Laborator 3

Calculul funcţiilor de matrice Exponenţiala matriceală

3.1 Tema

Înţelegerea conceptului de ”funcţie de matrice” şi însuşirea principalelor metode şi algoritmi de calcul al funcţilor de matrice.

3.2 Funcţii de matrice

Considerăm o funcţie $f : D \subset \mathbb{C} \to \mathbb{C}$ şi fie $A \in \mathcal{C}^{n\times n}$ o matrice dată. Ne propunem mai întâi să definim noţiunea de *funcţie de matrice* adică semnificaţia expresiei

$$F = f(A). \tag{3.1}$$

Fie $\mu_A(z)$ polinomul minimal al matricei A, i.e. polinomul monic de grad minim cu proprietatea $\mu_A(A) = 0$, şi $\lambda_i \in \mathcal{C},\ i = 1 : l,$ zerourile acestuia având ordinele de multiplicitate m_i. Avem

$$\mu_A(z) = z^m + \sum_{k=0}^{m-1} \alpha_i z^i = \prod_{i=1}^{l} (z - \lambda_i)^{m_i} \tag{3.2}$$

unde $m \stackrel{\text{def}}{=} \text{grad}(\mu_A) = \sum_{i=1}^{l} m_i$.

Definiţia 3.1 *Fie* $A \in \mathcal{C}^{n\times n}$ *şi*

$$\Lambda = \{(\lambda_i, m_i) \,|\, i = 1 : l,\ \lambda_i \in \mathcal{C},\ m_i \in \mathcal{N}\} \tag{3.3}$$

mulţimea zerourilor polinomului minimal μ_A *al matricei* A *împreună cu multiplicităţile respective. Dacă funcţia* $f : D \subset \mathcal{C} \to \mathcal{C}$ *este* analitică *pe o mulţime* deschisă D *ce conţine punctele* λ_i , $i = 1 : l$ *atunci spunem că* f *este definită pe spectrul matricei* A *iar* mulţimea valorilor funcţiei f pe spectrul matricei A *este*

$$f(\Lambda) = \left\{ f^{(k)}(\lambda_i) \,|\, i = 1 : l,\ k = 0 : m_i - 1 \right\}. \tag{3.4}$$

În particular, o funcţie întreagă f (i.e. analitică pe $D = \mathcal{C}$) este definită pe spectrul oricărei matrice $A \in \mathcal{C}^{n\times n}$.

Deşi introducerea noţiunii de funcţie de matrice prin intermediul polinomului de interpolare Lagrange-Sylvester este, probabil, mai intuitivă pentru un inginer, pentru evidenţierea rapidă a unor proprietăţi utile în elaborarea unor proceduri de calcul efectiv, preferăm următoarea definiţie.

Definiţia 3.2 *Fie $\lambda(A)$ spectrul matricei $A \in \mathcal{C}^{n\times n}$ şi $D \subset \mathcal{C}$ un domeniu cu frontiera Γ suficient de netedă astfel încât $\lambda(A) \subset D$. Dacă f este o funcţie analitică pe $D \cup \Gamma$ atunci*

$$F \stackrel{\text{def}}{=} f(A) = \frac{1}{2\pi i}\oint_\Gamma (zI-A)^{-1}f(z)dz\ . \tag{3.5}$$

Calculul elementelor matricei F, i.e. a integralei Cauchy

$$f_{ij} = \frac{1}{2\pi i}\oint_\Gamma e_i^T(zI-A)^{-1}e_j f(z)dz \tag{3.6}$$

poate fi efectuat, de exemplu, cu ajutorul teoremei reziduurilor, deşi această cale nu este recomandată decât, eventual, pentru matrice de dimensiuni nesemnificative.

Menţionăm, de asemenea, posibilitatea exprimării funcţiilor de matrice prin serii matriceale de puteri.

În continuare, prezentăm câteva proprietăţi ale funcţiilor de matrice, utile în dezvoltările procedurale care fac obiectul metodelor de calcul recomandate ca fiind cele mai bune în momentul actual.

Propoziţia 3.1 *Dacă funcţia f este definită pe spectrul matricei A atunci*

$$f(TAT^{-1}) = Tf(A)T^{-1}. \tag{3.7}$$

oricare ar fi matricea nesingulară T.

Propoziţia 3.2 *Dacă matricea $A \in \mathcal{C}^{n\times n}$ este superior (inferior) triunghiulară, iar f este o funcţie definită pe spectrul lui A atunci*

a) $F = f(A)$ este superior (inferior) triunghiulară;

b) $f_{ii} = f(a_{ii}), \quad i = 1:n$.

Propoziţia 3.3 *Fie $\lambda(A) = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ spectrul matricei $A \in \mathcal{C}^{n\times n}$. Atunci pentru orice funcţie f definită pe spectrul lui A avem*

$$\lambda(f(A)) = \{f(\lambda_1), \lambda_2, \ldots, f(\lambda_n)\}\,. \tag{3.8}$$

Propoziţia 3.4 *Matricele $A \in \mathcal{C}^{n\times n}$ şi $F = f(A)$ comută i.e.*

$$Af(A) = f(A)A. \tag{3.9}$$

3.3 Calculul funcţiilor de matrice. Algoritmi

Tehnicile numerice de evaluare a funcţiilor de matrice, recomandate de experienţa numerică acumulată se pot împărţi în două categorii:

a) metode bazate pe calculul valorilor proprii;

b) metode aproximative bazate pe trunchierea unor dezvoltări în serie.

Prezentăm, în continuare, procedurile bazate pe calculul valorilor proprii; metodele aproximative, bazate pe trunchierea unor dezvoltări în serie, vor fi aplicate în paragraful următor, dedicat cazului particular dar important pentru teoria sistemelor liniare, al calculului exponenţialei matriceale.

Pentru matricele simple (i.e. diagonalizabile) calculul funcţiilor de matrice prin evaluarea valorilor şi vectorilor proprii se bazează pe propoziţiile 3.2 şi 3.3. Într-adevăr, în acest caz, dacă $V = [\, v_1 \; v_2 \; \ldots \; v_n \,]$ este matricea vectorilor proprii ai matricei date $A \in \mathcal{C}^{n \times n}$ atunci $A = V \Lambda V^{-1}$ unde $\Lambda = \mathrm{diag}(\lambda_1,\, \lambda_2,\, \ldots,\, \lambda_n)$.

Prin urmare, pentru orice funcţie definită pe spectrul matricei A rezultă

$$F = f(A) = V f(\Lambda) V^{-1} = V diag(f(\lambda_1),\; f(\lambda_2),\; \ldots,\; f(\lambda_n)) V^{-1}. \tag{3.10}$$

Folosind funcţia MATLAB **eig** de calcul a valorilor şi vectorilor proprii ai unei matrice având sintaxa

$$[\, V,\, L \,] = \mathbf{eig}(A)$$

unde $L = \mathrm{diag}(\lambda_1,\; \lambda_2,\; \ldots,\; \lambda_n)$ este forma Jordan a matricei diagonalizabile A, iar coloanele matricei de transformare V sunt vectorii proprii asociaţi. Relaţia (3.10) stă la baza următorului algoritm de calcul al funcţiilor de matrice diagonalizabile.

Algoritmul 3.1 (Date matricea simplă $A \in \mathcal{C}^{n \times n}$ şi funcţia $f : D \subset \mathcal{C} \to \mathcal{C}$ definită pe spectrul matricei A, algoritmul calculează $F = f(A)$ prin determinarea valorilor şi vectorilor proprii.)

1. $[\, V,\, L \,] = \mathbf{eig}(A)$
2. $D = \mathbf{zeros}(n, n)$
3. `Pentru` $i = 1 : n$
   1. $D(i, i) = f(L(i, i))$
4. $F = V D / V$, i.e. se rezolvă în raport cu F sistemul matriceal liniar nesingular $FV = VD$.
5. Dacă se ştie a priori că rezultatul este real atunci $F = \mathrm{Re}(F)$.

Evident, efortul de calcul principal este destinat calculului valorilor şi vectorilor proprii şi, într-o oarecare măsură, rezolvării sistemului matriceal liniar.

Proprietăţi numerice mult mai bune, în toate cazurile, se obţin dacă în locul formei canonice Jordan se utilizează forma Schur complexă (sau reală) a matricei A. Această alternativă este condiţionată de existenţa unui algoritm eficient de calcul al funcţiilor de matrice triunghiulare (sau cvasitriunghiulare). Un astfel de algoritm, propus de B.N. Parlett [1], are la bază propoziţia 3.3 şi proprietatea de comutativitate formulată în propoziţia 3.4. Vom deduce algoritmul pentru cazul generic al matricelor cu valori proprii distincte.

Fie $T \in \mathcal{C}^{n \times n}$ o matrice superior triunghiulară cu $t_{ii} \neq t_{jj}$ pentru orice $i \neq j$ şi f o funcţie definită pe spectrul lui T. Dacă $F = f(T)$ atunci avem

$$FT = TF. \tag{3.11}$$

Ţinând cont că matricea F este, de asemenea, superior triunghiulară şi scriind (3.11) pe elemente obţinem

$$\sum_{k=i}^{j} t_{ik} f_{kj} = \sum_{k=i}^{j} f_{ik} t_{kj}, \quad j = 1 : n,\; i = 1 : j, \tag{3.12}$$

de unde, în ipotezele menţionate, avem

$$f_{ij} = \frac{1}{t_{jj} - t_{ii}} [t_{ij}(f_{jj} - f_{ii}) + \sum_{k=i+1}^{j-1} (t_{ik} f_{kj} - f_{ik} t_{kj})], \quad j = 1 : n, \ i = 1 : j. \tag{3.13}$$

Relaţia (3.13) este utilă numai în măsura în care se poate găsi o ordine de calcul a elementelor matricei F astfel încât, la fiecare moment al procesului de calcul, în membrul drept al expresiei (3.13) să apară numai elemente deja calculate. O astfel de ordine există şi ea poate fi evidenţiată observând că elementele diagonale sunt calculabile cu formula

$$f_{ii} = f(t_{ii}) \tag{3.14}$$

şi că în membrul drept al relaţiei (3.13) apar elementele $F(i, i : j-1)$ din ”stânga” elementului f_{ij} şi $F(i+1 : j, j)$ de ”sub” elementul f_{ij} (vezi diagrama din figura 3.1).

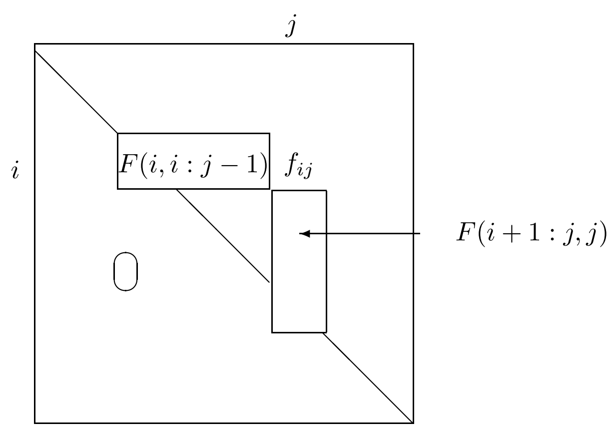

Figura 3.1: Matricea $F = f(T)$.

De exemplu, se poate adopta o ordine diagonală de calcul al elementelor triunghiului superior, după cum este indicat în diagrama din figura 3.2 pentru $n = 5$, numărul înscris în matrice marcând numărul de ordine pentru calculul elementului de pe poziţia respectivă.

1	6	10	13	15
	2	7	11	14
		3	8	12
			4	9
				5

Figura 3.2: Ordinea diagonală de calcul a elementelor matricei $F = f(T)$.

Aceasta nu este singura ordine posibilă. Într-adevăr, dacă efectuăm calculele pe coloane în ordinea $j = 1, 2, \ldots, n$, pe fiecare coloană calculele efectuându-se de jos în sus (vezi diagrama a) din figura 3.3) sau pe linii în ordinea $i = n, n-1, \ldots, 1$, pe fiecare linie ordinea de calcul fiind de la stânga la dreapta (diagrama b) din figura 3.3), atunci în momentul calculului unui element curent oarecare, elementele din stânga şi de sub elementul curent sunt deja calculate.

1	3	6	10	15
	2	5	9	14
		4	8	13
			7	12
				11

(a)

11	12	13	14	15
	7	8	9	10
		4	5	6
			2	3
				1

(b)

Figura 3.3: Ordinea ”pe coloane” (a) şi ordinea ”pe linii” (b) de calcul a elementelor matricei $F = f(T)$.

Prezentăm în continuare algoritmul corespunzător ordinii diagonale de calcul, celelalte variante făcând obiectul unor exerciţii. Pentru a urmări mai uşor indexarea, observăm că, atribuind indicele q direcţiilor paralele cu diagonala principală a matricei, rezultă următoarea schemă de calcul.

1. Se calculează $f_{ii} = f(t_{ii}),\ i = 1 : n$.
2. `Pentru` $q = 2 : n$
   1. `Pentru` $i = 1 : n - q + 1$
      1. Se calculează $f_{i,i+q-1}$ cu relaţia (3.13).

Pentru a nu modifica indexarea folosită în expresia (3.13) a elementului f_{ij} facem următoarele schimbările de indici

$$p = q - 1, \qquad j = i + p.$$

Schema de mai sus ne conduce la următorul algoritm de calcul al funcţiilor de matrice triunghiulare cu elementele diagonale distincte.

Algoritmul 3.2 (**Parlett**)(Date o matrice superior triunghiulară $T \in \mathcal{C}^{n\times n}$ cu $t_{ii} \neq t_{jj}$ pentru orice $i \neq j$ şi o funcţie $f : D \subset C \to C$ definită pe spectrul matricei T, algoritmul calculează, în ordine diagonală, elementele matricei superior triunghiulare $F = f(T)$.)

1. `Pentru` $i = 1 : n$
   1. $f_{ii} = f(t_{ii})$
2. `Pentru` $p = 1 : n - 1$
   1. `Pentru` $i = 1 : n - p$
      1. $j = i + p$
      2. $s = t_{ij}(f_{jj} - f_{ii})$
      3. `Dacă` $p > 1$ `atunci`
         1. `Pentru` $k = i + 1 : j - 1$
            1. $s = s + t_{ik}f_{kj} - f_{ik}t_{kj}$
      4. $f_{ij} = \dfrac{s}{t_{jj} - t_{ii}}$

Cu acest algoritm, completat cu procedura de aducere a unei matrice date la forma Schur complexă (superior triunghiulară) prin transformări ortogonale de asemănare, se obţine o procedură cu bune calităţi numerice pentru calculul funcţiilor de matrice. Utilizând funcţiile MATLAB **schur** şi **rsf2csf** se obţin matricea unitară de transformare U şi forma Schur complexă S a matricei A legate prin relaţiile

$$S = U^H A U, \quad A = U S U^H. \tag{3.15}$$

Introducând pentru utilizarea algoritmului 3.2 sintaxa

$$F = \mathbf{parlett}('f', T)$$

având ca parametrii de intrare şirul de caractere $'f'$ ce desemneazaa funcţia f şi matricea superior triunghiulară T, iar ca parametru de ieşire matricea $F = f(T)$ obţinem următorul algoritm.

> **Algoritmul 3.3** (Date o matrice A cu valori proprii distincte şi o funcţie f definită pe spectrul matricei A, algoritmul calculează matricea $F = f(A)$ prin metoda aducerii matricei A la forma Schur complexă.)
>
> 1. $[U, S] = \mathbf{schur}(A)$
> 2. $[U, S] = \mathbf{rsf2csf}(U, S)$
> 3. $T = \mathbf{parlett}(f, S)$
> 4. $F = UTU^H$
> 5. Dacă se ştie a priori că rezultatul este real atunci $F = \mathrm{Re}(F)$.

Calculul exponenţialei matriceale

Deşi metodele de calcul prezentate în secţiunea precedentă sunt aplicabile pentru toate funcţiile definite pe spectrul matricei argument, pentru anumite funcţii, de un interes aplicativ deosebit, au fost dezvoltate proceduri alternative, cu calităţi numerice superioare.

În acest paragraf prezentăm principalele metode pentru calculul exponenţialei matriceale

$$\Phi(t) = e^{tA}, \tag{3.16}$$

unde $t > 0$ este un parametru scalar. După cum se ştie, funcţia $\Phi(t)$ este matricea de tranziţie a stărilor sistemului liniar $\dot{x} = Ax$, adică satisface ecuaţia diferenţială matriceală liniară $\dot{\Phi} = A\Phi$ cu condiţia iniţială $\Phi(0) = I$.

Pentru a limita sensibilitatea lui $\Phi(t)$ în raport cu variaţiile lui A e necesar să limităm $t\|A\|$, de exemplu astfel încât

$$t\|A\| \leq 1, \tag{3.17}$$

fie alegând t suficient de mic, fie (în cazul în care t este impus) utilizând proprietatea

$$e^{tA} = \left(e^{\frac{t}{2^m}A}\right)^{2^m}, \quad m \geq 1. \tag{3.18}$$

Într-adevăr, oricare ar fi $t\|A\|$ există m astfel încât $\frac{t}{2^m}\|A\| \leq 1$ iar dacă $e^{\frac{t}{2^m}A}$ e cunoscut atunci, conform (3.18), e^{tA} poate fi calculat printr-un proces de ridicare succesivă la pătrat. O evaluare a lui m rezultă observând că dacă $t\|A\| \geq 1$ atunci trebuie să avem $\log_2(t\|A\|) < m$, deci

$$m = 1 + [\,\log_2(t\|A\|)\,], \tag{3.19}$$

unde $[\cdot]$ este partea întreagă a argumentului.

În practică m se poate alege printr-un procedeu de înjumătăţire succesivă. Schema de calcul, numită "de înjumătăţire şi ridicare la pătrat" (*scalling and squaring*), pe scurt $(1/2)^2$, este următoarea:

Schema $(1/2)^2$

1. Se calculează $\|A\|$.
2. Se iniţializează $m = 0$.
3. `Cât timp` $t\|A\| \geq 1$
   1. $t \leftarrow t/2$
   2. $m \leftarrow m + 1$
4. Se calculează $F = e^{tA}$.
5. `Cât timp` $m \geq 1$
   1. $F \leftarrow F^2$
   2. $m \leftarrow m - 1$

În ipoteza că $t\|A\|$ e limitată (de exemplu cu schema $(1/2)^2$), metodele uzuale de calcul al exponenţialei matriceale constau în construcţia unei aproximări *locale* în jurul lui $z = 0$ a funcţiei e^z.

De regulă aproximările utilizate sunt de tip polinomial (Taylor) sau raţional (Padé) şi permit calculul lui e^{tA} prin evaluarea aproximării considerate pentru $z = tA$.

I. Aproximaţia polinomială (Taylor) de ordin p este de forma

$$T_p(z) = \sum_{k=0}^{p} \frac{1}{k!}\, z^k \tag{3.20}$$

şi asigură cea mai bună aproximare polinomială locală de ordin p. Eroarea relativă de trunchiere comisă de aproximarea (3.20) este mărginită superior de $e_r \leq \frac{(t\|A\|)^{p+1}}{(p+1)!}$ astfel încât p se poate determina din relaţia

$$\frac{(t\|A\|)^{p+1}}{(p+1)!} \leq \mathrm{tol} \tag{3.21}$$

unde tol este o toleranţă acceptată.

Evaluarea polinomului matriceal $T_p(A)$ are loc după schema

$$T_{k+1}(tA) = T_k(tA) + X_k\,, \quad X_k = \frac{1}{k!} t^k A^k\,, \tag{3.22}$$

unde între doi termeni succesivi X_k şi X_{k-1} are loc relaţia

$$X_k = \frac{1}{k} tAX_{k-1}\,, \tag{3.23}$$

iar iniţializările sunt, evident, $T_0(tA) = I$ şi $X_0 = I$.

Ţinând seama de cele spuse mai sus, algoritmul de calcul al exponenţialei matriceale poate fi formulat în felul următor.

Algoritmul 3.4 (Date $A \in \mathcal{R}^{n\times n}$ şi $t \in \mathcal{R}$ algoritmul calculeaza $F = e^{tA}$, în ipoteza $t\|A\| < 1$, utilizând aproximaţia Taylor).

1. Se determină p din condiţia (3.21).
2. $X = I$
3. $F = I$
4. `Pentru` $k = 1 : p$
   1. $X \leftarrow \frac{1}{k} tAX$
   2. $F \leftarrow F + X$

Observaţia 3.1 În absenţa limitării lui $t\|A\|$, la pasul 4.2 pot avea loc fenomene de anulare prin scădere cu consecinţe catastofale asupra preciziei. Prin urmare, algoritmul de mai sus se asociază *în mod obligatoriu* cu **schema** $(1/2)^2$. În orice caz, pentru siguranţa calculului, se recomandă monitorizarea normei termenilor X. $\diamond$

II. Aproximaţia raţională (**Padé**) de grade $(p\,,\,q)$ şi de ordin $p+q$ este de forma

$$R_{pq}(z) = \frac{N_p(z)}{D_q(z)} = \frac{c_0 + c_1 z + \cdots + c_p z^p}{1 + d_1 z + \cdots + d_q z^q}\,, \qquad (d_0 \stackrel{\text{def}}{=} 1)\,, \tag{3.24}$$

unde coeficienţii c_k , d_k sunt

$$c_k = \frac{(p+q-k)!p!}{(p+q)!k!(p-k)!}\,, \quad d_k = \frac{(p+q-k)!q!}{(p+q)!k!(q-k)!}(-1)^k\,. \tag{3.25}$$

Eroarea de trunchiere poate fi limitată alegînd convenabil gradele p, q ale a proximării (3.24). Se poate arăta că dacă $t\|A\| < \frac{1}{2}$ atunci

$$R_{pq}(tA) = e^{t(A+E)} \tag{3.26}$$

unde $AE = EA$ iar

$$\|tE\| \le 8\frac{p!q!}{(p+q)!} \cdot \frac{(t\|A\|)^{p+q+1}}{(p+q+1)!}\,. \tag{3.27}$$

Eroarea relativă de trunchiere comisă prin utilizarea aproximaţiei Padé $R_{pq}(tA)$ poate fi evaluată acoperitor prin

$$\frac{\|e^{t(A+E)} - e^{tA}\|}{\|e^{tA}\|} \le \|e^{tE} - I\| = \|tE + \frac{t^2E^2}{2!} + \cdots\| \le t\,\|E\|\,e^{t\|E\|}\,. \tag{3.28}$$

Din motive de eficienţă se recomandă $p = q$, cu care valoarea concretă se alege din condiţia $\|tE\| <$ eps, unde eps este o toleranţă prescrisă, mai precis

$$8\frac{(p!)^2}{(2p)!} \cdot \frac{(t\|A\|)^{2p+1}}{(2p+1)!} \le eps\,, \tag{3.29}$$

În MATLAB se consideră că este suficient să se aleagă $p = q = 6$.

Polinoamele $N_p(tA)$ şi $D_p(tA)$ se evaluează după schema cunoscută, adică

$$N_{k+1}(tA) = N_k(tA) + X_k\ ,\ X_k = c_k t^k A^k \tag{3.30}$$

unde

$$X_k = \frac{p-k+1}{k(2p-k+1)} tAX_{k-1}. \tag{3.31}$$

Rezultă următorul algoritm.

Algoritmul 3.5 (**Padé**) (Date $A \in \mathcal{R}^{n\times n}$ şi $t \in \mathcal{R}$ astfel încât $t\|A\| < \frac{1}{2}$, algoritmul calculează $F = e^{tA}$ utilizând aproximaţia Padé).

1. Se determină p din condiţia (3.29) sau se consideră $p = 6$.
2. $X = I$
3. $N = I$
4. $D = I$
5. `Pentru` $k = 1 : p$
    1. $X \leftarrow \dfrac{p-k+1}{k(2p-k+1)} tAX$
    2. $N \leftarrow N + X$
    3. $D \leftarrow D + (-1)^k X$
6. $F = d\ N$, i.e. se rezolvă ecuaţia matriceală $DF = N$ în raport cu F.

Observaţia 3.2 Si algoritmul de mai sus se asociază *în mod obligatoriu* cu **schema** $(1/2)^2$. Pentru siguranţa calculului, în afară de monitorizarea normei termenilor X, se recomandă testarea lui cond(D) la pasul 6. $\diamond$

Programe MATLAB disponibile

Pentru calculul unei funcţii de matrice cu valori proprii distincte se poate folosi funcţia **funm**. Pentru calculul exponenţialei matriceale este disponibilă funcţia **expm**.

3.4 Sarcini de lucru

A. În laborator

1. Se va scrie programul MATLAB pentru implementarea algoritmului de calcul al funcţiilor de matrice bazate pe calculul valorilor şi vectorilor proprii şi se vor testa pe exemple numerice semnificative $(m, n > 20)$ pentru funcţiile exp, sin, cos, sqrt. Se vor compara soluţiile calculate cu programul propriu cu cele oferite de funcţiile MATLAB disponibile.

2. Se vor scrie programele MATLAB pentru implementarea algoritmului Parlett precum şi pentru implementarea algoritmului 3.5. Se vor compara soluţiile calculate pentru funcţiile exp, sin, cos, sqrt cu programe proprii cu cele oferite de funcţiile MATLAB disponibile şi cu rezultatele oferite de programul de la punctul 1.

3. Se va scrie programul MATLAB pentru implementarea algoritmului de calcul al exponenţialei matriceale bazat pe aproximarea Pad'e incluzând schema $(1/2)^2$. Se vor compara soluţiile calculate cu programe proprii cu cele oferite de funcţiile MATLAB disponibile şi cu rezultatele oferite de programul de la punctele 1 şi 2.

4. Se vor analiza sursele funcţiilor MATLAB **funm** şi **expm** şi se vor identifica metodele folosite.

B. Acasă

1. Se vor scrie programele MATLAB pentru implementarea algoritmilor Parlett în versiunile "pe linii" şi "pe coloane" şi se vor compara rezultatele oferite de cele trei versiuni testate pe exemple numerice semnificative ($m, n > 20$) pentru funcţiile exp, sin, cos, sqrt.

2. Se va scrie programul MATLAB pentru implementarea algoritmului de calcul al exponenţialei matriceale bazat pe aproximaţia Taylor şi se va testa pe exemple numerice semnificative ($m, n > 20$). Se vor compara soluţiile calculate cu acest program cu soluţiile calculate de celelalte programe. Ce constataţi?

3. **Exerciţii şi teme de casă.**

   E 3.1 Adaptaţi algoritmul Parlett pentru calculul funcţiilor de matrice inferior triunghiulare cu elementele diagonale distincte.

   E 3.2 Scrieţi un algoritm eficient pentru calculul funcţiei $F = f(A)$, unde A este superior bidiagonală cu elemente diagonale distincte.

   E 3.3 Scrieţi un algoritm eficient pentru calculul unei funcţii $F = f(A)$, unde $A \in \mathcal{R}^{n\times n}$ este simetrică. Consideraţi cazurile $f(z) = e^z,\ f(z) = ln\, z,\ f(z) = \sqrt{z}$.

   E 3.4 Scrieţi un algoritm eficient pentru calculul matricei de tranziţie $\Phi(k) = A^k$ a sistemului liniar discret $x(k+1) = Ax(k)$.

   E 3.5 Stabiliţi legătura dintre funcţiile considerate mai sus şi exponenţiala matriceală e^{tA^0}, unde
   $$A^0 = \begin{bmatrix} 0 & I_n \\ -A & 0 \end{bmatrix}.$$
   Ce implicaţii calculatorii are constatarea făcută?

Bibliografie

[1] **Jora B., Popeea C., Barbulea S.** *Metode de Calcul Numeric în Automaticăa*, Ed. Enciclopedică, Bucureşti 1996.