ΣΥΣ 201 Εργαστήριο

Matlab
(με έμφαση στο Control Toolbox)

ΕΔΙΠ Μανόλης Ντουντουνάκης

Skype: manolis.doudounakis

Φεβρουάριος 2021
Ver. 1.0

Μετρικές (norm) διανυσμάτων

```
a= [ 1 2 3];
>> norm(a,1)

ans =

     6

>> norm(a,2)

ans =

    3.7417

>> norm(a,inf)

ans =

     3
```

Βαθμός (rank) Πίνακα

```
a=[0 1 1 2; 1 2 3 4; 2 0 2 0];
>> rank(a)

ans =

     2
```

ΓΡΑΜΜΙΚΗ ΑΛΓΕΒΡΑ

d = det(A), επιστρέφει την ορίζουσα ενός τετραγωνικού πίνακα A.

Y = inv(X), X. επιστρέφει τον αντίστροφο Πίνακα ενός τετραγωνικού πίνακα

```
>> b=[1 2 5; 2 4 6; 7 3 11];
>> inv(b)

ans =

   -0.5909    0.1591    0.1818
   -0.4545    0.5455   -0.0909
    0.5000   -0.2500         0
```

<u>ilaplace,</u>

Υπολογίζει τον αντίστροφο Laplace μετασχηματισμό της F όσον αφορά την μεταβλητή trans_var στο σημείο eval_point.

ilaplace(F,trans_var,eval_point)

```
>> syms x y
>> F = 1/y^2;
>> ilaplace(F, y, x)

ans =

x
```

The MATLAB System Control Toolbox

1 Δημιουργία μοντέλων

- Συναρτήσεις μεταφοράς (Transfer Function - TF)

- Μηδενικά, πόλοι και κέρδος (zeros, poles, and gain - ZPK)

- Μεταβλητές κατάστασης (State space models - SS)

2 Συναρτήσεις μεταφοράς

Ορίζονται από τα πολυώνυμα του αριθμητή και του παρανομαστή

$$H(s) = \frac{p_1 s^n + p_2 s^{n-1} + \cdot + p_{n+1}}{q_1 s^m + q_2 s^{m-1} + \cdot + q_{m+1}}$$

Παράδειγμα

$$H(s) = \frac{s}{s^2 + 2s + 10}$$

```
>> num = [1 0];        % Numerador -> s
>> den = [1 2 10]; %Denominador:-> s^2 + 2s + 10
>> H = tf (num,den)

Transfer function:
    s
--------------
s^2 + 2 s + 10
```

Εναλλακτικά

```
>> s = tf('s');% Create Laplace variable
>> H = s/(s^2+2*s+10)

Transfer function:
    s
--------------
s^2 + 2 s + 10
```

3 Zeros, Poles, and Gain

$$H(s) = k \frac{(s - z_1) \cdot (s - z_n)}{(s - p_1) \cdot (s - p_m)}$$

Παράδειγμα

$$H(s) = 2\frac{(s+1)}{(s-2)(s^2-2s+2)}$$

```
>> z = [-1]; % Zeros
>> p = [2 1+i 1-i]; % Polos
>> k = 2; % Gain
>> H = zpk(z,p,k)

Zero/pole/gain:
       2 (s+1)
---------------------
(s-2) (s^2  - 2s + 2)
```

Εναλλακτικά

```
>> s = zpk('s');
>> H = 2*(s+1)/((s-2)*(s^2-2*s+2))

Zero/pole/gain:
   2 (s+1)
-----------
(s-2) (s^2  - 2s+2)
```

4 State Space

$$\frac{dx}{dt} = A\,x + B\,u$$
$$y = C\,x + D\,u$$

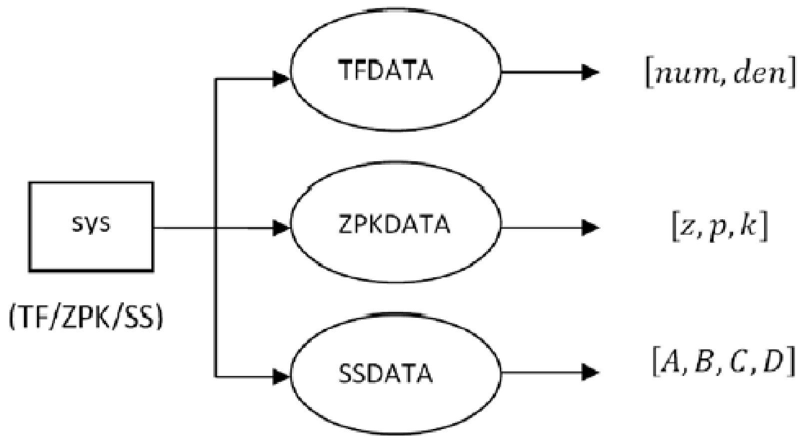
Παράδειγμα

$$\frac{dx}{dt} = Ax + BI \quad ; \quad A = \begin{bmatrix} 0 & 1 \\ -5 & -2 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$
$$y = Cx + Du \quad ; \quad C = [1 \quad 0] \quad D = [0]$$

```
>> A = [0 1;-5 -2];
>> B = [0; 3];
>> C = [1 0];
>> D = 0;
>> H = ss(A,B,C,D)
a =

       x1  x2
   x1   0   1
   x2  -5  -2


b =
       u1
   x1   0
   x2   3

c =
       x1  x2
   y1   1   0

d =
       u1
   y1   0

Continuous-time model.
```

5

```
>> [num,den] = tfdata (sistema, 'v')
>> [z,p,k] = zpkdata(sistema,'v')
>> [A,B,C,D] = ssdata(sistema)
```



Παράδειγμα

Εστω:

```
>> h = tf([1 1],[1 2 5])
>> [num,den] = tfdata(h,'v')
```

[**z,p,k]=tf2zp(num,den)**

```
>> num=[1];
>> den=[110 1000 200];
>> [z,p,k]=tf2zp(num,den)

z =

   Empty matrix: 0-by-1


p =

   -8.8863
   -0.2046


k =

    0.0091
```

[a,b,c,d]=tf2ss(num,den)

```
>> num=[0 1 0];
>> den=[1 1 1];
>> [a,b,c,d]=tf2ss(num,den)

a =

  -1  -1
   1   0


b =

   1
   0


c =

   1   0


d =

   0
```

[num,den]=ss2tf(a,b,c,d,i)

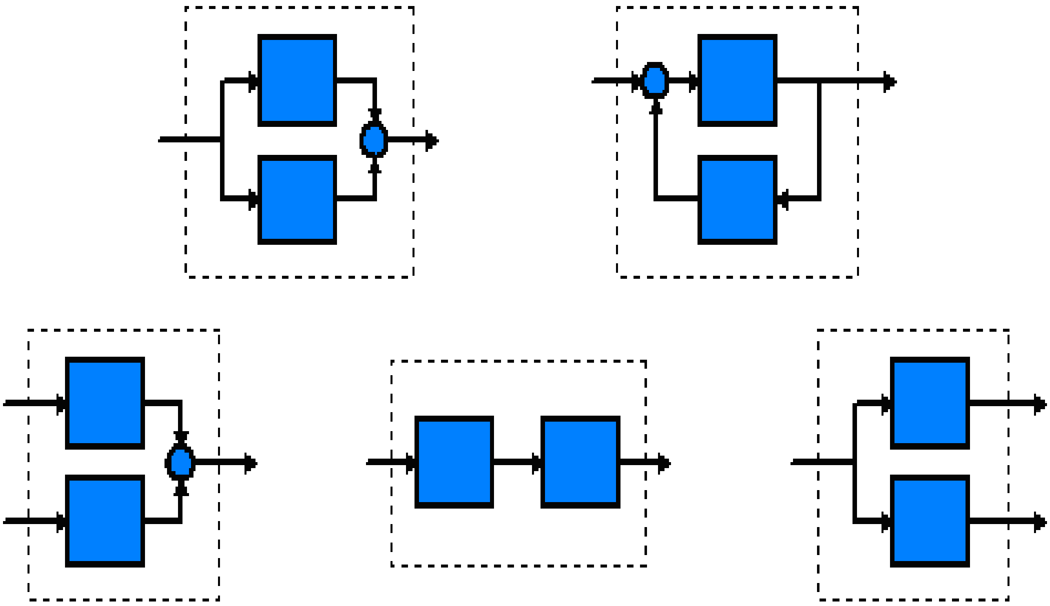
Υπολογίζει την i-στήλη του Πίνακα μεταφοράς.

Δημιουργία σύνθετων μοντέλων

•*series, parallel*

•*Feedback*

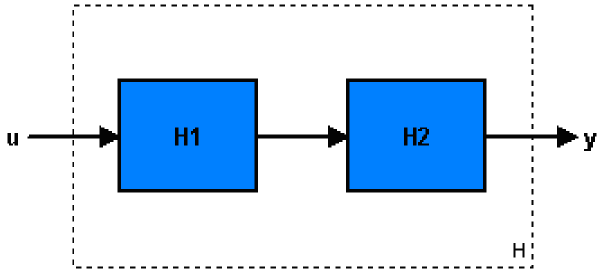
•Concatenation ([,], [;] and *append*)



```
>> H = series(H1, H2);
```

or else

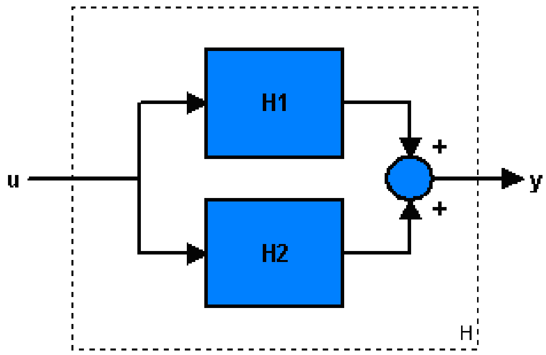
```
>> H = H2 * H1;
```



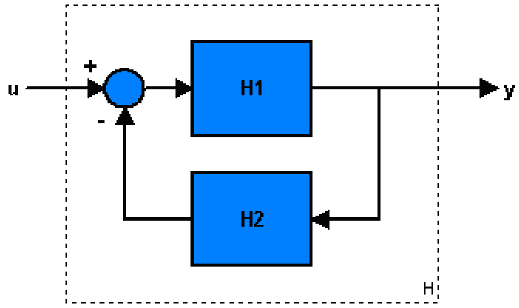

```
>> H = parallel(H1, H2);
```

or else

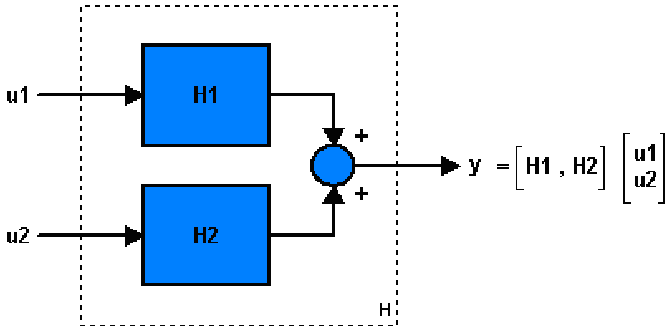
```
>> H = H2 + H1;
```



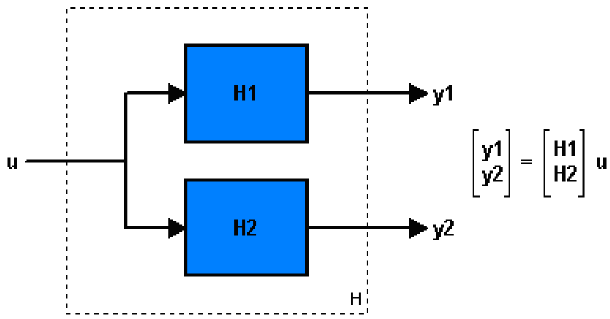

```
>> H = feedback (H1, H2);
```



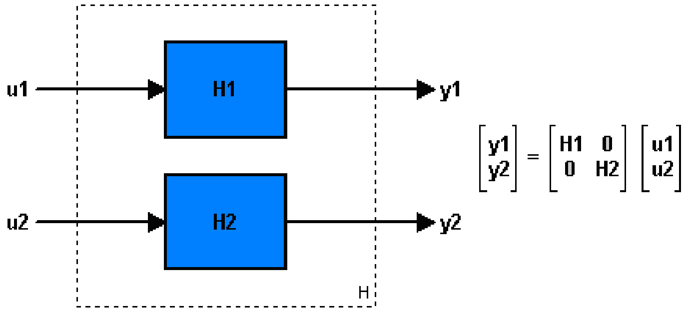

```
>> H = [H1,H2];
```




```
>> H = [H1; H2];
```

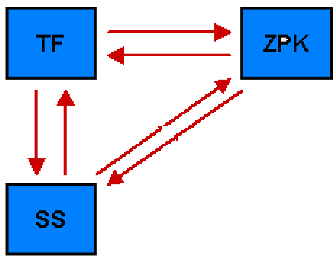



```
>> H = append(H1,H2);
```



7 Μετατροπές μορφών

```
>> System = tf(system);      % TF model converts
>> System = ZPK(system);     % ZPK model converts
>> System = ss (system);     % converts the SS model
```



Παράδειγμα

```
>> Gs= ss (-2,1,1,3);
```

```
>> Gz = zpk(Gs)
```

Βηματικη απόκριση

```
>> step(sys,t)
```

$t = t_0: \Delta t: t_f$

```
>> [y,t,x] = step(sys)
```

Impulse

```
>> impulse(sys,t)
>> [y,t,x] = impulse(sys)
```

$u(t), t = t_o: \Delta t: t_f$

```
>> lsim(sys,u,t)
>>[y,t,x] = lsim(sys,u,t)
```

Παράδειγμα

```
>> s=zpk('s');
>> G=10/((s+2)*(s+5));
>> t=0:0.1:5;
>> step(G,t)
>> grid
```

Παράδειγμα

$G(s) = \frac{5}{s^3+2s^2+s+1}$ $u(t) = 2e^{-t} \sin(\pi t),\ 0 < t < 10.$

```
>> s=tf('s');
>>
G=5/(s^3+2*s^2+s+1);
>> t=0:0.2:30;
```

```
>> u=2*exp(-t).*sin(pi*t);
>> lsim(G,u,t)
>> grid
```

.

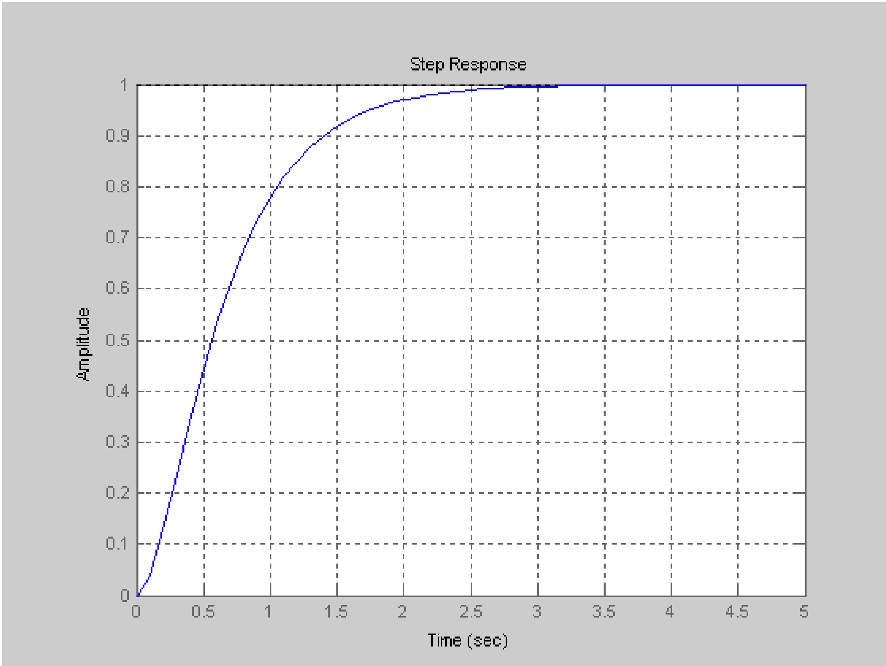


Fig. A.11 Step response of $G(s) = \frac{10}{(s+2)(s+5)}$

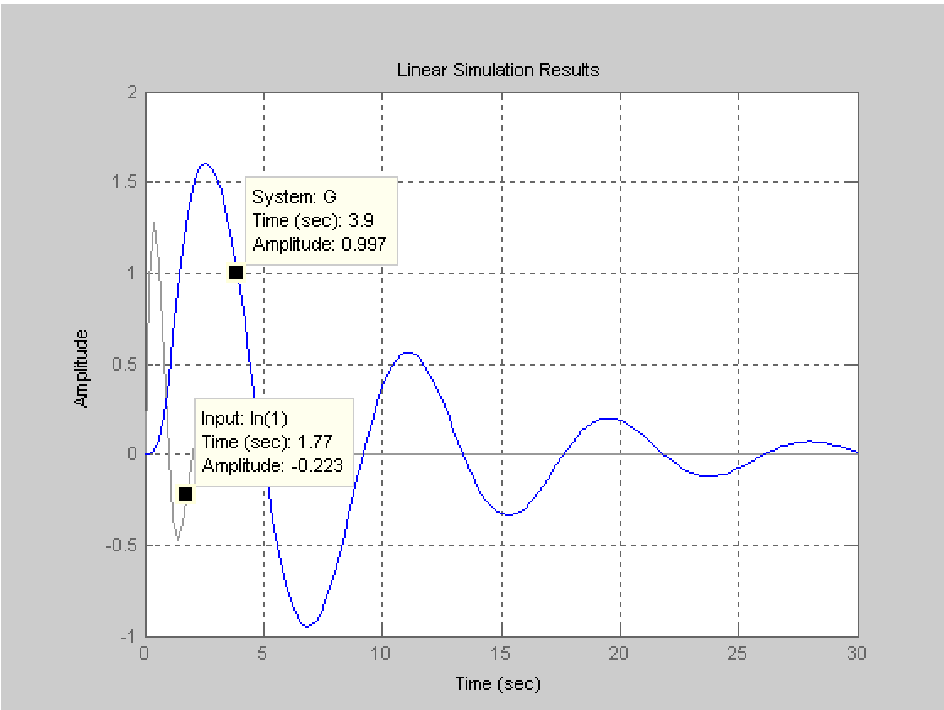


Fig. A.12 Time response for $G(s) = \frac{5}{s^3+2s^2+s+1}$.

"**lsim“, "gensig“, "square", "sawtooth", and "stepfun".**

"lsim" μπορεί να δεχτεί κάθε σήμα εισόδου.

"gensig", παράγει ημίτονα, τετραγωνικούς, ή περιοδικούς παλμούς.
Let us as an example simulate H1c with a sinusoidal input, u, having
Περίοδος: Tp=0.5,
τελικός χρόνος: Tf=10,
βήμα: Tstep=0.01:
Αρχικός χρόνος:Tp=0.5;

```
[u,t]=gensig('sin',Tp,Tf,Tstep);
H= tf([1 0], [2 1 1]);
lsim(H1c,u,t)
```

Άλλες γεννήτριες σημάτων είναι:

- "square",
- "sawtooth", και
- "stepfun".

Ορισμός και πρόσβαση σε ιδιότητες ΓΧΑ συστημάτων

Name	Description	Applies to LTI-object of type
Ts	Sample Time	All
Td	Input delays	All (only continuous systems)
Inputname	Input names	All
Outputname	Output names	All
Notes	Notes on model history	All
Userdata	Additional data	All
Num	Numerator	TF
Den	Denominators	TF
Variable	Transfer function variable (e.g. 's' or 'z')	TF, ZPK
Z	Zeros	ZPK
P	Poles	ZPK
K	Gains	ZPK
A	Matrix A	SS
B	Matrix B	SS
C	Matrix C	SS
D	Matrix D	SS

Με τη "get" μπορούμε να δούμε τις ιδιότητες, π.χ. μιας συνάρτησης μεταφοράς :

```
get(H)
```

MATLAB's response is

num={1x2 cell}

den={1x2 cell}

Variable='s'

Ts=0

Td=[1 2]

InputName={2x1 cell}

OutputName={''}

Notes={}

UserData=[]

Μπορούμε να έχουμε πρόσβαση σε συγκεκριμένες ιδιότητες σε ΓΧΑ συστήματα:

```
Tdvec=get(H,'Td')
```

Tdve

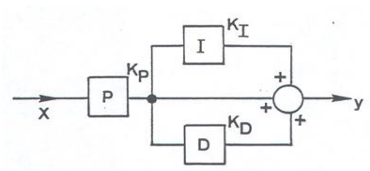
c= 1

2

Αντίστοιχα με τη *set μπορούμε να δώσουμε τιμή σε συγκεκριμένες ιδιότητες ΓΧΑ συστημάτων.* :

```
set(H,'InputName',{'Voltage';'Flow'})
```

```
H
```



$$F_R = \frac{K_p}{s}\left[s + \frac{1}{T_i} + T_d \cdot s^2\right]$$

PID Ελεγκτής του εργαστηρίου

Υλοποιείται στο Control Toolbox του Matlab με τη συνάρτηση pidstd(Kp,Ti,Td).

Εναλλακτικά υλοποιείται με το apps PID Tuner, Form Standard.

Συνήθη λάθη

- Λάθος επιλογή συνάρτησης
- Λάθος χρήση συνάρτησης (Λάθος παραδοχές)
- Εσφαλμένη επιλογή χώρου παρατήρησης