

Lezione n.4 - Soluzioni

Laboratorio di Telecomunicazioni

L.Verdoliva

1 Sistemi LTI

Di seguito è riportata la funzione che realizza l'operazione di convoluzione:

```
function [y,ny] = convoluzione(x,nx,h,nh);  
% [y,ny] = convoluzione(x,nx,h,nh);  
% funzione che calcola la convoluzione tra x e h  
% e restituisce il risultato in y  
Lx = length(x);  
Lh = length(h);  
ny = [nx(1)+nh(1):nx(Lx)+nh(Lh)];  
y = conv(x,h);
```

a) *Convoluzione tra due finestre rettangolari.*

```
N = 6;  
x = boxcar(N);  
nx = [0:N-1];  
M = [6:2:14];  
for i = 1:length(M),  
    h = boxcar(M(i));  
    nh = [0:M(i)-1];  
    [y,ny] = convoluzione(x,nx,h,nh);  
    figure; stem(ny,y);  
    title(sprintf('convoluzione N = 6, M = %d',M(i)));  
    pause;  
end
```

b) *Convoluzione tra un gradino e una sequenza esponenziale.*

```
nx = [0:30]; x = (nx>=0);
nh = nx; h = ((1/2).^nh);
[y,ny] = convoluzione(x,nx,h,nh);
figure; stem(ny,y);
check = y(length(nx)) - 1/(1-1/2)
```

c) *Convoluzione tra sequenze di durata finita.*

```
nx = [0:4]; x = ones(1,5);
nh = [0:6]; h = 2.^nh;
[y,ny] = convoluzione(x,nx,h,nh);
figure; stem(ny,y);
```

d) *Sistemi FIR.* Consideriamo il segnale $x(n) = 4\mathcal{R}_{11}(n) + 12\mathcal{R}_5(n)$ e determiniamo l'uscita con un sistema avente risposta impulsiva $h_1(n) = \frac{1}{2}\delta(n) + \frac{1}{2}\delta(n-1)$, quindi confrontiamo il risultato con l'uscita ottenuta da un sistema con $h_2(n) = \frac{1}{2}\delta(n) - \frac{1}{2}\delta(n-1)$.

```
nx = [0:15];
x = [4*ones(1,11) 12*ones(1,5)];
subplot(311); stem(nx,x); axis([-3 18 0 15]);
nh = [0:1];
h = [1/2 1/2];
[y,ny] = convoluzione(x,nx,h,nh);
subplot(312); stem(ny,y); axis([-3 18 0 15]);
h = [1/2 -1/2];
[y,ny] = convoluzione(x,nx,h,nh);
subplot(313); stem(ny,y); axis([-3 18 -13 10]);
```

Notate come l'uscita del primo sistema sia una versione più grossolana del segnale (le discontinuità sono state smussate), mentre l'uscita del secondo sistema presenti una risposta diversa da zero solo in corrispondenza delle discontinuità (tanto più forte quanto più il salto è grande).

2 Sistemi ARMA

a) *Sistema AR.*

```
n = [-10:30]; x = [(n == 0)];
a = [1 -2/3]; b = [1]; h = filter(b,a,x);
subplot(211); stem(n,x); title('Ingresso x(n)');
subplot(212); stem(n,h); title('Risposta impulsiva');
```

b) *Sistema ARMA.*

```
n = [0:50]; x = [(n == 0)];
a = [1 -2/3]; b = [1 1/2]; h = filter(b,a,x);
subplot(211); stem(n,h); title('Risposta impulsiva');
y = (2/3).^n + (1/2)*((2/3).^(n-1)).*(n>=1);
subplot(212); stem(n,y); title('Verifica');
```

c) *Filtraggio di un'interferenza sinusoidale.*

```
n = [0:40];
s = 5*(n>=0); % segnale costante
i = cos(pi*n); % interferenza
x = s + i;
subplot(211); stem(n,x); axis([-2 42 0 10]);
title('Segnale con interferenza');
nh = [0 1]; h = [1/2 1/2];
[y,ny] = convoluzione(x,n,h,nh);
subplot(212); stem(ny,y); axis([-2 42 0 10]);
title('Segnale filtrato');
```

Notate che solo ai bordi del segnale filtrato è presente una lieve distorsione a causa del fatto che non è possibile filtrare perfettamente una sinusoidale (che ha durata teoricamente infinita) da un segnale costante (anch'esso di durata infinita) quando si opera con segnali di durata finita.

```
n = [0:40];
s = (1.02).^n; % segnale utile
subplot(311); stem(n,s); axis([-2 42 0 5]);
i = (1/2)*cos(pi*n/4 + pi/4); % interferenza
x = s + i;
subplot(312); stem(n,x); axis([-2 42 0 5]);
title('Segnale con interferenza');
M = [2:10];
for k = 1:length(M),
    h = (1/M(k))*boxcar(M(k));
    nh = [0:M(k)-1];
    [y,ny] = convoluzione(x,n,h,nh);
    subplot(313); stem(ny,y); axis([-2 42 0 5]);
    title(sprintf('Segnale filtrato con M = %d',M(k)));
    pause;
end
```

Notate come all'aumentare di M il filtro riesca a rimuovere meglio l'interferenza, sebbene il segnale risulti attenuato.