]$$

$$Let \Omega_0 = \frac{2\pi}{N} = the fundamental frequency$$

This permits us to write the synthesis equation as:

$$x[n] = \sum_{k=0}^{N-1} \left[\frac{X(k\Omega_0)}{N}\right] \exp(jkn\Omega_0)$$

since, also by definition, $X = Na_k$.

Since $\Omega_0/2\pi = 1/N$, where Ω_0 = fundamental frequency we can write:

$$x[n] = \frac{1}{2\pi} \sum_{k=0}^{N-1} X(k\Omega_0) \exp(jkn\Omega_0)\Omega_0$$

Digital Fourier Transform

In addition:

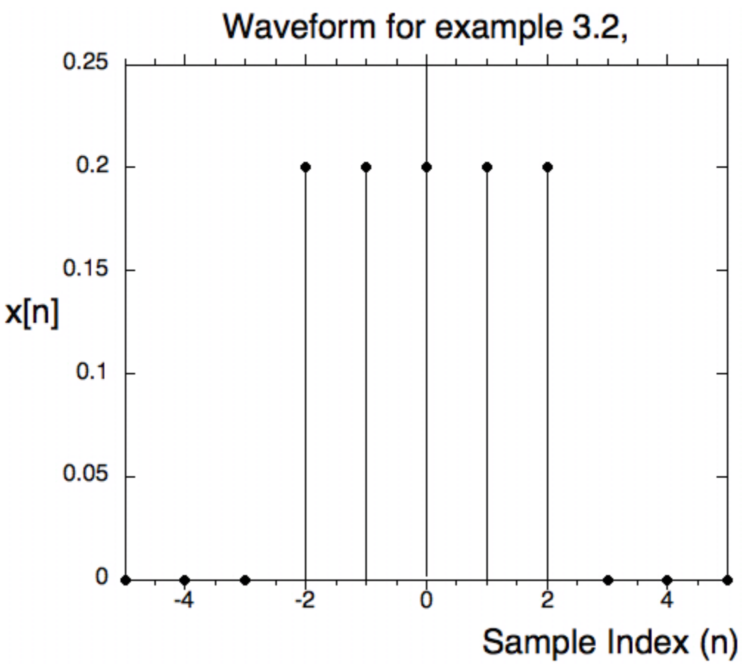
as $N \rightarrow \infty$, $\Omega_0 \rightarrow d\Omega$, $k\Omega_0 \rightarrow \Omega$ and $\sum_{k=0}^{N-1} \rightarrow \int_{2\pi}$

So finally we can write the 'Synthesis Transform'

$$x[n] = \frac{1}{2\pi}\int_0^{2\pi} X(\Omega)\exp(jn\Omega)d\Omega$$

Digital Fourier Transform

Example 3.2(a) (Paul and Fuerst)



Single isolated pulse with 5 non-zero sample values - find its FT

$$X(\Omega) = \sum_{n=-\infty}^{n=\infty} x[n]\exp(-jn\Omega)$$

x[n] is a running sum of weighted impulses δ[n]

Ergo -

$X(\Omega) = 0.2\{\delta[n - 2] + \delta[n - 1] + \delta[n] + \delta[n + 1] + \delta[n + 2]\} \text{x} \exp(-jn\Omega)$

Using the sifting property of the unit impulse, i.e., $\delta[n - n_0] \rightarrow \exp(-jn_0\Omega)$, we can write $X(\Omega) = 0.2\{\exp(-j2\Omega)+ \exp(-j\Omega) + 1 + \exp(j\Omega) + \exp(j2\Omega)\}$

Digital Fourier Transform

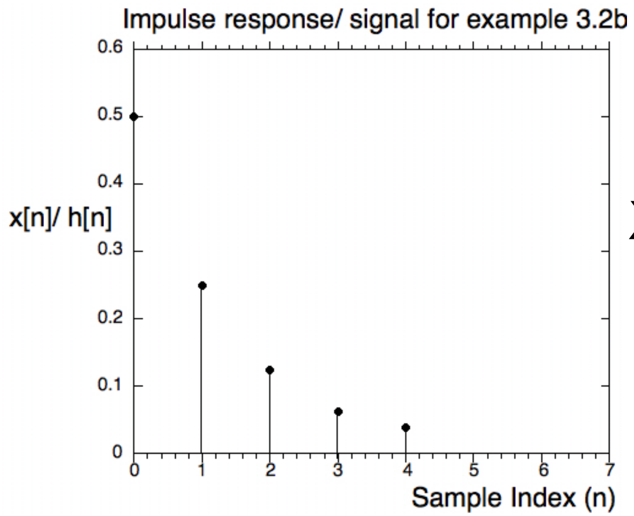
Example 3.2(a) (Paul and Fuerst) cont'd

x[n] is an even function of 'n' we have that exp(jΩ) → Cos(Ω),

$$\boxed{X(\Omega) = 0.2[1 + 2\mathrm{Cos}(\Omega) + 2\mathrm{Cos}(2\Omega)]}$$

Note that X(Ω) is a repetitive and periodic in Ω with period 2p (-π → +π)

Example 3.2(b) (Paul and Fuerst)



x[n] = 0.5, 0.25, 0.125, 0.0625,...............

X(Ω) = 0.5 + 0.25 exp(-jΩ)+ 0.125 exp(-j2Ω) +.....

$$= 0.5\sum_{n=0}^{n=\infty}\left\{0.5\exp(-j\Omega)\right\}^n = \frac{0.5}{1-0.5\exp(-j\Omega)}$$

Digital Fourier Transform

X(Ω) is a complex function containing both sine and cosine sinusoids; hence it is best displayed as amplitude and phase spectra-

Using $$\left|X(\Omega)\right| = \sqrt{X(\Omega)^{*}\,X(\Omega)}$$

We get $$\left|X(\Omega)\right| = \frac{0.5}{\left(1.25 - Cos\Omega\right)^{\frac{1}{2}}}$$

@ $\Omega = 0$, $|X(\Omega)|$ = 1 (i.e., it is a maximum)
@ $\Omega = \pi$, $|X(\Omega)|$ = 1/3 (i.e., it is a minimum)

Cf: Fig 3.8 in Lynn and Fuerst for plot of X(Ω) -

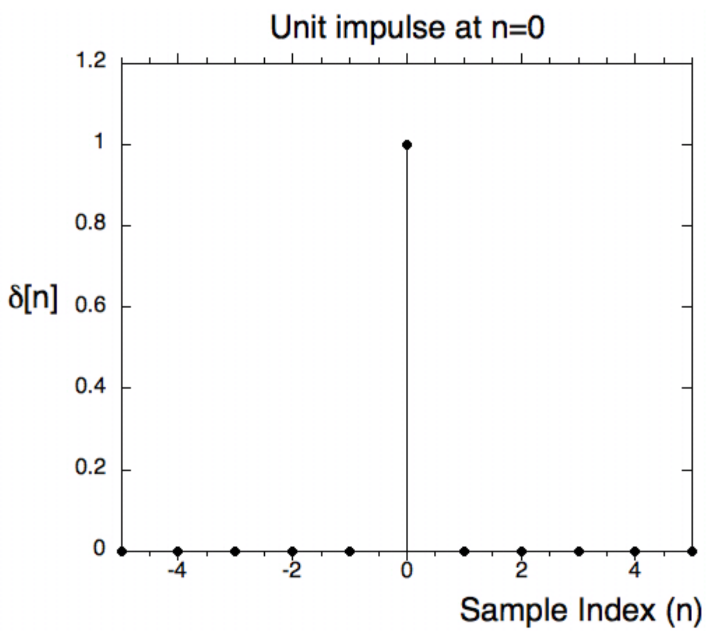
What does x[n] remind you of ?

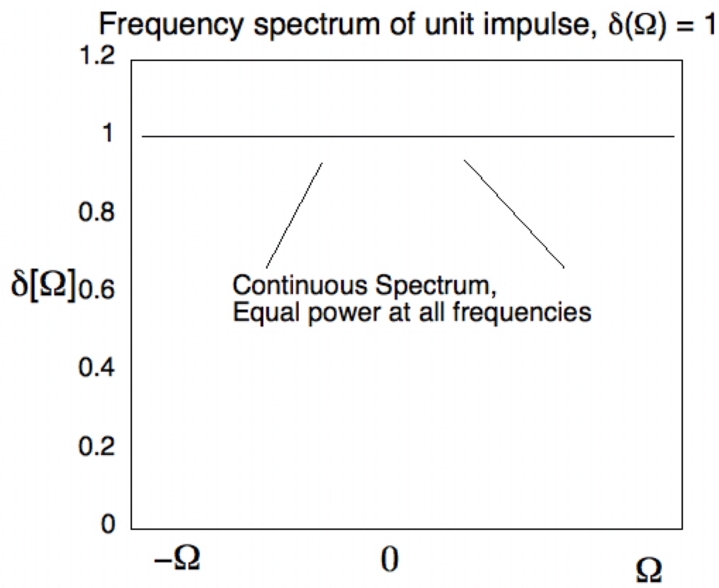
Digital Fourier Transform

Fourier Transform of a single 'isolated' unit impulse

$$X(\Omega) = \sum_{n=-\infty}^{n=\infty} x[n]\exp(-jn\Omega) \quad = \sum_{n=-\infty}^{n=\infty} \delta[n]\exp(-jn\Omega)$$

$$\boxed{= 1.\exp(0) = 1 \text{ !!!!!!!!!}}$$





Digital Fourier Transform

Since $\delta(\Omega) = 1$ it is real. Hence the $\text{Im}\{\delta(\Omega)=0\}$

$$|\delta(\Omega)| = \sqrt{\delta(\Omega)^* \delta(\Omega)} = 1$$

$$\Phi_\delta(\Omega) = Tan^{-1}\left\{\frac{\text{Im}[\delta(\Omega)]}{\text{Re}[\delta(\Omega)]}\right\} = 0$$

So the signal strength is constant at all frequencies and each single infinitely close Fourier component has exactly the same phase, i.e., the phase shift between them is zero.

Digital Fourier Transform

Usual Properties for Fourier Transforms in the Digital Domain

Linearity: $ax_1[n] + bx_2[n] \Leftrightarrow aX_1(\Omega) + bX_2(\Omega)$

Time Shifting: $x[n-n_0] \Leftrightarrow X(\Omega)\exp(-jn_0\Omega)$

Convolution: $x_1[n] * x_2[n] \Leftrightarrow X_1(\Omega) \times X_2(\Omega)$

NB: Time shift ≡ multiplying the FT by $\exp(-jn_0\Omega)$ in the frequency domain

Also: ***Frequency domain multiplication ≡ time domain convolution***

Since $\delta(\Omega) = 1$ it is real. Hence the $\mathrm{Im}\{\delta(\Omega)=0\}$

Digital Fourier Transform

Frequency Response of LTI Processors Cf: Fig 3.10

$x[n]$ Input Signal $\longrightarrow$ **Digital LTI Processor** $\longrightarrow$ Output Signal $y[n]$

't' domain	$x[n]$	$h[n]$	$y[n] = h[n]*x[n]$
'Ω' domain	$X(\Omega)$	$H(\Omega)$	$Y(\Omega) = H(\Omega) \times X(\Omega)$

Using polar (magnitude/phase) representation in the Ω plane we have:

$$X(\Omega) = |X(\Omega)|.\exp[-j\Phi_X(\Omega)]$$

Similarly: $$H(\Omega) = |H(\Omega)|.\exp[-j\Phi_H(\Omega)]$$

$H(\Omega)$ = LTI Processor Frequency Transfer Function,

$|H(\Omega)|$ = processor 'Gain' and $\Phi_H(\Omega)$ = processor Phase Transfer Function

Digital Fourier Transform

Example: Let's take 3.2 again but this time we designate x[n]'s as h[n]'s - Then 3.8(a) is a weighted moving or 5-point adjacent channel average filter

Then from our earlier solution we have: $H(\Omega) = 0.2\,\{1+2Cos(\Omega)+2Cos(2\Omega)\}$- Transfer Function for a 5 point moving average *low pass* filter - Fig 3.8(a)

Figure 3.8 (a)

Looking at positive Ω between 0 and π one can see that the filter transmits low frequencies most strongly - LOW PASS FILTER action

Notice that the gain = 0 @ $\Omega = 2\pi/5$ or 5 samples/cycle !

Looking at $H(\Omega)$ one can see that it is real-symmetric, hence no phase shifts are introduced

Digital Fourier Transform

Now consider figure 3.8 (b). This picture now refers to the impulse response of a low pass filter:

h[n] = 0.5δ[n] + 0.25δ[n-1] + 0.0625δ[n-2] + ...

Correspondingly: $H(\Omega) = \dfrac{0.5}{1 - 0.5\exp(-j\Omega)}$

And: $|H(\Omega)| = \dfrac{0.5}{\left(1.25 - Cos(\Omega)\right)^{0.5}}$

Figure 3.8 (b)

One can see that this is also clearly a low pass filter - but not a very good one. Significant transmission at $\Omega = \pi$ ($|H(\Omega)| = 1/3$)

Digital Fourier Transform

General specification of LTI processors in the 'Ω' domain

In general we know that a LTI processor can be specified by a difference equation of the form:

$$\sum_{l=0}^{l=L} c_l\, y[n-l] = \sum_{l=0}^{l=I} d_l\, x[n-l]$$

where 'L' is the order of the system and c_l are the recursive multiplier coefficients. Taking Fourier Transforms of both sides:

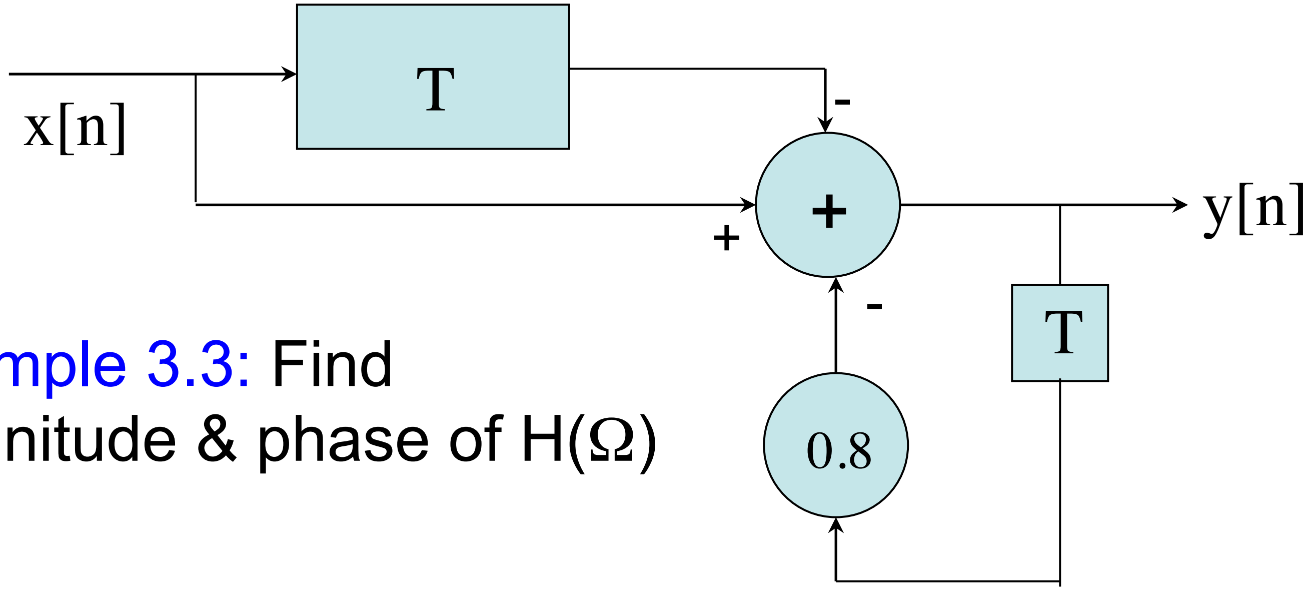
$$\sum_{l=0}^{l=L} c_l \exp(-jl\Omega)\, Y(\Omega) = \sum_{l=0}^{l=I} d_l \exp(-jl\Omega)\, X(\Omega)$$

using linearity and time shifting properties of the Fourier Transform

Digital Fourier Transform

We also have that: $Y(\Omega) = H(\Omega) \times X(\Omega)$ and hence that: $H(\Omega) = Y(\Omega)/X(\Omega)$

So we can write: $$H(\Omega) = \frac{Y(\Omega)}{X(\Omega)} = \frac{\sum_{l=0}^{l=I} d_l \exp(-jl\Omega)}{\sum_{l=0}^{l=L} c_l \exp(-jl\Omega)}$$



Example 3.3: Find magnitude & phase of $H(\Omega)$

Digital Fourier Transform

Inspection of the flow of the block diagram:
y[n] = -0.8y[n-1] + x[n] - x[n-1] which can be written

$$\underbrace{y[n]}_{c_0} + \underbrace{0.8y[n-1]}_{c_1} = \underbrace{x[n]}_{d_0} - \underbrace{x[n-1]}_{d_1}$$

By inspection: $c_0 = 1.0$, $c_1 = 0.8$, $d_0 = 1.0$, $d_1 = -1.0$

Using: $$H(\Omega) = \frac{\sum_{l=0}^{l=I} d_l \exp(-jl\Omega)}{\sum_{l=0}^{l=L} c_l \exp(-jl\Omega)}$$

Digital Fourier Transform

We get: $$H(\Omega)=\frac{\left[1.\exp(-j0)\right]+\left[-1.\exp(-j\Omega)\right]}{\left[1.\exp(-j0)\right]+\left[0.8.\exp(-j\Omega)\right]}$$

$$= \qquad H(\Omega)=\frac{1-\exp(-j\Omega)}{1+0.8.\exp(-j\Omega)}$$

$$\Rightarrow H(\Omega)=\frac{1-Cos\Omega+jSin\Omega}{1+0.8Cos\Omega-0.8\,jSin\Omega}$$

$$\Rightarrow \left|H(\Omega)\right|=\frac{\left[(1-Cos\Omega)^2+Sin^2\Omega\right]^{1/2}}{\left[(1+0.8Cos\Omega)^2+0.64\,Sin^2\Omega\right]^{1/2}}$$

Digital Fourier Transform

So we can write: $$|H(\Omega)| = \left[\frac{2 - 2Cos\Omega}{1.64 - 1.6Cos\Omega}\right]^{1/2}$$

Phase Spectrum: $$\Phi_H(\Omega) = Tan^{-1}\left[\frac{\mathrm{Im}\,H(\Omega)}{\mathrm{Re}\,H(\Omega)}\right]$$

$$\Rightarrow \Phi_H(\Omega) = Tan^{-1}\left[\frac{Sin\Omega}{1 - Cos\Omega}\right] - Tan^{-1}\left[\frac{-0.8Sin\Omega}{1 + 0.8Cos\Omega}\right]$$

Cf: Fig 3.11 for plots of magnitude $|H(\Omega)|$ and phase $\Phi_H(\Omega)$ transfer functions or 'gain profiles'

The processor is a High Pass Filter with a gain of 10 @ $\Omega=\pi$

Program no 9 to investigate 90, 91 & 92. Figs 3.12 & 3.13