

# Laboratorio (Automatica)

**Silvio Simani**

Dipartimento di Ingegneria

Università di Ferrara

Tel. 0532 97 4844

Fax. 0532 97 4870

E-mail: `ssimani@ing.unife.it`

URL: `http://www.ing.unife.it/simani`



## Informazioni generali sul corso



### Organizzazione delle lezioni

1. Informazioni generali sul corso
2. Introduzione a *Matlab*<sup>®</sup>
3. Introduzione a *Simulink*<sup>®</sup>
4. Sistemi di Controllo Digitale (...)



## Sistemi di Controllo Digitale

1. Introduzione al controllo digitale (generalità sui sistemi di controllo digitale, dispositivi di interfaccia e anello di controllo digitale).
2. Introduzione agli strumenti matematici per l'analisi di sistemi a tempo discreto (equazioni alle differenze, esempi di controllo digitale e importanza del campionamento).
3. Strumenti matematici per l'analisi di sistemi a tempo discreto ( $\mathcal{Z}$ -Trasformate, funzioni di trasferimento,  $\mathcal{Z}$ -trasformate di segnali elementari, esempi di funzioni)
4.  $\mathcal{Z}$ -Trasformate (proprietà, teoremi e metodi di antistrasformazione).
5. Campionamento e ricostruzione di segnali (dispositivi di campionamento, campionario, ricostruttore, Teorema di Shannon, aliasing, esempi).
6. Sistemi a tempo discreto (funzione di trasferimento discreta, composizione di schemi a blocchi).
7. Stabilità dei sistemi discreti (definizione, piano complesso, luogo delle radici, piano  $w$ ).
8. Specifiche di progetto di sistemi di controllo e progetto di regolatori (tecniche di discretizzazione, progetto con metodi analitici, progetto mediante luogo delle radici, progetto nel piano  $w$ , regolatori standard PID).



## Bibliografia essenziale e strumenti didattici

1. Dispense e Lucidi del Corso di Automatica I (Laboratorio).
2. Matlab, The Language of Technical Computing. Getting Started with MATLAB. *Version 5.2* (In formato pdf su CD Matlab)
3. MATLAB Primer. Second Edition. Kermit Sigmon. Department of Mathematics. University of Florida.
4. The MathWorks Inc., Matlab User's Guide, 1993.
5. The MathWorks Inc., Simulink User's Guide, 1995.
6. G. Marro, TFI: insegnare e apprendere i controlli automatici di base con matlab. Bologna: Zanichelli, I ed., Ottobre 1998.



## Introduzione a MATLAB

⇒ **Linguaggio per risolvere problemi di calcolo numerico** (*MAThematical LABoratory*)

⇒ **Marchio registrato da *MathWorks Inc.* (U.S.A.)**

⇒ **Pacchetto software più diffuso tra progettisti e ricercatori**

⇒ **Può essere ampliato da pacchetti specifici (*toolbox*)**

⇒ Es. Control System Toolbox, Identification Toolbox, Simulink

⇒ **È un *interprete* in grado di eseguire**

⇒ Istruzioni native (*build-in*, definite dal Matlab)

⇒ Contenuti in files (*m-files*, *function-files*, *script-files*)



## Elementi di base di Matlab

⇒ L'elemento di base è la matrice (elementi interi, reali o complessi)

⇒ `>> A = [1,2,3;4,5,6]`

⇒ `A =`  
$$\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{matrix}$$


⇒ `>> 7*3+2`


⇒ `ans =`  
$$23$$



## Istruzioni elementari

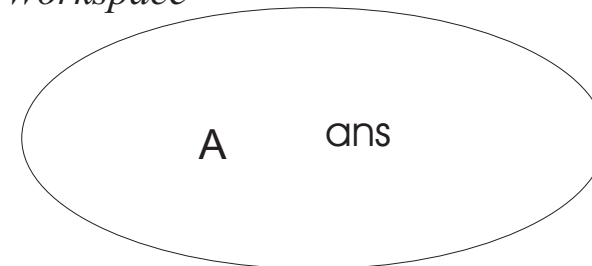
 >> **who**

 **Your variables are:**  
**A**            **ans**

 >> **whos**

Name	Size	Bytes	Class
<b>A</b>	<b>2x3</b>	<b>48</b>	<b>double array</b>
<b>ans</b>	<b>1x1</b>	<b>8</b>	<b>double array</b>

*Workspace*



**Grand total is 7 elements using 56 bytes**



## Istruzioni “DOS-like”



**Direttorio di lavoro: Z:\...\auto\_?**

⇒ dir

⇒ type

⇒ delete, ...



>> **help** <topic>



**Creazione del file *pippo.mat* che contiene la matrice A**

⇒ >> save pippo A (*salva la matrice A nel file “pippo.mat”*)

⇒ >> clear A (*rimuove dal workspace la matrice A*)

⇒ >> load pippo (*carica da file la matrice A*)





## Operazioni sulle matrici



**Date le matrici A e B di dimensioni opportune, si possono definire le seguenti operazioni:**

$$\Rightarrow \gg S = A + B$$

$$\Rightarrow \gg P = A * B$$

$$\Rightarrow \gg At = A'$$

$$\Rightarrow \gg Ai = \text{inv}(A)$$

$$\Rightarrow \gg Ap = \text{pinv}(A), \text{ con } Ap = (A^T * A)^{-1} * A^T$$



## Calcolo dei valori

⇒ Definita la matrice rettangolare  $A_{m \times n}$  e la matrice quadrata  $B_{n \times n}$ , Matlab definisce le seguenti funzioni:

⇒ `>> det(B)`

⇒ `>> rank(A)`

⇒ `>> [V,D] = eig(B)`, con  $V*B*V' = D$

⇒ `>> Bi = inv(B)`

⇒ `>> [m,n] = size(A)`

⇒ Selezione degli elementi della matrice A

⇒ `A(i,j)`, `A(1:2,2:3)`, `A(1,:)`, `A(:,3:5)`



## Risoluzione di sistemi lineari

⇒ **Calcolare il valore di  $x$ , con  $Ax = B$**

$$\Rightarrow x = A^{-1}B$$

$$\Rightarrow \gg x = A \setminus B$$

$$\Rightarrow \gg x = \text{inv}(A) * B$$

⇒ **Calcolare il valore di  $x$ , con  $xC = D$**

$$\Rightarrow x = DC^{-1}$$

$$\Rightarrow \gg x = D / C$$

$$\Rightarrow \gg x = D * \text{inv}(C)$$



## Script files e function files

```
% File quad.m
```

```
A = B * B;
```

```
B = A;
```

```
function r = rank(A)
```

```
%RANK Matrix rank.
```

```
% RANK(A) provides an estimate
```

```
% of the number of linearly
```

```
% independent rows or columns
```


```
% of a matrix A.
```


```
s = svd(A);
```

```
r = sum(s > 0);
```




## Istruzioni di controllo

 for

 input

 disp

 while

 end

 if

 ...



## Numeri complessi e formato dell'output

 `>> x = [1 , 3 , 7.5 + 4*i, 6.3]`

`x =`

`1      3      7.5 + 4i      6.3`

 **Formato dell'output**

⇒ `format short`: 5 cifre

⇒ `format long`: 15 cifre

⇒ `format exe`: formato esadecimale

⇒ `format long e`: floating point format with 15 digits.



## Generazione di matrici e polinomi



### Matrice casuale

⇒ `>> A = rand(n,m)`

⇒ Generazione di una matrice ad elementi casuali secondo alcuni parametri definibili dall'utente (distribuzione, valore medio, varianza, seme)



### Rappresentazione di polinomi

⇒  $p(x) = x^3 - 6x + 3$

⇒ `>> p = [1 , 0 , -6 , 3]`



## Operazioni sui polinomi



### Radici di un polinomio

$\Rightarrow$  `>> r = roots(p)`

`r =`

`-2.6691`

`2.1451`

`0.5240`



### Prodotto c di due polinomi (coefficienti a e b)

$\Rightarrow$  `>> c = conv(a,b)`





## Operazioni sui polinomi

### ⇒ Deconvoluzione (divisione) di polinomi

⇒ `>> [q,r] = deconv(a,b)`, con q, quoziente e r, resto

### ⇒ Sviluppo in fratti

⇒ `>> [r,p,k] = residue(n,d)`

⇒ con  $n = s^2 + 6s + 7$  e  $d = s^2 + 5s + 6$

⇒ Sviluppo in fratti:  $\frac{s^2+6s+7}{s^2+5s+6} = \frac{2}{s+3} - \frac{1}{s+2} + 1$

r =	p =	k =
2	-3	1
-1	-2	



## Esercizi Proposti



### Costruzione di una funzione Matlab *obsv()*

- Scrivere un programma che, date le matrici  $A_{n \times n}$  e  $C_{m \times n}$ , costruisca la matrice

$$Q = [C^T, A^T * C^T, \dots, A^{T^{n-1}} * C^T]^T.$$

Successivamente effettuare il test del rango.



## Risoluzione degli esercizi proposti

⇒ **Costruzione della matrice Q**

⇒ **Calcolare**  $Q = [C^T, A^T * C^T, \dots, A^{T^{n-1}} * C^T]^T =$   
 $= [C; C * A; \dots; C * A^{n-1}]$

```
>>Q = [C]; % calcolo passo passo
```

```
>>Q = [Q; C*A];
```

```
>>Q = [Q; C * A^2];
```

```
·      ·  
·      ·  
·      ·
```

```
>>Q = [Q; C * A^(n-1)];
```



## Risoluzione degli esercizi proposti

⇒ **Costruzione della matrice Q in maniera automatica**

⇒ **Utilizzo della funzione `obsv(A,C)`**

```
function Q = obsv(A,C)
```

```
n = size(A,1);
```

```
Q = C;
```

```
for i=1:n-1
```

```
    Q = [C; Q*A];
```

```
end
```

