Lezione n.3 - Soluzioni

Laboratorio di Telecomunicazioni

L.Verdoliva

1 Calcolo media, energia e potenza

- a) *Media*. Di seguito trovate lo script per il calcolo della media nel caso in cui $a = 0.9$ nell'intervallo $(-10, 10)$ e la verifica che il risultato coincide con l'espressione analitica:

```
% script per il calcolo della media in un intervallo
% del segnale esponenziale monolatero

a = 0.9;
N = 10;
n = [-N:N];
x = (a.^n).*(n>=0);
media1 = mean(x);
media2 = (1/(2*N+1))*(1-a^(N+1))/(1-a);
check = media1-media2
```

- b) *Segnale rampa*. Script per il calcolo dell'energia di un segnale con andamento a rampa e potenza di corrispondente segnale periodico.

```
nx = [-19:20];
x = n.*(n>=0);
figure; stem(nx,x);
xlabel('nx'); ylabel('x'); title('Rappresentazione del segnale');
Ex = sum(x.^2);
y = [x x];
ny = [-19:60];
figure; stem(ny,y);
xlabel('ny'); ylabel('y'); title('Segnale periodico');
Px = Ex/20;
```

- c) *Segnale esponenziale*. Verifichiamo che al crescere di N l'energia di un segnale esponenziale monolatero tende a $\frac{1}{1-a^2}$:

```

clear all; close all; clc;
a = input('a=');
N = [0:60];
for i = 1:length(N),
    n = [-N(i):N(i)];
    x = (a.^n).*(n>=0);
    E(i) = sum(x.^2);
end
stem(N,E); xlabel('N'); ylabel('Energia');
E_inf1 = E(length(N));
E_inf2 = 1/(1-a^2);
check = E_inf1 - E_inf2

```

d) *Segnale alternato*. Se un segnale periodico ha generatore $x_g(n)$ la funzione che valuta la potenza è:

```

function P = potenza(xg);
% Calcolo della potenza P per un segnale periodico con generatore xg
% P = potenza(xg)

E = sum(xg.^2);
P = E/(length(xg));

```

Scriviamo uno script in cui richiamiamo tale funzione per il calcolo della potenza dei seguenti segnali $x(n) = \cos(\pi n/4)$ e di $x(n) = \text{rep}_8[\mathcal{R}_4(n)]$, entrambi periodici di periodo 8:

```

n = [0:7];
x1 = cos(pi*n/4);
P1 = potenza(x1)
x2 = [ones(1,4) zeros(1,4)];
P2 = potenza(x2)

```

e) *Energia mutua*. Utilizziamo la formula $E_{x+y} = E_x + E_y + 2E_{xy}$ per calcolare l'energia di $z(n) = x(n) + y(n)$ dove $x(n) = \mathcal{R}_9(n)$ e $y(n) = (1 - |n - 5|/5)\mathcal{R}_{11}(n)$.

```

clear all; close all; clc;
% Generazione di x e y
nx = [0:8];
x = ones(1,9);
ny = [0:10];
y = (1-abs(ny-5))/5;

```

```

% Calcolo dell'energia di x e y
Ex = sum(x.^2);
Ey = sum(y.^2);

% Calcolo dell'energia Exy, energia del segnale prodotto
nz = [min(min(nx),min(ny)): max(max(nx),max(ny))];
z1 = zeros(1,length(nz));
z2 = z1;
z1(find((nz>=min(nx)) & (nz<=max(nx)))) = x;
z2(find((nz>=min(ny)) & (nz<=max(ny)))) = y;
xy = z1.*z2;
Exy = sum(xy);

% Energia complessiva
E = Ex + Ey +2*Exy

```

Per verificare la correttezza del risultato possiamo prima sommare i due segnali e poi calcolare l'energia del segnale somma:

```

[z,nz] = somma(x,nx,y,ny);
E = sum(z.^2)

```

2 Funzioni di auto/mutua correlazione

- a) *Mutua correlazione di due segnali esponenziali.* La funzione per il calcolo della mutua correlazione è la seguente:

```

function [r,nr] = mutuacorr(x,nx,y,ny);
% calcola la correlazione fra due vettori
% [r,nr] = mutuacorr(x,nx,y,ny);
r = xcorr(x,y);
nri = nx(1)-ny(length(ny));
nrf = nx(length(nx))-ny(1);
nr = [nri:nrf];
figure; stem(nr,r);

```

Usiamo questa funzione per calcolare e diagrammare la funzione di mutua correlazione dei seguenti segnali

$$\begin{aligned}
 x(n) &= (0.9)^n & 0 \leq n \leq 20 \\
 y(n) &= (0.8)^{-n} & -20 \leq n \leq 0
 \end{aligned}$$

```

nx = [0:20];
x = (0.9).^nx;
ny = [-20:0];
y = (0.8).^(-ny);
[r,nr] = mutuacorr(x,nx,y,ny); stem(nr,r);

```

Fate attenzione all'uso della funzione `xcorr` quando i segnali hanno diversa lunghezza. Infatti in tal caso vengono resi di uguale lunghezza aggiungendo degli zeri per cui anche $R_{xy}(m)$ risulta essere estesa con degli zeri a destra o a sinistra. La funzione per il calcolo della mutua correlazione va modificata come segue:

```

function [r,nr] = mutuacorr(x,nx,y,ny);

r = xcorr(x,y);
Delta = length(nx)-length(ny);
if Delta > 0
    r = r(Delta+1:length(r));
else r = r(1:length(r)+Delta);
end
nri = nx(1)-ny(length(ny));
nrf = nx(length(nx))-ny(1);
nr = [nri:nrf];
figure; stem(nr,r);

```

b) *Applicazione 2.*

```

clear all; close all; clc;
nx = [1:200];
x = [ones(1,20) zeros(1,180)];
Lx = length(x);
nrx = [1-Lx:Lx-1];
rx = xcorr(x);
figure; stem(nrx,rx);

alpha = 0.1; k = 50;
nxt = nx + k;
xt = alpha * x;
[y,ny] = somma(x,nx,xt,nxt);
Lr = length(y);
nr = [1-Lr:Lr-1];
r = xcorr(y);
figure; stem(nr,r);

```