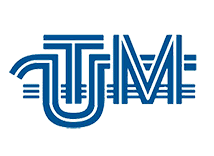
Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Ingineria Software și Automatică



**RAPORT**

Lucrare de laborator Nr.3

Disciplina: Prelucrarea semnalelor

Tema: Analiza spectrală a semnalelor

A efectuat:

st.gr.TI-201FR

Dascal Dumitru

A verificat :

Romanenko Alexandru

conf. univ., dr.

Chișinău 2023

**Scopul lucrarii:** *studiul semnalelor periodice prin dezvoltare în serie Fourier şi a semnalelor aperiodice prin aplicarea transformatei Fourier.*

**ConsideraŢii teoretice**

Pentru a înțelege cât mai bine modul în care circuitele electronice prelucreaza semnalele electrice (de tensiune sau de curent), este importantă în primul rând studierea semnalelor. Studierea semnalelor în domeniul timp, de exemplu cu ajutorul osciloscopului, poate fi completată cu o metoda complet diferită: *studiul în domeniul frecvențelor (analiza spectrală).* Studiul în frecvenţă a semnalelor și circuitelor s-a dovedit a fi o abordare extrem de productivă în electronică și automatizări. Metoda are ca punct de pornire constatarea că semnalele periodice pot fi descompuse în sume de componente armonice cu ajutorul dezvoltării în serie Fourier.

**Semnale periodice. Serii Fourier.**

Pentru a putea fi dezvoltat în serie Fourier semnalul *s(t)* trebuie să fie uniform, finit și să aibă un număr finit de discontinuități și de maxime într-o perioadă.

Semnale periodice sunt acele semnale pentru care

,

unde  o constantă numită perioadă.

Seria Fourier are următoarele forme.

***Forma trigonometrică.***

,

unde  reprezintă frecvenţa de repetiţie a semnalului periodic (frecvenţa fundamentală).

Coificienții  se calculează cu relațiile

 , .

***Forma armonică.*** Este o formă mai compactă, utilizând numai funcţii cosinusoidale.

,

; .

Deci, seria Fourier armonică dă o descompunere a semnalului periodic  într-o sumă de semnale cosinusoidale ale căror frecvenţe sunt multipli frecvenţei de repetiţie a semnalului periodic. Aceste componente se mai numesc armonici.

***Forma complexă sau exponenţială.***

,

.

Coeficienţii  sunt mărimi complexe şi deci pot fi reprezentaţi prin modul şi fază

.

Relaţiile de legătură dintre coeficienţii  şi coeficienţii seriilor trigonometrică şi armonică sunt

;

;

; ;

.

Alegerea limitelor de integrare în evaluarea coeficienţilor seriilor Fourier este arbitrară, esenţial este ca integrarea să se se facă pe durata unei perioade (de la  la  sau de la  la  etc.). Pentru semnale pare seria trigonometrică coincide cu cea armonica, deoarece . Pentru semnale impare, respectiv .

Caracterizarea în domeniul frecvenţă a semnalului periodic se face prin reprezentarea spectrelor de amplitudini şi faze. Se pot reprezenta fie diagramele spectrale asociate seriei armonice,  şi , fie diagramele spectrale asociate seriei complexe,  şi . Deoarece semnalele periodice sunt exprimate prin sume discrete de semnale elementare, rezultă că spectrele de amplitudini şi faze vor fi discrete. La spectrele asociate seriei armonice, liniile spectrale vor fi localizate la frecvenţele: etc., în timp ce la spectrele asociate seriei exponenţiale liniile spectrale vor fi localizate la: 

Teoretic spectrele semnalelor periodice se întind de la  la . Practic, spectrele sunt limitate din cauza descreşterii amplitudinilor componentelor, ceea ce permite limitarea seriei la un termen, începând cu care amplitudinea componentelor este neglijabilă. Trunchierea seriei la un anumit termen depinde de cerinţele impuse tipului de comunicaţie care utilizează semnalul respectiv. Prin urmare analiza spectrală a unui semnal ne permite să stabilim lăţimea benzii de frecvenţe efectiv ocupată de semnal.

**Semnale aperiodice. Transformata Fourier.**

Trecerea la limită,  aduce ultima relație la forma



Cunoscută ca transformata Fourier directă a semnalului s(t), de data aceasta aperiodic. Integrala este convergentă dacă semnalul satisface condițiile Dirichlet pe orice interval finit. Există și transformarea inversă, analogă seriilor Fourier pentru semnalele periodice



Care reconstituie semnalul în domeniul timp, s(t), din exprimarea lui absolute echivalentă informațional din domeniul frecvențelor , .

Atît pentru semnalele periodice cît și pentru cele aperiodice prin tratarea Fourier se obțin spectrele semnalelor, care sunt proiecții ale acestor semnale pe baze în spațiul semnalelor alcătuite din sinusoide.

**Mersul lucrării**

1. Descompunerea unui semnal periodic s(t) in serie Fourier.

Se vor analiza mai multe semnale:

* semnale armonice și combinații liniare de semnale armonice:
* semnale dreptunghiulare;
* semnale de tip dinte de ferestrău.

La execuţie, programul cere următoarele argumente: T-perioada, N-numarul de armonici pentru aproximare precum şi tipul semnalului s(t): sinusoidal, dreptunghiular sau dinte de ferestrău. Timpul este modelat printr-un vector de dimensiunea 1024.

Rezultatele se obțin prin intermediul a patru diagrame:

* semnalul s(t);
* descompunerea în armonici;
* fazele fiecărei armonici;
* recompunerea semnalului s(t) prin însumarea armonicilor calculate. Această diagramă permite compararea semnalului recompus cu cel inițial și validarea rezultatelor obținute.

%Descompunerea unui semnal periodic s(t) in serie Fourier:

%T=perioada [sec], N=nr. de armonici

T = input('Setati perioada T [sec]: ');

N = input('Setati nr. de armonici: ');

tip = input('Alegeti tipul semnalului (sin[s], dreptunghiular[d], sau ferestrau[f]): ', 's');

W=2\*pi/T; %pulsatia fundamentala

t=0:T/1022:T+T/1022;

if strcmp(tip,'s')

s=sin(W\*t); % semnal s(t) sinusoidal)

else

for j=1:1024

if strcmp(tip,'d')

if j<512 %semnal dreptunghiular

s(j)=1;

else

s(j)=-1;

end

elseif strcmp(tip,'f')

s(j)=j/500-1; %semnal s(t) dinte de ferestrau

end

end

end

val\_medie=trapz(t,s)/T; %valoarea medie

val\_efectiva=sqrt(trapz(t,s.^2)/T); %valoarea efectiva

timp=t-T/2;

for i=1:N

a(i)=2\*trapz(t,s.\*cos(i\*W\*t))/T; %primii N coef. trigonometrici

b(i)=2\*trapz(t,s.\*sin(i\*W\*t))/T;

A(i)=sqrt(a(i)^2+b(i)^2); %coeficientii formei armonice

F(i)=atan2(b(i),a(i)); %defazajele formei armonice

f(i)=i/T;

end

r=val\_medie;

for j=1:N

r=r+A(j)\*cos(j\*W\*t-F(j));

end

subplot(223); plot(t,r);

title('semnalul reconstruit (verificare)');

xlabel('t [sec]');

axis([min(t) max(t) (min(r)-0.02\*(max(r)-min(r))) (max(r)+0.02\*(max(r)-min(r)))]);

grid;

subplot(221); plot(t,s);

title('semnalul s(t)'); xlabel('t [sec]'); grid;

axis([min(t) max(t) (min(r)-0.02\*(max(r)-min(r))) (max(r)+0.02\*(max(r)-min(r)))]);

subplot(222); stem(f,A);

title('Armonicile A(n)\*cos[n\*2\*pi\*f\*t-Fi(n)]');

xlabel('f [Hz]'); grid;

subplot(224); stem(f,F/(pi));

title('defazajele Fi(f)'); xlabel('f [Hz]'); ylabel('x pi [rad]');

grid;

Pentru exemplificarea execuției funcției au fost construite reprezentările grafice pentru semnalul reprezentat printr-o succesiune de impulsuri dreptunghiulare de perioadă T=7s, aproximat prin 20 și prin 60 armonici.

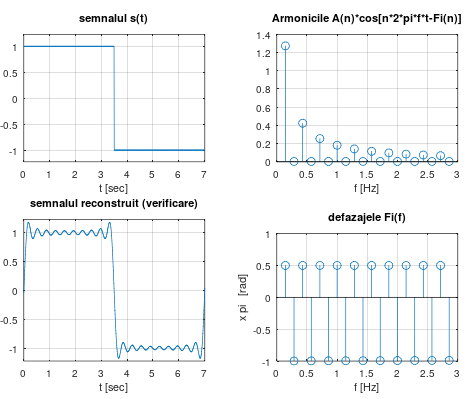


Figura 1.1 Descompunerea în serie Fourier a unui semnal dreptunghiular   
aproximat prin 20 de armonici.

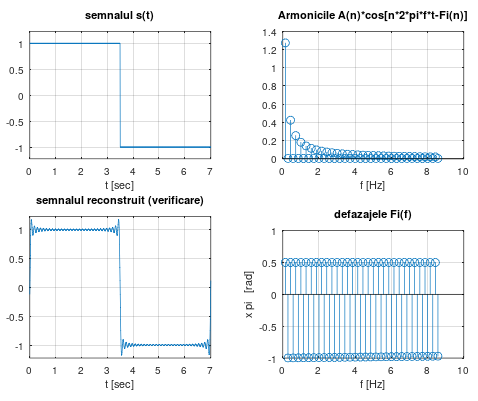


Figura 1.2 Descompunerea în serie Fourier a unui semnal dreptunghiular   
aproximat prin 60 de armonici.

2. Studiaţi spectrul unui tren de impulsuri pentru diferite valori ale parametrilor:

clear all; clg;

%parametrii trenului de impulsuri

T=3;tau=0.5;Amplit=5;

% Numarul de armonici pentru aproximarea initiala

Ni=8;

% Pasul de selectare a numarului de armonici

n=Ni;

% numarul de armonici pentru aproximarea finala

Nf=3\*n;

w0=2\*pi/T;

f0=1/T;

B=Nf+1;

% calculul parametrilor modelului spectral

A=zeros(1,B);phi=zeros(1,B);

for i=1:B,

alf=(i-1)\*w0\*tau/2;

alf=alf/pi;

A(1,i)=abs(Amplit\*tau\*sinc(alf)/T);

phi(1,i)=-angle(sinc(alf));

end;

%se calculeaza vectorul ind, necesar în reprezentarea grafica a spectrului

for i=1:B,

ind(i)=(i-1)\*f0;

end;

%reprezentarea spectrului SFC (numai pentru frecvenţe pozitive)

subplot(221);

stem(ind,A(1,:));

title('spectrul SFC al trenului de impulsuri');

xlabel('f [Hz]');

grid;

subplot(222);

stem(ind,phi(1,:));

title('defazajele Fi(f)');

xlabel('f [Hz]'); ylabel('x pi [rad]');

grid;

%generarea trenului de impulsuri si reprezentarea lui grafica

x1=zeros(1,((T\*1000/2)-(tau\*1000/2)));

x2=Amplit\*ones(1,(tau\*1000));

x3=zeros(1,((T\*1000/2)-(tau\*1000/2)));

x=[x1 x2 x3];

dt=0.001;t=[-T/2+dt:dt:T/2];

subplot(223);

h=plot(t,x); %set(h,'LineWidth',T);

axis([-T/2 T/2 -1.5 1.2\*Amplit]);grid;hold on;

%calculul semnalelor deduse pe baza spectrului determinat

%se utilizeaza Ni, 2\*Ni si 3\*Ni armonici in spectru;

%aceste semnale se reprezinta pe un grafic comun

%cu cel al trenului de impulsuri

for j=Ni:n:Nf,

xy=A(1)\*ones(1,(T\*1000));

for i=1:j,

xy=xy+2\*A(1,i+1)\*cos(i\*w0\*t+phi(1,i+1));

end;

plot(t,xy,'k');grid;

title('semnalul initial si reconstruit');

xlabel('t [sec]');

axis([-T/2 T/2 -1.5 1.2\*Amplit]);

end;

grid;

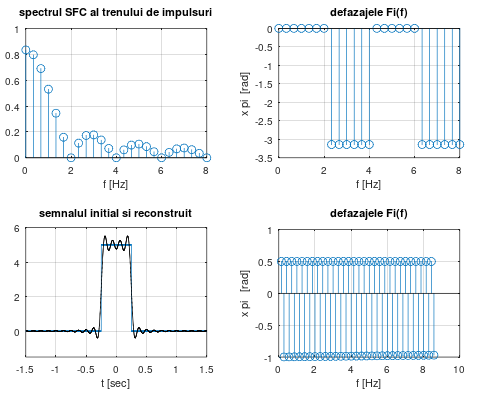


Figura 1.3 Spectrul unui tren de impulsuri

3. Pentru câteva semnale aperiodice au fost pregătite câteva *script-uri* **Matlab: FRECT, FTRIUNGHI, FSINUS, FCOSINUS, FSINUSFI, FTRENSIN.**

**FRECT**

% Scriptul FRECT calculeaza transformata Fourier a

% semnalului rectangular, simetric fata

% de origine, de durata a si de arie unitara

%parametrii impulsului rectangular

tau=1;Amplit=1/tau;

%generarea impulsului rectangular

a=tau;

tm=6;

x1=zeros(1,((tm\*1000/2)-(tau\*1000/2)));

x2=Amplit\*ones(1,(tau\*1000));

x3=zeros(1,((tm\*1000/2)-(tau\*1000/2)));

x=[x1 x2 x3];

dt=0.001;t=[-tm/2+dt:dt:tm/2];

subplot(211);

h=plot(t,x); %set(h,'LineWidth',T);

title('impuls rectangular s(t)');

xlabel('t [sec]');

axis([-tm/2 tm/2 -0.1 1.2\*Amplit]);

grid;hold on;

syms x w % sunt declarate variabilele simbolice

% se calculeaza integrala Fourier

wmax=30;

int(Amplit\*exp(-j\*w\*x),-a/2,a/2);

subplot(212);

ezplot(ans,[-wmax wmax]) % se reprezinta grafic

title('transformata Fourier a impulsului rectangular s(w)');

xlabel('w');

axis([-wmax wmax -0.5 1]);

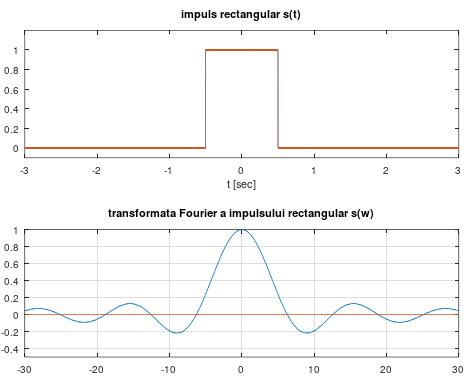
grid;hold on

u=-wmax:wmax:wmax;

y=0.0\*u;

plot(u,y) % se traseaza orizontala y=0

hold off;



**FTRIUNGHI**

% Scriptul FTRIUNGHI calculeaza transformata Fourier a

% semnalului triunghiular, simetric fata

% de origine, de durata a si de pante b si -b

a=0.5;

b=2;

syms x w f f1 f2 % sunt declarate variabilele simbolice

wmax=50;

f1=int(b\*(x+a/2)\*exp(-j\*w\*x),-a/2,0); % se calculeaza

%integrala Fourier pentru jumatatea stanga

f2=int(b\*(-x+a/2)\*exp(-j\*w\*x),0,a/2); % se calculeaza

%integrala Fourier pentru jumatatea dreapta

z=strcat(char(f1),char(f2)); % se concateneaza cele doua

%expresii f1 si f2

if strncmp(char(f2),'-',1)

z=strcat(char(f1),char(f2));

else

z=strcat(char(f1),'+',char(f2));

end

f=sym(z); % se revine la simbolic f='z'

ezplot(f,[-wmax wmax]) % se reprezinta grafic

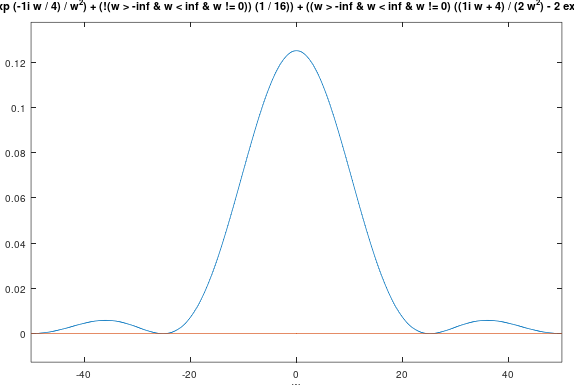
hold on

u=-wmax:wmax:wmax;

y=0.0\*u;

plot(u,y) % se traseaza orizontala y=0

hold off



**FSINUS**

% Scriptul FSINUS calculeaza transformata Fourier a unui

%puls sinusoidal de durata unei

% perioade, centrat pe origine, de durata a si de arie

%totala unitara

a=6;

w0=2\*pi

syms x w % sunt declarate variabilele simbolice

int(pi/(4\*a)\*sin(w0\*x/a)\*exp(-j\*w\*x),-a/2,a/2); % se

%calculeaza integrala Fourier

ezplot(real(ans),[-5 5]) % se reprezinta grafic partea

%reala a transformatei Fourier

hold on

ezplot(imag(ans),[-5 5]) % se reprezinta grafic partea

%imaginara a transformatei Fourier

hold on

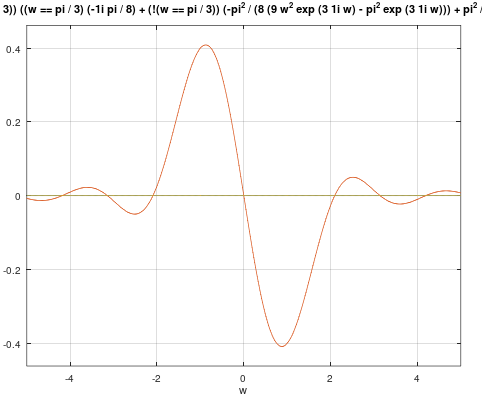
u=-5:1:5;

y=0.0\*u;

plot(u,y) % se traseaza orizontala y=0

grid;

hold off



FCOSINUS

% Scriptul FCOSINUS calculeaza transformata Fourier a unui

%puls sinusoidal, simetric fata

% de origine, de durata a si de arie unitara

a=7;

syms x w % sunt declarate variabilele simbolice

int(pi/(2\*a)\*cos(pi\*x/a)\*exp(-j\*w\*x),-a/2,a/2); % se

%calculeaza integrala Fourier

ezplot(ans,[-5 5]) % se reprezinta grafic

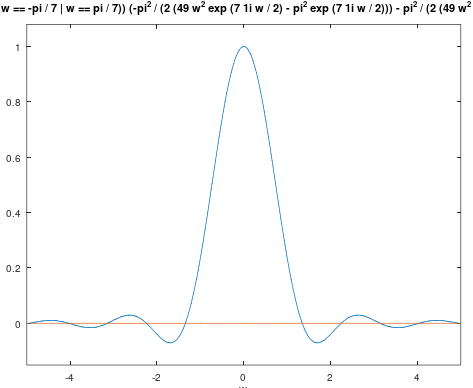
hold on

u=-5:1:5;

y=0.0\*u;

plot(u,y) % se traseaza orizontala y=0

hold off



FSINUSFI

% Scriptul FSINUSFI calculeaza transformata Fourier a unui

%puls sinusoidal de durata unei

% perioade, deplasat fata de origine cu b, de durata a si

%de arie totala unitara

a=3;

b=2;

syms x w % sunt declarate variabilele simbolice

int(pi/(4\*a)\*sin(2\*pi\*(x-b)/a)\*exp(-j\*w\*x),-a/2-b,a/2-b);

% se calculeaza integrala Fourier

ezplot(real(ans),[-5 5]) % se reprezinta grafic partea

%reala a transformatei Fourier

hold on

ezplot(imag(ans),[-5 5]) % se reprezinta grafic partea

%imaginara a transformatei Fourier

hold on

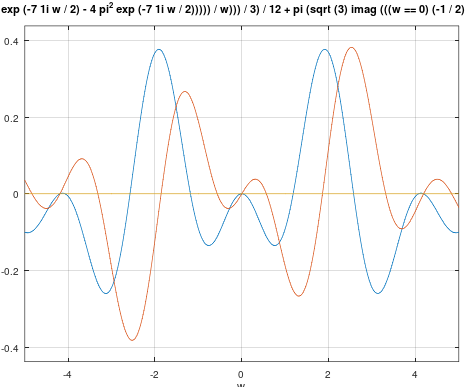
u=-5:1:5;

y=0.0\*u;

plot(u,y); % se traseaza orizontala y=0

grid;

hold off



**FMODTIMP**

Programul dat serveşte pentru identificarea proprietăţi de modificare în timp a transformatei DTFT.

% Program P3\_2

% Proprietatea DTFT de modificare în timp

clf;

w=-pi:2\*pi/255:pi;

wo=0.4\*pi;

D=10;

num=[1 2 3 4 5 6 7 8 9];

h1=freqz(num,1,w);

h2=freqz([zeros(1,D) num],1,w);

subplot(2,2,1);

plot(w/pi,abs(h1));

grid

title('Spectrul valorilor a secvenţei iniţiale');

subplot(2,2,2);

plot(w/pi,abs(h2));

grid

title('Spectrul valorilor secvenţei modificate în timp');

subplot(2,2,3);

plot(w/pi,angle(h1));

grid

title('Spectrul de faze a secvenţei iniţiale');

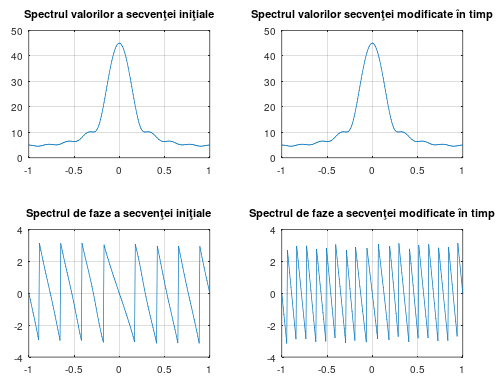
subplot(2,2,4);

plot(w/pi,angle(h2));

grid

title('Spectrul de faze a secvenţei modificate în timp');

Rezultatul programului 3\_2:



**FMODFRECV**

Programul dat serveşte pentru identifcarea proprietăţii de modificare a frecvenţei a DTFT.

% Program P3\_3

% Proprietatea de modificare a frecvenţei a DTFT

clf;

w=-pi:2\*pi/255:pi; wo=0.4\*pi;

num1=[1 3 5 7 9 11 13 15 17];

L=length(num1);

h1=freqz(num1,1,w);

n=0:L-1;

num2=exp(wo\*i\*n).\*num1;

h2=freqz(num2,1,w);

subplot(2,2,1);

plot(w/pi,abs(h1)); grid

title('Spectrul valorilor secvenţei iniţiale');

subplot(2,2,2);

plot(w/pi,abs(h2)); grid

title('Spectrul valorilor secveţei cu frecvenţa modificată');

subplot(2,2,3);

plot(w/pi,angle(h1)); grid

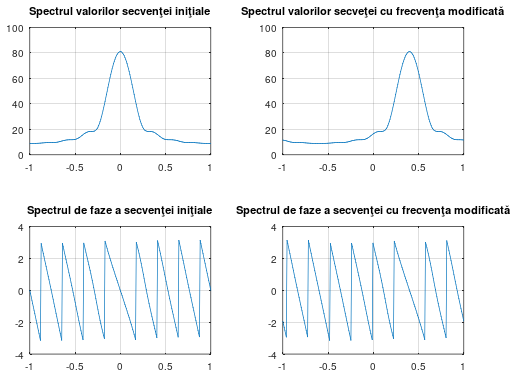
title('Spectrul de faze a secvenţei iniţiale');

subplot(2,2,4);

plot(w/pi,angle(h2)); grid

title('Spectrul de faze a secvenţei cu frecvenţa modificată');

Rezultatul programului 3\_3:



**Concluzie:**

În prelucrarea semnalelor digitale, este adesea necesar să separăm un semnal de zgomot sau să îmbunătățim calitatea semnalului în prezența zgomotului, ceea ce reprezintă o provocare în domenii precum telecomunicațiile și prelucrarea semnalelor audio. Pentru a atinge acest obiectiv, se folosesc diverse tehnici și algoritmi.Un exemplu specific de prelucrare a semnalelor implică un semnal distorsionat de zgomot aleatoriu, iar obiectivul este să obținem un semnal cât mai aproape de original eliminând zgomotul. Pentru aceasta, se utilizează un algoritm simplu de filtrare, în care fiecare punct al semnalului rezultat este calculat ca medie a sa și a două puncte vecine.Semnalele sinusoidale sunt esențiale în prelucrarea semnalelor și pot fi generate în MATLAB folosind funcțiile sin și cos. Aceste semnale sunt utile în telecomunicații, audio și alte domenii pentru analiză, sinteză și manipulare. De asemenea, secvențele exponențiale sunt semnale digitale importante, generate în MATLAB cu operatorii .^ și exp. Acestea joacă un rol semnificativ în analiza semnalelor, în special în transformatele Fourier și în domeniul filtrelor.