

Corso di Segnali e Sistemi
Ingegneria Biomedica-Ingegneria Elettronica
Università degli Studi di Padova
(Proff. C. Dalla Man e T. Erseghe)

Laboratorio 03

- Convoluzione di segnali a supporto limitato a tempo discreto e continuo

Richiami di Teoria

Convoluzione a tempo discreto

Consideriamo due segnali $x[n]$ e $y[n]$ a tempo discreto.

Dalla teoria sappiamo che la loro convoluzione è definita come:

$$x[n] * y[n] = \sum_{k=-\infty}^{+\infty} y[n-k] x[k]$$

Nella pratica non si può eseguire la sommatoria da $-\infty$ a $+\infty$!...
ma è anche vero che nella pratica tutti i segnali sono a supporto limitato!

(Si ricorda che se x è diverso da zero in N campioni ed y è diverso da zero in M campioni, $x*y$ è diverso da zero in $N+M-1$ campioni).

Richiami di Teoria

Convoluzione a tempo continuo

Consideriamo i segnali a tempo continuo $f(t)$ e $g(t)$.

La convoluzione a tempo continuo è definita da:

$$f * g(t) = \int_{-\infty}^{+\infty} f(t - \tau)g(\tau)d\tau$$

Attenzione: in Matlab si può solo approssimare un segnale a tempo continuo scegliendo il passo di campionamento sufficientemente piccolo.

Scegliendo il passo di campionamento **T** sufficientemente piccolo, possiamo scrivere

$$f(t) * g(t) = \int_{-\infty}^{+\infty} f(t - \tau)g(\tau)d\tau \approx T \sum_{k=-\infty}^{+\infty} f(nT - kT)g(kT)$$

e ricondurci al caso a tempo discreto

Esercizio 1

Consideriamo due segnali $a[n]$ e $b[n]$ a tempo discreto a supporto limitato.

$$a[n] = \begin{cases} -1 & n = 0 \\ 3 & n = 1 \\ -5 & n = 2 \\ 2 & n = 3 \\ 0 & \text{altrove} \end{cases} \quad b[n] = \begin{cases} 1 & n = 0 \\ 2 & n = 1 \\ -1 & n = 2 \\ 0 & \text{altrove} \end{cases}$$

- Calcolare la convoluzione discreta $c[n]=(a*b)[n]$ e plottarla

Per calcolare la convoluzione discreta $c[n]=(a*b)[n]$ si può usare il comando *conv*

CONV Convolution and polynomial multiplication.

C = CONV(A, B) convolves vectors A and B. The resulting vector is length MAX([LENGTH(A)+LENGTH(B)-1,LENGTH(A),LENGTH(B)]).

Traccia di Soluzione

```
%% convoluzione
```

```
c = conv (a,b);
```

Esegue la convoluzione tra a e b

```
%% Grafico della convoluzione c = a*b
```

```
figure
```

```
n= ... ;
```

```
...
```

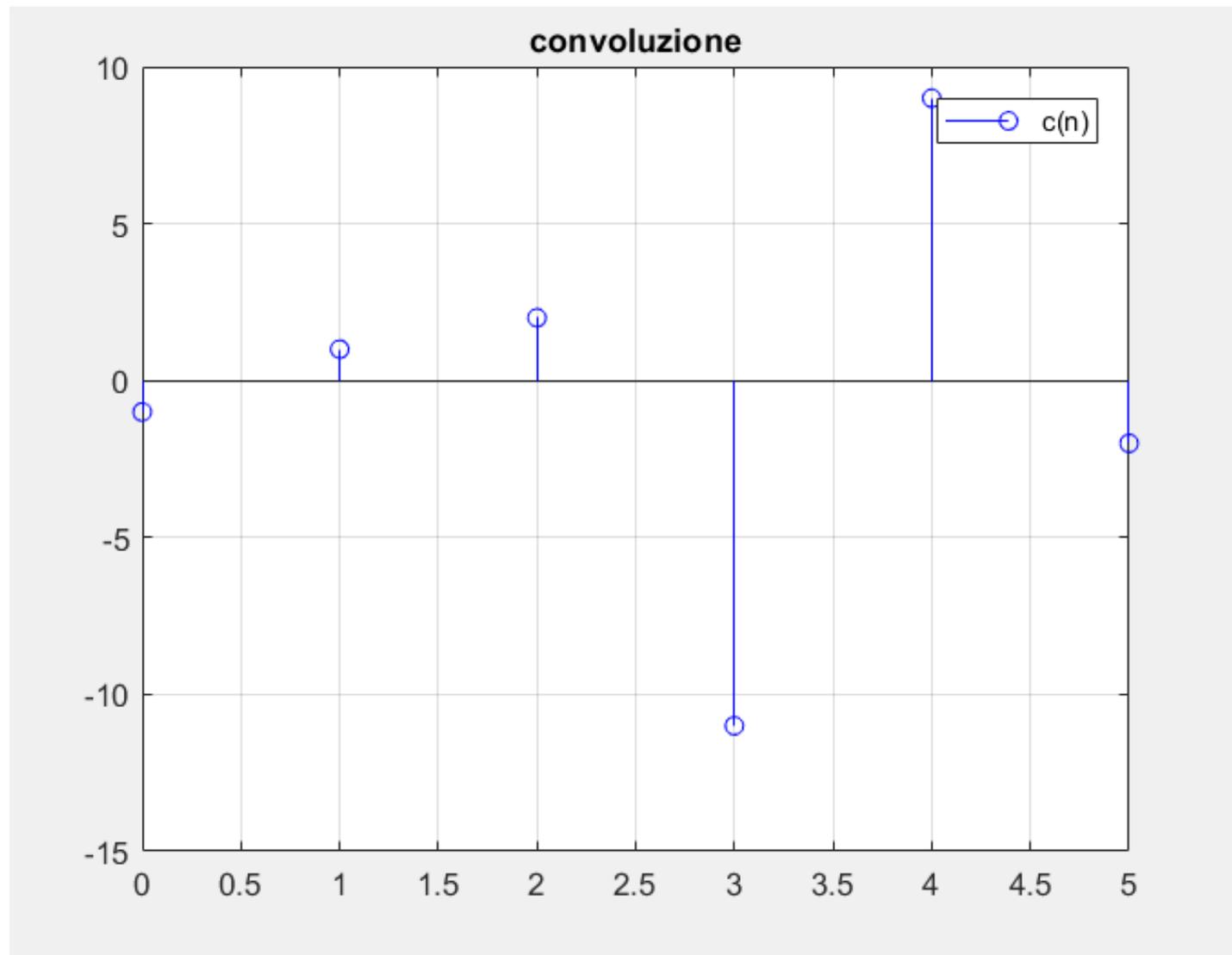
```
...
```

```
...
```

```
...
```

Per plottare correttamente $c(n)$
Devo costruire opportunamente l'asse dei tempi.
Sfrutto la relazione tra i supporti dei segnali nota dalla teoria:
Se il supporto di $\mathbf{a(n)}$ è $[n_1 \ n_2]$
e il supporto di $\mathbf{b(n)}$ è $[n_3 \ n_4]$, il supporto di $\mathbf{c(n)}$ è?

Soluzione



Esercizio 2

Dati i segnali a tempo continuo (per $t \in [-5,5]$):

$$f(t) = A_f \cdot \text{rect}\left(\frac{t-t_f}{D_f}\right) \quad g(t) = A_g \cdot \text{rect}\left(\frac{t-t_g}{D_g}\right)$$

con $A_f=A_g=1$; $t_f=t_g=1$; $D_f=D_g=1$

- Plottare i segnali $f(t)$ e $g(t)$
- Calcolare e plottare la convoluzione tra i due segnali

Ricordando che, scegliendo un passo di campionamento T **sufficientemente piccolo**, possiamo scrivere:

$$f(t) * g(t) = \int_{-\infty}^{+\infty} f(t-\tau)g(\tau)d\tau \approx T \sum_{k=-\infty}^{+\infty} f(nT - kT)g(kT)$$

Possiamo ricondurci al caso discreto, e possiamo quindi utilizzare il comando *conv* di Matlab visto nell'esercizio

precedente....**ATTENZIONE: RICORDARSI DI MOLTIPLICARE PER T!!!**

Per costruire i segnali «rect», usare la function **rectpuls**:

rectpuls Sampled aperiodic rectangle generator.

`rectpuls(T)` generates samples of a continuous, aperiodic, unity-height rectangle at the points specified in array `T`, centered about `T=0`. By default, the rectangle has width 1. Note that the interval of non-zero amplitude is defined to be open on the right, i.e., `rectpuls(-0.5)=1` while `rectpuls(0.5)=0`.

`rectpuls(T,W)` generates a rectangle of width `W`.

Traccia di Soluzione

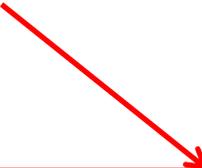
```
%% Segnali di interesse
clc
close all
% Parametri che definiscono i segnali
Af = 1;
Ag = 1;
tf = 1;
tg = 1;
Df = 1;
Dg = 1;
% Approssimazione discreta
T= ...
t=-5:T:5;
f= ...
g= ...

%% Grafici
subplot(211)
plot(t,f,'LineWidth',2);
legend('f')
xlabel('t')
ylabel('f(t)')
grid on

subplot(212)
plot(t,g,'LineWidth',2);
legend('g')
xlabel('t')
ylabel('g(t)')
grid on
```

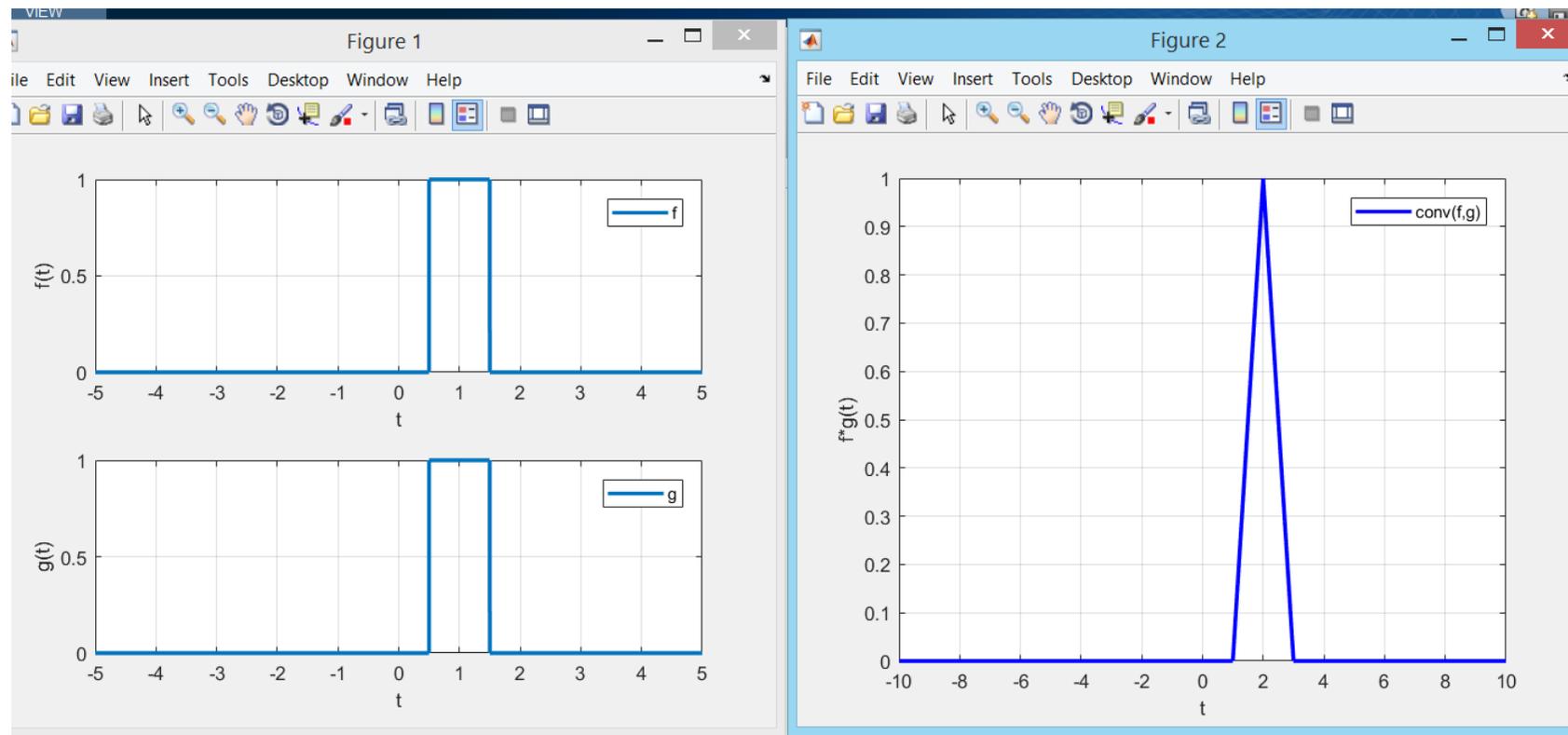
Traccia di Soluzione

```
%% Convoluzione ( approssimazione discreta )
% Attenzione a moltiplicare per il passo di campionamento
conv_fg= ...
t_conv= ...
figure
plot(t_conv,conv_fg,'b','LineWidth',2)
legend('conv(f,g)')
xlabel('t')
ylabel('f*g(t)')
grid on
```



Per plottare correttamente `conv_fg` devo costruire opportunamente l'asse dei tempi `t_conv`.
Ai vettori `f` e `g`, dati in ingresso alla function `conv`, corrisponde l'asse dei tempi `t`.
Sfruttando la relazione tra i supporti dei segnali nota dalla teoria,

Traccia di Soluzione



Esercizio 3

Si consideri il sistema LTI che descrive la cinetica dell'ormone C-peptide. La risposta impulsiva è la somma di due esponenziali:

$$h(t) = Ae^{-at} + Be^{-bt}$$

con $A=0.76$, $B=0.24$, $a=0.14 \text{ min}^{-1}$, $b=0.02 \text{ min}^{-1}$

Il file `lab3_es3_data.mat` contiene la secrezione di C-peptide (in pmol/l/min) campionata con passo $T=1 \text{ min}$ in $[0,420] \text{ min}$ (l'ingresso del sistema).

- Simulare l'uscita del sistema, cioè la concentrazione di C-peptide (in pmol/l) [in \[0,420\] min.](#)
- Simulare l'uscita del sistema quando $b=0.2$ e $b=0.002$ (sovrapporre i grafici con il precedente)

Traccia di Soluzione

```
clear all
clc
close all
%
load lab3_es3_data
A=0.76;
B=0.24;
a=0.14;
b=0.02;
passo=1;
t=[0:passo:tSR(end)]';
h=...
interpSR=interp1(tSR, SR,t);
Cpep=passo*conv(interpSR,h);
Cpep=Cpep(1:length(t)); %% prende solo i primi elementi del vettore corrispondenti
```

Costruisco un asse dei tempi con passo =1.
Osservando la risposta del sistema in [0 420], si può considerare 1 sufficientemente piccolo da approssimare il sistema con un sistema a tempo continuo

Definisco la risposta impulsiva sull'asse dei tempi

Interpolo i dati di secrezione sullo stesso asse dei tempi in modo da poter eseguire la convoluzione

Eseguo la convoluzione e mi limito a considerare l'uscita del sistema corrispondente ai tempi in t

Traccia di Soluzione

