

Matlab

COMANDI UTILI PER AUTOMATICA

HELP	RTFM
EIG(M)	AUTOVETTORI/VALORI
REAL(M)	Re(z)
IMAG(M)	Im(z)
PLOT(T,V)	GRAFICO DEI VALORI
PLOT3(V1,V2,V3)	TRAIECTORIE SPAZIALI
MAX(M)	MASSIMO
MIN(M)	MINIMO
INV(M)	A^{-1} MATRICE INVERSA
ZEROS(SIZE)	$0_{M,N}$ TUTTI ZERI
ONES(SIZE)	$1_{M,N}$ TUTTI UNI
LSIM(SYS,U,T,X0)	USCITA SISTEMA DINAMICO (SIMULAZIONE)
SS(A,B,C,D)	SISTEMA DINAMICO (STATE SPACE)
TF(NUM,DEN)	FUNZIONE DI TRASFERIMENTO
SS2TF(SYS)	DA SS A TRASFERIMENTO
TF2SS(TF)	DA TRASFERIMENTO A SS
ABS(M)	MODULO / VALORE ASSOLUTO
EYE(SIZE)	MATRICE IDENTITÀ
LOG(M)	LOGARITMO NATURALE
FIGURE	APRE FINESTRA PER PLOT
DCGAIN(SYS)	GUADAGNO SISTEMA
CTRL(M)	RAGGIUNGIBILITÀ
OBSV(M)	OSSERVABILITÀ
DET(M)	DETERMINANTE
TRACE(M)	TRACCIA
NORM(M)	NORMA
POLY(M)	POLINOMIO CARATTERISTICO
ACKER(A,-B,AVAL)	POSIZIONA POLI ANELLO CHIUSO (K,L)
RANK(M)	RANGO MATRICE
ROOTS(V)	RADICI EQUAZIONE GRADO N
INITIAL(SYS)	MOVIMENTO LIBERO
STEP(SYS)	RISPOSTA ALLO SCALINO
IMPULSE(SYS)	RISPOSTA ALL'IMPULSO
BODE(SYS)	DIAGRAMMA BODE
NYQUIST(SYS)	DIAGRAMMA NYQUIST
RLOCUS(SYS)	LUOGO DELLE RADICI

STUDIO STABILITÀ $T_R = 5T_D = -\frac{1}{2\max}$
A = [...]
AVAL = EIG(A)
AVAL-RE = REAL(AVAL) SE CONTINUO
T-DOM = -1 / MAX(AVAL-RE)
AVAL-MOD = ABS(AVAL) SE DISCRETO
T-DOM = -1 / LOG(AVAL-MOD)
T-RES = 5 * T-DOM

STUDIO EQUILIBRIO $\begin{cases} \dot{x} = (I - A)^{-1} b \\ \dot{y} = c^T \dot{x} + d u \end{cases}$
A = [...] , B = [...]
C = [...] , D = [...]
U = [...]
X-EQ = - INV(EYE(SIZE(A)-A)) * B * U
Y-EQ = C * X-EQ + D * U

SIMULAZIONE
SYS = SS(A,B,C,D)
T = 0:0.001:(2*T-RES) \nwarrow SE DISCRETO
U = ONES(SIZE(T))
GAIN = DCGAIN(SYS) * ONES(SIZE(T))
[Y, TIME, STATE] = LSIM(SYS, U, T)

INGRESSO / USCITA
PLOT(TIME, Y)
PLOT(TIME, U)
PLOT(TIME, GAIN)

SPAZIO DEGLI STATI
PLOT(TIME, STATE(:,1))
PLOT(TIME, STATE(:,2))
PLOT(TIME, STATE(:,3))

TRAIECTORIA SPAZIALE
PLOT(STATE(:,1), STATE(:,2), STATE(:,3))

STATO E TRASFERIMENTO
SYS = SS(A,B,C,D)
TF = [NUM, DEN]

CONVERSIONE
SYS [A,B,C,D] = TF2SS(TF)
TF [NUM, DEN] = SS2TF(SYS)

REGOLATORE
A = [...] , B = [...]
C = [...] , D = [...]
AVAL = [...]
R = CTRL(A, B)
D = OBSV(A, C)
DET-R = DET(R) CR SE DET(R) ≠ 0
DET-D = DET(D) CO SE DET(D) ≠ 0
K = ACKER(A, -B, AVAL) CONTROLLORE
L = ACKER(A', -C', AVAL) RICOSTRUZIONE
EIG(A+B*K) VERIFICA CONTROLLORE
EIG(A+L*C) VERIFICA RICOSTRUZIONE

RISPOSTE CANONICHE
SYS = SS(A,B,C,D)
[Y, TIME] = STEP(SYS) SCALINO
PLOT(TIME, Y)
[Y, TIME] = IMPULSE(SYS) IMPULSO
PLOT(TIME, Y)

SIMULINK