CURS 2

1. Matrice, vectori şi scalari. Notaţii

MATLAB-ul este un pachet de programe care lucrează numai cu un singur tip de obiecte, matrice numerice rectangulare, cu elemente reale sau complexe. În acest sens, scalarii sunt asimilaţi matricelor cu o linie şi o coloană (1x1), iar vectorii sunt asimilaţi matricelor cu o linie (1xn) sau o coloană (nx1). Operaţiile şi comenzile în MATLAB sunt aproape naturale, în sens matriceal, asemănător modului de calcul obişnuit. Astfel entităţile:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}; \; B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}; \; C = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}; \; D = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

sunt toate matrice în accepţiunea MATLAB. A este o matrice 2x2, B este 2x3, C este 3x1, D este 3x1.

Elementele unei matrice, fie aceasta A, pot fi identificate prin una dintre notaţiile: A[i,j], A(i,j) etc. şi semnifică elementul de la intersecţia liniei i cu coloana j. Ultima notaţie, A(i,j), este cea care a fost adoptată si în MATLAB. Dimensiunea unei matrice este precizată de o pereche de numere care arată numărul de linii si coloane al matricei respective. (D este o matrice 3x1). O matrice cu o singură linie sau o singură coloană se numeşte vector linie (D) sau vector coloană (C), elementele acesteia putând fi identificate cu un singur indice. O matrice cu o singură linie si o singură coloană este un scalar.

Pentru a face referire la un element A(i,j) al, unei matrice A, sunt necesari doi indici, indicele de linie şi indicele de coloană, în această ordine. Referirea unui element al unui vector poate fi făcută numai cu un singur indice. Astfel:

A(2,1) = 3;   B(1,3) = 3,   C(2) = 2;   D(3) = 3.

2. Definirea matricelor simple

Definirea matricelor se face prin una dintre următoarele metode:

- introducerea explicită a listei de elemente;

- generarea prin instrucţiuni si funcţii;

- crearea de fişiere M;

- încărcarea din fişiere de date externe.

MATLAB-ul nu conţine instrucţiuni de dimensionare şi declaraţii de tip, iar memoria este alocată automat, până la valoarea maxim disponibilă. Cea mai simplă metodă de definire a matricelor mici constă în utilizarea unei liste explicite. La introducerea unei astfel de liste trebuie respectate următoarele reguli:

- elementele unei linii trebuie separate prin blanc-uri sau virgule;

- liniile se separă prin semnul punct-virgulă ";";

- elementele matricei sunt cuprinse între paranteze drepte „[ ]".

Astfel, matricea A din paragraful anterior poate fi introdusă cu secvenţa:

A=[1 2;3 4]

care returnează rezultatul:

$$A = \begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$$

Matricea A, astfel definită, poate fi utilizată în calcule sau poate fi salvată într-un fişier de date pentru o folosire ulterioară.

Pentru matricele mari, la care datele de intrare nu încap pe o singură linie, se poate proceda la înlocuirea semnului „;" cu „Enter", ca în exemplu următor:

A=[1 2

3 4]

Elementele matricelor

Elementele matricelor pot fi numere reale sau complexe, precum si orice expresie MATLAB. De exemplu, pentru:

x=[-1.3 sqrt(3) (l+2+3)*4/5]

rezultă:

x = [-1.3000 1.7321 4.8000]

Elementele unei matrice pot fi referite cu indici cuprinşi între paranteze rotunde „()", ca în exemplul:

a = x(2) care returnează:

a =1.7321

De remarcat că dacă se asignează o valoare unui element care ocupă o poziţie în afara dimensiunii maxime a matricei sau vectorului referit, dimensiunea acestuia este mărită automat până la valoarea indicelui noului element, iar elementele nedefinite sunt setate la valoarea zero. în acest sens, instrucţiunea:

x(5)= abs(x(l))

returnează rezultatul:

x = [-1.3000 1.7321 4.8000 0 1.3000]

iar instrucţiunea:

A(2,4)= 6

returnează rezultatul:

$$A = \begin{matrix} 1 & 2 & 0 & 0 \\ 3 & 4 & 0 & 6 \end{matrix}$$

În exemplele de mai sus s-au utilizat 2 funcţii MATLAB predefinite a căror denumire este rezervată:

abs – modulul unui număr; sqrt – rădăcină pătrată

O modalitate de a construi matrice mari constă în folosirea matricelor mici ca elemente. Spre exemplu, din două matrice 2x3 se poate construi o matrice 4x3; dimensiunile matricelor utilizate trebuie să fie astfel alese încât să realizeze tablouri rectangulare complete.

Fie A1 şi A2 cele două matrice utilizate. Cu secvenţa următoare:

Al = [l 2;3 4]

A2 = [5 6;7 8]

A = [Al; A2]

se obţine rezultatul:

$$A = \begin{matrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{matrix}$$

O matrice mai mică poate fi extrasă din matrice mai mari utilizând semnul „:" (două puncte). De exemplu:

B = A(2:3,:)

extrage liniile doi si trei şi toate coloanele din matricea curentă A, obţinându-se matricea B:

$B =$

3 4

5 6

Declaraţii şi variabile

MATLAB-ul este un limbaj de expresii. Expresiile tipărite de utilizator sunt interpretate şi evaluate. Instrucţiunile MATLAB sunt, de cele mai multe ori, de forma:

variabila = expresie

sau, mai simplu:

expresie

Expresiile sunt compuse din operatori sau alte caractere speciale, din funcţii şi nume de variabile. Evaluarea expresiei produce o matrice, care este afişată pe ecran şi atribuită unei variabile. Dacă numele variabilei şi semnul egal („variabila=”) sunt omise, MATLAB-ul crează automat o variabilă cu numele „ans”, ca în exemplul:

3/4

care returnează:

ans= 0.7500

Orice instrucţiune este în mod normal terminată cu „Enter". Dacă ultimul caracter al acesteia este punct-virgulă „;”, instrucţiunea este executată, dar tipărirea este suprimată. Utilizarea acestui caracter la sfârşitul unei instrucţiuni în fişiere-M este necesară în situaţiile în care nu se doreşte afişarea datelor intermediare. De exemplu:

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

introduce matricea A, dar nu o afişează. Tastarea numelui unei variabile urmată de „Enter" afişează valoarea acesteia.

Dacă expresia este aşa de mare încât declaraţia nu încape pe o singură linie, se utilizează semnul „...” (trei puncte), urmat de „Enter", pentru a preciza că instrucţiunea continuă pe linia următoare. Astfel, instrucţiunea:

S=1+2+3+...

4+5+6;

evaluează suma celor şase numere şi o atribuie variabilei S. Spaţiile dintre semnele „=”, „+”, „-” şi numere sunt opţionale.

Numele de variabile si funcţii au ca prim caracter o literă, urmată de litere, cifre sau caracterul „underscore” (adică „ _”). Deşi se pot folosi oricâte caractere, MATLAB-ul reţine ca nume de variabilă numai primele 19 caractere. MATLAB-ul face deosebirea între literele mari şi mici (case sensitive), astfel încât „a" si „A" sunt două variabile distincte.

1.OPERAŢII ARITMETICE

Calculele aritmetice asupra tablourilor de date în MATLAB pot fi: - operaţii după regulile calculului matriceal - operaţii cu matrice; - operaţii după regulile calculului scalar - operaţii cu tablouri.

Operatorii folosiţi în calculele aritmetice cu tablouri şi matrice sunt prezentaţi în tabelul 1:

Tabelul 1. Operatori aritmetici MATLAB

Operaţia	Scalari	Matrice	Tablouri
Adunarea	+	+	+
Scăderea	-	-	-
înmulţirea	*	*	.*
împărţirea la stânga	\	\	.\
împărţirea la dreapta	/	/	./
Ridicarea la putere	^	^	^
Transpunerea	A’	A’	A’

1.1. Operaţiile aritmetice cu scalari

Operaţiile aritmetice între doi scalari sunt prezentate în tabelul 2.

Tabelul 2. Forma MATLAB a operaţiilor cu scalari

Operaţia	Forma algebrică	Forma MATLAB
Adunare	a+b	a+b
Scădere	a-b	a-b
înmulţire	axb	a*b
împărţire la dreapta	a: b	a/b
împărţire la stânga	b: a	a\b
Ridicare la putere	a^b	a^b

Expresiile aritmetice pot fi evaluate şi rezultatul memorat în variabile specificate. Astfel, instrucţiunea:

x=a+b

atribuie variabilei x, suma dintre variabilele a si b. Instrucţiunea: **k=k+1** atribute variabilei k o nouă valoare, egală cu suma dintre vechea valoare şi constanta 1. În urma instrucţiunilor succesive: **a=1;a=2.5** în variabila a se află valoarea 2.5.

O variabilă introdusă fără nominalizare este asignată variabilei ***ans*** (answer). În variabila ***ans*** este memorată în permanenţă valoarea ultimei variabile căreia nu i s-a atribuit un nume.

1.1.1. Ordinea operaţiilor aritmetice

Ordinea operaţiilor în MATLAB, tabelul 7.3, este aceeaşi cu cea a operaţiilor aritmetice standard, cunoscută în matematica elementară.

Tabelul 3. Ordinea operaţiilor aritmetice

Ordinea	Operaţia
1	parantezele
2	ridicarea la putere
3	înmulţirea si împărţirea
4	adunarea si scăderea

1.1.2. Limitele calculelor

Deşi variabilele memorate au un interval foarte mare, cel mai adesea fiind între limitele 10^{-308} şi 10^{308} (vezi funcţiile MATLAB ***realmax*** si ***realmin***), totuşi, uneori este posibil ea rezultatul unei expresii să depăşească aceste limite, aşa cum se întâmplă în exemplul:

$x = 2.5*10^{200}$

$y = 10^{200}$

$z= x*y = 2.5*10^{400}$. Deoarece z este în afara limitelor de mai sus, valoarea calculată z nu poate fi memorată. MATLAB-ul înregistrează ∞. Se verifică acest lucru cu secvenţa:

x=2.5*10^200; y=10^200

z=x*y care retumează:

z=Inf

Rezultatul unui calcul este mai mic decât 10^{-308} calculatorul înregistrează valoarea zero.

în MATLAB rezultatul împărţirii cu zero este ∞. în acest caz se afişează mesajul de atenţionare „Warning:Divide by zero”, dar calculele continuă cu operandul ∞.

1.2. Operaţiile aritmetice cu tablouri

Operaţiile cu tablouri sunt operaţii aritmetice (înmulţire, împărţire, ridicare la putere, etc) între elementele situate în aceeaşi poziţie ale tablourilor, cunoscute sub numele de operaţii element cu element.

Pentru a preciza că înmulţirea se efectuează element cu element între componentele a două matrice de aceleaşi dimensiuni, se utilizează operatorul de înmulţire precedat de punct (.*), adică:

C=A.*B

Pentru efectuarea operaţiilor cu tablouri se folosesc aceiaşi operatori ca în operaţiile cu scalari, precedaţi de semnul punct „.”, semn ce indică efectuarea operaţiilor în ordinea element cu element. Dacă unul dintre operanzi este un scalar, acesta operează cu fiecare element al tabloului.

1.2.1. Adunarea şi scăderea

Exemplul 1. Fie: A=[2 5 6], B=[4 3 2], p=2. Să se calculeze: C=A-B, D=A-p şi E=p-A. Cu secvenţa MATLAB:

A=[2 5 6]; B=[4 3 2]; p=2; C=A-B

D=A-p

E=p-A

se obţin rezultatele:

C = [-2 2 4]

D = [0 3 4] E = [0 -3 -4]

1.2.2. înmulţirea tablourilor

Exemplul 2. Fie: A=[2 5 6], B=[4 3 2], p=2. Să se calculeze : C=A.*B, D=A.*p si E=p.*A . Cu secvenţa MATLAB:

A=[2 5 6}; B=[4 3 2];p=2; C=A.*B D=A.*p E=p.*A

se obţin rezultatele: C=[8 15 12] D=[4 10 12] E=[4 10 12]

1.2.3. Împărţirea la dreapta

Operaţia de împărţire Ia dreapta, element cu element, între două tablouri este simbolizată cu operatorul punct-slash (./). Instrucţiunea: Z=X./Y reprezintă împărţirea element cu element a tablourilor X si Y, rezultând elementele: Z(i,j) = X(i,j)/Y(i,j)

Exemplul 3. Fie: A=[2 5 6], B=[4 3 2], p=2. Să se calculeze: C=A./B, D=A./p şi E=p./A. Cu secvenţa MATLAB:

A=[2 5 6]; B=[4 3 2]; p=2; C=A./B D=A . /p E=p./A

se obţine rezultatul:

C=[0.5000 1.6667 3.0000]

D=[1.0000 2.5000 3.0000]

E=[1.0000 0.4000 0.3333]

1.2.4. Împărţirea la stânga ;

Operaţia de împărţire la stânga, element cu element, între două tablouri este simbolizată cu operatorul punct-bakslash (.\ ). Instrucţiunea: Z = X.\Y reprezintă împărţirea element cu element a tablourilor X şi Y, cu aceleaşi dimensiuni, rezultând un tablou cu elementele: Z(i,j) = Y(i,j)/X(i,j)

Prin urmare: Z = X.\Y = Y./X

Exemplul 4. Fie: A=[2 5 6] B=[4 3 2] p=2. Să se calculeze : C=A.\B , D=A.\p şi E=p.\A Cu secvenţa MATLAB:

A=[2 5 6]; B=[4 3 2]; p=2; C=A.\B D=A. \p E=p .\A se obţin rezultatele:

C=[2.0000 0.6000 0.3333] D=[1.0000 0.4000 0.3333] E=[1.0000 2.5000 3.0000]

1.2.5. Ridicarea la putere

Operaţia de ridicare la putere element cu element într-un tablou este simbolizată cu operatorul punct-^ (.^). Următoarea instrucţiune:

Z=X.^Y

reprezintă ridicarea fiecărui element din tabloul X la puterea indicată de valoarea elementului din aceeaşi poziţie a tabloului Y, adică: Z(i,j) = X(i,j)^Y(i,j)

Dacă X este un scalar, se lasă un blanc între scalar şi operatorul de ridicare la putere „ .^” pentru a nu interpreta punctul care indică operarea cu tablouri de elemente ca punctul zecimal.

Exemplul 5. Fie: A=[2 5 6], B=[4 3 2], p=2. Să se calculeze : C=A.^B, D=A.^p şi E=p.^A. Cu secvenţa MATLAB:

A=[2   5   6};   B=[4   3 2];   p=2; C=A.^B D=A.^p E=p .^A

se obţin rezultatele: C=[16 125 36] D=[4 25 36] E=[4 32 64]

1.2.6. Transpunerea tablourilor

Operaţia de transpunere a unui tablou este simbolizată de operatorul punct-apostrof. Cu instrucţiunea:

Z=Y.'

liniile tabloului Y devin coloanele tabloului transpus Z. Acest lucru face ca un tablou Y cu dimensiunea ***m x n*** să devină un tablou Z cu dimensiunea ***n x m***.

Exemplul 6. Să se determine transpusa tabloului: $Z = \begin{bmatrix} 1 & 1+i \\ i & 3-2*i \end{bmatrix}$

Cu secvenţa MATLAB:

Z=[1 1+i;i 3-2*i]; Z1=Z.'

se obţin rezumatele: Z1 =

1.0000 0 +1.0000i

1.0000 + 1.0000i 3.0000 - 2.0000i

1.3. Operaţiile aritmetice cu vectori

1.3.1. Produsul scalar

Produsul scalar a doi vectori de aceeaşi dimensiune este un scalar, egal cu suma produselor corespunzătoare aceloraşi poziţii:

$$C = \vec{A} \cdot \vec{B} = \sum_{i=1}^{N} a(i) \cdot b(i)$$

Produsul scalar a doi vectori, A şi B, se calculează cu instrucţiunea:

C=sum(A.*B)

Dacă A este de dimensiunea 1xN şi B de dimensiunea Nx1, atunci sunt posibile următoarele expresii pentru calculul produsului scalar:

E=sum(A'.*B) F=sum(A.*B')

Cosinusul unghiului vectorilor: $\vec{A} = a_x \vec{i} + a_y \vec{j} + a_z \vec{k}; \vec{B} = b_x \vec{i} + b_y \vec{j} + b_z \vec{k}$ se calculează cu relaţia:

$\cos\left(\vec{A}, \vec{B}\right) = \dfrac{\vec{A} \cdot \vec{B}}{\left|\vec{A}\right| \cdot \left|\vec{B}\right|} = \dfrac{a_x b_x + a_y b_y + a_z b_z}{\sqrt{a_x^2 + a_y^2 + a_z^2}\sqrt{b_x^2 + b_y^2 + b_z^2}}$ unde $\left|\vec{A}\right|$ şi $\left|\vec{B}\right|$ sunt lungimile, norma sau modulul.

Dacă produsul scalar este nul, cei 2 vectori sunt ortogonali.

Exemplul 7. Să se calculeze produsul scalar şi unghiul dintre vectorii: $\vec{a} = 3\vec{i} - 4\vec{j},\ \vec{b} = \vec{i} + 2\vec{j} - 2\vec{k}$

Cu secvenţa MATLAB:

a=[3 -4 0]; b=[l 2 -2] ; ab=sum(a.*b);% -produsul scalar

mod_a=norm(a);mod_b=norm(b);

alfa=acos(ab/(mod_a*mod_b))*180/pi se obţine rezultatul:

ab=-5

alfa= 109.4712 grade

1.3.2. Produsul vectorial

Produsul vectorial a doi vectori $\vec{A}$ şi $\vec{B}$:

$$\vec{C} = \vec{A} \times \vec{B} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix}$$

este un vector perpendicular pe planul celor doi vectori. Modulul produsului vectorial este:

$$\left|\vec{C}\right| = \left|\vec{A}\right|\left|\vec{B}\right| \sin\left(\vec{A}, \vec{B}\right)$$

De exemplu, dacă A si B descriu vectorii cu ajutorul coordonatelor (proiecţiile după axele Ox, Oy şi Oz), cu secvenţa MATLAB:

A=[ax ay az]; B=[bx by bz]; C= A'*B se obţine ca rezultat matricea C, cu dimensiunea 3x3.

$$\vec{C} = \begin{bmatrix} x & \vec{k} & -\vec{j} \\ -\vec{k} & x & \vec{i} \\ \vec{j} & -\vec{i} & x \end{bmatrix}$$

J

Componentele C(i,j) au următoarea semnificaţie:

- pentru $i \neq j$ reprezintă un vector orientat după versorul menţionat la pozif ia respectivă a matricei C;

- pentru $i = j$ reprezintă o componentă a produsului scalar (marcată cu x în matricea C)

Vectorul rezultant C are componentele: cx=C(2,3)-C(3,2), cy=C(3,l)-C(l,3) cz=C(l,2)-C(2,l) PV=[cx cy cz]; %Produsul vectorial

Suma elementelor diagonalei principale constituie produsul scalar al vectorilor $\vec{A}$ şi $\vec{B}$, şi se calculează cu instrucţiunea MATLAB:

PS=sum(diag (C)); %Produsul scalar. Prin urmare, produsul a doi vectori este:

- un scalar, dacă operaţia este un produs scalar;
- o matrice, dacă operaţia este un produs vectorial

Exemplul 8. Să se calculeze produsul vectorial si scalar al vectorilor: $\vec{a} = 5\vec{i} - 3\vec{j} - \vec{k}$, $\vec{b} = -\vec{i} - 2\vec{j} - 2\vec{k}$

a=[5 -3 -1]; b=[-l -l -2]; C=a'*b; cx=C(2,3)-C(3,2); cy=C(3,l)-C(l,3); cz=C(l,2)-C(2,l);

PV=[cx cy cz] %produsul vectorial

PSl=sum(a.*b) % produsul, scalar metoda 1

PS2=sum(diag(C)) %produsul scalar metoda 2

se obţin rezultatele:

PV = [5 11 -8] PS1=0 PS2=0

care reprezintă vectorul produs vectorial: $\vec{a} = 5\vec{i} + 11\vec{j} - 8\vec{k}$; produsul scalar nul evidenţiază faptul că cei doi vectori sunt perpendiculari.

1.4. Operaţiile aritmetice cu matrice

Operaţiile uzuale de algebră liniară cu matrice sunt simbolizate cu semnele grafice: * , \ , /, A , ', şi se efectuează după regulile cunoscute din calculul matriceal.

1.4.1. Adunarea şi scăderea

Operaţia de adunare a două matrice este simbolizată cu operatorul plus (+). Instrucţiunea: Z=X+Y reprezintă adunarea matricelor X şi Y, rezultând elementele: Z(i,j)=X(i,j)+Y(i,j)

Matricele X şi Y trebuie să aibă aceleaşi dimensiuni, în afara cazului când X sau Y este un scalar. Un scalar a poate fi adunat cu orice matrice, rezultând elementele: Z(i,j)=a+X(i,j) Operaţia de scădere a două matrice este simbolizată cu operatorul minus (-). Instrucţiunea: Z = X-Y reprezintă scăderea matricei Y din X, rezultând elementele: Z(i,j)=X(i,j)-Y(i,j)

Matricele X şi Y trebuie să aibă aceleaşi dimensiuni, în afara cazului când X sau Y este un scalar. Un scalar a poate opera cu orice matrice, rezultând elementele: Z(i,j)=X(i,j)-a sau Z(i,j)=a-X(i,j)

Exemplul 9. Fie: A =[1 2;3 4], B=[5 6;7 8], p=2

Să se calculeze: A+B, A-B, A+p .

7.4.2. înmulţirea matricelor

Operaţia de înmulţire este simbolizată cu operatorul (*). Următoarea instrucţiune: Z=X*Y reprezintă matricea produs a având elementele $Z(i,j)=\sum_k X(i,k)Y(k,j)$. Produsul matriceal este posibil dacă numărul coloanelor matricei X este egal cu numărul liniilor matricei Y, elementul Z(i,j) fiind suma produselor dintre elementele liniei i cu elementele corespondente din coloana j. Produsul matrice-vector este un caz special al cazului genera! al produsului matrice-matrice. De asemenea, un scalar poate poate fi înmulţit cu orice matrice, realizându-se înmulţirea cu fiecare element al matricei; Z(i,j) = a*X(i,j).

Exemplul 10. Fie matricele A, B şi scalarul p din exemplul anterior. Să se calculeze A*B, A*p.

1.4.3. Împărţirea la dreapta

Operaţia de împărţire la dreapta a două matrice este simbolizată cu operatorul slash (/). Următoarea instrucţiune: Z = X/Y

reprezintă împărţirea la dreapta a matricelor X şi Y, şi este identică cu $X*Y^{-1}$ (Y^{-1} este inversa matricei Y – în MATLAB funcţia de calcul a inversei unei matrice este ***inv***).

Exemplul 11. Fie: A =[1 2;3 4], B=[5 6;7 8]. Să se calculeze A/B în cele 2 moduri posibile

1.4.4. împărţirea la stânga

Operaţia de împărţire la stânga a două matrice este simbolizată cu operatorul backslash (\). Următoarea instrucţiune: Z=X\Y reprezintă împărţirea la stânga a matricelor X şi Y şi este identică cu $X^{-1}*Y$ (X^{-1} este inversa matricei X).

Dacă A este o matrice n x n, iar B este un vector coloană cu n componente, atunci X=A\B este soluţia sistemului de ecuaţii AX=B, obţinută prin eliminarea Gauss. Dacă unul dintre operanzi este scalar, operaţia nu este posibilă.

Exemplul 12. Fie sistemul $\begin{cases} 2x+5y=-3 \\ x-y=2 \end{cases}$ Să se rezolve prin 2 metode: cu ***solve*** şi cu împărţirea la stânga (X=[x ;y] ;şi identificaţi matricile A şi B

1.4.5. Ridicarea la putere

Operaţia de ridicare la putere a unei matrice este simbolizată cu operatorul (^). Următoarea instrucţiune: Z=X^p reprezintă ridicarea la puterea p a matricei X. Expresia X^p are sens doar pentru X matrice pătrată şi p scalar.

Dacă p este un întreg pozitiv, ridicarea la putere este obţinută prin înmulţiri repetate:

Z = X^p= X * X*...* X ,Vp>0 .

iar dacă p este un întreg negativ, X este mai întâi inversată şi apoi se înmulţesc inversele de p ori:

Z = X^p = inv(X) * inv(X)*...*inv(X) ,Vp<0

Exemplul 13. Fie: Z =[l 2;3 4] şi p=-2 . Să se calculeze Z^p.

1.4.6. Transpunerea matricelor

Operaţia de transpunere a unei matrice este simbolizată cu operatorul apostrof. Cu instrucţiunea: Z=Y'

liniile matricei Y devin coloanele matricei transpuse Z. Acest lucru face ca pentru o matrice Y cu dimensiunea m x n să se obţină o matrice Z cu dimensiunea n x m. Dacă elementele matricei Y sunt numere reale, operaţia de transpunere face ca: Z(i,j)=Y(j,i). Dacă elementele matricei Y sunt numere complexe, operaţia de transpunere returnează conjugata transpusei, adică: Z(i,j)=conj(Y(j,i))=real(Y(j,i))-i*imag(Y(j,i))

Z(i,j) = conj(Y(j,i)) = real(Y(j,i))- i*imag(YG,i))

Exemplul 14. Să se determine transpusa matricei Z = [1 1+i;i 3-2*i]. I.

1.4.7. Alte instructiuni in prelucrarea matricelor si vectorilor

- Determinarea rangului unei matrice.

Fie o matrice A. Rangul matricei se determina cu instrucţiunea: rank(A)

- Determinantul unei matrice.

Fie o matrice A. Determinantul matricei se calculeazaă cu instrucţiunea: det(A)

- Inversa unei matrice.

Fie o matrice A. Inversa matricei se determina cu instrucţiunea: inv(A)

- Dimensiunea unei matrice

Fie o matrice A. Dimensiunea matricei se determina cu instrucţiunea: [m,n]=size(A)

- Lungimea unui vector

Fie un vector b. Lungimea vectorului se determina cu instrucţiunea: p=length (b)

GENERAREA VECTORILOR ŞI A MATRICELOR UZUALE

Funcţiile folosite pentru generarea vectorilor şi a matricelor uzuale sunt:

zeros	generează matricea nulă;
ones	generează matricea unitate;
eye	generează matricea identică;
rand	generează numere aleatoare cu distribuţie uniformă;
randn	generează numere aleatoare cu distribuţie normală;

linspace generează un vector cu pas liniar;

logspace generează un vector cu pas logaritmic;

meshgrid generează o matrice a reţelei în planul X-Y;

: generează un vector cu pas constant.

1. GENERAREA VECTORILOR

1.1. Generarea vectorilor cu pas liniar

Generarea vectorilor cu pas liniar implică cunoaşterea limitelor intervalului (*amin* şi *amax*) şi a pasului dintre două elemente (*pas*), sau numărul de elemente ale vectorului. Metoda de generare a vectorului se alege funcţie de datele de intrare.

- Dacă se cunosc limitele intervalului (*amin* şi *amax*) şi pasul (*pas*) dintre două elemente, se generează vectorul cu instrucţiunea:

x = amin : pas : amax

unde: *amin*, *amax* şi *pas* sunt scalari şi pot avea orice valoare reală. Numărul de elemente ale vectorului rezultant x este: $N = \left[\frac{a\max - a\min}{pas}\right] + 1$

unde „[ ]" semnifică partea întreagă a rezultatului expresiei dintre paranteze. Instrucţiunea presupune că:

- dacă *pas* > 0, atunci este necesar ca *amin* < *amax*,

- dacă *pas* < 0, atunci este necesar ca *amin* > *amax*. Spre exemplu:

x= 2:5:25

x= -20:3:10

x= 5:-2:-4

x =-15:-3:-25

x = 5:15

sunt corecte, în timp ce următoarele instrucţiuni: **x = 2:-l:5; x = -5:2:-10** sunt incorecte.

Dacă pasul se omite, valoarea acestuia este considerată implicit egală cu unitatea.

- Dacă se cunosc limitele intervalului (*amin* şi *amax*) şi numărul de elemente (N) ale vectorului generat cu pas liniar, atunci se foloseşte instrucţiunea:

x = linspace(amin, amax, N). Pasul dintre două elemente rezultă egal cu:

$$pas = \frac{a\max - a\min}{N-1}$$

Dacă valoarea N este omisă, atunci aceasta este considerată implicit egală cu 100. Valorile limitelor intervalului, *amin* şi *amax*, nu sunt supuse nici unei restricţii si pot fi date în orice ordine (dacă *amin* > *amax* vectorul generat va fi ordonat descrescător).

Exemplul 15. Să se genereze un vector cu pas liniar, cu limitele: amin =2.5, amax = 7 si pasul egal cu 1.25. Secvenţa:

x = 2.5 : 1.25 : 7 conduce la rezultatul:

X = [2,5000 3.7500 5,0000 6.2500]

Exemplul 16. Să se genereze un vector cu pas liniar, cu limitele: amin = 2.5, amax = 7 şi N = 4 elemente. Secvenţa:

x = linspace(2.5, 7, 4) conduce la rezultatul:

x = [2.5000 4.0000 5.5000 7.0000] , .

Deşi generează acelaşi număr de elemente, prima secvenţă controlează pasul şi poate modifica eventual limita superioară, iar a doua controlează numărul de elemente si menţine limitele impuse.

1.2. Generarea vectorilor cu pas logaritmic

Funcţia ***logspace*** generează vectori cu pas logaritmic; se apelează cu sintaxa: x = logspace(amin, amax, N). Vectorul x conţine N elemente distribuite logaritmic între decadele $[10^{amin}, 10^{amax}]$. Dacă numărul de elemente N este omis, se generează un vector cu 50 elemente distribuite logaritmic între decadele $[10^{amin}, 10^{amax}]$. Valorile limitelor intervalului, *amin* şi *amax*, nu au nici o restricţie şi pot fi date în orice ordine. Dacă *amin* > *amax*, vectorul generat va fi ordonat descrescător.

Exemplul 17. Să se genereze un vector cu N=5 elemente distribuite logaritmic pe intervalul $[10^{-2},10^{2}]$. Secvenţa:

x = logspace(-2,   2,   5) determină rezultatul:

x = [0.0100   0.1000   1.0000   10.0000 100.0000]

Exemplul 18. Să se genereze un vector cu N=6 elemente distribuite logaritmic pe intervalul $[10^{-2}, \pi]$. Secvenţa:

x = logspace(-2,   pi,   6) determină rezultatul:

x = [0.0100   0.0316   0.0997   0.3150   0.9948   3.1416].

Observaţie. Când amax=π, elementele vectorului sunt distribuite între *amin* şi pi.

2. GENERAEA MATRICELOR

2.1. Matricea goală

Deşi în MATLAB nu există instrucţiuni pentru declararea tipurilor de variabile, iar matricele se autodimensionează în timpul utilizării, pentru a creşte viteza de lucru se procedează la crearea unei matrice goale. Acest lucru se face la începutul sesiunii de lucru sau la apelarea unui program. Declararea unei matrice goale se face cu instrucţiunea:

X = [] care asignează lui X matricea de dimensiuni 0x0.

Orice matrice goală trebuie sa aibă cel puţin una din dimensiuni zero. Pentru a testa dacă o matrice X este goală, se foloseşte funcţia MATLAB isempty, care se apelează cu sintaxa:

r = isempty(X) şi retumează în variabila r valoarea 1 dacă X este matrice goală şi 0 în caz contrar.

2.2. Matricea unitate

Matricea unitate este o matrice cu toate elementele 1:

$$U = \begin{bmatrix} 1 & 1 & ... & 1 \\ 1 & 1 & ... & 1 \\ ... & ... & ... & ... \\ 1 & 1 & ... & 1 \end{bmatrix}$$

şi poate fi generată cu funcţia **ones**; se apelează cu una dintre sintaxele: **U=ones(n)** **U=ones(m,n)** **U=ones(size(A))**

unde m şi n sunt scalari, iar A este matrice. Dacă funcţia **ones** este apelată cu un singur argument scalar, matricea generată este o matrice pătrată având dimensiunea argumentului. Apelată cu două argumente scalare (m,n), matricea generată are m linii şi n coloane. Dacă funcţia ones are ca argument o matrice A, matricea generată este o matrice unitate de aceleaşi dimensiuni cu matricea A.

Exemplul 19. Să se genereze o matrice unitate cu dimensiunea matrice unitate cu dimensiunea 2x3. Cu secvenţa:

Al=ones(2) A2=ones(2,3)

Exemplul 20. Fie matricea: A = [1 4 8;0 2 3] .

Să se genereze o matrice unitate de aceleaşi dimensiuni cu matricea A.

2.3. Matricea zero

Matricea zero este o matrice cu toate elementele zero:

$$O = \begin{bmatrix} 0 & 0 & ... & 0 \\ 0 & 0 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & 0 \end{bmatrix}$$

şi poate fi generată cu funcţia **zeros**, care se apelează cu una dintre sintaxele:

O= zeros(n)   O= zeros(m,n)   O= zeros(size(A))

unde m şi n sunt scalari, iar A este matrice. Dacă funcţia **zeros** este apelată cu un singur argument scalar, matricea generată este o matrice pătrată având dimensiunea argumentului. Apelată cu două argumente scalare (m,n), matricea generată are m linii si n coloane. Dacă funcţia **zeros** are ea argument o matrice A, matricea generată este o matrice zero de aceleaşi dimensiuni-cu matricea A.

2.4. Matricea identitate

Matricea identitate este o matrice care are elementele de pe diagonala principală egale cu unu, iar toate celelalte egale cu zero:

$$I = \begin{bmatrix} 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & 1 \end{bmatrix}$$

Se generează cu funcţia **eye**, care se apelează cu una dintre sintaxele: I= eye(n); I= eye(m.n) I= eye(size(A))

unde m şi n sunt scalari, iar A este matrice. Dacă funcţia **eye** este apelată cu un singur argument scalar, matricea generată este o matrice pătrată având dimensiunea argumentului. Apelată cu două argumente scalare (m,n), matricea generată are m linii si n coloane. Dacă funcţia **eye** are ca argument o matrice A, matricea generată este o matrice identitate de aceleaşi dimensiuni cu matricea A.

Exemplul 21. Să se genereze o matrice identitate cu dimensiunea 2x2 şi o matrice identitate cu dimensiunea 2x3. Să se genereze o matrice identitate de aceleaşi dimensiuni cu matricea A: A=[1 3 9; 5 7 2];

2.5. Matricea aleatoare

Generarea matricelor cu numere aleatoare se poate face cu:

- funcţia **rand** pentru numere aleatoare cu distribuţie uniformă în intervalul (0.0, 1.0);

- funcţia **randn** pentru numere aleatoare cu distribuţie normală (Gaussiană), de medie zero şi varianţă unu. Sintaxele pentru generarea matricelor cu numere aleatoare sunt:

Ru=rand(n)   Rn=randn(n)

Ru=rand(m,n)   Rn=randn(m,n)

Ru=rand(size(A))   Rn=randn(size(A))

unde m şi n sunt scalari, iar A este matrice. Dacă funcţiile **rand** şi **randn** sunt apelate cu un singur argument scalar, matricea generată este o matrice pătrată având dimensiunea argumentului. Apelată cu două argumente scalare (m, n), matricea generată are m linii si n coloane. Dacă funcţiile **rand** şi **randn** au ca argument o matrice A, matricea generată este o matrice aleatoare de aceleaşi dimensiuni cu matricea A.

Pentru simularea experienţelor care comportă aceleaşi condiţii se generează serii de numere aleatoare la care se controlează un parametru de iniţializare al generatorului, ***seed***. rand('seed') sau randn('seed') - retumează valoarea curentă a numărului „seed"; rand('seed',n) sau randn('seed',n) - setează numărul „seed" la valoarea n.

Exemplul 22. Să se genereze o matrice aleatoare cu dimensiunea 2x2 şi o matrice aleatoare cu dimensiunea 2x3 cu elementele distribuite uniform si normal. Cu secvenţa:

Rlu=rand(2); Rln=rand-(2); R2u=rand(2,3);R2n=randn(2,3)

2.6. Generarea unei reţele (mesh)

Funcţia **meshgrid** transformă domeniul specificat prin vectorii x şi y în tablourile X şi Y care pot fi folosite atât pentru evaluarea funcţiilor de două variabile, cât şi pentru reprezentări 3D de tipul **mesh** sau **surface**. Se apelează cu una dintre sintaxele:

[X, Y]=meshgrid(x, y)   [X, Y]=meshgrid(x)

şi returnează datele în tablourile X si Y, care sunt sunt copii ale vectorului x si ale vectorului y. Cu alte cuvinte, funcţia **meshgrid** returnează în tablourile X şi Y perechile de coordonate ale tuturor punctelor din domeniul definit de vectori x şi y.

Exemplul 23. Să se genereze tablourile X şi Y pentru domeniul: $-2 \le x \le 2, -2 \le y \le 3$ cu pasul 2 pe axa Ox şi pasul 1 pe Oy. ([X,Y]=meshgrid(-2:2:2,-2:3))

Teme:

1) Se vor rula toae instrucţiunile şi exemplele din cadrul cursului.

2) 1) Fie matricea:

$$A = \begin{bmatrix} 1 & 7 & -7 \\ 1 & 2^4 & \sqrt{5} \\ |-2| & 5^2 & 0 \end{bmatrix}$$

Sa se scrie un program Matlab care realizeaza:

a) introducerea matricei A;
b) calculul matricei B astfel incat sa fie selectate coloanele 1 si 2, liniile 2 si 3 din matricea A.
c) calculul transpusei matricei B.
d) calculul determinantului matricei B.
e) calculul inversei matricei B.
f) calculul rangului matricei B.
g) calculul patratului matricei B.