METODE NUMERICE DE APROXIMARE A SOLUȚIILOR SISTEMELOR DE ECUAȚII LINIARE

13.1 Metoda Jacobi

Un sistem de ecuații liniare este caracterizat prin aceea că toate necunoscutele ecuațiilor sistemului sunt la puterea întâi. Rezolvarea unui astfel de sistem, se poate realiza folosind funcția spaecifică MATLAB, cu parametri de ieșire: ***[x_1, x_2, …, x_n]=solve(ecuația_1, ecuația_2, …, ecuația_n)***. De exemplu, sistemul $\begin{cases} 4\cdot x_1 - x_2 + x_3 = 7 \\ 4\cdot x_1 - 8\cdot x_2 + x_3 = -21 \\ -2\cdot x_1 + x_2 + 5\cdot x_3 = 15 \end{cases}$, admite soluțiile:

```
» syms x_1 x_2 x_3
»[x_1,x_2,x_3]=solve(4*x_1-x_2+x_3-
7,4*x_1-8*x_2+x_3+21,-2*x_1+x_2+
                        5*x_3-15)
x_1 =
  2
x_2 =
  4
x_3 =
  3
```

Dacă acest sistem se scrie, sub forma $\begin{cases} x_1 = \dfrac{7 + x_2 - x_3}{4} \\ x_2 = \dfrac{21 + 4\cdot x_1 + x_3}{8} \\ x_3 = \dfrac{15 + 2\cdot x_1 - x_2}{5} \end{cases}$, această formulă devine formula de iterare Jacobi, scrisă sub o formă mai sugestivă: $\begin{cases} x_{1,k+1} = \dfrac{7 + x_{2,k} - x_{3,k}}{4} \\ x_{2,k+1} = \dfrac{21 + 4\cdot x_{1,k} + x_{3,k}}{8} \\ x_{3,k+1} = \dfrac{15 + 2\cdot x_{1,k} - x_{2,k}}{5} \end{cases}$. Folosind această formulare a iterării se poate admite ca util un fișier-funcție, **met_jacobi_sist.m** care să realizeze iterarea numerică pentru calculul cu aproximație a soluției unui sistem de ecuații liniare.

```
function met_jacobi_sist
% Argumente de intrare
        %-A=matricea patratica a coeficientilor: size(A)=n
        %-B=matricea coloana a termenilor liberi: size(B)=(n,1)
        %-X0=matricea initiala
```

```
        %-toler=abaterea radacinilor
        %-iter=numarul de iteratii
% Argumente de iesire
        %-X=matricea coloana a solutiilor sistemului AX=B
        %-eps=eroarea de calcul implicita: eps=2.2204e-016
A=input('Introduceti matricea patratica a coeficientilor:');
B=input('Introduceti matricea coloana a termenilor liberi:');
X0=input('Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:');
toler=input('Introduceti abaterea fata de valoarea corecta:');
iter=input('Dati numarul maxim de iteratii:');
N=length(B);
it_car=num2str(iter);
num_de_car=length(it_car);
for k=1:iter
    for j=1:N
        X(j)=(B(j)-A(j,[1:j-1,j+1:N])*X0([1:j-1,j+1:N]))/A(j,j);
    end
    eroarea=abs(norm(X'-X0));
    eroarea_relativa=eroarea/(norm(X)+eps);
    X0=X';
    if(eroarea<toler)|(eroarea_relativa<toler)
        break
    end
end
disp('|........................................................|')
fprintf(1,'S O L U T I A    DUPA    ~~%g~~  ITE R A T I I  E S T E:\n',iter)
disp('|........................................................|')
X=X'
```

Rezolvarea sistemului $\begin{cases} 4\cdot x_1 - x_2 + x_3 = 7 \\ 4\cdot x_1 - 8\cdot x_2 + x_3 = -21 \\ -2\cdot x_1 + x_2 + 5\cdot x_3 = 15 \end{cases}$, folosind metoda iterării a lui Jacobi, oferă următoarele rezultate, obținute aplicând fișierul funcție **met_jacobi_sist.m**:

a)-După 9 iterații:

```
»met_jacobi_sist
Introduceti matricea patratica a coeficientilor:[4,-1,1;4,-
8,1;-2,1,5]
Introduceti matricea coloana a termenilor liberi:[7;-21;15]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;2;2]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:9
|........................................................|
 S O L U T I A    D U P A    ~~9~~  I T E R A T I I  E S T E:
|........................................................|
X =
    1.9999
    3.9999
    2.9999
```

b)-După 10 iterații:

```
» met_jacobi_sist
Introduceti  matricea  patratica  a  coeficientilor:[4,-1,1;4,-
8,1;-2,1,5]
Introduceti matricea coloana a termenilor liberi:[7;-21;15]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;2;2]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:10
|.........................................................|
 S O L U T I A   D U P A   ~~10~~  I T E R A T I I  E S T E:
|.........................................................|
X =
    2.0000
    4.0000
    3.0000
```

13.2 Metoda Gauss- Seidel

Metoda Gauss-Seidel pentru rezolvarea numerică prin iterare a sistemelor de ecuații liniare, implică anumite considerente de procedură mai eficiente, în ceea ce privește obținerea soluției celei mai convenabile pentru un sistem. Astfel, dacă valoarea $\mathbf{x}_{1,k+1}$, de exemplu, poate fi considerată o aproximare mai bună pentru soluția $\mathbf{x}_{1,k}$, atunci, este indicat ca în calculul soluției $\mathbf{x}_{2,k+1}$ să se folosească valoarea $\mathbf{x}_{1,k+1}$. În acest sens, se poate scrie noua exprimare analitică a metodei, prin extrapolarea la celelalte necunoscute:

$$\begin{cases} \mathbf{x}_{1,k+1} = \dfrac{7 + \mathbf{x}_{2,k} - \mathbf{x}_{3,k}}{4} \\ \mathbf{x}_{2,k+1} = \dfrac{21 + 4 \cdot \mathbf{x}_{1,k+1} + \mathbf{x}_{3,k}}{8} \\ \mathbf{x}_{3,k+1} = \dfrac{15 + 2 \cdot \mathbf{x}_{1,k+1} - \mathbf{x}_{2,k+1}}{5} \end{cases}.$$

Pentru aplicarea acestei formule de calcul iterativ un fișier funcție ar putea avea următoarea construcție:

```
function sist_gauss_seidel
% Argumente de intrare
        %-A=matricea patratica a coeficientilor: size(A)=n
        %-B=matricea coloana a termenilor liberi: size(B)=(n,1)
        %-X0=matricea initiala
        %-toler=abaterea radacinilor
        %-iter=numarul de iteratii
A=input('Introduceti matricea patratica a coeficientilor:');
B=input('Introduceti matricea coloana a termenilor liberi:');
X0=input('Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:');
toler=input('Introduceti abaterea fata de valoarea corecta:');
```

```
iter=input('Dati numarul maxim de iteratii:');
% Argumente de iesire
        %-X=matricea coloana a solutiilor sistemului AX=B
        %-eps=eroarea de calcul implicita: eps=2.2204e-016
N=length(B);
it_car=num2str(iter);
num_de_car=length(it_car);
for k=1:iter
    for j=1:N
        if j==1
            X(1)=(B(1)-A(1,2:N)*X0(2:N))/A(1,1);
        elseif j==N
            X(N)=(B(N)-A(N,1:N-1)*(X(1:N-1))')/A(N,N);
        else
            X(j)=(B(j)-A(j,1:j-1)*X(1:j-1)'-A(j,j+1:N)*X0(j+1:N))/A(j,j);
        end
    end
    eroarea=abs(norm(X'-X0));
    eroarea_relativa=eroarea/(norm(X)+eps);
    X0=X';
    if(eroarea<toler)|(eroarea_relativa<toler)
        break
    end
end
X=X';
disp('|..........................................................|')
fprintf(1,'S O L U T I A   D U P A   ~~%g~~  I T E R A T I I   E S T
E:\n',iter)
disp('|..........................................................|')
X
```

Rezolvarea sistemului $\begin{cases} 4\cdot x_1 - x_2 + x_3 = 7 \\ 4\cdot x_1 - 8\cdot x_2 + x_3 = -21 \\ -2\cdot x_1 + x_2 + 5\cdot x_3 = 15 \end{cases}$, folosind fișierul funcție **met_gauss_seidel_sist.m**, oferă următoarele rezultate:

a)-după 5 iterații

```
» met_gauss_seidel_sist
Introduceti matricea patratica a coeficientilor:[4,-1,1;4,-
8,1;-2,1,5]
Introduceti matricea coloana a termenilor liberi:[7;-21;15]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;2;2]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:5
|..........................................................|
S O L U T I A   D U P A   ~~5~~  I T E R A T I I   E S T E:
|..........................................................|
X =
    1.9999
    3.9999
    3.0000
```

b)-după 6 iterații

```
» met_gauss_seidel_sist
Introduceti  matricea  patratica  a  coeficientilor:[4,-1,1;4,-
8,1;-2,1,5]
Introduceti matricea coloana a termenilor liberi:[7;-21;15]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;2;2]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:6
|.............................................................|
S O L U T I A   D U P A   ~~6~~  I T E R A T I I   E S T E:
|.............................................................|
X =
    2.0000
    4.0000
    3.0000
```

13.3 Aplicații

[1] Să se aproximeze soluțiile sistemului $\begin{cases} x_1 + x_2 = 5 \\ 2 \cdot x_1 - x_2 + 5 \cdot x_3 = -9 \\ 3 \cdot x_2 - 4 \cdot x_3 + 2 \cdot x_4 = 19 \\ 2 \cdot x_3 + 6 \cdot x_4 = 2 \end{cases}$, folosind cele două fișiere funcție `met_jacobi_sist.m` respectiv `met_gauss_seidel_sist`:

```
» met_jacobi_sist
Introduceti  matricea  patratica  a  coeficientilor:[1,1,0,0;2,-
1,5,0;0,3,-4,2;0,0,2,6]
Introduceti matricea coloana a termenilor liberi:[5;-9;19;2]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:100
|..............................................................
S O L U T I A   D U P A   ~~100~~  I T E R A T I I   E S T E:
|..............................................................
X =
  1.0e+007 *
   -2.0782
   -0.4890
    1.3855
    0.0815
```

```
» met_gauss_seidel_sist
Introduceti  matricea  patratica  a  coeficientilor:[1,1,0,0;2,-1,5,0;0,3,-4,2;0,0,2,6]
Introduceti matricea coloana a termenilor liberi:[5;-9;19;2]
Introduceti  solutia  initiala  (matrice  coloana)  pentru  startul iterarii:[1;1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:100
|...........................................................
S O L U T I A    D U P A    ~~100~~  I T E R A T I I  E S T E:
|...........................................................
X =
  1.0e+013 *
   -4.6140
    6.1520
    4.1013
   -1.3671
```

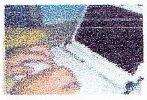
[2] Să se aproximeze soluțiile sistemului de ecuații liniare,

$$\begin{cases} \sqrt[6]{e^{\lg 3}}\cdot x_1 + \log_{\frac{6}{5}} e^{\begin{vmatrix}1 & 2 & 3\\ 0 & 3 & \pi\\ 4 & \ln 5 & 1\end{vmatrix}}\cdot x_2 - e^2\cdot x_3 = 1 \\ x_1 - e^{2+\lg 8}\cdot x_2 + \dfrac{\sqrt{1+\lg 15}}{2\pi}\cdot x_3 = \pi^3 \\ \dfrac{1}{2}\cdot x_1 - (2+\pi)\cdot x_2 + (1-\ln 8)\cdot x_3 = 0 \end{cases}$$

, folosind cele două fișiere funcție **met_jacobi_sist.m** respectiv **met_gauss_seidel_sist**:

```
» met_jacobi_sist
Introduceti matricea patratica a coeficientilor:
[exp(log10(3)/6),(log(exp(det([1,2,3;0,3,pi;4,log(5),1])))/log(5))^6,-exp(2);1,-exp(2+log10(8)),sqrt(1+log10(15));1/2,-2-pi,1-log(8)]
Introduceti matricea coloana a termenilor liberi:[1;pi^3;0]
Introduceti  solutia  initiala  (matrice  coloana)  pentru  startul iterarii:[1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:100
|...........................................................
S O L U T I A    D U P A    ~~100~~  I T E R A T I I  E S T E:
|...........................................................
X =
  1.0e+209 *
```

```
   2.5162
   0.0042
   0.0357
```



```
» met_gauss_seidel_sist
Introduceti matricea patratica a coeficientilor:[exp(log10(3)/6),(log(exp(det([1,2,3;0,3,pi;4,log(5),1])))/log(5))^6,-exp(2);1,-exp(2+log10(8)),sqrt(1+log10(15));1/2,-2-pi,1-log(8)]
Introduceti matricea coloana a termenilor liberi:[1;pi^3;0]
Introduceti solutia initiala (matrice coloana) pentru startul iterarii:[1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:100
|..............................................................|
S O L U T I A   D U P A   ~~100~~  I T E R A T I I   E S T E:
|..............................................................|
X =
   NaN
   NaN
   NaN
```

```
» met_gauss_seidel_sist
Introduceti matricea patratica a coeficientilor:[exp(log10(3)/6),(log(exp(det([1,2,3;0,3,pi;4,log(5),1])))/log(5))^6,-exp(2);1,-exp(2+log10(8)),sqrt(1+log10(15));1/2,-2-pi,1-log(8)]
Introduceti matricea coloana a termenilor liberi:[1;pi^3;0]
Introduceti solutia initiala (matrice coloana) pentru startul iterarii:[1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:10
|..............................................................|
S O L U T I A   D U P A   ~~10~~  I T E R A T I I   E S T E:
|..............................................................|
X =
  1.0e+042 *
    3.8844
    0.2131
    0.7844
```

[3] Să se aproximeze soluţiile sistemului de ecuaţii lineare,

$$\begin{cases} \sqrt[3]{e}\cdot\log_5^8 e^{\operatorname{rang}\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}} x_1 + (1-\lg 2)\cdot x_2 - e^{2\cdot+\lg 5}\cdot x_3 = 1 \\ (1-\lg 2\cdot)\cdot x_1 - \pi^{\ln 8}\cdot x_2 + \dfrac{\sqrt{1+\ln 15}}{2+\pi}\cdot x_3 = 2\pi \\ e^{\log_8\begin{vmatrix}1 & 2e^{\ln 6}\\ \sqrt[5]{3} & \ln 4\end{vmatrix}}\cdot x_1 - \left(\dfrac{1}{5e}+\lg 3\right)\cdot x_2 + \begin{vmatrix}\ln 3&2&3\\0&\pi&\lg 2\\4&5&1\end{vmatrix}\cdot x_3 = 10 \end{cases}$$

folosind cele două fişiere funcţie `met_jacobi_sist.m`:

```
met_jacobi_sist
Introduceti matricea patratica a coeficientilor:
[exp(1/3)*((log(exp(rank([1,2,3;4,5,6])))/log(5))^8),1-
log10(2),-exp(2+log10(5));1-log10(2),-pi^log(8),sqrt(1+
log(15))/(2+pi);exp(log(det([1,2*exp(log(6));3^(1/5),log(4)]))
),-1/(5*exp(1))-log10(3),det([log(3),2,3;0,pi,log10(2);
4,5,1])]
Introduceti matricea coloana a termenilor liberi:[1;2*pi;10]
Introduceti solutia initiala (matrice coloana) pentru startul
iterarii:[1;1;1]
Introduceti abaterea fata de valoarea corecta:eps
Dati numarul maxim de iteratii:100
|..........................................................
S O L U T I A   D U P A   ~~100~~  I T E R A T I I  E S T E:
|..........................................................
X =
  -0.2057 - 0.0000i
  -0.6017 + 0.0000i
  -0.2054 + 0.0000i
```