ΨΕΣ

2ο Σύνολο Αναλυτ. Ασκήσεων

Χρήστος Τσούδης
03117176
chris99ts@gmail.com

Άσκηση 2.1

α) Βέλτιστος γραμμικός προβλέπτης τάξης $p=3$ (μέθοδος Αυτοσυσχέτισης)

$r_x[0]=1{,}05$, $r_x[1]=0{,}7$, $r_x[2]=0{,}5$, $r_x[3]=0{,}4$

α.1) $$R_x=\begin{bmatrix} r_x[0] & r_x[1] & r_x[2] & r_x[3] \\ r_x[1] & r_x[0] & r_x[1] & r_x[2] \\ r_x[2] & r_x[1] & r_x[0] & r_x[1] \\ r_x[3] & r_x[2] & r_x[1] & r_x[0] \end{bmatrix} = \begin{bmatrix} 1{,}05 & 0{,}7 & 0{,}5 & 0{,}4 \\ 0{,}7 & 1{,}05 & 0{,}7 & 0{,}5 \\ 0{,}5 & 0{,}7 & 1{,}05 & 0{,}7 \\ 0{,}4 & 0{,}5 & 0{,}7 & 1{,}05 \end{bmatrix}$$

▷ Η παραπάνω ακολουθία αυτοσυσχέτισης είναι θετικά ορισμένη $\Leftrightarrow$ R_x θετικά ορισμένος.

Θα πρέπει όλες οι κύριες υποορίζουσες να είναι θετικές. Πράγματι,

$$\Delta_1=\det[1{,}05]=1{,}05>0$$

$$\Delta_2=\det\begin{bmatrix} 1{,}05 & 0{,}7 \\ 0{,}7 & 1{,}05 \end{bmatrix}=0{,}6125>0$$

$$\Delta_3=\det\begin{bmatrix} 1{,}05 & 0{,}7 & 0{,}5 \\ 0{,}7 & 1{,}05 & 0{,}7 \\ 0{,}5 & 0{,}7 & 1{,}05 \end{bmatrix}=0{,}356125>0$$

$$\Delta_4=\det\begin{bmatrix} 1{,}05 & 0{,}7 & 0{,}5 & 0{,}4 \\ 0{,}7 & 1{,}05 & 0{,}7 & 0{,}5 \\ 0{,}5 & 0{,}7 & 1{,}05 & 0{,}7 \\ 0{,}4 & 0{,}5 & 0{,}7 & 1{,}05 \end{bmatrix}=0{,}206781>0$$

α.2) Levinson-Durbin: $E^{(0)}=r_x[0]=1{,}05$

τάξη $i=1$: $$K_1=-\frac{r_x[1]}{E^{(0)}}=-\frac{0{,}7}{1{,}05}=-0{,}667$$

$$a_1^{(1)}=-K_1=0{,}667$$

$$E^{(1)}=(1-K_1^2)E^{(0)}=0{,}592$$

Τάξη $i=2$:

$$K_2 = -\frac{\left(r_x[2] - a_1^{(1)} r_x[1]\right)}{E^{(1)}} = -\frac{0,5 - 0,667\cdot 0,7}{0,592} = -0,055$$

$$a_2^{(2)} = -K_2 = 0,055$$

$$a_1^{(2)} = a_1^{(1)} + K_2 a_1^{(1)} = 0,667 - 0,055\cdot 0,667 = 0,629$$

$$E^{(2)} = (1 - K_2^2)E^{(1)} = 0,590$$

Τάξη $i=3=p$:

$$K_3 = -\frac{\left(r_x[3] - (a_1^{(2)} r_x[2] + a_2^{(2)} r_x[1])\right)}{E^{(2)}} =$$

$$= -\frac{(0,4 - (0,629\cdot 0,5 + 0,055\cdot 0,7))}{0,59} = -0,079$$

$$a_3^{(3)} = -K_3 = 0,079$$

$$a_2^{(3)} = a_2^{(2)} + K_3 a_1^{(2)} = 0,055 - 0,079\cdot 0,629 = 0,005$$

$$a_1^{(3)} = a_1^{(2)} + K_3 a_2^{(2)} = 0,629 - 0,079\cdot 0,055 = 0,624$$

$$E^{(3)} = (1 - K_3^2)\cdot E^{(2)} \simeq 0,578$$

Άρα $\underline{LPC} = (a_1, a_2, a_3) = (0,624,\ 0,005,\ 0,079)$

$\underline{PARCOR} = (K_1, K_2, K_3) = (-0,667,\ -0,055,\ -0,079)$

B) $p=4$

LPC: $a_1^{(4)} = 0,93$, $a_2^{(4)} = 0,338$, $a_3^{(4)} = -1,065$, $a_4^{(4)} = 0,5$

Τάξη $i=4=p$:

$$K_4 = -a_4^{(4)} = -0,5$$

$$a_1^{(3)} = \frac{a_1^{(4)} - K_4 a_3^{(4)}}{1 - K_4^2} = 0,53$$

$$a_2^{(3)} = \frac{a_2^{(4)} - K_4 \cdot a_2^{(4)}}{1 - K_4^2} = 0,676$$

$$a_3^{(3)} = \frac{a_3^{(4)} - K_4 \cdot a_1^{(4)}}{1 - K_4^2} = -0,8$$

Τάξη $i=3$:

$$K_3 = -a_3^{(3)} = 0,8$$

$$a_1^{(2)} = \frac{a_1^{(3)} - K_3 \cdot a_2^{(3)}}{1-K_3^2} = -0,03$$

$$a_2^{(2)} = \frac{a_2^{(3)} - K_3 \cdot a_1^{(3)}}{1-K_3^2} = 0,7$$

Τάξη $i=2$:

$$K_2 = -a_2^{(2)} = -0,7$$

$$a_1^{(1)} = \frac{a_1^{(2)} - K_2 \cdot a_1^{(2)}}{1-K_2^2} = -0,1$$

Τάξη $i=1$:

$$K_1 = -a_1^{(1)} = 0,1$$

Άρα PARCOR $= (K_1, K_2, K_3, K_4) = (0,1 \,,\; -0,7 \,,\; 0,8 \,,\; -0,5)$

Άσκηση 2.2

Παρακάτω φαίνεται η υλοποίηση στο jupyter.

In [65]:

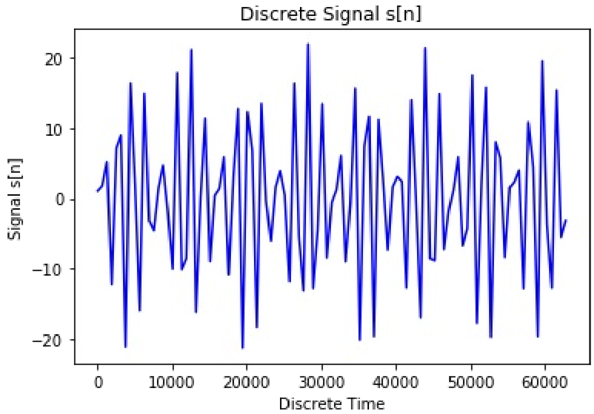
```
import numpy as np
import matplotlib.pyplot as plt
import scipy.signal
```

In [66]:

```
# Implementation of s[n] discrete signal

N = 101 # number of samples
n_s = np.linspace(0, 2*np.pi*10000, N)
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

plt.plot(n_s, s_n, color = 'blue')
plt.xlabel('Discrete Time')
plt.ylabel('Signal s[n]')
plt.title('Discrete Signal s[n]')
```

Out[66]:

```
Text(0.5, 1.0, 'Discrete Signal s[n]')
```

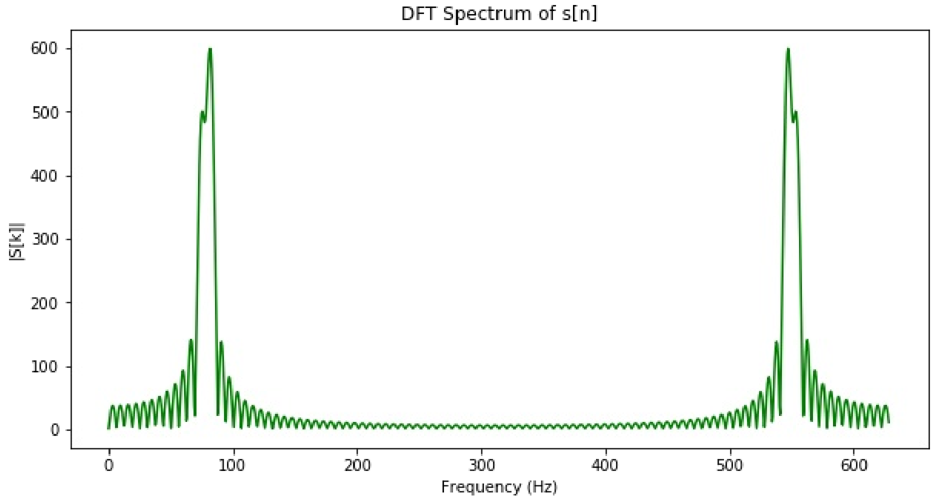


In [67]:

```
# Question (a)
# Calculation of DFT X[k] of x[n] for N>>100

N = 101 # number of samples
n_s = np.arange(N)
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

N_new = 1024 # new number of samples
n_dft = np.linspace(0, 2*np.pi*100, N_new)
s_dft = np.fft.fft(s_n, N_new)

plt.figure(figsize = (10, 5))
plt.plot(n_dft, np.abs(s_dft), color = 'green')
plt.xlabel('Frequency (Hz)')
plt.ylabel('|S[k]|')
plt.title('DFT Spectrum of s[n]')
```

Out[67]:

```
Text(0.5, 1.0, 'DFT Spectrum of s[n]')
```



In [68]:

```
# Question (b)
# Covariance Method for Linear Prediction of s[n] & LPC

# LPC factors
p = 4 # predictor's class
n_s = np.arange(-p, N) # conversion of the values of the initial signal in [-p, N-1]
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

synolo = np.zeros(shape = (p+1, p+1))
fi = np.zeros(shape = (p+1, p+1)) # array of Φ ("fi")
psi = np.zeros(p+1) # array of Ψ ("psi")

for i in range(0, p+1):
    for j in range(0, p+1):
        synolo = 0
        for k in range(p, N-1):
            synolo += s_n[k-i]*s_n[k-j]
        fi[i, j] = synolo

psi = fi[1:p+1, 0]
fi = fi[1:p+1, 1:p+1]

inv_fi = np.linalg.inv(fi) # inverse array of Φ

ak = np.zeros(p+1) # array of LPC factors
ak = inv_fi.dot(psi)

print("The LPC factors with Covariance Method are:")
for i in range(4):
    print(ak[i], sep = '\n')

print('\n')

# Polynomial of Tranfer Function with the Covariance Method

cov_pol = [1, -ak[0], -ak[1], -ak[2], -ak[3]]
roots = np.roots(cov_pol) # roots of polynomial

print("The roots of the polynomial with the Covariance Method are:")
for i in range(4):
    print(roots[i], sep = '\n')
```

```
The LPC factors with Covariance Method are:
2.827031466701328
-3.9960534568601815
2.8270314667033745
-1.0000000000015916


The roots of the polynomial with the Covariance Method are:
(0.6845471059315216+0.7289686274177213j)
(0.6845471059315216-0.7289686274177213j)
(0.7289686274191445+0.6845471059333642j)
(0.7289686274191445-0.6845471059333642j)
```

In [69]:

```python
# Question (c)
# Similar to (b) but with Autocorrelation Method using the same part of the
# initial signal of 101 points with a Hamming Window w[n]

# LPC factors
p = 4 # predictor's class
n_s = np.arange(-p, N+p) # conversion of the values of the initial signal in [-p, N+p-1]
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

# Hamming Window
win = np.hamming(N + 2*p)
s_n *= win

synolo2 = np.zeros(shape = (p+1, p+1))
fi2 = np.zeros(shape = (p+1, p+1)) # array of Φ ("fi")
psi2 = np.zeros(p+1) # array of Ψ ("psi")

for i in range(0, p+1):
    for j in range(0, p+1):
        synolo2 = 0
        for k in range(p, N+p-1):
            synolo2 += s_n[k-i]*s_n[k-j]
        fi2[i, j] = synolo2

psi2 = fi2[1:p+1, 0]
fi2 = fi2[1:p+1, 1:p+1]

inv_fi2 = np.linalg.inv(fi2) # inverse array of Φ

ak2 = np.zeros(p+1) # array of LPC factors
ak2 = inv_fi2.dot(psi2)

print("The LPC factors with Autocorrelation Method are:")
for i in range(4):
    print(ak2[i], sep = '\n')

print('\n')

# Polynomial of Tranfer Function with the Covariance Method

cov_pol2 = [1, -ak2[0], -ak2[1], -ak2[2], -ak2[3]]
roots2 = np.roots(cov_pol2) # roots of polynomial

print("The roots of the polynomial with the Autocorrelation Method are:")
for i in range(4):
    print(roots2[i], sep = '\n')
```

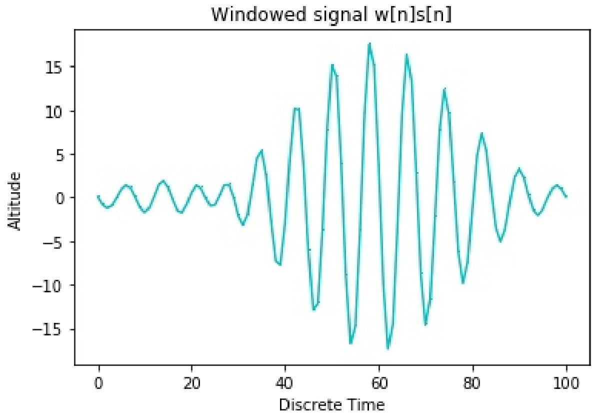
```
The LPC factors with Autocorrelation Method are:
2.81008137147046
-3.96788397503326
2.806618588426886
-0.9975195677652664


The roots of the polynomial with the Autocorrelation Method are:
(0.6718062546584069+0.7400231531877013j)
(0.6718062546584069-0.7400231531877013j)
(0.7332344310768233+0.6789163610868757j)
(0.7332344310768233-0.6789163610868757j)
```

In [61]:

```
# Depiction of Hamming Windowed Signal

n_s = np.arange(0, N)
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

win = np.hamming(N)
s_n *= win

plt.plot(n_s, s_n, marker = ',', color = 'c')
plt.xlabel('Discrete Time')
plt.ylabel('Altitude')
plt.title('Windowed signal w[n]s[n]')
```

Out[61]:

```
Text(0.5, 1.0, 'Windowed signal w[n]s[n]')
```

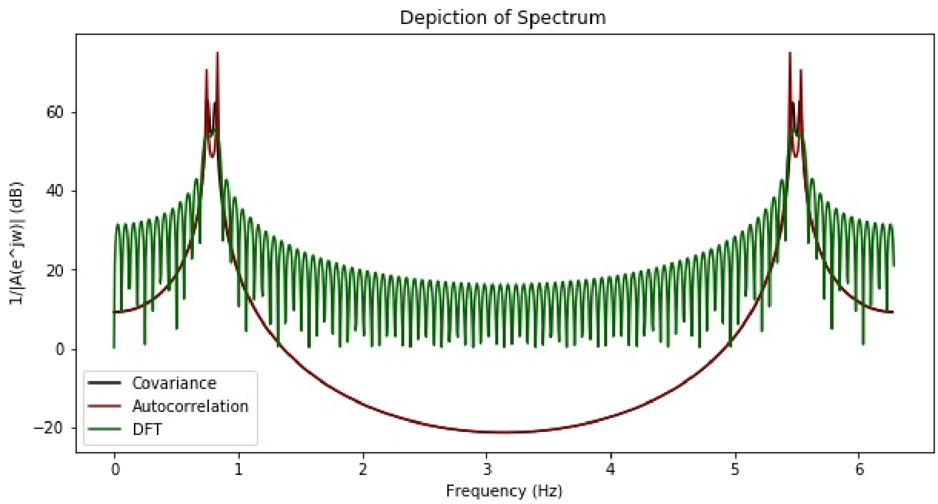


In [78]:

```
# Question (d)
# Depiction of the above approximations of spectrum

# N = 101 # number of samples
# N_new = 1024
n_s = np.arange(N)
s_n = 10*np.cos(0.24*np.pi*n_s + 0.2*np.pi) + 12*np.sin(0.26*np.pi*n_s - 0.8*np.pi)

covar,covar2 = scipy.signal.freqz([1], cov_pol, whole = True)

autoc,autoc2 = scipy.signal.freqz([1], cov_pol2, whole = True)

n_dft2 = np.linspace(0, 2*np.pi, N_new)
s_dft2 = np.fft.fft(s_n, N_new)

plt.figure(figsize = (10, 5))

plt.plot(covar, 20*np.log10(abs(covar2)), color = 'black')
plt.plot(autoc, 20*np.log10(abs(autoc2)), color = 'darkred')
plt.plot(n_dft2, 20*np.log10(abs(s_dft2)), color = 'darkgreen')

plt.xlabel("Frequency (Hz)")
plt.ylabel("1/|A(e^jw)| (dB)")
plt.legend(["Covariance", "Autocorrelation", "DFT"])
plt.title("Depiction of Spectrum")
```

Out[78]:

```
Text(0.5, 1.0, 'Depiction of Spectrum')
```



In [79]:

```
# Question (e)

# By comparing the above the spectrums of Question (d), it can be concluded
# that the approximations were able to find the real frequencies of the signal s[n].
# From the above depiction, it can also be seen that the areas around these
# two approximations are very close to the DFT Spectrum, which means that they
# are both accurate methods.
```

In [ ]:

Άσκηση 2.3

αιτιατό ΓΧΑ σύστημα : $H(z) = \dfrac{(1-0{,}8z^{-1})\cdot(1+6{,}25z^{-2})}{(1-0{,}49z^{-2})}$

α) $H_{1,min}(z) = ;$, $H_{ap}(z) = ;$ $\Rightarrow$ $H(z) = H_{1,min}(z)\cdot H_{ap}(z)$

Έχουμε: $H(z) = \dfrac{(1-0{,}8z^{-1})(1+6{,}25z^{-2})}{(1-0{,}49z^{-2})} = \dfrac{(1-0{,}8z^{-1})(1+2{,}5jz^{-1})(1-2{,}5jz^{-1})}{(1-0{,}7z^{-1})(1+0{,}7z^{-1})}$

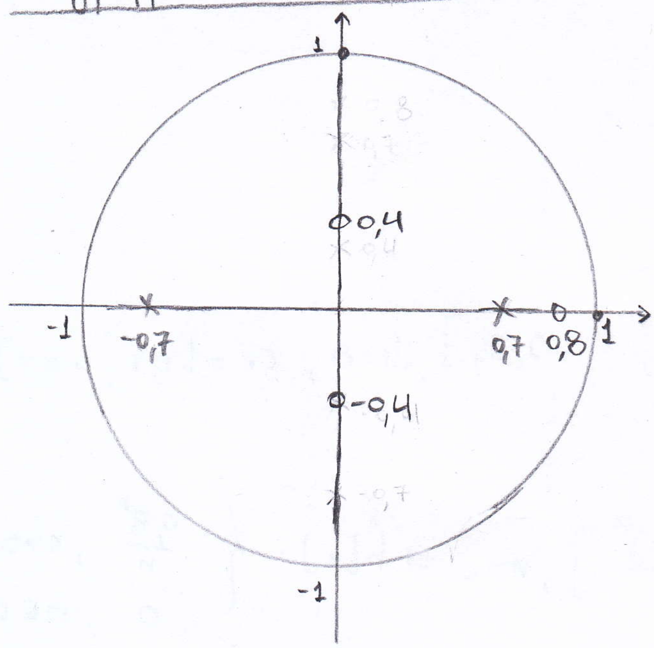
Κάθε σύστημα ελάχιστης φάσης έχει όλους τους πόλους και τα μηδενικά του μέσα στον μοναδιαίο κύκλο $|z|=1$. Για τα συστήματα all-pass ισχύει ότι οι πόλοι και τα μηδενικά είναι σε ζεύγη αντίστροφων συζυγών θέσεων.

Οπότε: $H_{1,min}(z) = \dfrac{(1-0{,}8z^{-1})(1+0{,}16z^{-2})}{(1-0{,}49z^{-2})} = \dfrac{(1-0{,}8z^{-1})(1+0{,}4jz^{-1})(1-0{,}4jz^{-1})}{(1-0{,}7z^{-1})(1+0{,}7z^{-1})}$

$\frac{1}{6{,}25} = 0{,}16$

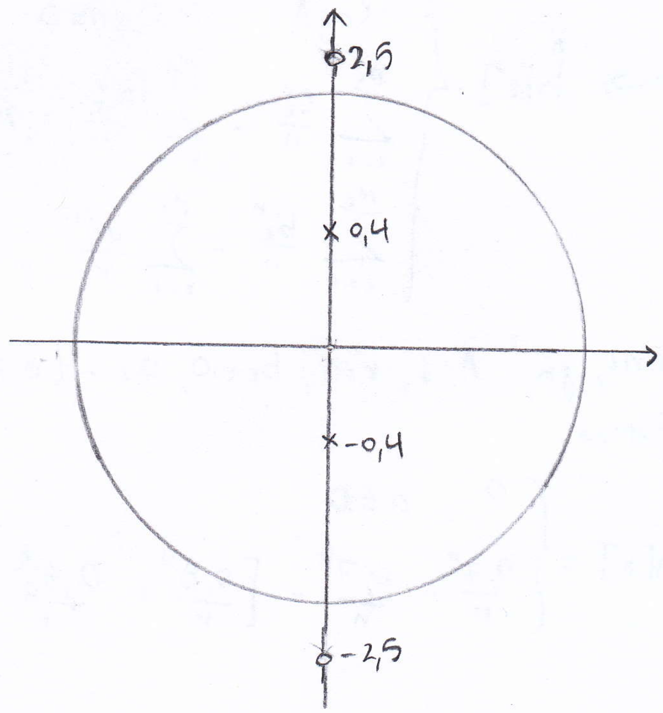
$H_{ap}(z) = \dfrac{(1+6{,}25z^{-2})}{(1+0{,}16z^{-2})} = \dfrac{(1+2{,}5jz^{-1})(1-2{,}5jz^{-1})}{(1+\underbrace{0{,}4j}_{=-b}z^{-1})(1-\underbrace{0{,}4j}_{=-a}z^{-1})}$

β) Διάγραμμα Πόλων - Μηδενικών:



$H_{1,min}(z)$

περιοχή σύγκλισης: $|z| > 0{,}7$



$H_{ap}(z)$

$|z| > |a| = |b| = 0{,}4$

γ) Ένα FIR σύστημα γενικευμένης γραμμικής φάσης έχει πόλους & μηδενικά στις θέσεις $z = 1, -1, 0, \infty$ ή σε αντίστροφα συζυγή ζεύγη. Επομένως:

$$H_2(z) = \frac{(1-0{,}8z^{-1})}{(1-0{,}49z^{-2})(1+0{,}16z^{-2})} \quad \text{και}$$

$$H_{lin}(z) = (1+0{,}16z^{-2})(1+6{,}25z^{-2})$$

Άσκηση 2.4

a) complex cepstrum $\hat{h}[n]$; & απλός cepstrum $c[n]$;

cepstrum

$$H(z) = \frac{(1-0{,}8z^{-1})(1+6{,}25z^{-2})}{(1-0{,}49z^{-2})} = \frac{(1-0{,}8z^{-1})(1+2{,}5jz^{-1})(1-2{,}5jz^{-1})}{(1-0{,}7z^{-1})(1+0{,}7z^{-1})}$$

complex cepstrum

Γενικά, $H(z) = Az^r \dfrac{\prod_{k=1}^{Mi}(1-a_kz^{-1})\prod_{k=1}^{Mo}(1-b_kz)}{\prod_{k=1}^{Ni}(1-c_kz^{-1})\prod_{k=1}^{No}(1-d_kz)}$, $|a_k|, |b_k|, |c_k|, |d_k| < 1$

$$\rightsquigarrow \hat{h}[n] = \begin{cases} \log A & , n=0 \\ \sum_{k=1}^{Ni}\frac{c_k^n}{n} - \sum_{k=1}^{Mi}\frac{a_k^n}{n} & , n>0 \\ \sum_{k=1}^{Mo}\frac{b_k^{-n}}{n} - \sum_{k=1}^{No}\frac{d_k^{-n}}{n} & , n<0 \end{cases}$$

Οπότε, για $A=1$, $r=0$, $b_k=0$, $a_k=[0{,}8 \;,\; 2{,}5j \;,\; -2{,}5j]$, $d_k=0$, $c_k=[0{,}7, -0{,}7]$ δίνεται:

$$\hat{h}[n] = \begin{cases} 0 & , n\le 0 \\ \cancel{\frac{0{,}7^n}{n}} - \cancel{\frac{0{,}7^n}{n}} - \left[\frac{0{,}8^n}{n} + \cancel{\frac{(2{,}5j)^n}{n}} - \cancel{\frac{(2{,}5j)^n}{n}}\right] & , n>0 \end{cases} \Rightarrow \hat{h}[n] = \begin{cases} -\frac{0{,}8^n}{n} & , n>0 \\ 0 & , n\le 0 \end{cases}$$

απλός cepstrum

$$c[n] = \frac{\hat{h}[n] + \hat{h}[-n]}{2}$$

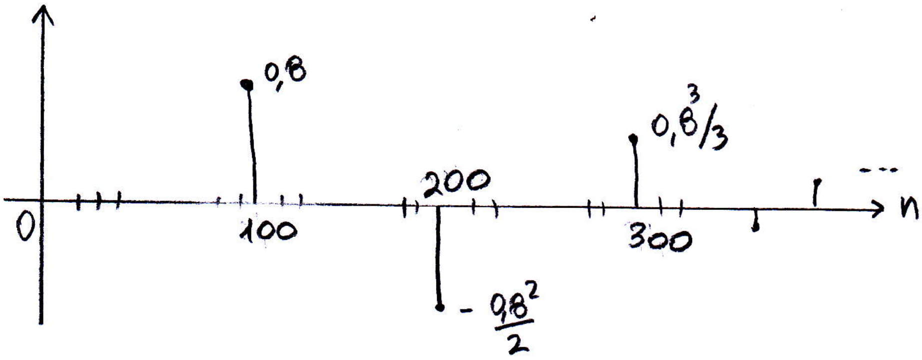
$$c[n] = \begin{cases} \frac{1}{2n}[0{,}8^{-n} - 0{,}8^n] & , n>0 \\ 0 & , n\le 0 \end{cases}$$

β) $x[n] = \delta[n] + 0{,}8\,\delta[n - N_p]$, $N_p = 100$

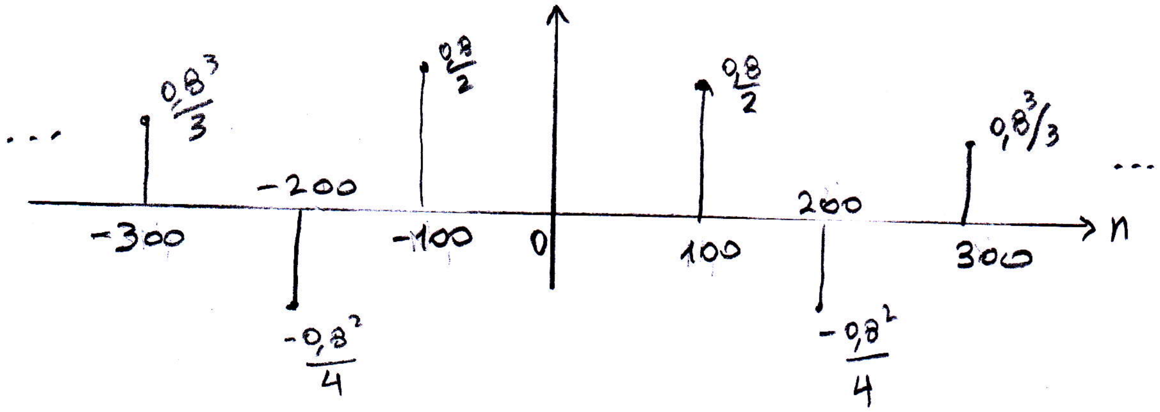
β.1) $\hat{x}[n]$;

$$X(z) = 1 + 0{,}8\,z^{-N_p} \rightsquigarrow \hat{X}(z) = \log X(z) = \sum_{n=1}^{\infty} (-1)^{n+1} \frac{0{,}8^n}{n} \cdot z^{-10n}$$

$$\hat{x}[n] = \sum_{m=1}^{\infty} (-1)^{m+1} \frac{0{,}8^m}{m} \cdot \delta[n - 10m] \quad (1)$$



β.2) $$c[n] = \frac{\hat{x}[n] + \hat{x}[-n]}{2} = \frac{1}{2} \sum_{k=1}^{\infty} (-1)^{k+1} \cdot \frac{0{,}8^k}{k} \left[\delta[n - 100k] + \delta[-n - 100k]\right]$$



β.3). $\hat{x}_p[n]$ με DFT N=500 σημείων του complex cepstrum:

$$\hat{x}_p[n] = \frac{1}{N} \sum_{k=0}^{N-1} \log(X[k]) \exp\left(j 2\pi \frac{kn}{N}\right), \quad n = 0, \dots, N-1$$

· για N=550;

• $\frac{N}{N_p} = \frac{500}{100} = 5$

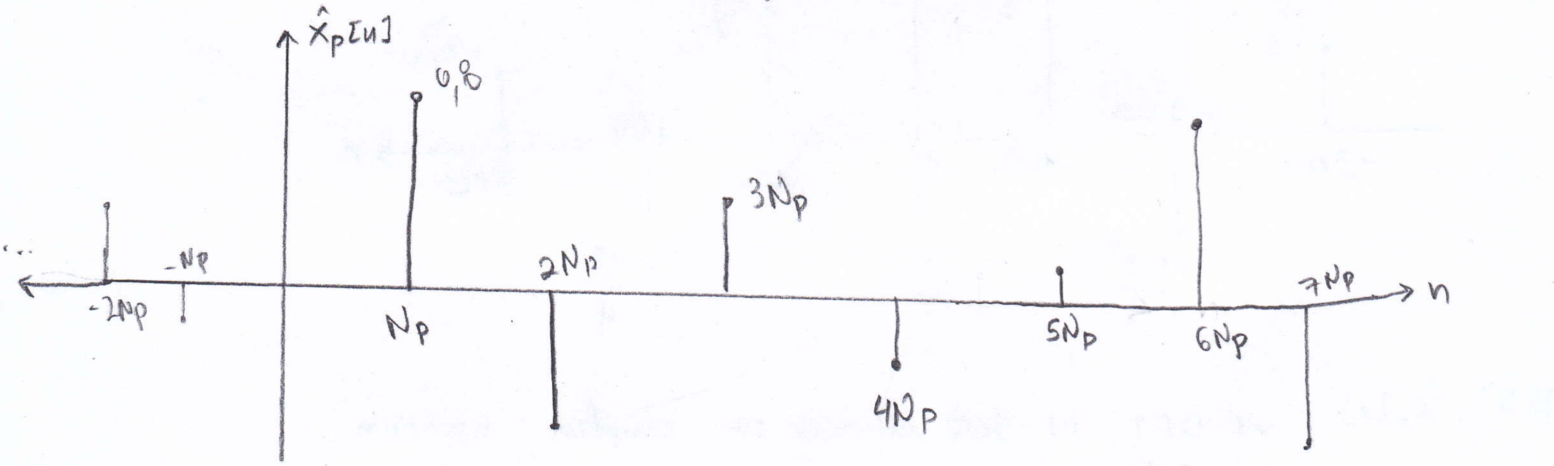
$\hat{x}_p[n] = \sum_{r=-\infty}^{+\infty} \hat{x}[n + 500r]$: περιοδική επανάληψη με περίοδο N=500 και υπέρθεση όλων των επαναλήψεων. Αυτό θα δημιουργήσει aliasing στο χρόνο.

$$\hat{x}_p[n] = \sum_{r=-\infty}^{+\infty} \hat{x}[n+500r] \xRightarrow{(1)}$$

$$\Rightarrow \hat{x}_p[n] = \sum_{r=-\infty}^{+\infty} \sum_{k=1}^{\infty} (-1)^{k+1} \cdot \frac{0,8^k}{k} \delta[n+500r-100k] \Rightarrow$$

$$\Rightarrow \hat{x}_p[n] = \sum_{k=1}^{\infty} (-1)^{k+1} \cdot \frac{0,8^k}{k} \cdot \sum_{r=-\infty}^{+\infty} \delta[n+500r-100k]$$

Παρατήρηση: Από (1), η προσέγγιση κατά DFT του Cepstrum αφορά ολίσθηση κατά $N=500$ του complex cepstrum πάνω σε όλο τον οριζόντιο άξονα. Το Cepstrum έχει μη μηδενικές τιμές στα πολλαπλάσια του $N_p=100$. Επομένως, αν το N_p δεν διαιρεί το N θα δημιουργηθούν τιμές λάθος σημείο και θα αλλοιωθεί το περιεχόμενο. Σε άλλη περίπτωση, δηλ. αν το N_p διαιρεί το N, η ολίσθηση κατά N πάνω σε όλο τον άξονα θα αλλάξει μόνο τα πλάτη των τιμών στα πολλαπλάσια του N_p. και έτσι, η πληροφορία δεν θα χαθεί. Για κάθε $\frac{N}{N_p}=5$ δείγματα, επαναλαμβάνεται:
Η γραφική παράσταση για $N_p=100$, $N=500$ είναι:



- Επομένως, για $N=550$: επειδή $N_p \nmid N$ το $\hat{x}_p[n]$ θα έχει μη-μηδενικές τιμές και εκτός από τα ακέραια πολλαπλάσια του N_p.

B.4) Όμοια με (B.3)

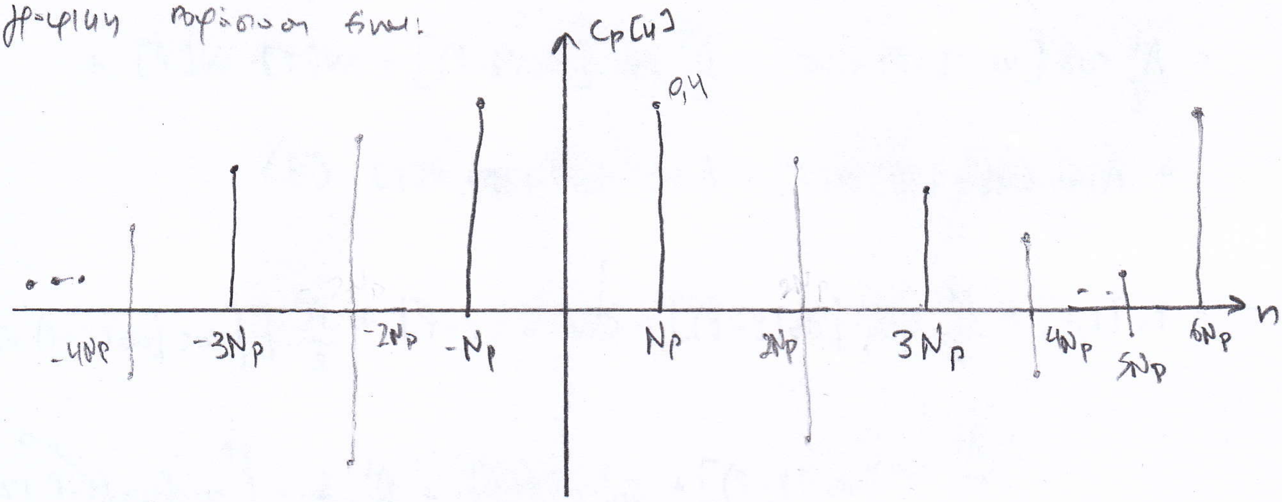
$$c_p[n] = \frac{1}{N}\sum_{k=0}^{N-1} \log|X[k]| \exp\left(j\, 2\pi \frac{kn}{N}\right), \quad n=0,\dots,N-1$$

- $\hat{c}_p[n] = \sum_{r=-\infty}^{+\infty} \hat{c}[n+500r]$

- $c_p[n] = \frac{1}{2} \sum_{k=1}^{\infty} (-1)^{k+1} \cdot \frac{0.8^k}{k} \cdot \sum_{r=-\infty}^{+\infty} \left[\delta[n-100k+500r]+\delta[-n+500r+100k]\right]$

<u>Παρατήρηση</u>: Υπάρχει ολίσθηση κατά $N=500$ πάνω σε όλο τον οριζόντιο άξονα, σε αυτή την περίπτωση του πραγματικού cepstrum. Για τους ίδιους λόγους με το (B.3), για την αποφυγή δημιουργίας λανθασμένων κορυφών, δηλ. εσφαλμένης πληροφορίας, πρέπει το N_p να διαιρεί το N.

Η γραφική παράσταση είναι:

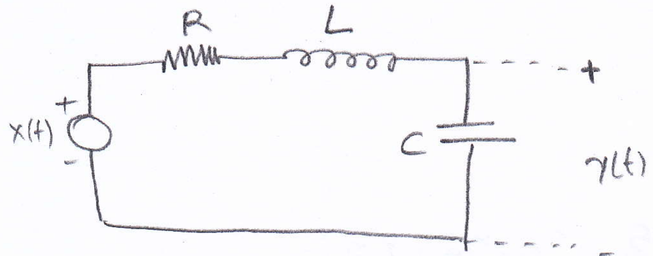


B.5) $N_p = 100$

- Θα πρέπει $N > 2 \cdot N_p$ για την αποφυγή του aliasing.

- Οι μεγαλύτερες κρουστικές αντιστοιχούν σε 2 spikes που απέχουν N_p από την αρχή των αξόνων (δεξιά & αριστερά αντίστοιχα). Όπως παρατηρήθηκε, η προσέγγιση από DFT γίνεται μέσω ολίσθησης κατά N πάνω στον οριζόντιο άξονα. Για να αποφευχθεί aliasing πρέπει το N να είναι αρκετά μεγάλο ώστε η ολίσθηση να μην πέσει πάνω από $2N_p$ και να δημιουργηθεί μεγαλύτερο spike. Συνεπώς, με άλλα λόγια θα πρέπει $N_{min} = 2N_p + 1$.

Άσκηση 2.5



α) in: $x(t)$ σήμα πηγής τάσης

out: $y(t)$ σήμα τάσης πυκνωτή

ΝΤΚ:

$$V_R(t)+V_L(t)+V_C(t)=U_n(t) \Rightarrow i(t)\cdot R + L\frac{di(t)}{dt}+V_c(t)=U_n(t)$$

Όμως, $V_c(t)=y(t)$ και $U_n(t)=x(t)$ οπότε:

$$i(t)\cdot R + L\frac{di(t)}{dt}+y(t)=x(t)$$

Επιπλέον, ισχύει: για τον πυκνωτή $i(t)=C\cdot\frac{dV_c(t)}{dt}=C\frac{dy(t)}{dt}$

Άρα, $x(t)=C\frac{dy(t)}{dt}R+L\frac{d}{dt}\left[C\frac{dy(t)}{dt}\right]+y(t)\Rightarrow$

$$\Rightarrow x(t)=CR\frac{dy(t)}{dt}+LC\cdot\frac{d^2y(t)}{dt^2}+y(t)\Rightarrow$$

$$\Rightarrow x(t)=LC\cdot\frac{d^2}{dt^2}y(t)+CR\frac{d}{dt}y(t)+y(t) \qquad (1)$$

Για $\frac{R}{L}=2$, $LC=0{,}2$ $\rightarrow$ $RC=0{,}4$

$\mathcal{L}\{(1)\}\Rightarrow X(s)=0{,}2[s^2Y(s)-sy(0)-y'(0)]+0{,}4[sY(s)-y(0)]+Y(s)$

Για μηδενικές αρχ. συνθ.: $X(s)=[0{,}2s^2+0{,}4s+1]Y(s)$

Και $H_a(s)=\frac{Y(s)}{X(s)}=\frac{1}{0{,}2s^2+0{,}4s+1}=\frac{5}{(s+1+2j)(s+1-2j)}$

$\mathcal{L}^{-1}$

$$h_a(t)=\mathcal{L}^{-1}\left\{\frac{5}{s^2+2s+5}\right\}=\frac{5}{2}e^{-t}\cdot\sin 2t$$

β) Από αναλογικό σε ψηφιακό με την μέθοδο impulse invariance ($h[n] = h_a(nT)$)

$h[n] = ;$ $\quad T=1$

▷ $h_a(t) = \frac{5}{2} e^{-t} \sin 2t$

▷ $h[n] = h_a(nT) = \frac{5}{2} e^{-nT} \sin 2nT \xrightarrow{T=1} h[n] = \frac{5}{2} \cdot e^{-n} \cdot \sin 2n \cdot u[n]$

in : $\quad x[n] = x_a(nT) = x[n]$

$\quad\quad y[n] = y_a(nT) = y[n]$

γ) $H(z) = ;$

Από (α) : $H_a(s) = \dfrac{1}{0{,}2s^2 + 0{,}4s + 1}$ και με την ανάλυση σε απλά κλάσματα, γίνεται:

(γενική μορφή : $H_a(s) = \sum_{k=1}^{N} \frac{c_k}{s - p_k}$) $\quad H_a(s) = \dfrac{5}{(s+1+2j)(s+1-2j)} \Rightarrow$

$$\Rightarrow H_a(s) = \frac{5}{4}j \frac{1}{s+1+2j} - \frac{5}{4}j \frac{1}{s+1-2j}$$

(γενική μορφή : $H(z) = \sum_{k=1}^{N} \frac{c_k}{1 - e^{p_k T} z^{-1}}$)

$$\Rightarrow H(z) = \frac{5}{4}j \frac{1}{1 - e^{-1-2j} z^{-1}} - \frac{5}{4}j \frac{1}{1 - e^{-1+2j} z^{-1}}$$

$$\Rightarrow H(z) = \frac{Y(z)}{X(z)} \Rightarrow \frac{Y(z)}{X(z)} = \frac{\frac{5}{4}j \left[e^{-(1+2j)} - e^{-(1-2j)} \right] z^{-1}}{e^{-2} z^{-2} - z^{-1} \left(e^{-(1+2j)} + e^{-(1-2j)} \right)} \Rightarrow$$

$$\Rightarrow y[n] - \left(e^{-(1+2j)} + e^{-(1-2j)} \right) y[n-1] + e^{-2} y[n-2] = \frac{5}{4} j \left(e^{-(1+2j)} - e^{-(1-2j)} \right) x[n-1]$$

δ) Απόκριση πλάτους φίλτρων;

$$H_a(j\Omega) = \frac{1}{0{,}2(j\Omega)^2 + 0{,}4j\Omega + 1} = \frac{5}{s^2 + 2s + 5}$$

Παρακάτω φαίνεται η υλοποίηση στο jupyter.

$$H(e^{jw}) = \frac{5}{4}j \frac{1}{1 - e^{-1-2j} (e^{jw})^{-1}} - \frac{5}{4}j \frac{1}{1 - e^{-1+2j} (e^{jw})^{-1}} = \frac{\frac{5}{4}j \left(e^{-(1+2j)} - e^{-(1-2j)} \right) z^{-1}}{1 - \left(e^{-(1-2j)} + e^{-(1+2j)} \right) z^{-1} + e^{-2} z^{-2}}$$

$$\rightarrow H(e^{jw}) = \frac{\frac{5}{4}j\, e^{-1} \left[e^{-2j} - e^{2j} \right] z^{-1}}{1 + e^{-1} \left[e^{2j} + e^{-2j} \right] z^{-1} + e^{-2} z^{-2}} = \frac{\frac{5}{2} e^{-1} \sin 2\, z^{-1}}{1 + 2e^{-1} \cos 2\, z^{-1} + e^{-2} z^{-2}}$$

Παρακάτω φαίνεται η υλοποίηση στο jupyter.

In [1]:

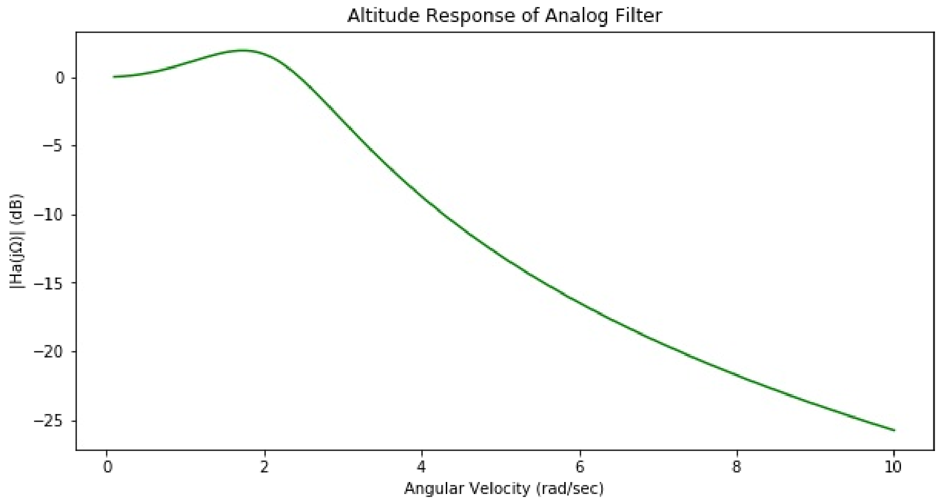
```
import numpy as np
import scipy.signal
import matplotlib.pyplot as plt
```

In [8]:

```
# Analog Filter

# factors
analog_a = [1, 2, 5]
analog_b = [0, 0, 5]
analog_w, analog_h = scipy.signal.freqs(analog_b, analog_a)

plt.figure(figsize = (10, 5))
plt.plot(analog_w, 20*np.log10(abs(analog_h)), color = 'green')
plt.xlabel('Angular Velocity (rad/sec)')
plt.ylabel('|Ha(jΩ)| (dB)')
plt.title('Altitude Response of Analog Filter')
```

Out[8]:

```
Text(0.5, 1.0, 'Altitude Response of Analog Filter')
```

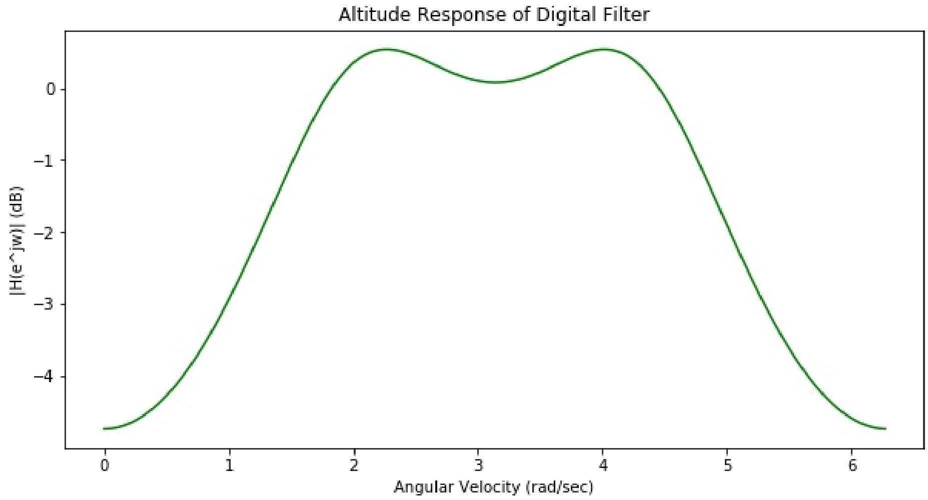


In [9]:

```
# Digital Filter

# factors
digital_a = [1, -2*np.cos(2)*np.exp(-1), np.exp(-2)]
digital_b = [0, 2.5*np.sin(2)*np.exp(-1), 0]
digital_w, digital_h = scipy.signal.freqz(digital_b, digital_a, whole = True)

plt.figure(figsize = (10, 5))
plt.plot(digital_w, 20*np.log10(abs(digital_h)), color = 'green')
plt.xlabel('Angular Velocity (rad/sec)')
plt.ylabel('|H(e^jw)| (dB)')
plt.title('Altitude Response of Digital Filter')
```

Out[9]:

```
Text(0.5, 1.0, 'Altitude Response of Digital Filter')
```



Άσκηση 2.6

α) $x[n] = A\cos(n\omega_0 + \phi) + w[n]$ όπου $w[n]$: λευκός θόρυβος μεταβλητότητας σ_w^2 κ' ασυσχέτιστος

α.1) A, ω_0: σταθ.

φ : τυχ. μεταβλ. ομοιόμ. $[-\pi, \pi]$

• $r_x[k,\ell] = E\{x[k]\, x^*[\ell]\} = E\{(A\cos(k\omega_0+\phi) + w[k])(A\cos(\ell\omega_0+\phi) + w[\ell])\} =$

$= E\{ \cancel{\frac{A^2}{2}\cos[\omega_0(k+\ell)+2\phi]} + \frac{A^2}{2}\cos[\omega_0(k-\ell)] + \cancel{w[k]\cdot w[\ell]} +$

$+ \cancel{A\cos(\omega_0 k+\phi) w[\ell]} + \cancel{A\cos(\omega_0\ell+\phi) w[k]} \}$

$= E\left\{\frac{A^2}{2}\cos(\omega_0(k-\ell))\right\} = \frac{A^2}{2}\cos[(k-\ell)\omega_0] + \sigma_w^2\,\delta[k-\ell]$

• Έλεγχος wss:

▷ $m_x[n] = E\{x[n]\} = E\{A\cos(n\omega_0+\phi) + w[n]\} = A\cdot E\{\cos(n\omega_0+\phi)\} = 0$ (σταθ.)

▷ $r_x[k,\ell] = \frac{A^2}{2}\cos[(k-\ell)\omega_0] + \sigma_w^2\,\delta[k-\ell]$ (εξαρτ. από $k-\ell$)

▷ $var[n] = \sigma_x^2[n] = E\{|x[n] - \overset{0}{m_x[n]}|^2\} = E\{x^2[n] - 2x[n]\, m_x[n] + m_x^2[n]\} =$

$= E\{x^2[n]\} - 2E\{\overset{0}{x[n]\cdot m_x[n]}\} + E\{\overset{0}{m_x^2[n]}\} =$

$= E\{A^2\cos^2(n\omega_0+\phi) + \cancel{2A\cos(n\omega_0+\phi)\cdot w[n]} + \overset{0}{w^2[n]}\} =$

$= E\left\{\frac{A^2}{2}\cancel{\cos(2n\omega_0+2\phi)}\right\} + \frac{1}{2} = \frac{1}{2} < \infty$ (πεπερασμένο)

Άρα είναι wss.

• Φάσμα Ισχύος:

$P_x(e^{j\omega}) = DTFT\{r_x[k]\} = \sigma_w^2 + \frac{\pi A^2}{2}\sum_{r=-\infty}^{+\infty}[\delta(\omega-\omega_0-2\pi r) + \delta(\omega+\omega_0-2\pi r)]$

α.2) A, φ: σταθ.

ω_0: τυχ. μεταβλ. ομοιομ. $[\omega_0-\Delta, \omega_0+\Delta]$

• $r_x[k,\ell] = E\{x[k]\cdot x^*[\ell]\} = E\{(A\cos(k\omega_0+\varphi)+w[k])(A\cos(\ell\omega_0+\phi) + w[\ell])\} =$

$= E\{A^2\cos(k\omega_0+\phi)\cos(\ell\omega_0+\phi)\} =$

$= \frac{1}{2}A^2 E\{\cos(k-\ell)\omega_0\} + \frac{1}{2}A^2\cdot E\{\cos[(k+\ell)\omega_0 + 2\phi]\} =$

$= \frac{1}{2}A^2\cdot\frac{1}{\Delta}\cdot\sum_{m=\omega_0-\Delta}^{\omega_0+\Delta}\cos(k-\ell)\cdot m + \frac{1}{2}A^2\cdot\frac{1}{\Delta}\sum_{m=\omega_0-\Delta}^{m_0+\Delta}\cos[(k+\ell)m+2\phi] = 0 + 0 = 0$

για $\Delta = (2k+1)\frac{\pi}{2}$

• Έλεγχος WSS:

Το $r_x[k,l]$ δεν εξαρτάται από τη διαφορά $k-l$.

Δεν είναι WSS.

β) Bernoulli ανέλιξη : $x[n] = \begin{cases} 0 & , 1-p \text{ πιθ.} \\ 1 & , p \text{ πιθ} \end{cases}$

$P_r(x[n]=1)=p$

$P_r(x[n]=0)=1-p \quad , 0<p<1$

• για $k \neq l$:

$$r_x[k,l] = E\{x[k]\cdot x^*[l]\} = E\{x[k]\}\cdot E\{x[l]\} = (1\cdot p + 0(1-p))\cdot(1\cdot p + 0(1-p)) = p^2$$

για $k=l$:

$$r_x[k,l] = E\{x[k]\cdot x[k]\} = E\{x^2[k]\} = [1^2\cdot p + 0^2(1-p)] = p$$

• Έλεγχος WSS.

▷ $m_x[n] = E\{x[n]\} = 1\cdot p + 0(1-p) = p$ (σταθ.)

▷ $r_x[k,l] = \begin{cases} p^2 & , k\neq l \\ 0 & , k=l \end{cases}$ (εξαρτ. από $k-l$)

▷ $var[x[n]] = \sigma_x^2[n] = E\{|x[n]-p|^2\} = E\{x^2[n]\} - 2E\{x[n]\cdot p\} + E\{p^2\} =$

$= p^2 - 2p + p^2 = p(1-p) < \infty$ (πεπερ.)

Άρα είναι WSS.

• Φάσμα Ισχύος:

$$P_x(e^{j\omega}) = \sum_{k=-\infty}^{+\infty} r_x[k]\cdot e^{-jk\omega} = (r_x[0] + r_x[1]e^{-j\omega} + \dots) = \quad r_x[x] = \begin{cases} p^2 & , x\neq 0 \\ p & , x=0 \end{cases}$$

$$= p + p^2 e^{-j\omega} + p^2 e^{-2j\omega} + \dots =$$

$$= p + p^2\left[\sum_{n=0}^{\infty} e^{-j\omega n} - 1\right] = p + p^2\left[\frac{1}{1+e^{-j\omega}} - 1\right]$$

δ) Τυχαίος περίπατος

$X[n] = X_1 + X_2 + \dots + X_n \quad , \forall n \geq 1$

$E[X_i] = 0$

$E[X_i^2] = s^2$

- $k \neq l$:

$r_x[k,l] = E\{x[k] \cdot x^*[l]\} = E\left\{\sum_{i=1}^{k} x_i\right\} \cdot E\left\{\sum_{i=1}^{l} x_i\right\} =$

$= \sum_{i=1}^{k} E\{x_i\} \cdot \sum_{i=1}^{l} E\{x_i\} = 0 \qquad$ (σταθ.)

$k = l$:

$r_x[k,l] = E\{x[k] \cdot x^*[k]\} = E\{x^2[k]\} = E\left\{\left(\sum_{i=1}^{n} x_i\right)^2\right\} = ns^2$

- Έλεγχος wss :
  - ▷ $m_x[n] = E\left\{\sum_{i=1}^{n} x_i\right\} = 0 \qquad$ (σταθ)
  - ▷ $r_x[k,l]$ υπολογίσαμε παραπάνω.
  - ▷ $var[x[n]] = \sum_{i=1}^{n} var[x_i] = n \cdot s^2 \qquad$ (μη σταθερ.)

Δεν είναι wss.