FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
DEPARTAMENTUL DE AUTOMATICĂ ŞI INFORMATICĂ INDUSTRIALĂ
Bd.Vasile Pârvan 2
1900 Timişoara
Romania
Tel./Fax.: +40 256 40 32 10; +40 256 40 32 41

TEORIA SISTEMELOR I – LUCRAREA DE LABORATOR 4 (2 şedințe)

DISCRETIZAREA SISTEMELOR

A. Obiective

- Fixarea cunoştințelor legate de:
  -structura sistemelor de reglare numerice şi problemele de discretizare corespunzătoare;
  -discretizarea proceselor ca realizare invariantă la semnal treaptă;
  -discretizarea algoritmelor de reglare.
- Familiarizarea cu unele dintre posibilitățile programelor Matlab şi Simulink pentru analiza sistemelor de reglare numerice.

B. Considerații pregătitoare

B.1. Tipuri de probleme de discretizare

La realizarea sistemelor de reglare numerică a proceselor analogice considerate ca sisteme în timp continuu conduse numeric, fig.1, se pot pune două probleme de discretizare:

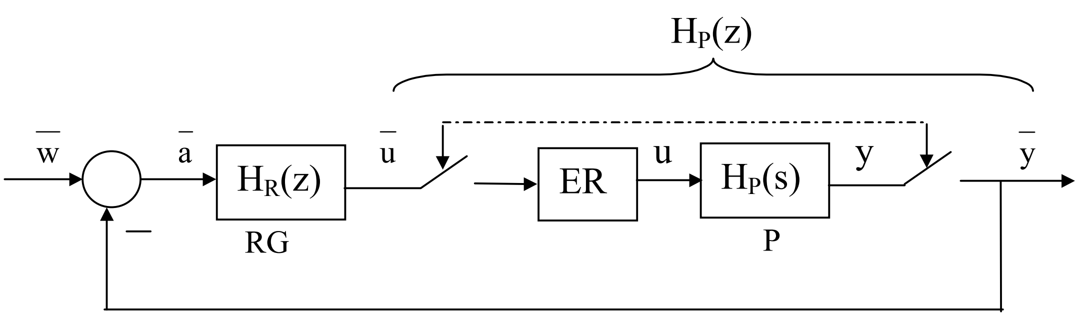


Fig.1. ***S**istem de **R**eglare **A**utomată **N**umerică* **($\overline{w}$ - mărime de conducere, $\overline{a}$ - eroare de reglare, $\overline{u}$ - mărime de comandă numerică, u - mărime de comandă (tip scară), y – mărime de ieşire, $\overline{y}$ - mărime de ieşire numerică, P – proces condus, RG – regulator numeric, ER - element de eşantionare şi reținere).**

- *Stabilirea unui model în timp discret ataşat modelului în timp continuu al procesului P, de exemplu ataşarea unei f.d.t. $H_P(z)$ unei f.d.t. $H_P(s)$, în cazul în care pentru procesul P se doreşte a se proiecta în mod direct un algoritm de reglare numerică, de exemplu sub forma unei f.d.t. $H_R(z)$.*

Atunci când mărimea de la intrarea procesului provine de la un element de reținere, având forma bine precizată de semnal în scară, este posibil să se determine un model în timp discret care să reproducă cu exactitate la momentele de discretizare comportarea procesului în timp continuu. Operația de discretizare se numeşte ***discretizare ca realizare invariantă la semnal treaptă*** *(r.i.s.t.).* Calculele se pot face atât în domeniul timp cât şi în domeniul complex.

- *Obținerea unui algoritm de reglare în timp discret, de exemplu sub forma unei f.d.t. H_R (z), care să aproximeze cât mai bine comportarea algoritmului de reglare în timp continuu, în cazul în care pentru procesul P a fost proiectat un algoritm de reglare în timp continuu, de exemplu sub forma unei f.d.t. H_R (s).*

Aproximarea se face prin aşa-numita operație de ***discretizare prin aproximare*** a algoritmului de reglare în timp continuu. Metodele de discretizare folosite provin din tehnicile de integrare numerică. Ele sunt: metoda dreptunghiului retardată (**MDR**), metoda dreptunghiului avansată (**MDA**) şi metoda trapezului (**MT**)[1)]. Metodele pot fi folosite sub diverse forme: în domeniul timp sau în domeniul complex. În cadrul lucrării se foloseşte forma din domeniul complex cunoscută sub denumirea de ***metoda substituției***.

Cu privire la cele prezentate sunt importante şi următoarele două aspecte:

i) Cele două metode se folosesc pentru subsisteme distincte ale sistemului de reglare.

ii) Pentru un sistem de reglare dat se foloseşte, în funcție de modul în care se proiectează acesta, numai unul din procedeele de discretizare.

B.2. Discretizarea proceselor ca realizare invariantă la semnal treaptă

Potrivit celor de la punctual B.1 "*principiul realizării invariante la semnal treaptă*" se exprimă astfel: o realizare invariantă la semnal treaptă asociată unui sistem în timp continuu pentru un pas de discretizare h este un model în timp discret al cărui răspuns la un semnal discret aplicat coincide cu răspunsul de la momentele de discretizare ale sistemului în timp continuu din care a provenit în situația în care la intrarea acestuia se aplică semnalul scară asociat semnalului discret printr-un element de reținere care lucrează cu perioada h. Pe durata h a unei perioade de eşantionare semnalul de la intrarea procesului are forma de semnal treaptă iar în ansamblu de "semnal în scară". Verificarea practică a principiului se face folosind schema bloc din fig. 2.

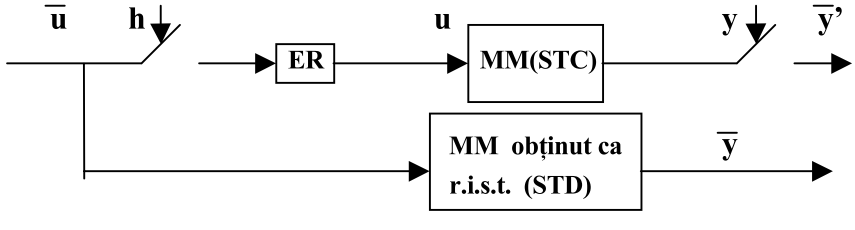


Fig.2

[1)] MDR este denumită şi metodă Euler iar MT este numită şi metodă Tustin.

Mărimea ū este un semnal în timp discret cu pasul de discretizare h ; la fel **ȳ** şi **ȳ'**; iar mărimile u şi y sunt semnale în timp continuu. Semnalele **y** şi **y'** trebuie să coincidă**.**

Discretizarea ca r.i.s.t. se poate face în domeniul timp sau în domeniul complex. În domeniul timp calculele se pot efectua analitic sau numeric, utilizând mediul MATLAB.

- *Dicretizarea analitică în domeniul timp ca r.i.s.t. a MM-ISI al unui sistem fără timp mort* descris prin relaţiile ($x \in \mathbf{R}^n$, $u \in \mathbf{R}^q$, $y \in \mathbf{R}^p$):

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \quad x(t_o) = x_o \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{1}$$

este reprezentată de realizarea sistemică :

$$\begin{cases} \dot{x}(t) = \Phi x(t) + \Gamma u(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{2}$$

în care:

$$\begin{cases} \Phi = \Phi(t)\big|_{t=h} = e^{Ah}, \ (= A_d) \\ \Gamma = \int_0^h e^{Av} B dv, \quad (= B_d) \\ C_d = C, \quad D_d = D \end{cases} \tag{3}$$

Pentru calculul matricei Γ se pot folosi diferite metode. Se pot folosi funcţii Matlab destinate integrării funcţiilor sau ecuaţiilor diferenţiale (exemplul 1) sau modelări numerice intermediare (exemplul 2).

Exemplul 1. Calculul matricei Γ din relaţiile (3) în cazul unui sistem de ordinul II.

Să se calculeze matricea Γ pentru STC cu f.d.t. $H(s) = \frac{1}{s(s+2)}$, *pasul de discretizare fiind* h = 0.2 sec.

Sistemului îi corespunde realizarea sistemică $\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -2 & -10 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot [u] \\ [y] = [1 \quad 0] \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{cases}$, respectiv matricea de tranziţie $e^{At} = \begin{bmatrix} e^{-2t} & 5(1-e^{-2t}) \\ 0 & 1 \end{bmatrix}$, $t \geq 0$. Pentru calculul celor două integrale care apar în acest caz în matricea Γ se poate folosi următorul fişier Matlab pe care îl vom numi exerc.m.

```
function dx=gamma(t,x)
%fisier functie in care se definesc cele doua functii de integrat;
%acestea sunt componentele unui vector coloana cu 2 elemente
dx=zeros(2,1);                          %generarea vectorului coloana
dx(1)=exp(-2*t)*0+5*(1-exp(-2*t))*1;    %prima functie
dx(2)=0*0+1*1;                          %a doua functie
```

Fişierul face uz de funcția **ode23** destinată integrării sistemelor de ecuații diferențiale de ordinul I folosind un algoritm compus din algoritme de tip Runge-Kutta de ordinul 2 şi 3. Funcțiile integrate sunt definite în fişierul funcție gamma.m, amplasat în acelaşi director cu fişierul exerc.m. Fişierul exerc.m conține şi instrucțiunea de reprezentare grafică a integrării pe intervalul [0,0.2] şi instrucțiuni de extragere a valorilor integralelor.

```
%fisier script pentru calculul integralelor
[tn,xn]=ode23('gamma',[0,0.2],[0;0]);  %integrarea celor doua functii
plot(tn,xn(:,1),'b',tn,xn(:,2),'r')          %reprezentarea grafica a celor doua functii
[r,c]=size(xn);
disp('gama1='); xn(r,1)                        %valoarea primei integrale
disp('gama2='); xn(r,2)                        %valoarea celei de-a doua integrale
```

- *Dicretizarea analitică în domeniul complex ca r.i.s.t. a MM-II al unui sistem fără timp mort* descris prin f.d.t. H(s) rațională, edste f.d.t. H(z) obținută cu formula:

$$H(z) = (1 - z^{-1}) Z\left\{\frac{1}{s}H(s)\right\} \tag{4}$$

$Z\{\bullet\}$ reprezintă „transformata Z“ a şirului care se obține prin discretizarea originalului funcției $\frac{1}{s}H(s)$ la momentele t = kh , k∈**N**. $Z\{f(s)\}$ se găseşte în tabelele de transformare pe coloana a III-a pentru expresiile lui f(s) identificate pe coloana a II-a.

- *Dicretizarea analitică în domeniul timp ca r.i.s.t. a MM-ISI al unui sistem cu timp mort* necesită descompunerea în prealabil a timpului mort τ sub forma

$$\tau = \theta\, h - \vartheta\, h, \qquad \theta \in N^*, \quad \vartheta \in [0,1) \ . \tag{5}$$

Dacă sistemul este descris prin MM – ISI :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{6}$$

atunci modelul obținut prin discretizare este:

$$\begin{cases} x(t+1) = \Phi\, x(t) + \Gamma_1\, u(t-\theta) + \Gamma_0\, u(t-\theta+1) \\ y(t) = C\, x(t) \end{cases} \tag{7}$$

unde:

$$\Phi = e^{Ah}, \quad \Gamma_1 = e^{A\vartheta h}\int_0^{h'} e^{Av}B\,dv, \quad \Gamma_0 = \int_0^{\vartheta h} e^{Av}B\,dv, \quad h' = (1-\vartheta)h \tag{8}$$

Se remarcă faptul că ecuația de stare din (7) nu are forma canonică. Aducerea la forma canonică se face prin introducerea a $\theta \cdot$m variabile de stare auxiliare. De exemplu, pentru cazul în care timpul mort este mai mic decît pasul de discretizare, ($\tau < h$), avem $\boldsymbol{\theta} = 1$ şi se introduce variabila de stare auxiliară $x_a(t)$ cu $1 \cdot m = m$ componente prin relația $x_a(t+1) = u(t)$ **.** Ordinul modelului discret rezultat devine *n+m* iar MM-ISI discret al procesului ia forma :

$$\begin{cases} \begin{bmatrix} x(t+1) \\ \ldots\ldots\ldots \\ x_a(t+1) \end{bmatrix} = \begin{bmatrix} \Phi & \vdots & \Gamma_1 \\ \cdots & \vdots & \cdots \\ 0 & \vdots & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ x_a(t) \end{bmatrix} + \begin{bmatrix} \Gamma_0 \\ \cdots \\ I \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} C & \vdots & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \ldots\ldots \\ x_a(t) \end{bmatrix} \end{cases} \tag{9}$$

♦ *Dicretizarea analitică în domeniul complex ca r.i.s.t. a MM-II al unui sistem cu timp mort* se bazează pe relația (5). Dacă procesul este descris prin f.d.t.

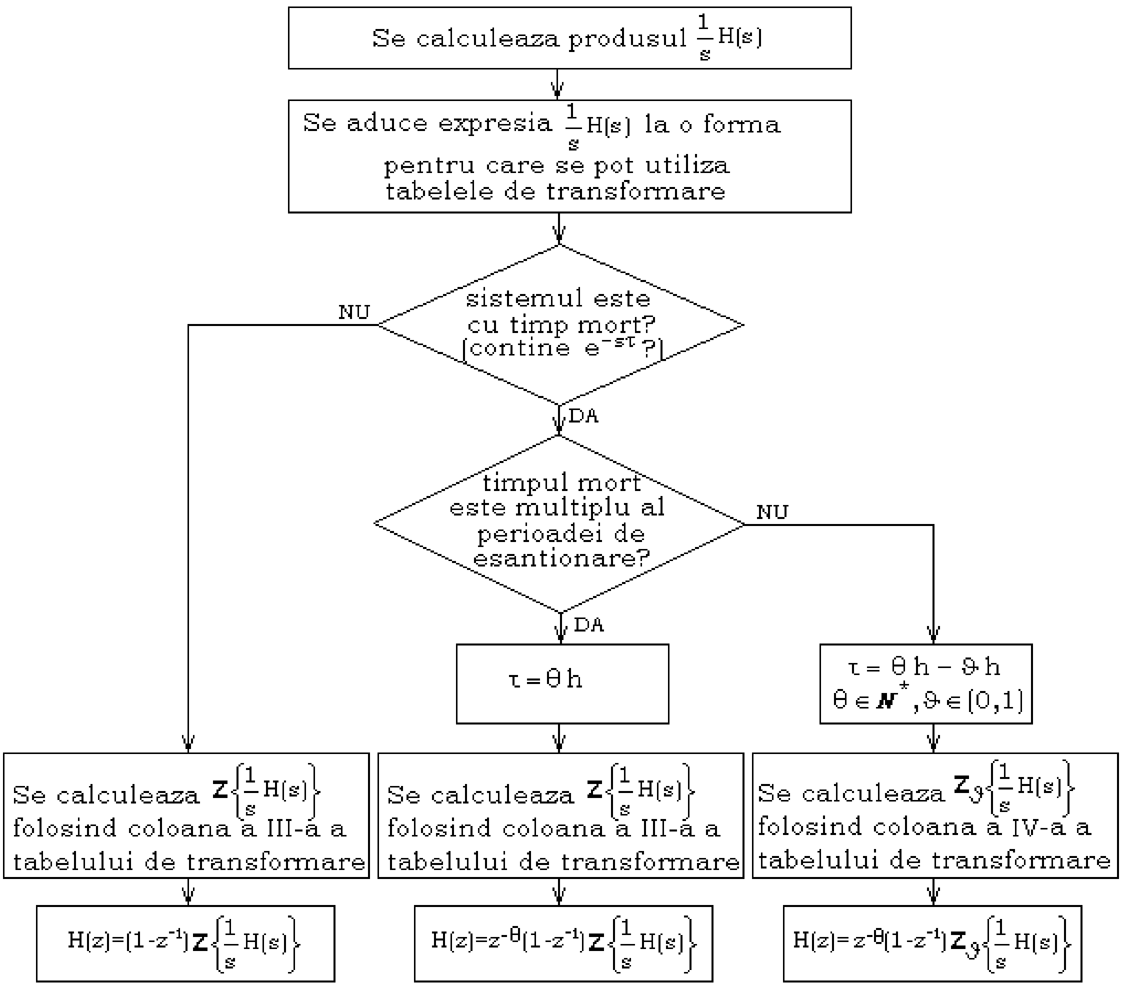
$$H(s) = H_1(s) \cdot e^{-\tau s} \tag{10}$$

unde $H_1(s)$ este o rațională strict proprie, atunci r.i.s.t. H(z) se obține cu relația

$$H(z) = \left(1 - z^{-1}\right) z^{-\theta} Z_{\vartheta}\left\{\frac{1}{s} H_1(s)\right\} \tag{11}$$

unde: $Z_{\vartheta}\{\bullet\}$ reprezintă „transformata z modificată". $Z_{\vartheta}\{f(s)\}$ se găseşte în tabelele de transformare pe coloana a IV-a şi se asociază expresiei f(s) identificate pe coloana a II-a.

Metoda de calcul este sintetizată în organigrama de mai jos:



♦ *Dicretizarea numerică în domeniul timp ca r.i.s.t. a MM-ISI al unui STC fără timp mort* se poate realiza folosind funcția Matlab **c2d** care calculează matricele ecuației de stare a modelului discret. Ea are sintaxa:

$$[Ad,Bd] = c2d\ (A,B,h) \tag{12}$$

unde: A, B - sunt matricele din MM-ISI (1), Ad, Bd - sunt matricele din MM - ISI discret (2)-(3) iar h este pasul de discretizare.

Ecuația de ieşire din (1) fiind o ecuație algebrică ea este valabilă în orice moment, prin urmare este valabilă şi în momentele de discretizare. Astfel, ecuația de ieşire din (2) are aceeaşi formă cu cea din (1).

Exemplu 2. Discretizarea numerică a unui STC de ordinul II folosind funcția Matlab c2d.

Să se determine MM discret asociat ca r.i.s.t. pentru un proces de ordinul 2 cu amplificarea unitară, factorul de amortizare 0.2 şi pulsația naturală 1 sec^{-1}, pentru un pas de discretizare h = 0.25 sec.

Întrucât prin enunț nu se dă un MM-ISI al STC ci parametri acestuia pentru forma canonică

$$\ddot{y}(t) + 2 \cdot \zeta \cdot \omega_n \cdot \dot{y}(t) + \omega_n^2 \cdot y(t) = \omega_n^2 \cdot u(t) \tag{13}$$

programul Matlab va utiliza în afara funcției c2d şi funcția Matlab ord2 pentru determinarea unui MM-ISI al STC (13). Sintaxa ei este: [A,B,C,D] = ord2(ω_n, ζ). Funcția ord2 returnează matricele A,B,C,D ale MM-ISI (1). Programul Matlab care soluționează problema pusă este:

```
[A,B,C,D] = ord2(1, .2)
[Ad,Bd] = c2d(A,B,0.25)
```

MM – ISI discret rezultat este:

$$\begin{cases} \begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 0.9699 & 0.2354 \\ -0.2354 & 0.8757 \end{bmatrix}}_{\Phi = A_d = Ad} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \underbrace{\begin{bmatrix} 0.0301 \\ 0.2354 \end{bmatrix}}_{\Gamma = B_d = Bd} \cdot u(t) \\ y(t) = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \underbrace{[0]}_{D} \cdot u(t) \end{cases} . \tag{14}$$

B.3. Discretizarea Algoritmelor de Reglare în Timp Continuu prin Metoda Substituției

Discretizarea prin aproximație a algoritmelor de reglare în timp continuu prin metoda substituției se bazează pe relațiile de substituție (abrevierele se referă la cele trei variante: metoda dreptunghiului retardată (**MDR**), metoda dreptunghiului avansată (**MDA**) şi metoda trapezului (**MT**)):

$$\frac{1}{s} \cong \begin{cases} \dfrac{h}{z-1} & \text{(MDR)} \\ \dfrac{h \cdot z}{z-1} & \text{(MDA)} \\ \frac{1}{2} h \cdot \dfrac{z+1}{z-1} & \text{(MT)} \end{cases} \tag{15}$$

Considerând pasul de discretizare h al regulatorului numeric dat, etapele de aplicare sunt:

- Se determină f.d.t. $H_R(s)$ a regulatorului
- Se calculează f.d.t. a regulatorului în timp discret $H_R(z)$ folosind din (15) o variantă aleasă pentru substituție. Bunăoară, în cazul metodei trapezului se foloseşte relația (16)

$$H(z) = H(s)\Big|_{\frac{1}{s}=\frac{hz}{z-1}} . \tag{16}$$

Alegerea unei metode de discretizare prin aproximare pentru o aplicație dată se face luând în considerare precizia de aproximare a algoritmului în timp continuu, complexitatea relațiilor de calcul a parametrilor algoritmului în timp discret şi capacitatea de structurare a algoritmului în timp discret în vederea implementării cât mai simple şi cât mai flexibilite.

Exemplul 3. Discretizarea unui algoritm PI

Să se discretizeze cu metoda trapezului algoritmul de reglare PI

$$H(s) = K(1 + \frac{1}{sT_i}) \tag{17}$$

Se face în (17) substituția corespunzătoare ultimei relații (15) şi se obține:

$$H(z) = H(s)\Big|_{\frac{1}{s}=\frac{2}{h}\frac{z-1}{z+1}} K\left(1 + \frac{1}{\frac{2}{h}\frac{z-1}{z+1}T_i}\right) = \frac{\frac{K(h+2T_i)}{2T_i}z + \frac{K(h-2T_i)}{2T_i}}{z-1} = \frac{\frac{K(h+2T_i)}{2T_i} + \frac{K(h-2T_i)}{2T_i}z^{-1}}{1-z^{-1}} \tag{18}$$

Considerând că H(z) = u(z) / a(z) din (18) rezultă MM-II

$$u(t) = u(t-1) = K\frac{h + 2\,T_i}{2\,T_i}a(t) + K\frac{h - 2\,T_i}{2\,T_i}a(t-1) \tag{19}$$

B.4. Scheme de Verificare a Modelelor Discretizate

- Pentru studiul discretizării ca r.i.s.t. se pot folosi scheme ca şi cea din fig. 3. Ea corespunde principiului prezentat în fig. 2. Cazul indicat corespunde unui STC cu f.d.t. $H(s) = \frac{2}{s+2}$ căruia i se ataşaza ca r.i.s.t. în cazul h = 0.2 sec. un STD cu f.d.t. $H(z) = \frac{1-\exp(-0.4)}{z-\exp(-0.4)}$.

Cu privire la utilizarea schemei Simulink din fig. 3 sunt importante următoarele observații şi precizări:

- În blocurile *step*, *zero-order hold* şi *discret transfer function* se introduce pasul de eşantionare sub formă literară *h*. Valoarea aleasă a pasului de eşantionare *h* se dă în fereastra Matlab înainte de simulare.
- În blocul *transfer function* se introduce MM-II al STC.
- În blocul *discret transfer function* se introduce MM-II obținut ca r.i.s.t. (STD) al sistemului din blocul *transfer function.*

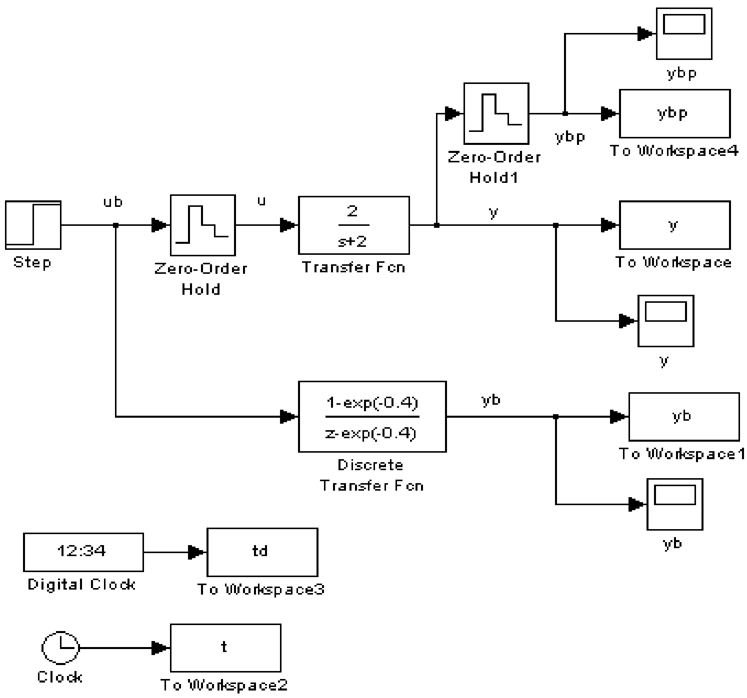


Fig. 3. Exemplu de schemă Simulink pentru studiul discretizării ca r.i.s.t.

- Variabilele *t* şi *td* din schema Simulink corespund timpului continuu şi respectiv timpului discret.
- Semnalele ale căror denumiri conțin litera b (barat) sunt semnale în timp discret corespunzătoare momentelor de eşantionare *td*, iar semnalele ale căror denumiri nu conțin litera *b* sunt semnale în timp continuu.
- Pentru vizualizare în schemă au fost prevăzute osciloscoape; modul de interpretare al imaginilor se va discuta la laborator. Pentru vizualizarea în mediul Matlab se procedează astfel:

```
» h=0.2;
»
» plot(t,y,'-',td,yb,'*'),grid
»
» plot(td,ybp,'*',td,yb,'*'),grid
```

♦ Pentru studiul discretizării ca r.i.s.t. se pot folosi scheme ca şi cea din fig. 4. Ea corespunde unui STC cu f.d.t. $H(s) = \frac{3}{5s+2}$ şi discretizatului acestuia obținut cu MDA în cazul unui pas de discretizare h $H(z) = \frac{3 \cdot h \cdot z}{(5+2 \cdot h) \cdot z - 5}$. Cu privire la utilizarea acestei scheme Simulink subliniem următoarele aspecte:

- Parametrii sistemului în timp continuu se introduc în blocul *transfer function*.
- Parametrii sistemului în timp discret se introduc în blocul *discret transfer function*.
- În blocurile *zero order hold* şi *discret transfer function* se introduce pasul de eşantionare sub formă literară *h*.
- Pentru vizualizarea formelor de variație ale semnalelor *u* şi *ub* se folosesc o instrucție similară ca în cazul exemplului din fig. 3.

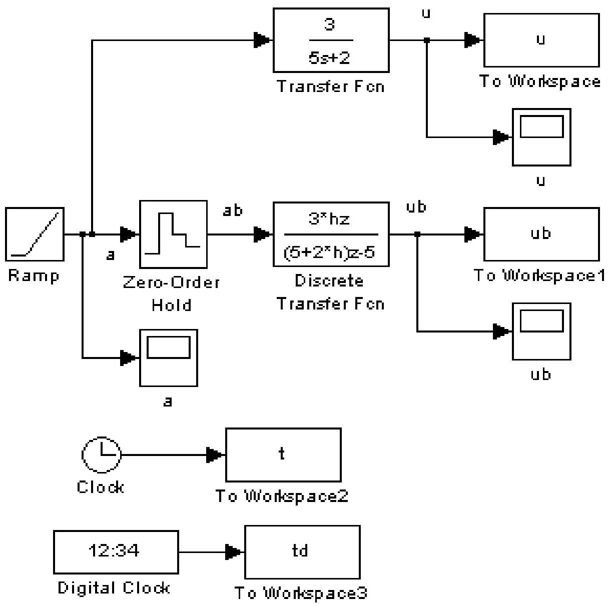


Fig. 4. Exemplu de schemă Simulink pentru studiul metodelor de discretizare prin aproximare.

♦ Pentru exemplificarea studiului sistemelor cu timp mort considerăm cazul unui proces cu *funcția de transfer*: $H(s) = \frac{1.5}{0.5s+1} e^{-1.5t}$. Sunt posibile două variante de abordare în funcție de succesiunea înserierii părții raționale $H(s) = \frac{1.5}{0.5s+1}$ şi părții iraționale $e^{-1.5t}$ ale lui H(s).

Prima variantă de abordare se bazează pe schema simulink din fig. 5:

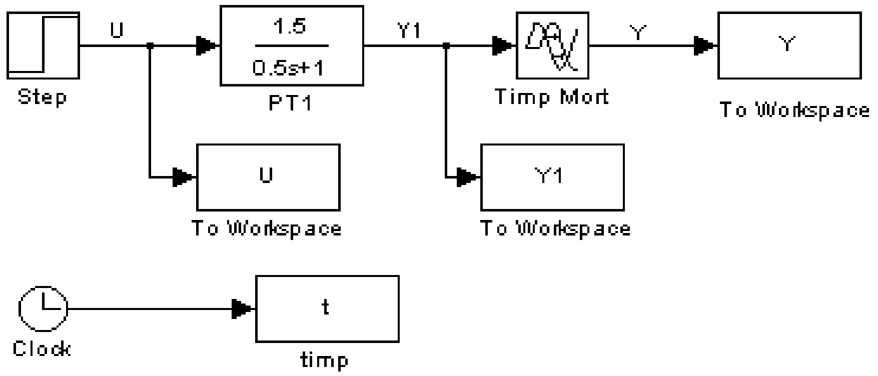
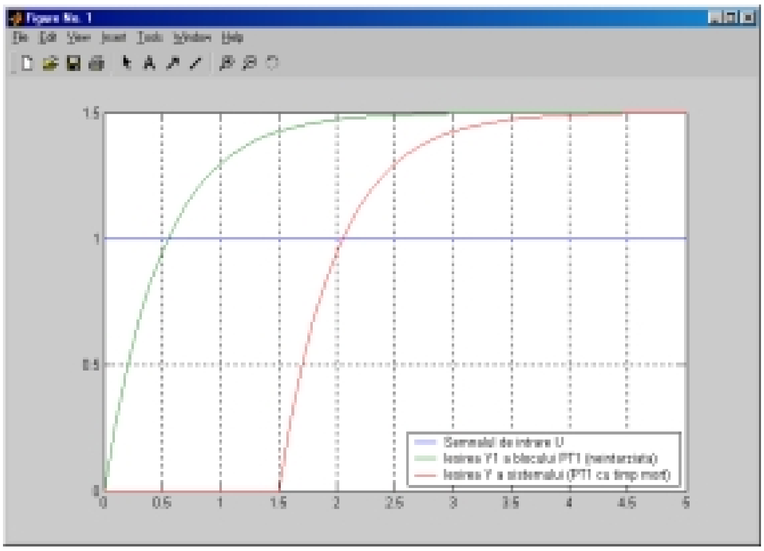


Fig. 5.

Reprezentarea grafică a variației mărimilor u, Y1 şi Y care intervin în proces este:



A doua variantă a modelului realizat în simulink este ilustrată în figura 6.

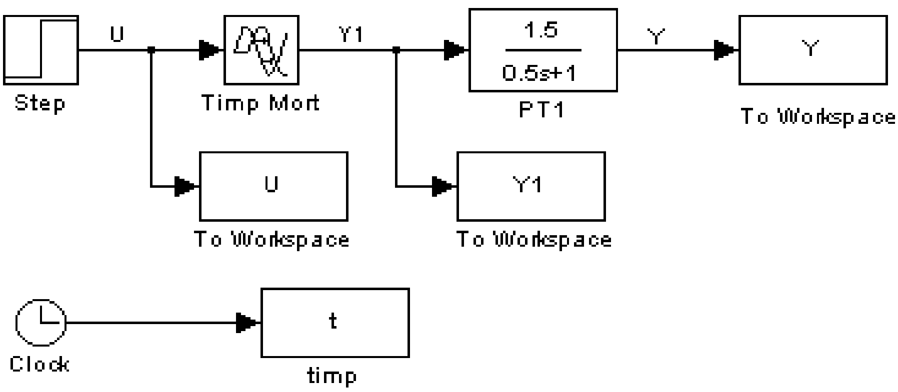
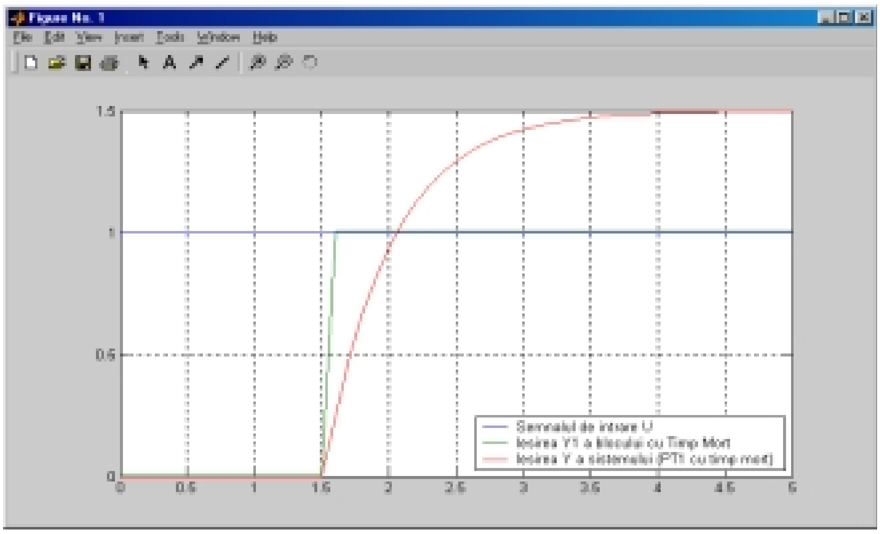


Fig. 6.

Reprezentarea grafică a variației mărimilor u, Y1 şi Y care intervin în process este:



C. Programul lucrării

1) Studiul primului exemplu de la punctual B.4. pe baza schemei din fig. 3.
2) Studiul r.i.s.t. pentru un proces cu f.d.t. $H_P(s) = \dfrac{3s}{s^2 + 4s + 3}$ în cazul h = 0.2 sec. (f.d.t. H(z), MM-ISI al r.i.s.t., studiul rezultatului folosind mediul Matlab-Simulink). Se va folosi experiența de la punctual 1) de mai sus.
3) Studiul exemplului de la pct. B.4 la care se referă fig. 5 și 6 și considerarea unui caz suplimentar corespunzător intrării $u(t) = \sigma(t\text{-}1) - \sigma(t\text{-}4)$.
4) Studiul r.i.s.t. pentru un proces cu f.d.t. $H_P(s) = \dfrac{10}{s(s+2)} e^{-1.5s}$ în cazul h=1 sec. folosind experiența de la punctele 2) și 3) de mai sus.
5) Studiul experimental folosind mediul Matlab-Simulink al influenței pasului de discretizare asupra r.i.s.t. corespunzătoare procesului $\begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & \omega_o \\ -\omega_o & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ \omega_o \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} 0 & k \end{bmatrix} x(t) \end{cases}$ cu $\omega_o = 10^2$ sec $^{-1}$ și k = 1. Pentru început se va considera h =0.01 sec.
6) Studiul celui de al doilea exemplu de la punctual B.4. pe baza schemei din fig. 4.
7) Studiul algoritmelor de reglare în timp discret obținute prin MT și prin MDR și MT din legea de reglare PID $u(t) = K[a(t) + \dfrac{1}{T_i}\int a(t)dt + T_d\,\dot{a}(t)]$ pentru valori ale parametrilor K, T_i și T_d precizate de conducătorul lucrării folosind experiența de la punctele 6) de mai sus.

D. Conținutul referatului (în succesiunea de la punctul C)

1) Consemnarea rezultatelor obținute cu comenzile
   » plot(t,y,'-',td,yb,'*'),grid
   » plot(td,ybp,'*',td,yb,'*'),grid
   şi compararea lor.
2) Precizarea f.d.t. H(z) şi MM-ISI corespunzătoare r.i.s.t. Consemnarea şi compararea rezultatelor în maniera de la punctual 1) de mai sus.
3) Prezentarea şi comentarea rezultatului din cazul suplimentar când u(t) = σ(t-1) - σ(t-4).
4) Calcului f.d.t. H(z) sau MM-ISI obținute ca r.i.s.t. pentru un proces studiat. Menționarea experimentelor efectuate.
5) Precizarea MM cu care s-a efectuat studiul experimental. Menționarea experimentelor efectuate şi comentarea rezultatelor lor.
6) Consemnarea rezultatelor experimentale şi explicarea pe baza lor a modului în care se manifestă caracterul "aproximativ".
7) Precizarea schemei Simulink implementate.
8) Indicarea unei căi mai simplă de rezolvare a problemei din exemplul 1.

E. Întrebări

1. În ce situații se apelează la discretizarea sistemelor?
2. Care este esența metodelor de discretizare studiate în lucrare?
3. Ce instrucții din Matlab tratează problema discretizării?
4. Care este diferența dintre coloana a IV-a şi a III-a din tabelele de transformare?
5. Care este semnificația simbolurilor din schemele din fig. 3,4,5 şi 6?
6. Ce modificări trebuie efectuate în schemele din fig. 3,4,5 şi 6 dacă trebuie să operăm cu MM-ISI ?
7. Ce înțelegeți prin „interpretarea" unui rezultat experimental ? Dar prin „compararea" unor rezultate experimentale ?