Laborator 9. Integrarea numerica a ecuatiilor diferentiale.

Scopul lucrării

In acest laborator vom prezenta functiile care rezolva ecuatiile diferentiale de ordin 1 si metode de rezolvare al celor de grad superior.

Consideratii Teoretice

O ecuatie diferentiala de ordinul intai are forma :

$$y' = \frac{dy}{dx} = g(x,y)$$

Metodele numerice cele mai cunoscute pentru rezolvarea ecuatiilor diferentiale sunt metoda Euler si metoda Runge-Kutta. Aceste metode aproximeaza functiile utilizand descompunerea in serii Taylor.

$$f(b) = f(a) + (b-a)f'(a) + \frac{(b-a)^2}{2!}f''(a) + \cdots + \frac{(b-a)^n}{n!}f^{(n)}(a) + \cdots$$

Aproximarea de ordinul intai foloseste functia si derivata de ordinul intai a acesteia:

$$f(b) = f(a) + (b-a)f'(a)$$

Aproximarea de ordinul doi foloseste functia si derivata de ordinul doi a acesteia:

$$f(b) = f(a) + (b-a)f'(a) + \frac{(b-a)^2}{2!}f''(a)$$

Prin cresterea numarului de termeni ai seriei Taylor creste precizia aproximarii.

Functiile MatLab pentru integrarea numerica a ecuatiilor diferentiale de ordin intai sau de un ordin mai mare se realizeaza cu ajutorul functiilor **ode23** si **ode45.**

Functia **ode23** rezolva ecuatiile diferentiale prin metoda Runge-Kutta de ordinul 2/3, iar functia **ode45** rezolva ecuatiile diferentiale prin metoda Runge-Kutta de ordinul 4/5.

Sintaxa acestor functii este :

[x,y] = ode45(‘fisier’,x_0,x_f,y_0);

[x,y] = ode45(‘fisier’,x_0,x_f,y_0,tol,trace)

Unde: fisier – este o variabila sir cu numele unui fisier **.m** care defineste derivata functiei

x0 – valoarea initiala a variabilei x

xf – valoarea finala a variabilei x

y0 – un vector coloana continand conditiile initiale

tol – parametru optional pentru setarea preciziei dorite trace – parametru optional care asigura tiparirea rezultatelor intermediare.

Exemplu 1:

Sa se integreze ecuatia diferentiala $y' = 3t^2$ pe intervalul [2,4], avand conditiile initiale y(2) = 0.5 si sa se afiseze solutia acesteia.

Se creaza un fisier de tip functie care rezolva ecuatia diferentiala :

```
function dy = f1(t,y)
dy=3*t^2;
```

Acest fisier se apeleaza dintr-un fisier de tip script cu urmatoarele linii de cod:

```
clear all;
close all;

[t,y]=ode23('f1',[2 4],[0.5]); % apelam functia de calcul a ecuatiei dif
plot(t,y,'-');                 % afisam rezultatul
grid on;
```

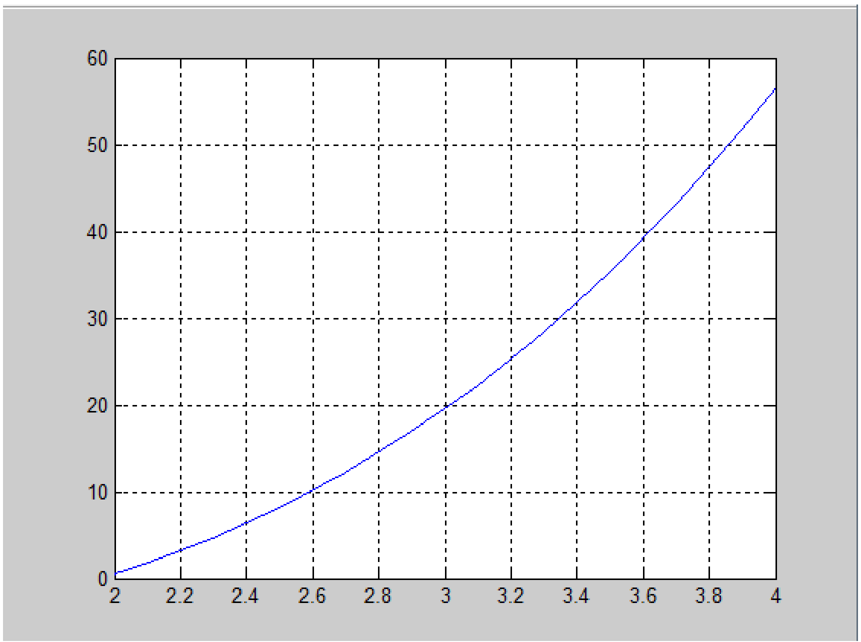

Fig. 1

Exemplu 2:

Fie ecuatia diferentiala de ordinul 2 : $y'' - (y^2 - 1)y' + y = 0$, avand conditiile initiale y(0) = 0.25; $y'(0) = 0$. Aceasta ecuatie se caracterizeaza prin sistemul:

$$\begin{cases} y' = y_1(1 - y_2^2) - y_2 \\ y_2' = y_1 \end{cases}$$

Pentru rezolvarea sistemului se creaza un fisier de tip functie in MatLab cu instructiunile:

```
function dy=ecdif(t,y)
dy=zeros(2,1);                 % definim un vector de 2 lini si o coloana
dy(1)=y(1).*(1-y(2).^2)-y(2); % calculam prima ecuatie a sistemului
```

```
dy(2)=y(1);                    % calculam a doua ecuatie a sistemului
```

Acest fisier il vom apela dintr-un fisier de tip script :

```
clear all;
close all;

t0=0;          % alegem prima valoarea a intervalului
tf=20;         % alegem valoarea finala a intervalului
ci=[0 0.25]; % punem conditiile initiale
[t,y]=ode45('ecdif',[t0 tf],[ci]);% apelam functia ecdif si calculam ec dif
plot(t,y(:,1),'--r',t,y(:,2),'-.b'); %afisam rezultatul
grid on;
```

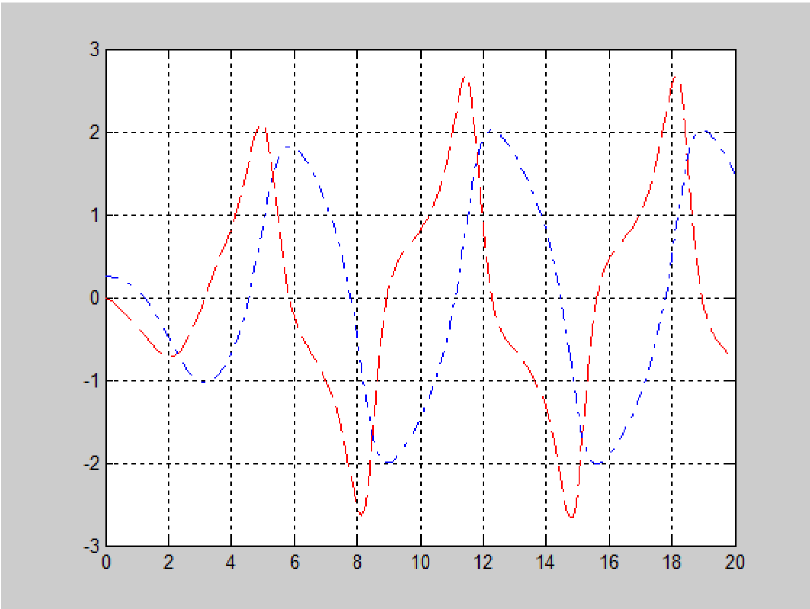

Fig. 2

Exemplu 3:

Se da circuitul RC din figura urmatoare :

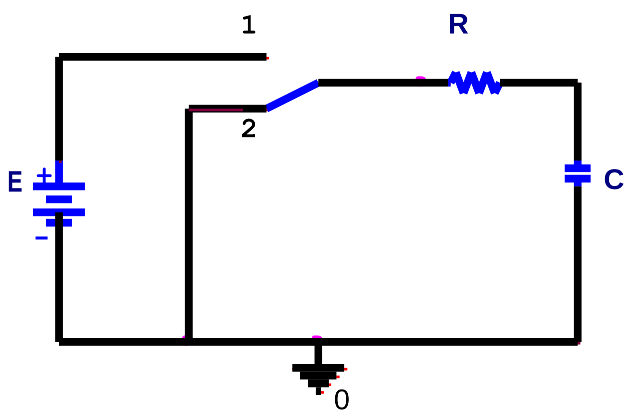

Fig. 3

Sa se reprezinte grafic tensiunea si curentul de la bornele, respectiv prin condensatorul din figura.

Inainte de rezolvarea acestei probleme trebuie sa cunoastem relatia ce face legatura dintre curentul ce trece prin condensator si tensiunea de la borne. Aceasta relatie este:

$$i_c = C\frac{du_c}{dt}$$

Din figura 3 se observa ca circuitul se poate afla in doua cazuri:

1. Comutatorul pe pozitia 1
2. Comutatorul pe pozitia 2

Daca comutatorul este pe pozitia 1 circuitul se reduce la :

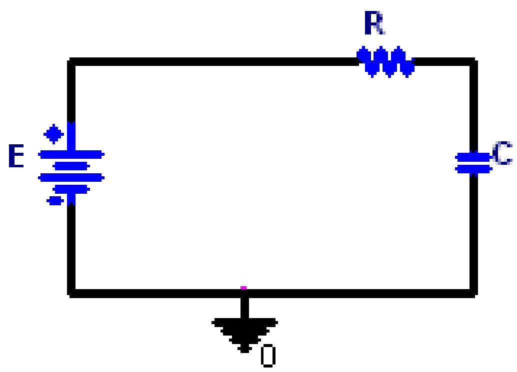

Fig. 4

Iar ecuatia ce descrie aceasta stare a circuitului este:

$$E = R * i_C + u_C$$

Inlocuind in relatia curentului vom obtine :

$$\frac{du_C}{dt} = \frac{E - u_C}{R * C}$$

Daca comutatorul este pe pozitia 2, circuitul devine :

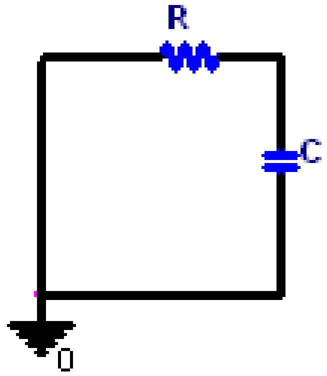

Fig. 5

Iar ecuatia ce descrie acesta stare este:

$$\frac{du_C}{dt} = \frac{-u_C}{R*C}$$

Avand cele doua ecuatii ale circuitului putem scrie un program MatLab cu ajutorul caruia sa reprezentam grafic curentul si tensiunea pe condensator.

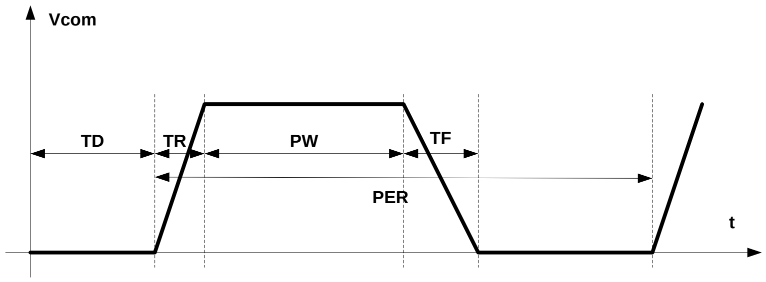

Fig. 6

In figura 6 am reprezentat un semnal de comanda al comutatorului pentru a putea defini timpul in care acesta este in pozitia 1 sau in pozitia 2. Elementul de legatura dintre timpul de conductie, in care vom pune comutatorul pe pozitia 1 si timpul de blocare, in care vom pune comutatorul pe pozitia 2, este factorul de umplere (D) care se defineste ca fiind:

$$D = \frac{PW}{PER} = \frac{T_C}{T}$$

, unde T_C reprezinta timpul de conductie si T reprezinta perioada semnalului.

Scriem un fisier script cu instructiunile:

```
clear all;
close all;

E=10;
R=3;
C=2;
T=6;
D=0.5;
N=7;

ecuatii_RC(E,R,C,T,D,N);
```

Scriem un fisier de tip functie in care calculam si afisam ecuatiile diferentiale ale circuitului:

```
function ecuatii_RC(E,R,C,T,D,N)

Fig=figure('Name','Comportarea in timp a circuitului RC ',...
           'Units','Normalized',...
           'Position',[0.17 0.17 0.7 0.7]);
Tc=D.*T;
y=0; %conditiile initiale,

for j=1:N;
```

```
    d(j)=min(y); %conditiile initiale,

    [t,y]=ode45('F1',[(j-1).*T (j-1).*T+Tc],[d(j)],[],E,R,C);
        i=(E-y)./R;
        a(j)=max(y); %conditiile initiale,

        subplot(211);
             plot(t,y,'r');hold on;grid on;
        subplot(212);
            plot(t,i,'r');hold on;grid on;

    [t,y]=ode45('F2',[(j-1).*T+Tc j.*T],[a(j)],[],E,R,C);
            i=(-y)./R;
       subplot(211);
            plot(t,y,'b');hold on;
       subplot(212);
       plot(t,i,'b');hold on;
end
subplot(211);
ylabel('uc [V]');
xlabel('timp [s]');
subplot(212);
ylabel('ic [A]');
xlabel('timp [s]');
```

Scriem cate un fisiere de tip functie pentru fiecare caz al circuitului nostru:

```
function dy=F1(t,y,flag,E,R,C)
dy=(E-y)./(R.*C);

function dy=F2(t,y,flag,E,R,C)
dy = -y./(R.*C);
```

Apeland din fisierul script vom obtine urmatorul rezultat:

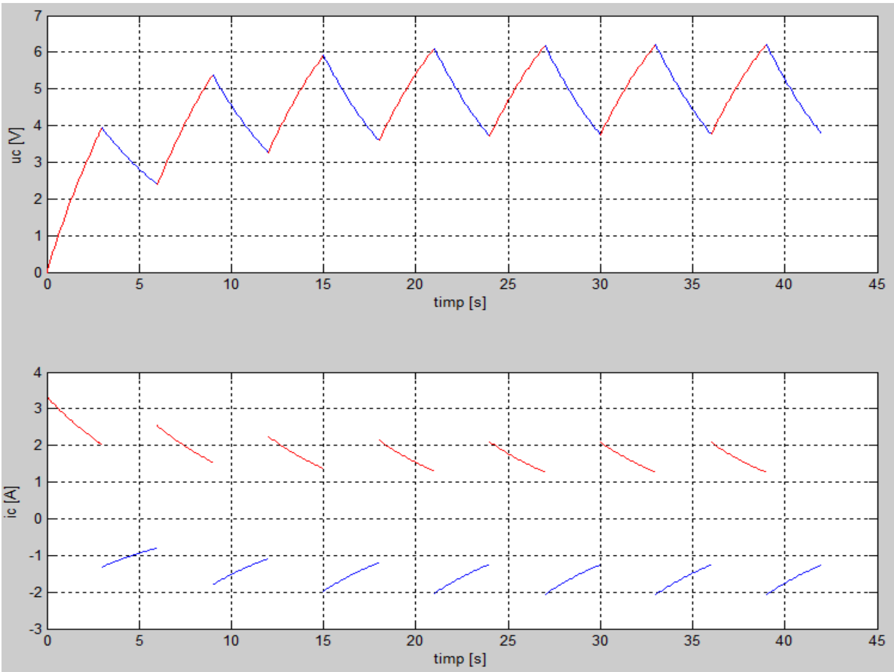

Fig. 7

Exemplul 4:

Folosind ecuatiile diferentiale sa se descrie comportarea in timp a unui circuit RL si a unui redresor cu sarcina RL.

Pentru un circuit RL serie care se cupleaza la o sursa ideala de curent continuu cu tensiunea E ecuatia diferentiala care descrie functionarea acestuia este:

$$E = R \cdot i + L \cdot \frac{di_L}{dt}$$

Programul MatLab descris :

```
function redresor_RL
close all;
R=10;       % valoarea rezistentei
L=2e-2;     % valoarea bobinei
E=10;       % valoarea tensiuni
%--------------------
f=50;        % frecventa
w=2.*pi.*f; % pulsatia
T=1./f;      % perioada semnalului
%------------------------
N=2;    % Numarul de perioade
%-------------------
% Circuitul RL
%-------------------
t=0;
y=0;
for k=1:N
    nt=length(t);
    t0=(k-1).*T;
    tf=k.*T;
    ci=y(nt);
[t,y]=ode45(@ec_redresor,[t0,tf],[ci],[],E,R,L,w);
    nt=length(t);
%--------------------------
% Tensiunea
e=E.*sin(w.*t);
%---------------------------
subplot(222);
plot(t,y,'r');hold on;
subplot(224);
uL=e-R.*y;
plot(t,uL,'r',t,e,'b');hold on;

end
%====================================================
subplot(222);
grid on;
ylabel('i   [ A ]');
title(['Circuit RL ',' R = ',num2str(R),' L = ',num2str(L)]);
subplot(224);
grid on;
ylabel('e  uL  [ V ]');
xlabel(['timp [ s ]']);

%====================================================
% Redresor RL
%========================================================
```

```
t=0;
y=0;
for k=1:N
    nt=length(t);
t0=(k-1).*T;
tf=k.*T;
ci=y(nt);
options=odeset('Events',@Trecere_zero);
[t,y,te,ye,ie]=ode45(@ec_redresor,[t0,tf],[ci],[options],E,R,L,w);
nt=length(t);
te;
%--------------------------
% Tensiunea
e=E.*sin(w.*t);
%---------------------------
subplot(221);
plot(t,y,'r',te,0,'ro');hold on;
subplot(223);
uL=e-R.*y;
plot(t,uL,'r',t,e,'b');hold on;
%-------------------------------- iL=0
t0=t(nt);
tf=k.*T;
ci=y(nt);
[t,y]=ode45(@ec_redresor,[t0,tf],[ci],[],0,R,L,w);
nt=length(t);
%--------------------------
% Tensiunea
e=E.*sin(w.*t);
%---------------------------
subplot(221);
plot(t,y,'b');hold on;
subplot(223);
uL=0.*ones(nt,1);
plot(t,uL,'r',t,e,'b');hold on;

end
%====================================================
subplot(221);
grid on;
ylabel('i  [ A ]');
title(['Redresor cu sarcina RL ',' R = ',num2str(R),'  L = ',num2str(L)]);
subplot(223);
grid on;
ylabel('e  uL  [ V ]');
xlabel(['timp [ s ]']); %,'  th = ',num2str(theta_r),' grade']);
%=============================================================
function dy=ec_redresor(t,y,E,R,L,w)
% Ecuatia diferentiala
 dy=zeros(1,1);
 e=E.*sin(w.*t);
 dy=(e-R.*y)./L;
 %====================================================
function [value,isterminal,direction] = Trecere_zero(t,y,E,R,L,w)
% Determina anularea curentului -trecerea prin zero-a curentului
%  trecerea prin zero-de la plus la minus
% opreste  integrarea
value = y;     % detecteaza iL = 0
isterminal = 1;   % % opreste  integrarea
direction = -1;   % directia de la + la -
```

Rezultatul acestui program este prezentat in figura urmatoare:

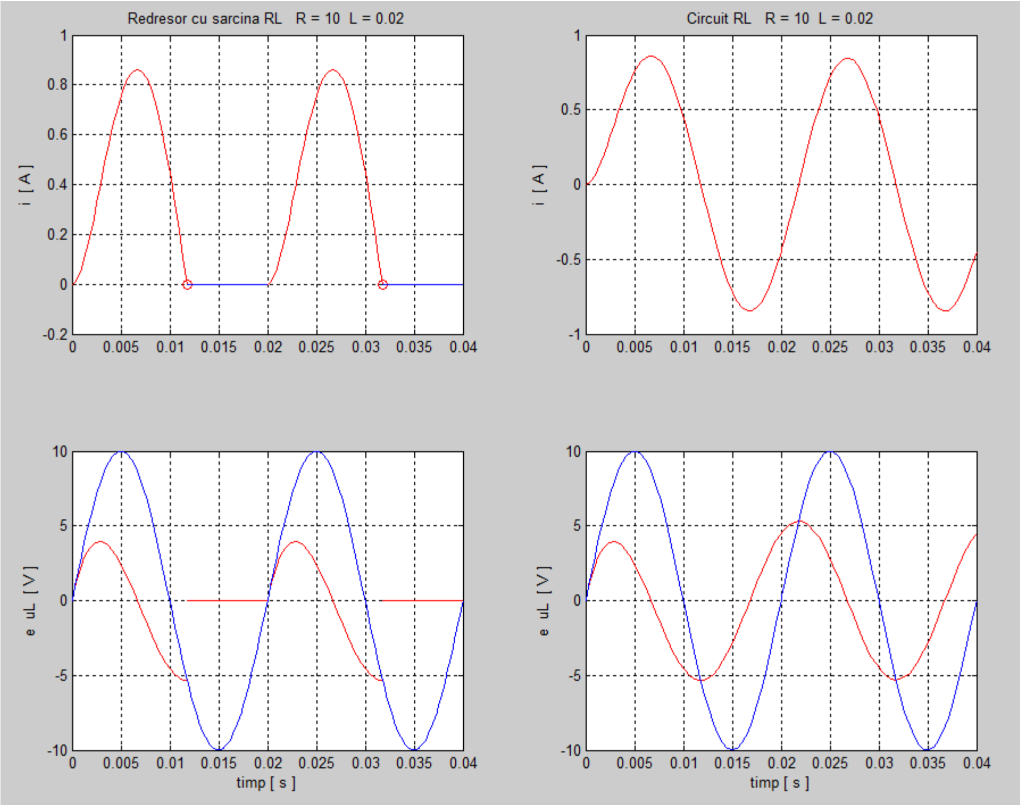

Fig. 8

Acest program a fost implementat pentru a recunoaste trecerile prin 0 ale curentului prin bobina.

Tema:

Sa se implementeze un program care sa permita modificarea parametrilor de intrare ai problemei de la exemplul 3.

Sa se implementeze un program care sa permita modificarea parametrilor de intrare ai problemei de la exemplul 4.

Sa se scrie un program care sa descrie functionarea unui redresor cu sarcina RC si care sa determine trecerile prin 0 ale tensiuni pe condensator.