



UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

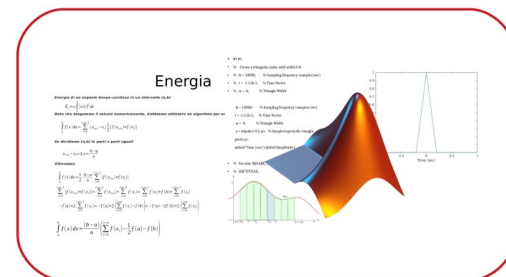
Programmazione Informatica

Ingegneria Civile e Ambientale
Ingegneria Navale

Matlab 6 – Funzioni nel dominio del tempo e della
frequenza. Analisi ed elaborazione dei segnali.

Antonio Fiumara
antonio.fiumara@dia.units.it

A. A. 2025-26





Struttura del Corso

Gli argomenti trattati nel corso saranno:

Lezione Teoria 1 – Introduzione, presentazione del corso, codifica dell'informazione

Lezione Teoria 2 – Rappresentazione dell'informazione in un computer

Lezione Teoria 3 – Algebra booleana, introduzione e applicazioni nella programmazione

Lezione Teoria 4 – Architettura dei calcolatori. HW, SW, Sistemi Operativi

Lezione Teoria 5 – Algoritmi e strutture dati - Implementazione di algoritmi notevoli (ordinamento , ricerca, ecc)

Lezione Programmazione - Matlab 1 – Introduzione, installazione, comandi di base, matrici

Lezione Programmazione - Matlab 2 – Input/Output (I/O), strutture di controllo

Lezione Programmazione - Matlab 3 – Funzioni, ciclo for

Lezione Programmazione - Matlab 4 – Grafici. Sistemi Lineari, Algebra lineare, Analisi matematica

Lezione Programmazione - Matlab 5 – Strutture dati avanzate. Assegnazione esercitazione di gruppo n.1.

Lezione Programmazione - Matlab 6 – Funzioni nel dominio del tempo e della frequenza. Analisi ed elaborazione dei segnali.

Lezione Programmazione - Matlab 7 – Applicazioni ingegneristiche avanzate. Assegnazione esercitazione di gruppo n.2

Lezione Programmazione - Matlab 8 – Correzione esercitazioni di gruppo. Ripasso e approfondimenti



Introduzione

Numeri complessi in Matlab

Segnali

Serie di Fourier

Trasformata di Fourier

FFT

Spettro di un segnale



Numeri complessi in Matlab

Definizioni e terminologia

Dato un numero complesso

$$z = x + jy \quad z \in \mathbb{C}, \quad x, y \in \mathbb{R}$$

x è la **parte reale** di z , y è la **parte immaginaria** di z .

Si definisce complesso coniugato di z e si indica con z' , il numero complesso che ha la stessa parte reale di z e l'opposto della parte immaginaria di z .

$$z' = x - jy$$

Modulo di z

$$|z| = |x + jy| = \sqrt{x^2 + y^2}$$

Angolo o fase di z

$$\phi(z) = \arctan\left(\frac{y}{x}\right)$$

In Matlab

$$x = \text{real}(z)$$

$$y = \text{imag}(z)$$

$$|z| = \text{abs}(z)$$

$$\phi(z) = \text{angle}(z)$$

$$z' = \text{conj}(z) \quad \text{o anche} \quad z'$$



Numeri complessi in Matlab

In MATLAB®, **i** e **j** rappresentano l'unità immaginaria di base.

Possono essere utilizzati per creare numeri complessi come

5+2j

È inoltre possibile determinare le parti reali e immaginarie dei numeri complessi e calcolare altri valori comuni, come il modulo e la fase (angolo).

Funzioni

abs Valore assoluto e magnitudine complessa

angle Angolo di fase

complex Crea un complex array

conj Coniugato complesso

cplxpair Sort complex numbers into complex conjugate pairs

i j unità immaginaria

imag Parte immaginaria di un numero complesso

isreal Determina se un array contiene numeri complessi

real Parte reale di un numero complesso

sign Funzione segno (funzione signum)

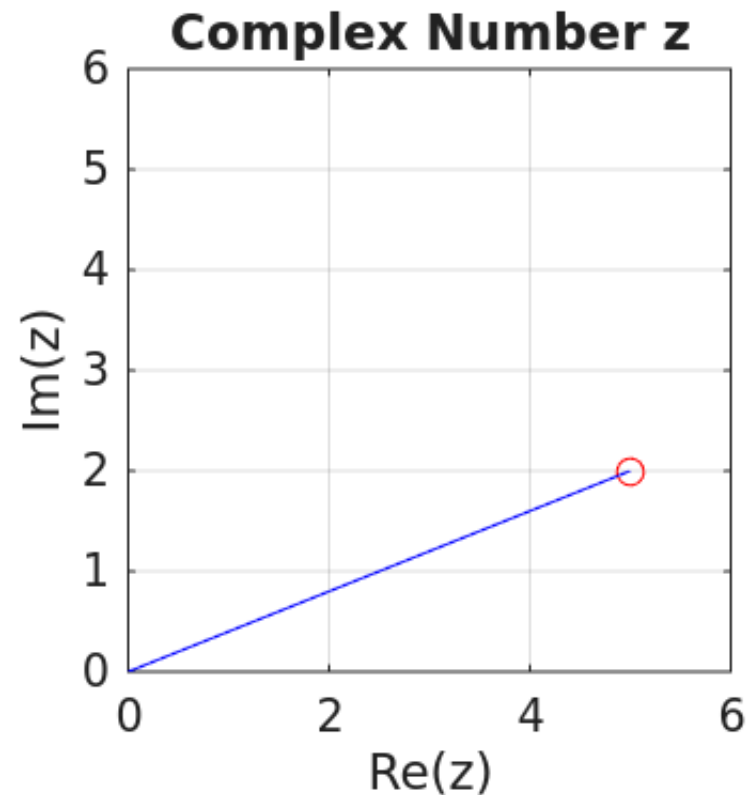
unwrap Shift phase angles



Numeri complessi in Matlab

Esempio

```
clear all
close all
z=5+2j
fid=figure (1);
plot(real(z), imag(z), 'or')
hold on
plot([0,5],[0, 2], 'b')
xlabel('Re(z)');
ylabel('Im(z)');
title('Complex Number z');
grid on;
axis([0 6 0 6], 'square')
```

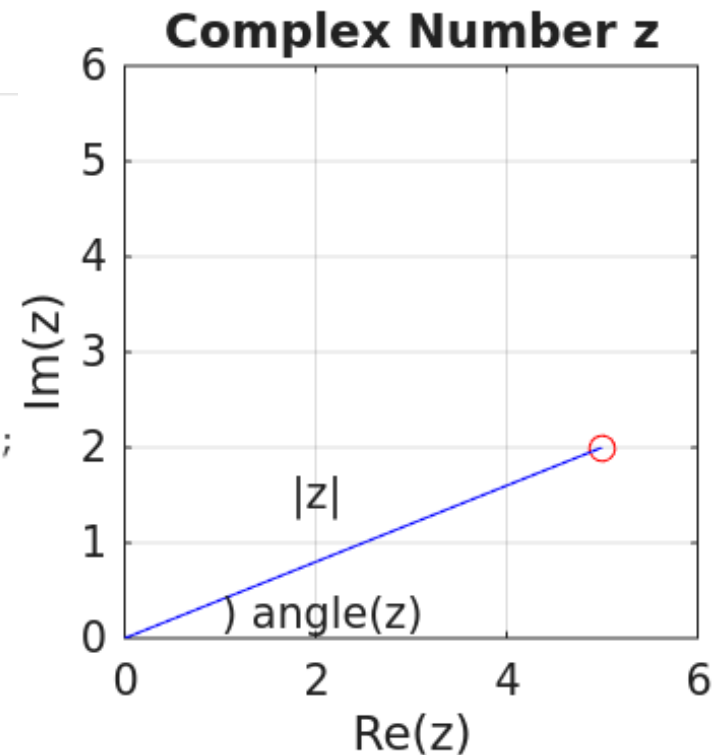




Numeri complessi in Matlab

Esempio

```
plot([0,5],[0, 2], 'b')  
xlabel('Re(z)');  
ylabel('Im(z)');  
title('Complex Number z');  
grid on;  
axis([0 6 0 6], 'square')  
disp(abs(z))  
disp(angle(z))  
text(2, 1.5, '|z|', 'HorizontalAlignment', 'center');  
text(1, 0.25, ') angle(z)');
```





Numeri complessi

Forma esponenziale

$$z = |z| e^{j\phi}$$

Matlab:

$$\text{abs}(z) \exp(j\phi)$$

Si ricorda che

$$e^{j\phi} = \cos(\phi) + j \cdot \sin(\phi)$$

$$\cos(\phi) = \frac{e^{j\phi} + e^{-j\phi}}{2}$$

$$\sin(\phi) = \frac{e^{j\phi} - e^{-j\phi}}{2j}$$

Numeri complessi

Forma esponenziale $\leftrightarrow$ Forma cartesiana $\leftrightarrow$ Forma modulo e fase

$$z = |z| e^{j\phi} = |z| (\cos(\phi) + j \sin(\phi))$$

$$z = x + jy$$

$$x = \text{real}(z) = |z| \cos(\phi)$$

$$y = \text{real}(z) = |z| \sin(\phi)$$

$$|z| = \sqrt{x^2 + y^2}$$

$$\phi = \text{atan} \frac{y}{x}$$



Funzione complessa di variabile complessa

Una funzione complessa di variabile complessa è un'applicazione che associa a ogni numero complesso del dominio un altro numero complesso (immagine).

N.B. Un numero complesso è formato da una coppia ordinata di numeri reali

$$z \rightarrow f(z) \quad z \in \mathbb{C}; \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$f(z) = u(x, y) + j v(x, y)$$



Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = x + j0 \quad \text{con: } x \in [-\pi, \pi]$$

$$u + jv = f(z) = e^{jx}$$

```
clear all
close all
x=-pi:0.05:pi;
f=exp(j*x)
u=real(f);
v=imag(f);
fid1=figure (1);
plot(real(x), imag(x), 'b')
xlabel('Re(z)');
ylabel('Im(z)');
title('Complex Number z=x+j0');
grid on;
axis([-pi pi -pi pi], 'square')
fid2=figure (2);
plot(real(f), imag(f), 'r')
xlabel('Re(f(z))');
ylabel('Im(f(z))');
title('Complex Number f(z)=exp(j*x)');
grid on;
axis([-2 2 -2 2], 'square')
```



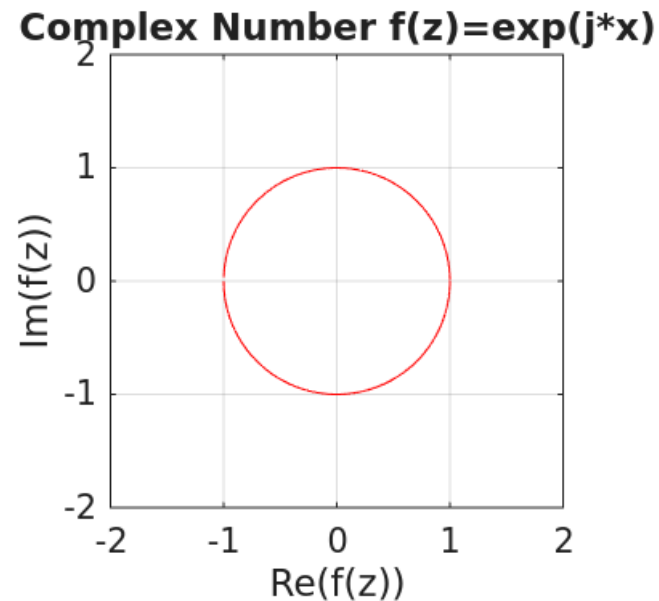
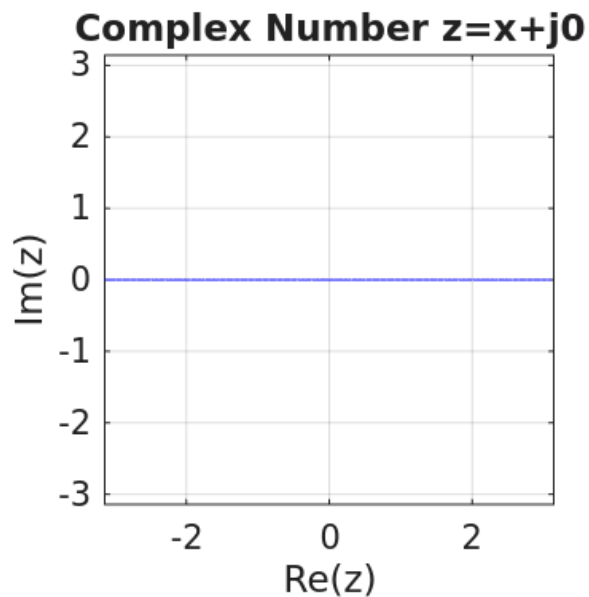
Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = x + j0 \quad \text{con: } x \in [-\pi, \pi]$$

$$u + jv = f(z) = e^{jx}$$





Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = x + j 0.5 \quad \text{con: } x \in [-\pi, \pi]$$

$$u + j v = f(z) = e^{jz}$$

```
clear all
close all
x=-pi:0.05:pi;
z=x+0.5j;
f=exp(j*z)
u=real(f);
v=imag(f);
fid1=figure (1);
plot(real(x), imag(x), 'b')
xlabel('Re(z)');
ylabel('Im(z)');
title('Complex Number z=x+0.5j');
grid on;
axis([-pi pi -pi pi], 'square')
fid2=figure (2);
plot(real(f), imag(f), 'r')
xlabel('Re(f(z))');
ylabel('Im(f(z))');
title('Complex Number f(z)=exp(j*z)');
grid on;
axis([-2 2 -2 2], 'square')
```



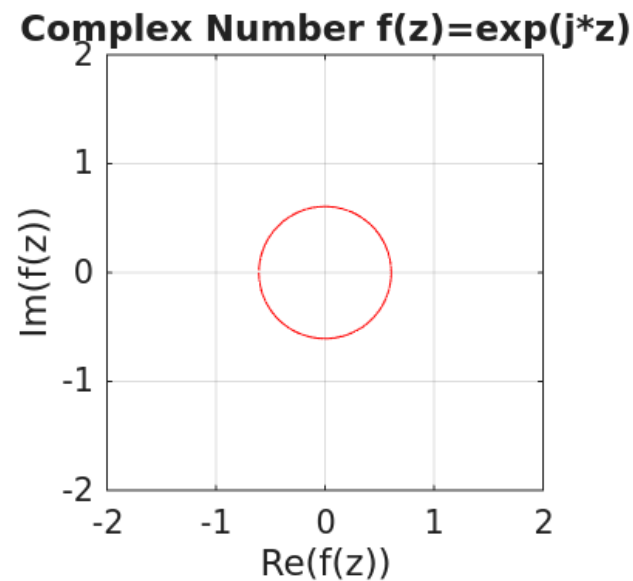
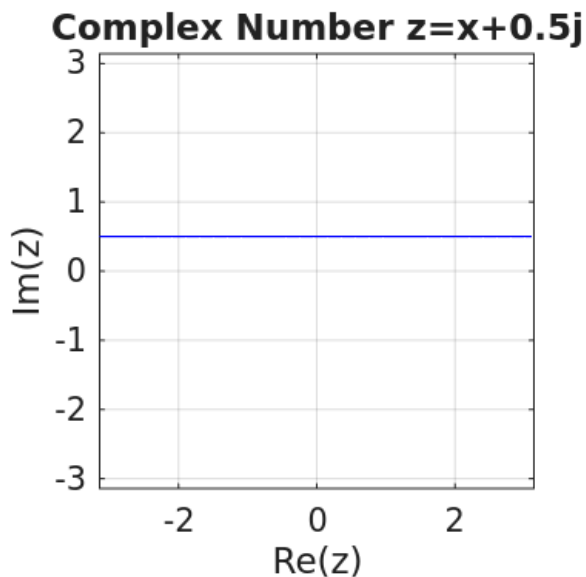
Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = x + j 0.5 \quad \text{con: } x \in [-\pi, \pi]$$

$$u + j v = f(z) = e^{j z}$$





Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = \frac{\pi}{4} + j y \quad \text{con: } y \in [-\pi, \pi]$$

$$u + j v = f(z) = e^{j z}$$

```
clear all
close all
x=-pi:0.05:pi;
z=x+0.5j;
f=exp(j*z)
u=real(f);
v=imag(f);
fid1=figure (1);
plot(real(x), imag(x), 'b')
xlabel('Re(z)');
ylabel('Im(z)');
title('Complex Number z=x+0.5j');
grid on;
axis([-pi pi -pi pi], 'square')
fid2=figure (2);
plot(real(f), imag(f), 'r')
xlabel('Re(f(z))');
ylabel('Im(f(z))');
title('Complex Number f(z)=exp(j*z)');
grid on;
axis([-2 2 -2 2], 'square')
```



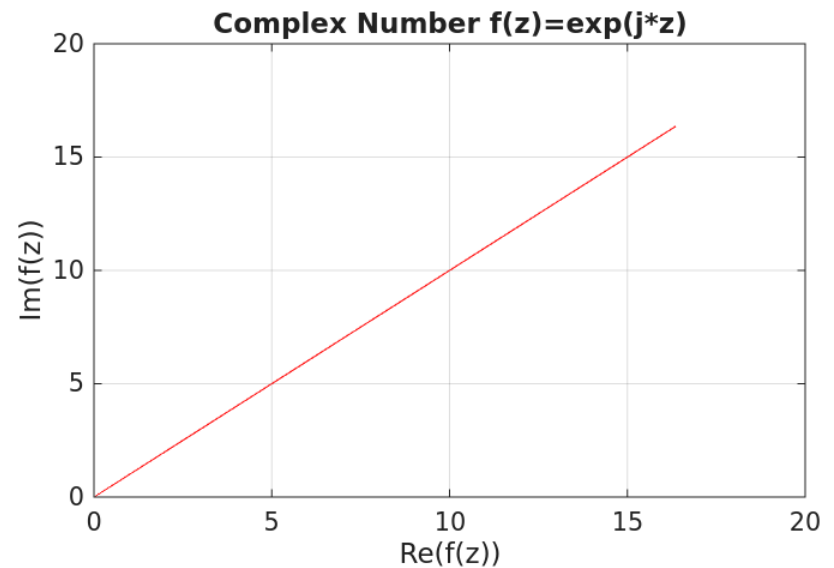
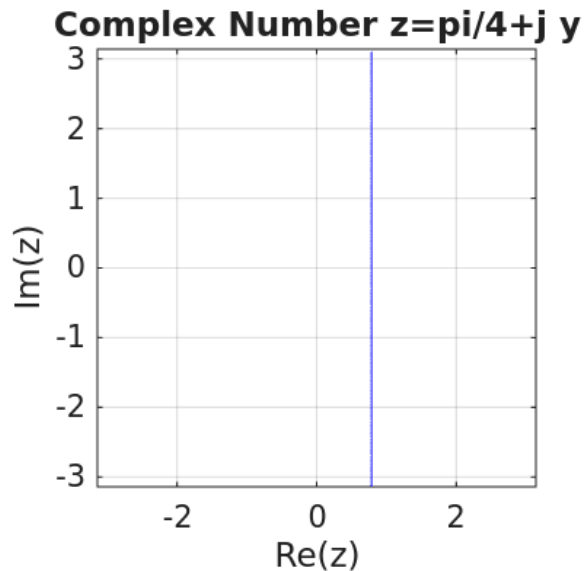
Funzione complessa di variabile complessa

Esempio

$$z \rightarrow f(z); \quad f: \mathbb{C} \rightarrow \mathbb{C}$$

$$z = \frac{\pi}{4} + j y \quad \text{con: } y \in [-\pi, \pi]$$

$$u + j v = f(z) = e^{j z}$$





UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Rappresentazione dei segnali

Rappresentazione dei segnali

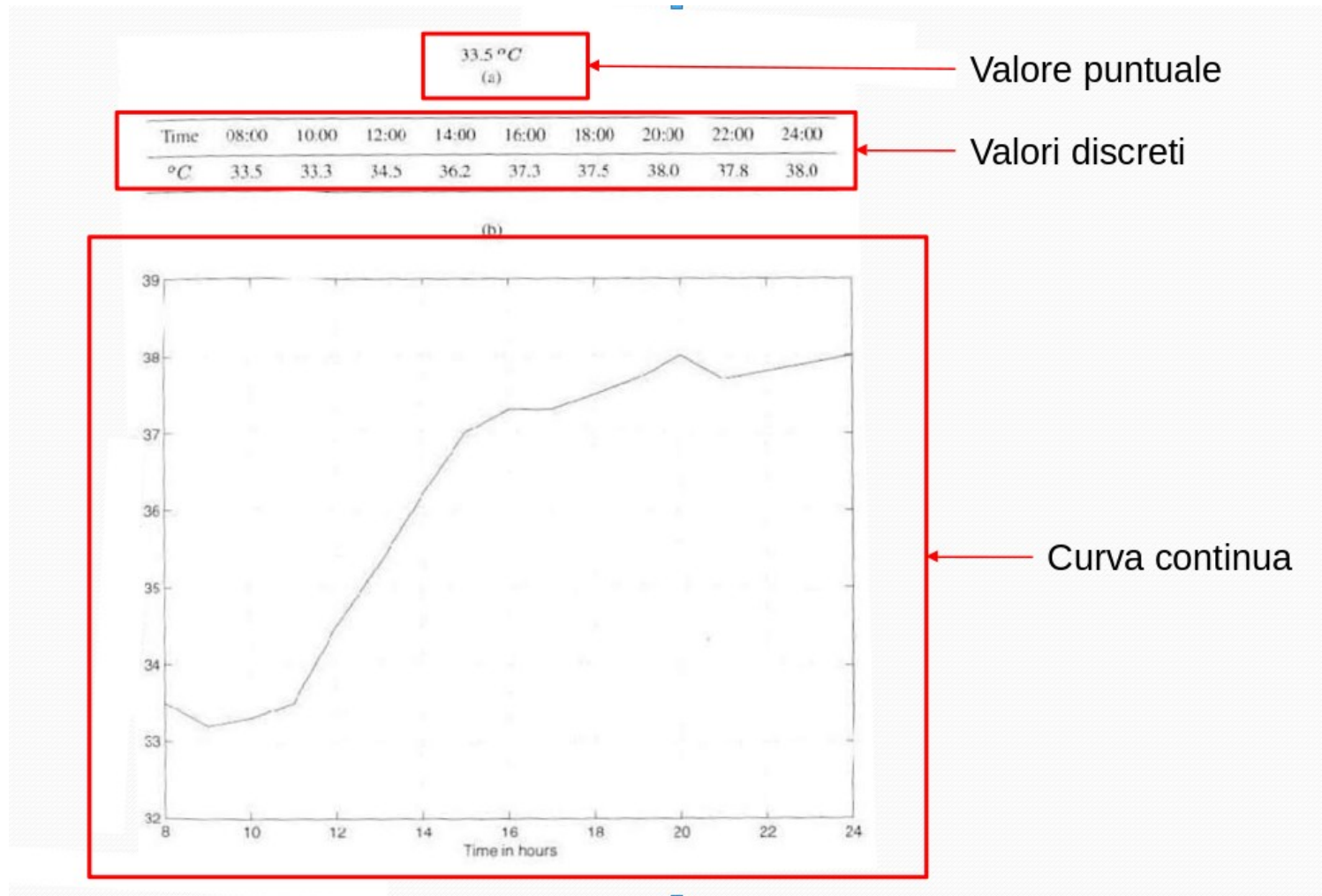


Rappresentazione dei segnali

- Un **segnale** è generato da un **evento** e contiene **informazione** relativa all'evento stesso.
- Esempi: segnale acustico, corrente, elettrocardiogramma, elettroencefalogramma, immagine radiologica, immagine fotografica, risonanza magnetica, ecc.
- In particolare un segnale biomedico porta informazione relativa ad un sistema biologico. Il processo di estrazione dell'informazione può essere da molto semplice (il medico che stima la frequenza cardiaca tastando il polso del paziente) a molto complesso (esempio: l'elaborazione di immagini ottenute mediante risonanza magnetica).
- La descrizione (modellazione) matematica di ognuno di questi esempi utilizza il concetto di funzione: un segnale è una funzione del tempo. Noi considereremo esempi in cui la variabile in uscita è una sola
- $t \rightarrow x(t)$ o $f(t)$
- In generale ce ne sono di più: 2 o 3 variabili spaziali, variabili che contengono informazioni aggiuntive (gradazione del colore, gradazione della luminosità, volume, ecc).



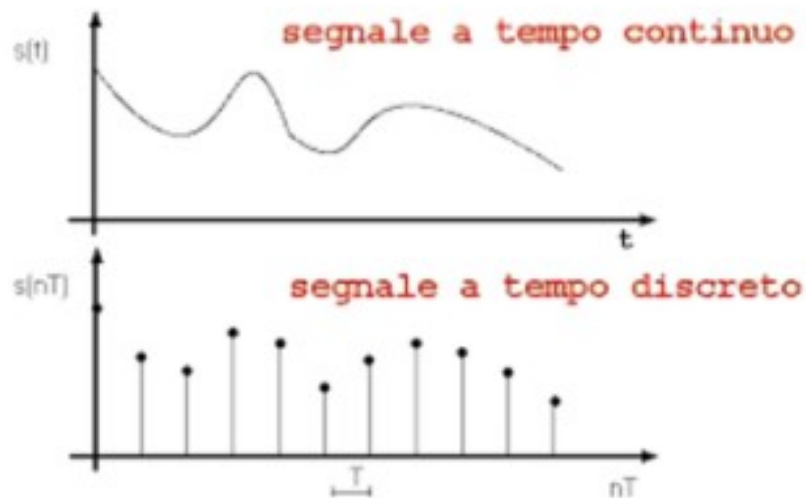
Rappresentazione dei segnali





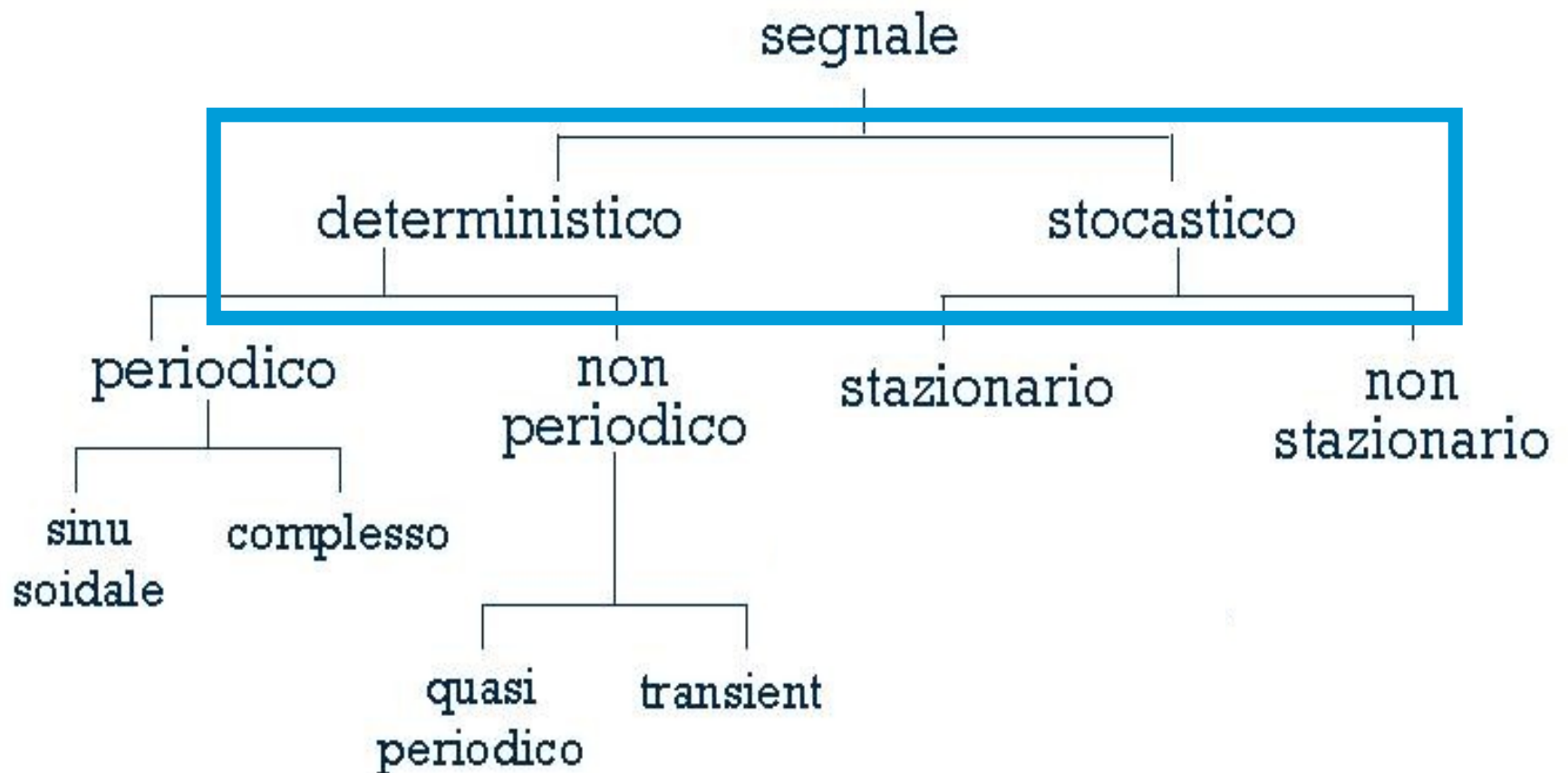
Rappresentazione dei segnali

- Segnali a tempo continuo: la variabile indipendente è una variabile che assume valori nell'intervallo $[T_1, T_2]$ dell'asse reale (estremi anche $\pm\infty$)
- Segnali a tempo discreto: la variabile indipendente è una variabile che assume valori discreti $[t_0, t_1, \dots, t_N]$, in numero finito o infinito numerabile. Tra due punti dell'asse tempi il segnale non è definito





Rappresentazione dei segnali

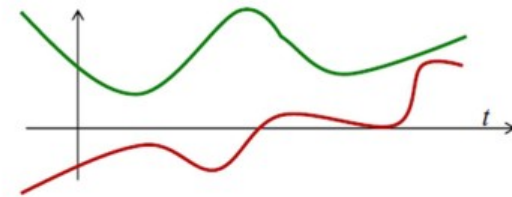
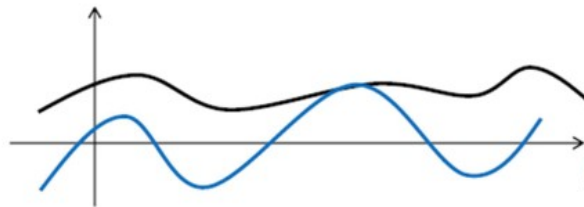
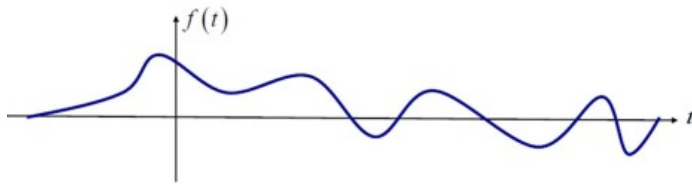




Rappresentazione dei segnali

- Segnale analogico a tempo continuo: modello di una grandezza fisica che varia nel tempo
- Il valore di $f(t)$ può essere reale o complesso. Se complesso possiamo considerare separatamente la parte reale e la parte immaginaria, oppure il modulo e la fase

$$f(t) = f_R(t) + jf_I(t)$$



dove

$$f(t) = |f(t)| \exp(j \arg(f(t)))$$

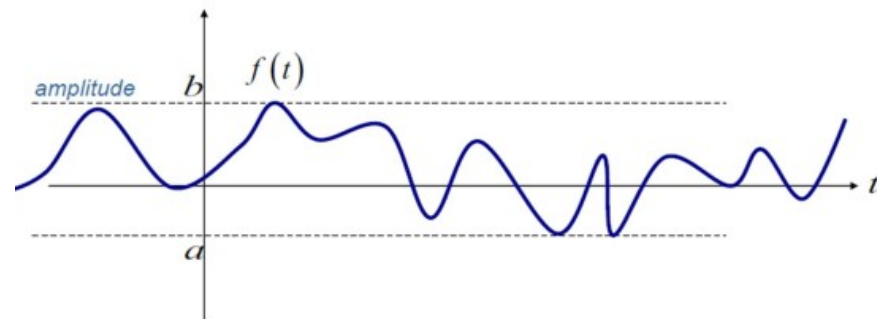
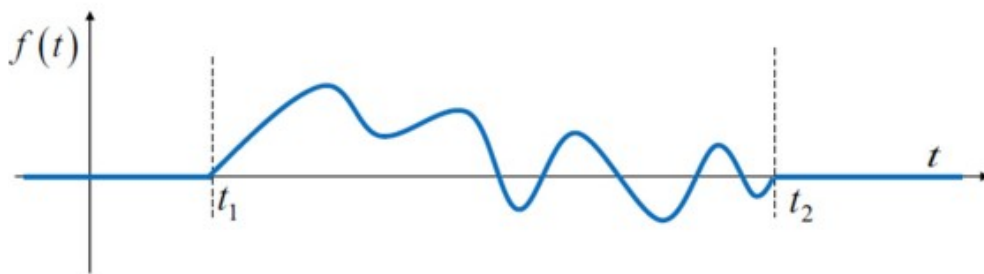
$$|f(t)| = \sqrt{f_R^2(t) + f_I^2(t)}$$

$$\arg(f(t)) = \arctan\left(\frac{f_I(t)}{f_R(t)}\right)$$



Rappresentazione dei segnali

- Nella realtà ogni segnale ha un inizio e una fine (**supporto limitato o finito**) e quindi viene descritto da una funzione con dominio limitato in un intervallo $[t_1, t_2]$
- Anche l'ampiezza di un segnale fisico è limitata in un intervallo $[a, b]$
- Per motivi teorici spesso conviene trattare i segnali in un dominio illimitato (**supporto illimitato o infinito**)
- Anche l'utilizzo dei numeri complessi è giustificato dalla necessità di semplificare alcune trattazioni teoriche.

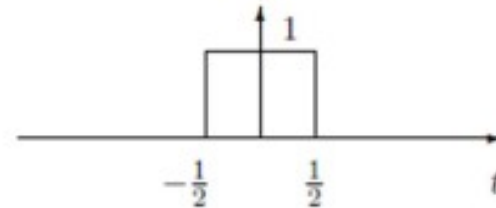




Rappresentazione dei segnali

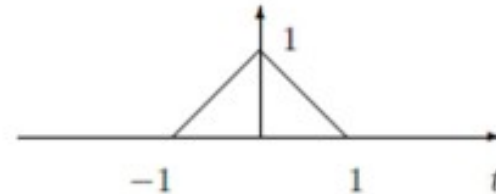
a) *Impulso o finestra rettangolare:*

$$\Pi(t) \triangleq \begin{cases} 1 & |t| \leq 1/2 \\ 0 & \text{altrimenti} \end{cases}$$



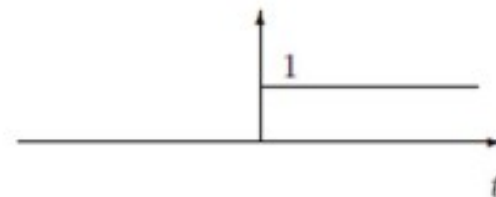
b) *Impulso o finestra triangolare:*

$$\Lambda(t) \triangleq \begin{cases} 1 - |t| & |t| \leq 1 \\ 0 & \text{altrimenti} \end{cases}$$



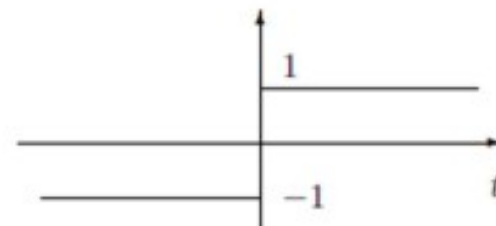
c) *Gradino unitario:*

$$u(t) = \begin{cases} 1 & t \geq 0 \\ 0 & \text{altrimenti} \end{cases}$$



d) *Signum:*

$$\text{sign}(t) = \begin{cases} 1 & t \geq 0 \\ -1 & t < 0 \end{cases}$$





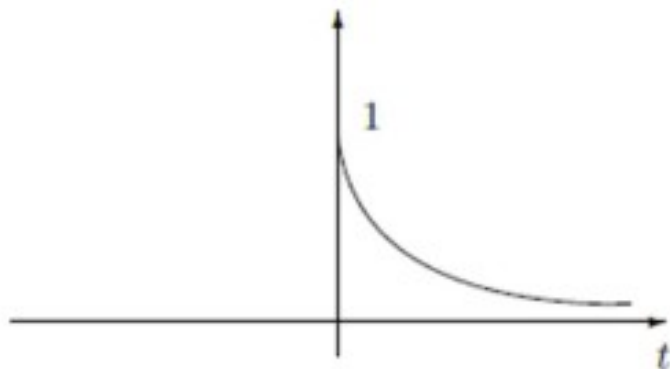
Rappresentazione dei segnali

e) *Impulso esponenziale monolatero:*

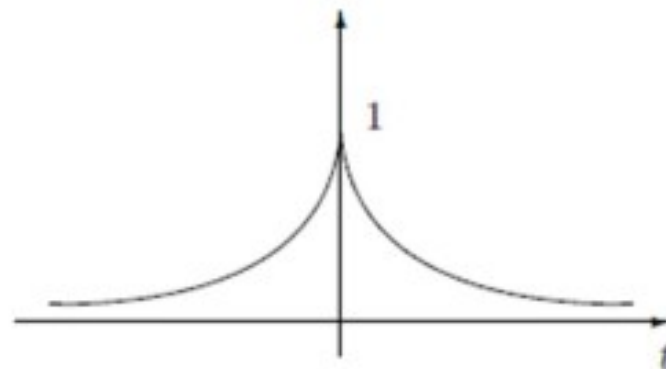
$$x(t) = \begin{cases} e^{-t} & t \geq 0 \\ 0 & \text{altrimenti} \end{cases}$$

f) *Impulso esponenziale bilatero:*

$$x(t) = e^{-|t|} = \begin{cases} e^{-t} & t \geq 0 \\ e^t & t < 0 \end{cases}$$



e)



f)

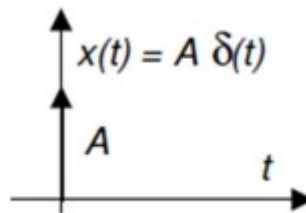


Rappresentazione dei segnali

La Delta di Dirac $\delta(t)$ o impulso matematico è un segnale che vale zero ovunque tranne per $t=0$ dove vale infinito. L'area di $\delta(t)$ è unitaria.

$$\delta(t) = \begin{cases} \infty & \text{con } t = 0 \\ 0 & \text{altrove} \end{cases} \quad \text{e} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1$$

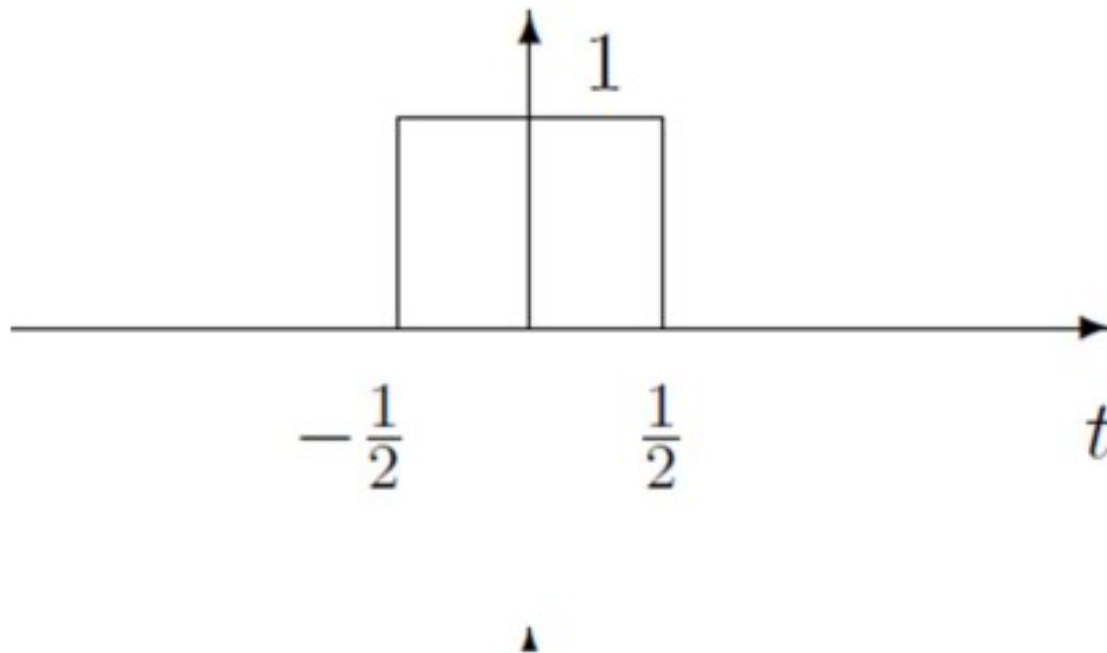
Da un punto di vista analitico, $\delta(t)$ non è una funzione, ma una “distribuzione” definita come il limite a cui tende una successione di funzioni. E' prassi rappresentare graficamente $A \cdot \delta(t)$ come una freccia con scritto accanto il valore dell'area A .





Elaborazione dei segnali

1. Impulso rettangolare standard $\Pi(t)$: Ampiezza 1 , centrato in 0 , durata 1





Elaborazione dei segnali

Supponiamo di considerare l'impulso rettangolare $x(t) = \Pi(t)$ e di voler rappresentare graficamente il segnale

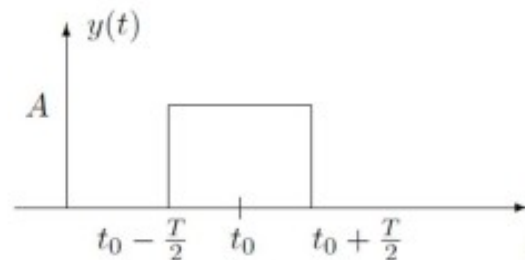
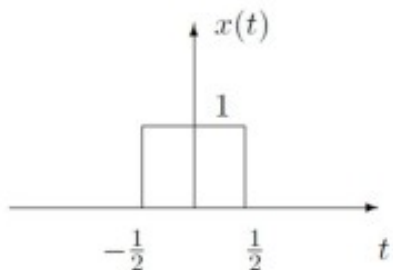
$$y(t) = A \Pi\left(\frac{t - t_0}{T}\right)$$

Con $t_0 > 0$ e $T > 1$. Si ottiene un impulso rettangolare di :

ampiezza A

centrato in t_0

durata T





Rappresentazione dei segnali in Matlab

Generazione di segnali notevoli

- Rect
- Tri
- Delta di Dirac
- Treno di impulsi



Rappresentazione dei segnali in Matlab

- **rect.m**

```
fs = 10000;                % Number of samples per second
ts=1/fs;
t = -1:ts:1;                % Time vector
y=(abs(t)<0.5)+0.5*(abs(t)==0.5);
figure(1)
plot(t,x2), axis([-0.1 0.1 -0.2 1.2]) % Plotting the pulse
xlabel('Time (sec)');
ylabel('Amplitude');
```

- **tri.m**

```
y=(1-abs(t)).*(abs(t)<1);
```



Rappresentazione dei segnali in Matlab

- %delta(t)

```
y=zeros(1,length(t));
```

```
[~,i]=min(abs(t));
```

```
y(i)=1/(t(2)-t(1));
```

- %treno(t,T)

```
i=T;
```

```
y=delta(t); %delta in 0
```

```
while i<=max(t)
```

```
    y=y+delta(t-i)+delta(t+i);
```

```
    i=T+i;
```

```
end
```



Rappresentazione dei segnali in Matlab

Segnali.m

```
clear all;
fs = 1000; % Number of samples per second
tMin=-1;
tMax=1;
t = tMin:1/fs:tMax;          % Time vector
Tau=0.5
A=1
% rettangolare
x1=A*(abs(t)<Tau/2)+0.5*(abs(t)==Tau/2);
figure(1)
plot(t,x1)      % Plotting the pulse
axis([tMin tMax min(x1) max(x1)*1.1])
xlabel('Time (sec)');
ylabel('Amplitude');
```



Rappresentazione dei segnali in Matlab

```
clear all;  
fs = 1000; % Number of samples per second  
tMin=-1;  
tMax=1;  
t = tMin:1/fs:tMax; % Time vector  
Tau=0.5  
A=1
```

%triangolare

```
x2=A*(1-abs(t/(Tau/2))).*(abs(t)<Tau/2);  
figure(2)  
plot(t,x2) % Plotting the pulse  
axis([tMin tMax min(x2) max(x2)]*1.1)  
xlabel('Time (sec)');  
ylabel('Amplitude');
```



Rappresentazione dei segnali in Matlab

```
clear all;  
fs = 1000; % Number of samples per second  
tMin=-1;  
tMax=1;  
t = tMin:1/fs:tMax; % Time vector  
Tau=0.5  
A=1
```

```
% impulso  
x3=zeros(1,length(t));  
[~,k]=min(abs(t));  
x3(k)=1/(t(2)-t(1));  
figure(3)  
plot(t,x3) % Plotting the pulse  
axis([tMin tMax min(x3) max(x3)*1.1])  
xlabel('Time (sec)');  
ylabel('Amplitude');
```



Rappresentazione dei segnali in Matlab

```
clear all;
fs = 1000; % Number of samples per second
tMin=-1;
tMax=1;
t = tMin:1/fs:tMax;          % Time vector
Tau=0.5
A=1
%treno di impulsi
T=0.25;
k=T;
x4=delta(t); %delta in 0
    while k<=max(t)
        x4=x4+delta(t-k)+delta(t+k);
        k=T+k;
    end
figure(4)
plot(t,x4)      % Plotting the pulse
axis([tMin tMax min(x4) max(x4)]*1.1)
xlabel('Time (sec)');
ylabel('Amplitude');
```

```
function [y]=delta(t)
y=zeros(1,length(t));
[~,k]=min(abs(t));
y(k)=1;  %1/(t(2)-t(1));
end
```



Rappresentazione dei segnali in Matlab

```
clear all;  
fs = 1000; % Number of samples per second  
tMin=-1;  
tMax=1;  
t = tMin:1/fs:tMax;          % Time vector  
Tau=0.5  
A=1
```

```
%gradino  
x5=zeros(1,length(t));  
x5=(t>=0);  
figure(5)  
plot(t,x5)      % Plotting the pulse  
axis([tMin tMax min(x5) max(x5)*1.1])  
xlabel('Time (sec)');  
ylabel('Amplitude');
```



Rappresentazione dei segnali in Matlab

```
clear all;  
fs = 1000; % Number of samples per second  
tMin=-1;  
tMax=1;  
t = tMin:1/fs:tMax;          % Time vector  
Tau=0.5  
A=1
```

```
%semirampa  
x6=zeros(1,length(t));  
x6=(t+1).*(t/2<0);  
figure(6)  
plot(t,x6)      % Plotting the pulse  
axis([tMin tMax min(x6) max(x6)*1.1])  
xlabel('Time (sec)');  
ylabel('Amplitude');
```



Rappresentazione dei segnali in Matlab

```
clear all;
fs = 1000; % Number of samples per second
tMin=-1;
tMax=1;
t = tMin:1/fs:tMax;          % Time vector
Tau=0.5
A=1

% exp
x7=A*((t)>=0).*exp(-6*t);
figure(7)
plot(t,x7)      % Plotting the pulse
axis([tMin tMax min(x1) max(x1)*1.1])
xlabel('Time (sec)');
ylabel('Amplitude');
```



UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Serie di Fourier in Matlab

Serie di Fourier



Serie di Fourier in Matlab

- Una funzione periodica è una funzione che si ripete ogni T secondi. In altre parole, una funzione periodica soddisfa alla condizione

$$f(t) = f(t + nT)$$

dove n è un intero e T è il periodo della funzione.

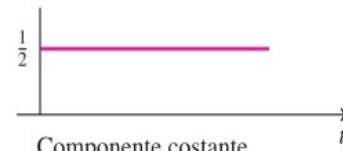
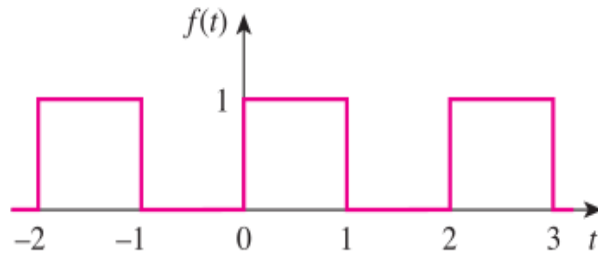
- **Teorema di Fourier**, qualunque funzione periodica di frequenza ω_0 si può esprimere come somma di infinite funzioni seno e coseno a frequenze multiple intere di ω_0 .

$$f(t) = a_0 + \sum_{k=1}^{\infty} [a_k \cos(k \omega_0 t) + b_k \sin(k \omega_0 t)]$$

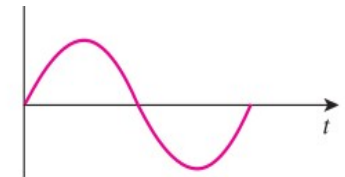
$$\omega_0 = \frac{2\pi}{T}$$



Serie di Fourier in Matlab

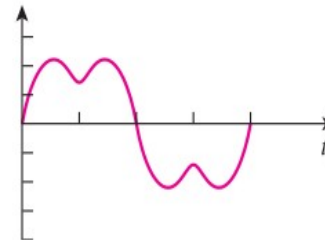


Componente costante

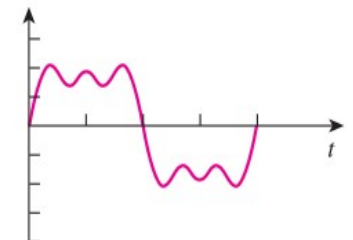


Componente AC fondamentale

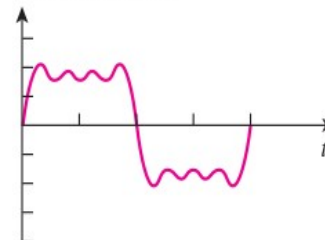
(a)



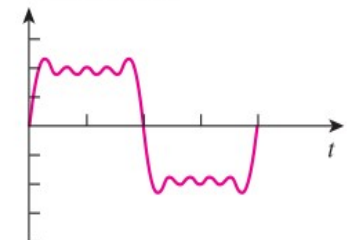
Somma delle prime due
componenti AC



Somma delle prime tre
componenti AC



Somma delle prime quattro
componenti AC



Somma delle prime cinque
componenti AC

(b)



Serie di Fourier in Matlab

- Forma trigonometrica

$$f(t) = a_0 + \sum_{k=1}^{\infty} [a_k \cos(k \omega_0 t) + b_k \sin(k \omega_0 t)]$$

$$a_0 = \frac{1}{T} \int_0^T f(t) dt$$

$$a_k = \frac{2}{T} \int_0^T f(t) \cos(n \omega_0 t) dt$$

$$b_k = \frac{2}{T} \int_0^T f(t) \sin(n \omega_0 t) dt$$

- Forma esponenziale

$$f(t) = \sum_{-\infty}^{\infty} (c_n e^{(j \omega_0 t)})$$

$$c_n = \frac{1}{T} \int_{-\infty}^{+\infty} f(t) e^{-jn \omega_0 t} dt$$

Serie di Fourier in Matlab

- Ricordare che

$$e^{j\omega_0 t} = \cos(\omega_0 t) + j \cdot \sin(\omega_0 t)$$

$$\cos(\omega_0 t) = \frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2}$$

$$\sin(\omega_0 t) = \frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j}$$



UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Trasformata di Fourier in Matlab

Trasformata di Fourier



Trasformata di Fourier in Matlab

La trasformata di Fourier è una trasformazione integrale di $f(t)$ dal dominio del tempo al dominio delle frequenze.

$$F(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-j\omega t} dt$$

$$f(t) = F^{-1}[F(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega) e^{j\omega t} d\omega$$

$$f(t) \leftrightarrow F(\omega)$$



Trasformata di Fourier in Matlab

Funzione nel dominio del tempo $f(t)$	Trasformata di Fourier $F(f)$
$\delta(t)$	1
1	$\delta(f)$
$e^{j2\pi f_0 t}$	$\delta(f - f_0)$
$e^{j-2\pi f_0 t}$	$\delta(f + f_0)$
$\cos(2\pi f_0 t)$	$\frac{1}{2} [\delta(f - f_0) + \delta(f + f_0)]$ <u>Dimostrazione</u>
$\sin(2\pi f_0 t)$	$\frac{1}{2j} [\delta(f - f_0) - \delta(f + f_0)]$ <u>Dimostrazione</u>



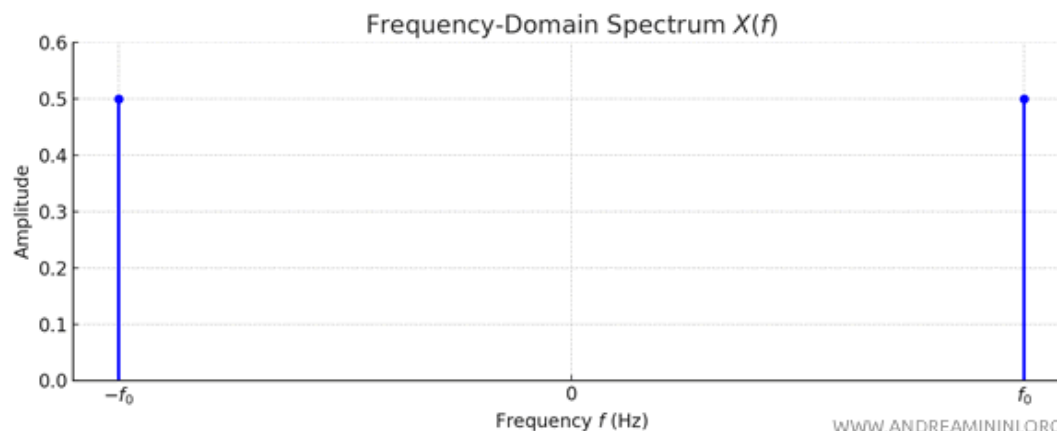
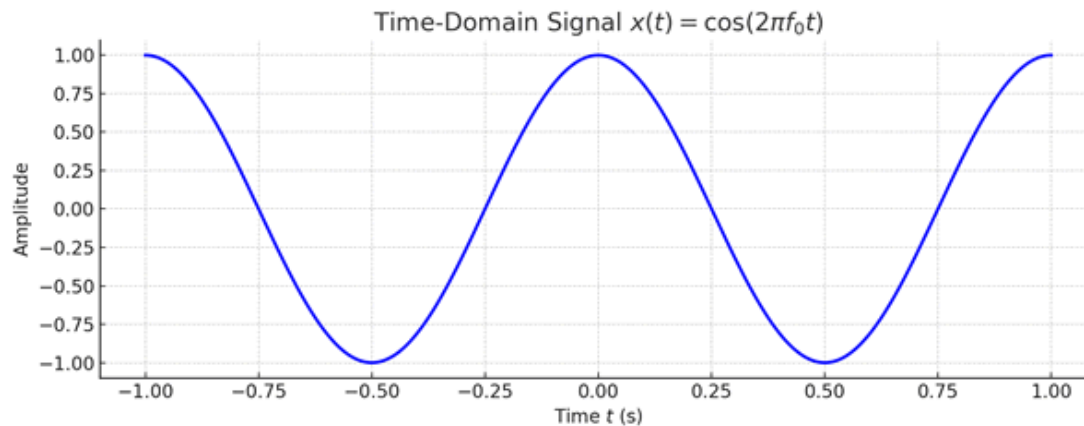
Trasformata di Fourier in Matlab

Funzione nel dominio del tempo $f(t)$	Trasformata di Fourier $F(f)$
$u(t)$	$\frac{1}{j2\pi f} + \frac{1}{2}\delta(f)$
$e^{-at}u(t), a > 0$	$\frac{1}{a+j2\pi f}$
$e^{at}u(-t), a > 0$	$\frac{1}{a-j2\pi f}$
$\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$	$\text{rect}(f)$
$\text{rect}(t)$	$\text{sinc}(f)$
$e^{-(t/T)^2}$	$T\sqrt{\pi}e^{-\pi^2 f^2 T^2}$
$\frac{1}{\pi(t^2+1)}$	$e^{-2\pi f }$
$\text{tri}(t)$	$\text{sinc}^2(f)$
$t \cdot e^{-at}u(t)$	$\frac{1}{(a+j2\pi f)^2}$



Trasformata di Fourier in Matlab

$$F(\cos(2\pi f_0 t)) = X(f) = \frac{1}{2}(\delta(f - f_0) + \delta(f + f_0))$$





Trasformata di Fourier in Matlab

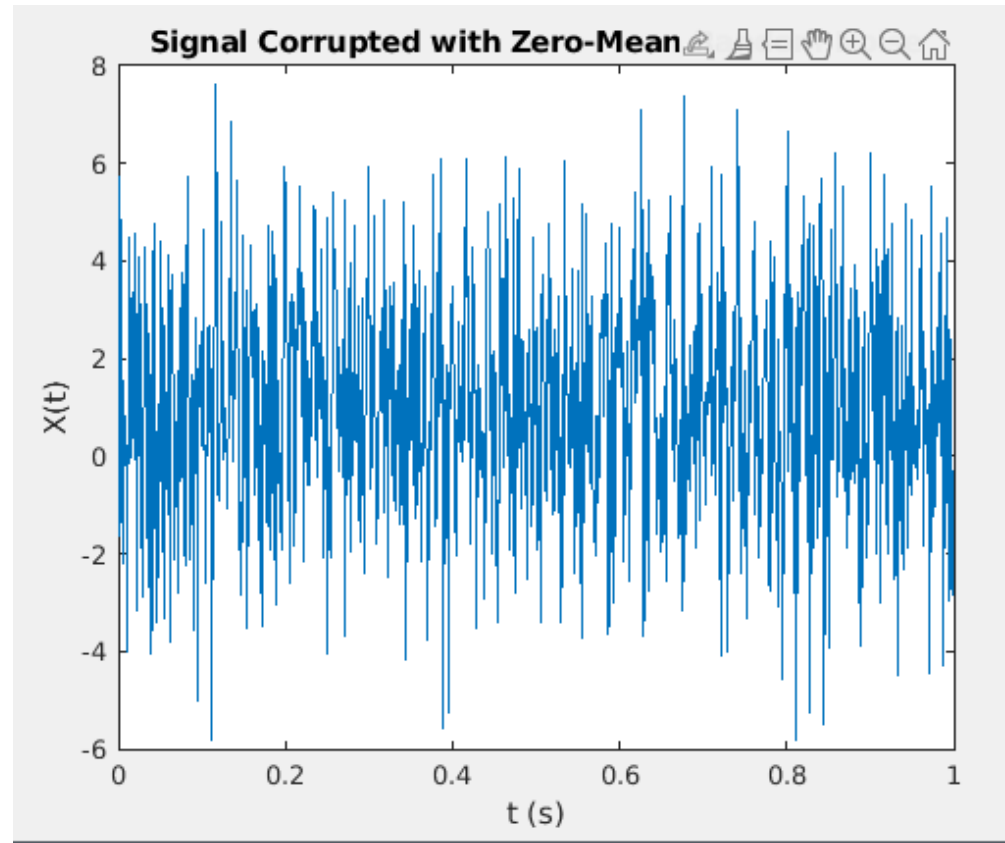
- $A_0=1$; %componente costante
- $A_1=1$; %segnale 1:ampiezza
- $f_1=50$; %segnale 1:frequenza
- $A_2=1$; %segnale 2:ampiezza
- $f_2=120$; %segnale 2:ampiezza
- $A_3=1$; %segnale 3:ampiezza
- $f_3=300$; %segnale 4:ampiezza
- $n(t)=2*\text{randn}(\text{size}(t))$; %rumore

$$x(t) = A_0 + A_1 \cdot \cos(2 \pi f_1 t) + A_2 \cdot \cos(2 \pi f_2 t) + A_3 \cdot \cos(2 \pi f_3 t) + n(t)$$



Trasformata di Fourier in Matlab

- $A_0=1$; %componente costante
- $A_1=1$; %segnale 1:ampiezza
- $f_1=50$; %segnale 1:frequenza
- $A_2=1$; %segnale 2:ampiezza
- $f_2=120$; %segnale 2:ampiezza
- $A_3=1$; %segnale 3:ampiezza
- $f_3=300$; %segnale 4:ampiezza
- $n(t)=2*\text{randn}(\text{size}(t))$; %rumore



$$x(t) = A_0 + A_1 \cdot \cos(2\pi f_1 t) + A_2 \cdot \cos(2\pi f_2 t) + A_3 \cdot \cos(2\pi f_3 t) + n(t)$$

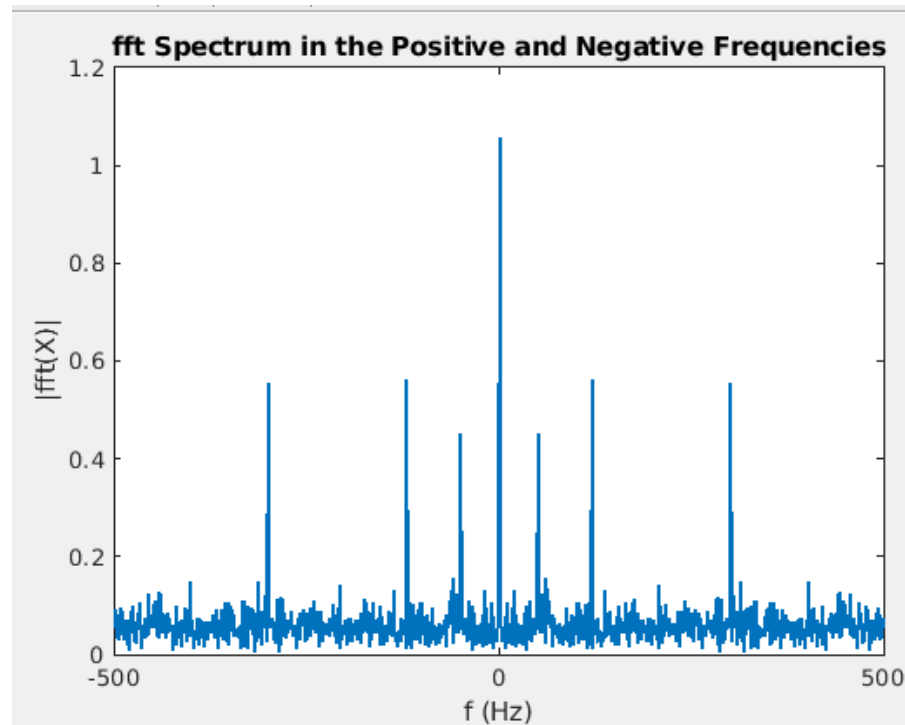


Trasformata di Fourier in Matlab

Ricordiamo che

$$F(\cos(2\pi f_0 t)) = X(f) = \frac{1}{2}(\delta(f - f_0) + \delta(f + f_0))$$

$$X(f) = A_0 + \frac{A_1}{2}(\delta(f - f_1) + \delta(f + f_1)) + \frac{A_2}{2}(\delta(f - f_2) + \delta(f + f_2)) + \frac{A_3}{2}(\delta(f - f_3) + \delta(f + f_3)) + N_0(f)$$





Trasformata di Fourier in Matlab (FFT)

- FFT: Fast Fourier Transform
- $Y = \text{fft}(x)$ calcola la trasformata di Fourier discreta (DFT) di x utilizzando un algoritmo della trasformata di Fourier veloce (FFT). Y ha la stessa grandezza di x .
- Se x è un vettore, $\text{fft}(x)$ restituisce la trasformata di Fourier del vettore.
- Se x è una matrice, $\text{fft}(x)$ tratta le colonne di x come vettori e restituisce la trasformata di Fourier di ciascuna colonna.
- **$Y = \text{fft}(x,n)$** restituisce la DFT a n punti
- Se x è un vettore e la lunghezza di x è meno di n , allora x è esteso alla dimensione n con degli zeri in coda.
- Se x è un vettore e la lunghezza di x è maggiore di n , allora x è troncato alla lunghezza n .



Trasformata di Fourier in Matlab (FFT)

- **$Y = \text{fft}(x,n)$** restituisce lo spettro di ampiezza mono direzionale (solo frequenze positive) del segnale, scalato di un fattore pari a n .
- Per ottenere i valori di ampiezza corretti bisogna considerare

$$Y = \frac{\text{fft}(x,n)}{n}$$

- A noi interessa sempre lo spettro completo, comprensivo delle frequenze negative, quindi dobbiamo “ribaltare” lo spettro rispetto all’asse a 0 Hz, mediante la funzione **$\text{fftshift}(Y)$**
- Per ottenere lo spettro completo di dimensione corretta utilizziamo le seguenti istruzioni



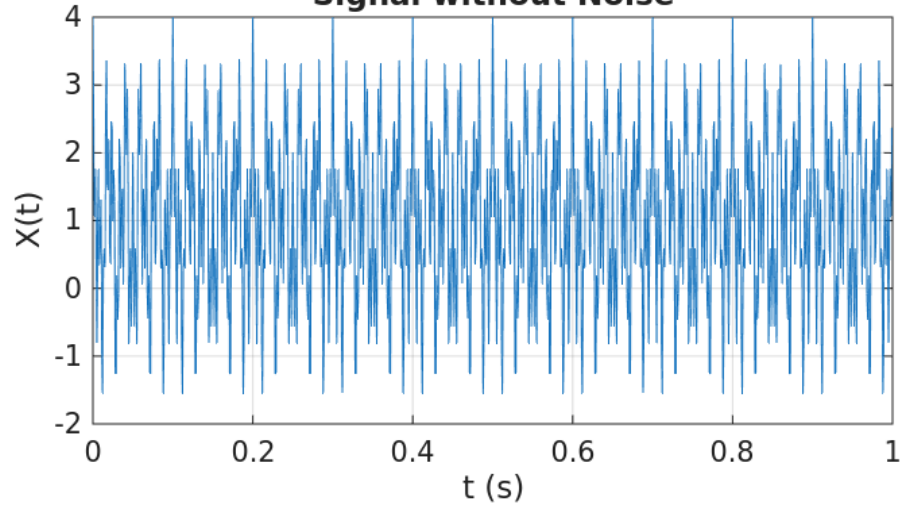
Trasformata di Fourier in Matlab (FFT)

```
clear all;
close all;
Fs=1000; %frequenza di campionamento
Ts=1./Fs; %Periodo di campionamento
L=1000; %nuero di campioni
n=L;
t=(0:L-1)*Ts;
f=(-Fs/2:Fs/L:Fs/2-1);
A0=1; %componente costante
A1=1; %segnale 1:ampiezza
f1=50; %segnale 1:frequenza
A2=1; %segnale 2:ampiezza
f2=120; %segnale 2:ampiezza
A3=1; %segnale 3:ampiezza
f3=300; %segnale 4:ampiezza
rumore=2*randn(size(t)); %rumore
x=A0+A1*cos(2*pi*f1*t)+A2*cos(2*pi*f2*t)+A3*cos(2*pi*f3*t);%+rumore;
figure(1)
plot(t,x) % segnale nel dominio del tempo
Y=fft(x,L)/L; % trasformata FFT
title("Signal Corrupted with Zero-Mean Random Noise")
xlabel("t (s)")
ylabel("X(t)")
grid on
figure(2)
plot(f,abs(fftshift(Y)),"LineWidth",3)
title("fft Spectrum in the Positive and Negative Frequencies")
xlabel("f (Hz)")
ylabel("|fft(X)|")
grid on
```

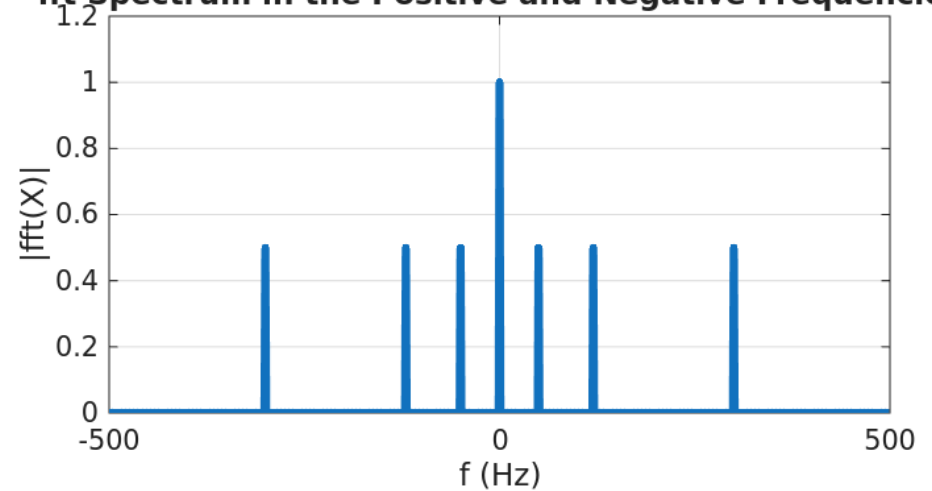


Trasformata di Fourier in Matlab (FFT)

Signal without Noise



fft Spectrum in the Positive and Negative Frequencies





Energia

Energia di un segnale tempo-continuo in un intervallo $[a,b]$

$$E_x = \int_a^b |x(t)|^2 dt$$

Dato che eseguiamo il calcolo numericamente, dobbiamo utilizzare un algoritmo per:

$$\int_a^b f(x) dx \approx \sum_{i=0}^{i=n-1} (x_{i+1} - x_i) \frac{1}{2} [f(x_{i+1}) + f(x_i)]$$

Se dividiamo $[a,b]$ in parti n parti uguali

$$x_{i+1} - x_i = \Delta x = \frac{b-a}{n}$$

Otteniamo

$$\begin{aligned} \int_a^b f(x) dx &\approx \frac{1}{2} \frac{b-a}{n} \sum_{i=0}^{i=n-1} [f(x_{i+1}) + f(x_i)] \\ \sum_{i=0}^{i=n-1} [f(x_{i+1}) + f(x_i)] &= \sum_{i=0}^{i=n-1} f(x_{i+1}) + \sum_{i=0}^{i=n-1} f(x_i) = \sum_{i=0}^{i=n-1} f(x_i) + f(b) + \sum_{i=0}^{i=n-1} f(x_i) \\ &= -f(a) + 2 \sum_{i=0}^{i=n-1} f(x_i) = -f(a) + 2 \left(\sum_{i=0}^{i=n-1} f(x_i) - f(b) \right) = -f(a) - 2f(b) + 2 \left(\sum_{i=0}^{i=n-1} f(x_i) \right) \\ \int_a^b f(x) dx &\approx \frac{(b-a)}{n} \left(\sum_{i=0}^{i=n-1} f(x_i) - \frac{1}{2} f(a) - \frac{1}{2} f(b) \right) \end{aligned}$$

- `W1.m`
- % Create a triangular pulse with width 0.4.
- % `fs = 10000;` % Sampling frequency (samples/sec)
- % `t = -1:1/fs:1;` % Time Vector
- % `w = .4;` % Triangle Width

```
fs = 10000; % Sampling frequency (samples/sec)
t = -1:1/fs:1; % Time Vector
w = .4; % Triangle Width
x = tripuls(t-0.5,w); % Sampled aperiodic triangle
plot(t,x);
xlabel('Time (sec)'); ylabel('Amplitude');
```

- % See also SQUARE,...
- % RECTPULSE,

