# 4. REZOLVAREA SISTEMELOR DE ECUAŢII LINIARE

**Obiectivele lucrării:**

- recapitularea unor elemente legate de rezolvarea sistemelor de ecuații liniare,
- fixarea de cunoştințe privitoare la rezolvarea sistemelor de ecuații liniare folosind mediul de programare Matlab,
- familiarizarea cu calculul simbolic în Matlab şi cu rezolvarea sistemelor de ecuații liniare folosind calea simbolică,

prin studierea unor exemple şi prin rezolvarea unor probleme.

Se recomandă parcurgerea anexei M4 înaintea studierii paragrafelor 4.1 şi 4.2.

## 4.1. Elemente despre rezolvarea sistemelor de ecuații liniare în Matlab

### *Rezolvarea sistemelor de ecuații liniare pe cale numerică*

Pe cale numerică se pot rezolva în Matlab sisteme compatibile determinate şi se pot găsi maxim două soluții particulare pentru sisteme compatibile nedeterminate.

În rezolvarea sistemelor liniare de `p` ecuații şi `n` necunoscute folosind mediul Matlab, se presupune că acestea au forma matriceală:

`A·X = b`

cu `A` matricea coeficienților, având `p` linii şi `n` coloane, `b` vectorul coloană al termenilor liberi şi `X` vectorul coloană al necunoscutelor.

Alternativ, se poate opera şi cu forma matriceală:

`X1·A1 = b1`

unde $A1=A^T$, $b1=b^T$ şi $X1=X^T$, formă care foloseşte vectorii termenilor liberi, respectiv al necunoscutelor, sub formă de vectori-linie.

Un sistem de ecuații liniare este definit în Matlab prin definirea matricei `A` şi a vectorului `b` (sau a matricei `A1` şi a vectorului `b1`).

#### a. Rezolvarea sistemelor compatibile determinate

Sistemele compatibile determinate pot fi rezolvate prin două metode:

1. ***metoda inversării matriceale***:

În cazul unui sistem pătratic, soluția sistemului este obținută prin inversarea matricei coeficienților şi înmulțirea ei cu vectorul termenilor liberi:

```
X = inv(A) * b
```

respectiv,

```
X1 = b1 * inv(A1)
```

2. ***metoda împărțirii la stânga / dreapta***:

Această metodă utilizează unul din operatorii de împărțire la stânga sau la dreapta, în funcție de forma matriceală utilizată:

```
X = A \ b
```

respectiv,

```
X1 = b1 / A1
```

Cele două metode de rezolvare se bazează pe metode numerice diferite. Metoda împărțirii la stânga / dreapta foloseşte pentru rezolvarea sistemelor de ecuații liniare metoda eliminării în versiunea Gauss. Această metodă nu necesită determinarea inversei matricei coeficienților.

**b. Determinarea unei soluţii particulare pentru sistemele compatibile nedeterminate**

În cazul sistemelor compatibile nedeterminate nu se pot determina pe cale numerică toate soluțiile, acestea fiind în număr infinit. Se poate determina o soluție particulară folosind una din următoarele două metode:

1. ***metoda împărțirii la stânga / dreapta***:

```
X = A \ b
```

respectiv,

```
X1 = b1 / A1
```

În acest caz, această metodă realizează căutarea acelei soluții a sistemului care minimizează în sensul celor mai mici pătrate norma euclidiană a vectorului `A·X-b` şi care are cel mult rang `A` componente nenule.

2. ***metoda pseudo-inversării matriceale***:

Soluția sistemului este obținută prin înmulțirea pseudo-inversei Moore-Penrose a matricei coeficienților cu vectorul termenilor liberi. Pseudo-inversa Moore-Penrose se obține prin apelul funcției Matlab *`pinv`*:

```
X = pinv(A) * b
```

respectiv,

```
X1 = b1 * pinv(A1)
```

## *Rezolvarea sistemelor de ecuații liniare pe cale simbolică*

Pe cale simbolică se pot rezolva în Matlab sisteme pătratice compatibile nedeterminate, sisteme subdeterminate şi sisteme cu parametri.

### a. Calcul simbolic în Matlab

Pentru calcule simbolice, Matlab pune la dispoziția utilizatorului toolbox-ul ***Symbolic Math***. Tabelul 4.1. prezintă câteva din funcțiile utilizate în calculul simbolic din mediul Matlab:

***Tabelul 4.1.*** *Funcții Matlab pentru calcul simbolic.*

| **Funcţia** | **Utilizare** |
|---|---|
| `det` | calculează determinantul unei matrice simbolice |
| `factor` | descompune în factori o expresie simbolică |
| `inv` | calculează inversa unei matrice simbolice |
| `rank` | calculează rangul maxim al unei matrice simbolice |
| `simplify`, `simple` | simplifică expresii simbolice |
| `solve` | rezolvă ecuații şi sisteme de ecuații |
| `subs` | substituie un simbol cu un alt simbol sau cu o valoare numerică |
| `sym` | crează un obiect simbolic |
| `syms` | crează mai multe obiecte simbolice |

Lista completă a funcțiilor Matlab pentru calcul simbolic se poate afişa prin apelul help-lui asociat directorului Matlab `symbolic`.

Rezolvarea unei probleme pe cale simbolică trebuie să înceapă cu definirea obiectelor simbolice (simbolurilor). Apoi se trece la implementarea propriu-zisă a soluționării problemei.

**b. Rezolvarea sistemelor de ecuaţii liniare folosind Symbolic Math Toolbox**

Metodele de rezolvare precizate la începutul lucrării sunt valabile şi pentru rezolvarea pe cale simbolică a sistemelor de ecuații liniare:

1. metoda inversării / metoda pseudo-inversării;
2. metoda împărțirii la stânga / dreapta.

Rezolvarea pe cale simbolică necesită de obicei studiul sistemului şi efectuarea soluționării pe cazuri.

## 4.2. Exemple de studiat

Toate exemplele corespund formei canonice $A \cdot X = b$ (a se vedea paragraful 4.1).

Înainte de a rezolva un sistem de ecuații liniare trebuie verificat dacă acest sistem este compatibil (a se vedea anexa M4).

**Exemplul 4.1:** Să se rezolve sistemul de mai jos utilizând metoda inversării matriceale:

$$\begin{cases} 5x_1 + 4x_2 + x_3 = 0 \\ 6x_1 + 3x_2 + 2x_3 = 5 \\ x_1 + x_2 + x_3 = -7 \end{cases}$$

*Soluție:* Se execută următoarea secvență de program Matlab (de exemplu, fişier-M):

```
A=[5 4 1; 6 3 2; 1 1 1]; % matricea coeficientilor
% rezolvarea sistemului
if det(A)~=0 % daca sistemul este compatibil determinat
    b=[0; 5; -7]; % vectorul-coloana al termenilor liberi
    X=inv(A)*b
else
    disp('Sistemul nu este compatibil determinat.')
end
```

În urma execuției secvenței de mai sus se obține:

```
X =
    6.2500
   -6.0000
   -7.2500
```

Adică: $x_1 = 6.25$ , $x_2 = -6$ , $x_3 = -7.25$.

**Exemplul 4.2:** Să se rezolve următorul sistem folosind metoda împărțirii la stânga:

$$\begin{cases} 2x_1 - 3x_2 = 7 \\ -6x_1 + 8x_2 - x_3 = -5 \\ 3x_2 + 4x_3 = 1 \end{cases}$$

*Soluție.*: Se execută următoarea secvență de program Matlab (de exemplu, fişier-M):

```
A=[2 -3 0; -6 8 -1; 0 3 4]; % matricea coeficientilor
% rezolvarea sistemului
if det(A)~=0
    b=[7 -5 1]'; % vectorul-coloana al termenilor liberi
    X=A\b
else
    disp('Sistemul nu este compatibil determinat.')
end
```

În urma execuției secvenței de mai sus se obține:

```
X =  -94.0000
     -65.0000
      49.0000
```

Adică: $x_1 = -94$ , $x_2 = -65$ , $x_3 = 49$.

**Exemplul 4.3:** Să se determine pentru sistemele de mai jos una-două soluții particulare, folosind metodele pseudo-inversării şi împărțirii la stânga.

a) $$\begin{cases} 2x_1 - 3x_2 = 7 \\ -6x_1 + 8x_2 - x_3 = -5 \\ 3x_2 + 4x_3 = 1 \\ -4x_1 + 8x_2 + 3x_3 = 3 \end{cases}$$

b) $$\begin{cases} 3x - y + z - 2t = 6 \\ -4x + 4y + 2z + t = 0 \end{cases}$$

*Soluţie.*:

a) Se execută următoarea secvenţă de program Matlab (de exemplu, fişier-M):

```
A=[2 -3 0; -6 8 -1; 0 3 4; -4 8 3]; % matricea coeficientilor
b=[7; -5; 1; 3]; % vectorul-coloana al termenilor liberi
% rezolvarea sistemului
if rank(A)==rank([A b]) % daca sistemul este compatibil
    disp('metoda pseudo-inversarii')
    X=pinv(A)*b
    disp('metoda impartirii la stanga')
    X=A\b
else
    disp('Sistemul nu este compatibil.')
end
```

În urma execuţiei secvenţei de mai sus se obţine:

```
metoda pseudo-inversarii
X =  -94.0000
     -65.0000
      49.0000
metoda impartirii la stanga
X =  -94.0000
     -65.0000
      49.0000
```

b) Se execută următoarea secvenţă de program Matlab (de exemplu, fişier-M):

```
A=[3 -1 1 -2; -4 4 2 1]; % matricea coeficientilor
b=[6; 0]; % vectorul-coloana al termenilor liberi
% rezolvarea sistemului
if rank(A)==rank([A b])
    disp('metoda pseudo-inversarii')
    X=pinv(A)*b
    disp('metoda impartirii la stanga')
    X=A\b
else
    disp('Sistemul nu este compatibil.')
end
```

Executând secvenţa de mai sus, se obţine:

```
metoda pseudo-inversarii
X =    0.9431
       0.5418
       1.3846
      -1.1639
metoda impartirii la stanga
X =    1.2000
       0
       2.4000
       0
```

*Comentarii:* 1. Pentru testarea compatibilităţii s-a folosit teorema lui Kronecker-Capelli. Matricea extinsă s-a obţinut în Matlab prin concatenarea matricei *A* cu vectorul-coloană *b*, `[A b]`.

2. În cazul primului sistem, care este compatibil determinat s-au obținut, evident, soluții identice prin utilizarea celor două metode. În cazul celui de-al doilea sistem, care este compatibil nedeterminat, prin utilizarea celor două metode s-au obținut două soluții particulare distincte. Rezolvarea completă a acestui sistem apare în exemplul 4.4.

**Exemplul 4.4:** Să se rezolve sistemul de ecuații liniare:

$$\begin{cases} 3x - y + z - 2t = 6 \\ -4x + 4y + 2z + t = 0 \end{cases}$$

*Soluție:* Sistemul de ecuații liniare considerat este un sistem subdeterminat. El poate fi rezolvat doar pe cale simbolică. Se execută următoarea secvență de program Matlab (de exemplu, fişier-M):

```
clc
% pas 1: se determina rangul matricei sistemului
A=[3 -1 1 -2;-4 4 2 1];
r=rank(A) % se obtine 2
% prin urmare, 2 variabile sunt independente, si 2
% variabile sunt dependente de primele

% pas 2: se cauta un minor de ordinul 2 nenul, pentru a
% stabili variabilele dependente; de exemplu:
rminor=rank(A(:,[1 2])) % se obtine 2
% x,y devin variabilele dependente, in raport cu care se
% rezolva sistemul;
% acesta se rescrie sub forma:
% 3x-y=6-z+2t;  -4x+4y=-2z-t;

disp('sistemul este compatibil nedeterminat')
disp(blanks(1)')

% pas 3: se rezolva sistemul rescris in noua forma:
% se creeaza obiectele simbolice
syms z t;
% matricea sistemului
Aredus=A(:,[1 2]);
% vectorul termenilor liberi
bredus=[6-z+2*t; -2*z-t];
% rezolvarea sistemului cu metoda inversarii
s=inv(Aredus)*bredus
% rezolvarea sistemului cu operatorul de impartire la
% stanga
ss=Aredus\bredus
disp(blanks(1)')
pause
disp('Solutia sistemului dat este:')
x=s(1)
y=s(2)
disp('z,t numere reale oarecare')
```

Se obțin rezultatele:

```
r =      2
```

```
rminor =     2
sistemul este compatibil nedeterminat

s =
 3-3/4*z+7/8*t
 3-5/4*z+5/8*t
ss =
 3-3/4*z+7/8*t
 3-5/4*z+5/8*t

Solutia sistemului dat este:
x =
3-3/4*z+7/8*t
y =
3-5/4*z+5/8*t
z,t numere reale oarecare
```

Prin urmare, a fost obținută soluția:

$$x=3-\frac{3}{4}\cdot z+\frac{7}{8}\cdot t,\ y=3-\frac{5}{4}\cdot z+\frac{5}{8}\cdot t,\ z\in\mathbf{R},\ t\in\mathbf{R}.$$

*Comentarii:* 1. Funcția Matlab `blanks` creează spații între şirurile de caractere.

2. Comanda Matlab `pause` are ca efect suspendarea momentană a execuției programului. Execuția se continuă numai după apăsarea unei taste.

**Exemplul 4.5:** Să se rezolve sistemul de ecuații liniare:

$$\begin{cases} mx-nz=q \\ nx-my=0 \\ my+mz-2q=0 \end{cases}$$ , în necunoscutele $x$, $y$, $z$,

cu parametri $m$, $n$ şi $q$, dintre primii doi parametri cel puțin unul fiind nenul.

*Soluție:* Sistemul dat fiind un sistem cu parametri, el se poate rezolva doar cu ajutorul toolbox-ului de calcul simbolic. Este necesar un studiu de compatibilitate al sistemului, în funcție de diverse valori ale parametrilor, rezolvarea efectuându-se pe cazuri, mediul Matlab fiind utilizat doar pentru efectuarea de calcule simbolice sau numerice şi de substituții ale parametrilor cu valori particulare.

Se execută următoarea secvență de program Matlab (de exemplu, fişier-M):

```
% se creeaza obiectele simbolice
m=sym('m'); n=sym('n'); q=sym('q');
% matricea sistemului
A=[m 0 -n;n -m 0;0 m m];
% vectorul coloana al temenilor liberi
b=[q; 0; 2*q];

% ** Discutie **
% determinantul sistemului
d=det(A)    % se obtine d = -m^3-n^2*m
factor(d)   % se obtine -m*(m^2+n^2)
% se observa ca d==0 daca si numai daca m==0
```

```
% cazul d~=0: sistem compatibil determinat
disp('Cazul: m~=0 -> sistem compatibil determinat')
% solutia calculata cu metoda inversarii matriceale
s=inv(A)*b
% solutia calculata cu operatorul de impartire la stanga
ss=A\b
disp(blanks(2)')
pause

% cazul d==0
% substituirea lui m cu valoarea 0
A=subs(A,m,0)
% se observa ca ultima linie a lui A contine doar elemente
% nule, dar, ultimul element al lui b este 2*q
% prin urmare sistemul este incompatibil daca q~=0
% si compatibil nedeterminat, daca q==0
disp('Cazul: m==0 si q~=0 -> sistem incompatibil')
disp(blanks(2)')
pause

% cazul sistem compatibil nedeterminat
disp(['Cazul: m==0,n~=0,q==0 -> sistem compatibil '...
    'nedeterminat'])
% substituirea lui q cu valoarea 0
b=subs(b,q,0);
% sistemul devine:
% -n*z==0; n*x==0;
% m fiind 0, rezulta din ipoteza ca n este nenul
disp('Solutia: (0,y,0) cu y real oarecare')
```

Se obțin rezultatele:

```
d =-m^3-n^2*m
ans =-m*(m^2+n^2)
Cazul: m~=0 -> sistem compatibil determinat
s =
     m/(m^2+n^2)*q+2*n/(m^2+n^2)*q
 n/(m^2+n^2)*q+2*n^2/m/(m^2+n^2)*q
    -n/(m^2+n^2)*q+2*m/(m^2+n^2)*q
ss =
     q*(m+2*n)/(m^2+n^2)
 q*n*(m+2*n)/m/(m^2+n^2)
    -q*(n-2*m)/(m^2+n^2)

A =
[  0,   0, -n]
[  n,   0,  0]
[  0,   0,  0]
Cazul: m==0 si q~=0 -> sistem incompatibil

Cazul: m==0,n~=0,q==0 -> sistem compatibil nedeterminat
Solutia: (0,y,0) cu y real oarecare
```

**Exemplul 4.6:** Să se compare metodele studiate de rezolvare a sistemelor de ecuații liniare din punct de vedere al timpului de execuție şi al preciziei soluției pentru cazul unui sistem pătratic compatibil determinat de ecuații liniare de dimensiuni mari.

*Soluție:* Pentru generarea unei matrice de dimensiuni mari s-a optat pentru folosirea funcției Matlab *rand*, care generează o matrice de numere aleatoare uniform distribuite.

Se execută următoarea secvență de program Matlab (de exemplu, fişier script):

```
% matricea sistemului, solutia exacta, vectorul termenilor
% liberi:
A=rand(700); x=rand(700,1); b=A*x;
%  timpii de executie:
tic; y=inv(A)*b; timp1=toc
tic; z=A\b; timp2=toc
%  precizia solutiilor calculate:
n1=norm(A*y-b)
n2=norm(A*z-b)
n1/n2
```

Executând secvența de mai sus, se obține:

```
timp1 =
    0.1671
timp2 =
    0.0685
n1 =
  3.3005e-010
n2 =
  5.9342e-012
ans =
    55.6182
```

Întrucât în primul caz – utilizarea metodei inversării – timpul de execuție este 0.1671 secunde şi eroarea $3.3005 \cdot 10^{-10}$, iar în al doilea caz – utilizarea metodei împărțirii la stânga – timpul de execuție este 0.0685 secunde şi eroarea $5.9342 \cdot 10^{-12}$, se observă că metoda prin împărțirea la stînga a matricelor este mai performantă decât metoda inversării matriceale, atât din punct de vedere al timpului de execuție, cât şi al preciziei soluției obținute.

## 4.3. Probleme de rezolvat

**P4.1.** Folosind mediul Matlab, să se analizeze dacă următorul sistem de ecuații liniare este compatibil determinat şi, în caz afirmativ, să se rezolve sistemul:

$$\begin{cases} 4x+3y-z=2 \\ -x+y+z=0 \\ x+3z=-1 \end{cases}.$$

**P4.2.** Folosind mediul Matlab, să se determine rangul matricei coeficienților şi câte o soluție aproximativă pentru sistemele de ecuații liniare de mai jos:

a) $$\begin{cases} 4x+3y-z=2 \\ -x+y+z=0 \\ x+3z=-1 \\ 4x+4y+3z=1 \end{cases};$$

b) $$\begin{cases} -6x+8y-z=-5 \\ 2x-3y=7 \end{cases}.$$

**P4.3.** Să se scrie un program care primeşte ca parametri matricea coeficienților unui sistem oarecare de ecuații liniare şi vectorul termenilor liberi şi returnează soluția sistemului, în cazul în care este compatibil determinat, sau un mesaj corespunzător, în cazul în care este compatibil nedeterminat sau incompatibil.

**P4.4.** Folosind mediul Matlab, să se rezolve sistemul de ecuații liniare:

$$\begin{cases} -6x+8y-z=-5 \\ 2x-3y=7 \end{cases}.$$

**P4.5.** Folosind mediul Matlab, să se rezolve următoarele sisteme de ecuații liniare cu parametri şi cu necunoscutele *x*,*y*,*z*:

a) $$\begin{cases} \alpha x-\beta y+z=\gamma \\ -\gamma x+y+\beta z=\alpha \\ x+\gamma y-\alpha z=\beta \end{cases};$$

b) $$\begin{cases} ax-by=p \\ -bx+by-cz=-2q \\ cy+az=p+q \end{cases}.$$

**P4.6.** Să se scrie un program care primeşte ca argumente matricea coeficienților şi vectorul termenilor liberi ai unui sistem oarecare de ecuații liniare şi care clasifică sistemul într-una din categoriile: i) compatibil determinat, ii) compatibil nedeterminat sau iii) incompatibil, şi afişează un mesaj corespunzător.

**P4.7.** Să se realizeze un studiu de caz asemănător exemplului 4.6. pentru cazul unui sistem compatibil nedeterminat pentru care se doreşte determinarea unei soluții particulare.

## 4.4. Întrebări recapitulative

Î4.1. Câte metode de rezolvare a sistemelor de ecuații liniare în Matlab cunoaşteți şi care sunt acestea?

Î4.2. Care este metoda numerică pe care se bazează metoda de împărțire la stânga / dreapta de rezolvare a sistemelor compatibile determinate de ecuații liniare?

Î4.3. Cum se pot rezolva în Matlab sistemele compatibile nedeterminate de ecuații liniare?

Î4.4. Ce returnează funcția Matlab *pinv*?

Î4.5. Ce este un sistem de ecuații liniare supradeterminat?

Î4.6. Ce este un sistem de ecuații liniare subdeterminat?

Î4.7. Care este condiția ca un sistem oarecare de ecuații liniare să fie compatibil?

Î4.8. Care este condiția ca un sistem compatibil (oarecare) de ecuații liniare să fie determinat?

Î4.9. Ce puteți spune legat de sisteme de ecuații liniare omogene?

Î4.10. Definiți noțiunea de "pseudo-inversă Moore-Penrose a unei matrice".

Î4.11. Care sunt proprietățile pseudo-inversei Moore-Penrose a unei matrice?

# ANEXA M4. ELEMENTE DESPRE REZOLVAREA SISTEMELOR LINIARE

## M4.1. Noţiuni generale

Fie sistemul de $p$ ecuații liniare cu $n$ necunoscute:

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,n}x_n = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,n}x_n = b_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{p,1}x_1 + a_{p,2}x_2 + \cdots + a_{p,n}x_n = b_p \end{cases}$$

cu $a_{i,j} \in \mathbf{R}, \quad b_i \in \mathbf{R}, \quad i = \overline{1,p},\ j = \overline{1,n}$.

Introducând notațiile:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{p,1} & a_{p,2} & \cdots & a_{p,n} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{pmatrix}, \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

se obține ***forma matriceală a sistemului de ecuații liniare***:

$$A \cdot X = b$$

Matricea *A* se numeşte ***matricea coeficienților***, matricea *b* ***vectorul-coloană al termenilor liberi*** şi *X* ***vectorul-coloană al necunoscutelor sistemului de ecuații liniare***.

Matricea (*A*,*b*) cu *p* linii şi *n+1* coloane obținută prin alipirea la dreapta matricei *A* a vectorului *b*, se numeşte ***matricea extinsă a sistemului***.

În funcție de rezultatul comparației lui *p* cu *n* se disting următoarele tipuri de sisteme:

- ***sistem pătratic***, dacă $p = n$;
- ***sistem supradeterminat***, dacă $p > n$;
- ***sistem subdeterminat***, dacă $p < n$.

Un sistem de ecuații liniare care are cel puțin o soluție se numeşte ***sistem compatibil***. Un sistem compatibil care are o singură soluție se numeşte ***sistem compatibil determinat***. Un sistem compatibil care are mai multe soluții se numeşte ***sistem compatibil nedeterminat***. Un sistem care nu are nici o soluție se numeşte ***sistem incompatibil***.

Soluția unui sistem pătratic de ecuații liniare compatibil determinat, se poate obține matriceal prin relația:

$$X = A^{-1} \cdot b$$

Un sistem compatibil nedeterminat are o infinitate de soluții.

## M4.2. Criterii de compatibilitate

***Teorema lui Kronecker – Capelli***: Un sistem de ecuații liniare este compatibil dacă şi numai dacă rangul matricei sistemului este egal cu rangul matricei extinse a sistemului.

Un sistem compatibil de ecuații liniare este determinat dacă şi numai dacă rangul matricei sistemului este egal cu numărul necunoscutelor sistemului.

Un sistem pătratic de ecuații liniare este compatibil determinat dacă şi numai dacă determinantul matricei coeficienților este nenul.

Verificarea compatibilitații unui sistem de ecuații liniare se poate face în următorul mod:

a) cazul sistemelor pătratice:
   - dacă determinantul matricei coeficienților este nenul, sistemul este compatibil determinat;
   - în caz contrar:
     - dacă rangul matricei sistemului este egal cu rangul matricei extinse, atunci sistemul este compatibil nedeterminat;
     - altfel, sistemul este incompatibil.

b) cazul sistemelor oarecare:
   - dacă rangul matricei sistemului este egal cu rangul matricei extinse, atunci sistemul este compatibil; distingem două subcazuri:
     - dacă rangul matricei sistemului este egal cu numărul necunoscutelor, atunci sistemul este compatibil determinat;
     - altfel, sistemul este compatibil nedeterminat;
   - în caz contrar, sistemul este incompatibil.

## M4.3. Sisteme de ecuaţii liniare omogene

Un caz particular de sisteme de ecuații liniare îl reprezintă ***sistemele de ecuații liniare omogene***, adică sisteme pentru care $b$=0.

Un sistem omogen este întotdeauna compatibil, deoarece el admite întotdeauna ***soluția nulă***: $X = 0$.

Dacă rangul matricei sistemului omogen este egal cu numărul necunoscutelor, atunci sistemul este compatibil determinat. În caz contrar (rangul este mai mic decât numărul necunoscutelor) sistemul este compatibil nedeterminat.

## M4.4. Pseudo-inversa Moore-Penrose a unei matrice

***Pseudo-inversa Moore-Penrose a matricei A*** este o matrice $B$ de aceeaşi dimensiune ca şi $A^T$, care îndeplineşte următoarele condiţii:

(a) $A \cdot B \cdot A = A$ şi $B \cdot A \cdot B = B$,

(b) $A \cdot B$ şi $B \cdot A$ sunt matrice hermitiene.

(***O matrice hermitiană*** este o matrice pătratică cu proprietatea că ea coincide cu transpusa conjugatei sale.)

Proprietăţile pseudo-inversei Moore-Penrose sunt :

1. Orice matrice are o unică pseudo-inversă Moore-Penrose.
2. Pseudo-inversa unei matrice inversabile coincide cu inversa matricei.
3. Pseudo-inversa pseudo-inversei unei matrice este matricea însăşi.

Fie sistemul compatibil de ecuaţii liniare cu $n$ necunoscute:

$$A \cdot X = b$$

Fie $A^+$ pseudo-inversa Moore-Penrose a matricei $A$. Atunci:

$$Xa = A^+ \cdot b$$

este o soluţie aproximativă a sistemului de ecuaţii liniare, cu proprietatea că are norma euclidiană minimă dintre toate $n$-uplurile $X$ pentru care $\|A \cdot X - b\|^2$ este minimă, unde cu $\|\cdot\|$ s-a notat norma euclidiană.