B. Jora C. Popeea S. Barbulea

METODE DE CALCUL NUMERIC ÎN AUTOMATICĂ

SISTEME LINIARE

EDITURA ENCICLOPEDICĂ
Bucureşti 1996

Cuprins

Cuvânt introductiv

Lucrarea de faţă reprezintă o primă parte a cursului universitar **Metode Numerice**, destinat studenţilor anului III al facultăţii Automatică şi Calculatoare din Universitatea POLITEHNICA din Bucureşti şi, evident, poate fi utilă tuturor studenţilor de la facultăţile de acelaşi profil din ţară. Desigur, ea poate interesa şi pe alţi utilizatori potenţiali (studenţi, cadre didactice, cercetători, ingineri, economişti, etc.) ai metodelor moderne de calcul în analiza şi proiectarea sistemică.

Cititorul *ideal* al lucrării trebuie să posede cunoştinţe de bază privind metodele de calcul numeric matricial – enumerate în finalul acestui cuvânt introductiv – precum şi unele informaţii sigure asupra principalelor noţiuni de teoria sistemelor liniare. De asemenea, el trebuie să aibă obişnuinţa unei gândiri limpezi, "algoritmice", precum şi pasiunea lucrului la calculator.

Desigur, cititorul *real*, dispunând într-un grad mai mare sau mai mic de calităţile menţionate mai sus, şi le poate perfecţiona prin studiul lucrării, confirmând, şi pe această cale, viabilitatea principiilor sistemice.

Lucrarea este structurată în 8 capitole. Ea este însoţită de o bibliografie de bază (cu lucrări referite prin utilizarea cifrelor romane), utilă în general pentru abordarea tuturor temelor propuse. Bibliografii specifice selective, dar care pot fi extinse cu uşurinţă de către însuşi cititorul interesat prin cumularea bibliografiilor lucrărilor citate, sunt ataşate fiecărui capitol. De asemenea, lucrarea este însoţită de o listă de notaţii uzuale iar cuprinsul poate constitui, în acelaşi timp, indice de noţiuni.

Capitolul 1 completează cunoştinţele cititorului privind metodele de rezolvare a sistemelor de ecuaţii liniare cu cazul matricial în care necunoscutele sunt structurate într-o matrice $X \in \mathcal{R}^{m \times n}$, $m > 1$, $n > 1$, fapt care permite dezvoltarea unor tehnici specifice. În particular, ecuaţiile matriciale Liapunov vor fi întâlnite aproape în toate capitolele lucrării.

Capitolul 2 tratează problema importantă a calculului funcţiilor de matrici. Un accent special este pus pe calculul exponenţialei matriciale, uti-

lizate intensiv în capitolele 4 şi 7.

Capitolul 3 expune principalele tehnici de procesare a modelelor sistemice liniare (conexiuni, realizări, conversii de modele, algoritmi de calcul polinomial) fără de care nu este posibilă abordarea niciunei probleme de analiză sau sinteză sistemică asistate de calculator.

Capitolul 4 descrie metodele numerice de simulare, i.e. de calcul al răspunsului în timp al sistemelor liniare la stimuli externi caracteristici pentru diverse regimuri de funcţionare. Aceste tehnici de simulare sunt absolut necesare pentru validarea demersului teoretic şi calculatoriu de analiză şi sinteză sistemică şi reprezintă o punte de legătură esenţială către lumea reală a experimentului fizic.

Capitolul 5 prezintă procedurile numerice de analiză a proprietăţilor sistemice fundamentale (stabilitate, controlabilitate, observabilitate, etc.) precum şi principalele metode de construcţie a realizărilor minimale şi echilibrate. În acest fel conţinutul acestui capitol se constituie într-o colecţie de algoritmi fundamentali, componente de bază ale oricărei activităţi de concepţie asistate de calculator a unor sisteme de conducere complexe.

Capitolul 6 realizează o trecere în revistă a principalelor proceduri numerice de alocare a polilor sistemelor liniare. În contextul general al tehnicilor de sinteză, alocarea polilor reprezintă un instrument necesar pentru asigurarea unei dinamici dorite sistemelor automate elaborate. Dacă în cazul sistemelor simple reacţia după stare ce realizează alocarea este unic determinată şi, deci, libertăţile de calcul sunt strict procedurale, în cazul sistemelor multiple apar posibilităţi suplimentare de optimizare sau robustificare a spectrului alocat.

Capitolul 7 abordează, din punctul de vedere al metodelor numerice de calcul, problema de sinteză liniar-patratică, care vizează optimizarea unui sistem liniar cu indice de calitate patratic. În aplicaţii problema liniar-patratică generală apare sub forma unor probleme cu finalităţi sau cu tehnici de tratare specifice. În acest context sunt prezentate metodele de calcul cele mai performante pentru rezolvarea ecuaţiilor matriciale Riccati, însoţite de analize comparative ale eficienţei şi stabilităţii lor numerice.

Capitolul 8 consideră unele generalizări ale problemei de sinteză liniar-patratice şi ilustrează aplicarea algoritmilor de calcul stabiliţi la construcţia compensatoarelor H_2 şi H_∞ (sub)optimale.

Principalele rezultate ale expunerii sunt concretizate sub formă de algoritmi de calcul direct implementabili iar fiecare capitol este însoţit de un set de probleme.

Ne exprimăm opinia că pentru însuşirea materialului prezentat este absolut necesară *rezolvarea* problemelor propuse şi, mai ales, *implementarea* algoritmilor, urmată de *experimentarea* lor pe exemple numerice concrete şi semnificative.

Autorii mulţumesc colegului lor prof. Paul Flondor pentru comentariile constructive făcute pe marginea lucrării. De asemenea, mulţumesc domnului Marcel Popa, directorul Editurii Enciclopedice, pentru atenţia plină de solicitudine cu care a urmărit apariţia promptă şi în cele mai bune condiţii a lucrării. Autorii apreciază în mod deosebit interesul manifestat de domnul ing. Sabin Stamatescu, directorul firmei ASTI Bucureşti, faţă de conţinutul lucrării şi posibilele aplicaţii ale procedurilor de calcul prezentate în conducerea cu calculator a proceselor industriale. În sfârşit, dar, desigur, nu în ultimul rând, mulţumiri se cuvin studenţilor Siana Petropol, Simona Chirvase, Ştefan Murgu, Cătălin Petrescu, Simona Ruxandu, Adrian Sandu, Laura Stan, Florin Roman, Mădălina Zamfir, Ion Vasile care au realizat redactarea computerizată a lucrării cu menţiunea expresă că fără munca lor plină de dăruire nimic din ce am realizat împreună nu ar fi fost posibil.

*
* *

Enumerăm în continuare principalele probleme de calcul numeric matricial, împreună cu succinte referiri la modul de rezolvare al acestora, pe care cititorul interesat de cuprinsul lucrării de faţă este bine să le aibă în vedere.

1. Pentru rezolvarea unui sistem de ecuaţii liniare $Ax = b$, în care matricea $A \in \mathcal{R}^{n\times n}$ este *inversabilă*, se utilizează *procedura de triangularizare prin eliminare gaussiană cu pivotare parţială*

$$A \leftarrow R = MA, \qquad b \leftarrow Mb,$$

unde R rezultă superior triunghiulară inversabilă iar M este o secvenţă de transformări elementare stabilizate, urmată de rezolvarea – prin substituţie înapoi – a sistemului superior triunghiular rezultat. Efortul de calcul necesar este $N_{op} \approx n^3/3$.

În cazul *simetric* $A = A^T$, se recomandă conservarea acestei proprietăţi prin utilizarea schemei de eliminare simetrizate $A \leftarrow D = MAM^T$, propuse de Parlett [III], în care D rezultă cvasidiagonală.

În cazul *simetric pozitiv definit* $A = A^T > 0$, se impune aplicarea factorizării Cholesky $A = LL^T$.

2. Pentru rezolvarea, în sensul celor mai mici patrate (CMMP), a sistemului supradeterminat $Ax = b$, unde matricea $A \in \mathcal{R}^{m\times n}$ este *monică*,

i.e. are coloanele liniar independente (rang$A = n$), se utilizează *procedura de triangularizare ortogonală*

$$A \leftarrow R = UA, \qquad b \leftarrow Ub,$$

unde R rezultă superior triunghiulară iar U este o secvenţă de transformări ortogonale elementare (reflectori sau rotaţii). Triangularizarea ortogonală este, evident, echivalentă cu factorizarea (sau descompunerea) $A = QR$, unde factorul ortogonal $Q = U^T$ se obţine în formă factorizată. Efortul de calcul necesar este $N_{op} \approx mn^2 - n^3/3$ iar eventuala formare explicită a matricii Q, prin acumularea transformărilor elementare parţiale, necesită încă $\approx 2n^3/3$ operaţii în cazul cel mai defavorabil ($m = n$).

Dacă matricea A nu este neapărat monică, atunci, pentru asigurarea stabilităţii numerice este necesară utilizarea unei strategii adecvate de *pivotare a coloanelor*, i.e. $A \leftarrow R = UAP$, unde P este o matrice de permutare iar R rezultă superior trapezoidală, vezi [III].

3. Pentru rezolvarea, în sens CMMP, a sistemului subdeterminat $Ax = b$, unde matricea $A \in \mathcal{R}^{m\times n}$ este *epică*, i.e. are liniile liniar independente (rang$A = m$), se utilizează *procedura de triangularizare ortogonală la dreapta*

$$A \leftarrow L = AV,$$

unde L rezultă inferior triunghiulară. (Această procedură este echivalentă cu procedura precedentă reformulată prin dualitate, i.e. aplicată lui A^T.) Efortul de calcul necesar este $N_{op} \approx nm^2 - m^3/3$.

4. Pentru calculul valorilor proprii ale unei matrici $A \in \mathcal{R}^{n\times n}$ se utilizează *algoritmul* **QR** care, în esenţă, construieşte iterativ forma Schur S a lui A, i.e.

$$A \leftarrow S = Q^T AQ, \qquad (N_{op} \approx 6n^3),$$

unde S rezultă cvasisuperior triunghiulară iar Q este o secvenţă de transformări ortogonale. În caz de necesitate valorile proprii pot fi ordonate pe diagonala principală a lui S, conform oricărui criteriu de ordonare impus, cu ajutorul unei secvenţe suplimentare de transformări ortogonale de asemănare. Calculul matricii Q prin acumularea transformărilor parţiale (necesară, e.g. pentru calculul vectorilor proprii) implică aproape dublarea numărului de operaţii menţionat.

În cazul *simetric* $A = A^T$, se utilizează avantajos versiunea simetrică a algoritmului **QR**, i.e. $A \leftarrow \Lambda = Q^T AQ$, unde Λ rezultă diagonală.

5. Pentru rezolvarea problemei CMMP generale în care matricea $A \in \mathcal{R}^{m\times n}$ este de rang nu neapărat maximal $r \stackrel{\text{def}}{=} \text{rang}A \leq \min(m, n)$, precum

şi a altor probleme importante de calcul numeric matricial (determinarea rangului, a pseudoinversei, operaţii cu subspaţii liniare, etc.) se utilizează *descompunerea valorilor singulare* **DVS**

$$U^T A V = \Sigma,$$

unde matricea Σ conţine pe diagonala principală valorile singulare nenule $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$ ale lui A (restul elementelor fiind nule) iar matricile U, V sunt ortogonale. (Algoritmul **DVS** utilizat pentru calculul descompunerii de mai sus constituie o adaptare ingenioasă a algoritmului **QR** simetric). Din nou, formarea factorilor ortogonali U, V necesită acumularea (relativ costisitoare) a transformărilor parţiale.

6. Pentru calculul valorilor proprii generalizate ale unui fascicol $\lambda B - A$, unde A, $B \in \mathcal{R}^{n\times n}$, se utilizează algoritmul **QZ** care, în esenţă, aduce perechea (A, B) la *forma Schur generalizată*

$$A \leftarrow S = Q^T A Z, \qquad B \leftarrow T = Q^T A Z,$$

unde S rezultă cvasisuperior triunghiulară, T rezultă superior triunghiulară iar Q şi Z sunt secvenţe de transformări ortogonale.

În cazul *simetric* $A = A^T$, $B = B^T > 0$ se utilizează factorizarea Cholesky $B = LL^T$ şi se aplică algoritmul **QR** simetric matricii $\tilde{A} = L^{-1}AL^{-T}$, ceea ce conduce la diagonalizarea simultană a ambelor matrici A, B.

Pentru detalii privind aspectele teoretice şi procedurale ale problemelor de calcul matricial enumerate mai sus recomandăm consultarea referinţelor bibliografice [V], [VI] sau [IX], [X] .

Bibliografie

- *Pentru programe de calcul şi indicaţii de utilizare:*

[I] **Smith B.T., Boyle J.M., Ikebe Y., Klema V.C., Moler C.B.** Matrix Eigensystem Routines: EISPACK Guide, 2-nd ed., Springer-Verlag, New York, 1970.

[II] **Garbow B.S., Boyle J.M., Dongarra J.J., Moler C.B.** Matrix Eigensystem Routines: EISPACK Guide Extension, Springer - Verlag, New York, 1972.

[III] **Dongarra J.J., Bunch J.R., Moler C.B., Stewart G.W.** LINPACK User's Guide, SIAM Publications, Philadelphia, PA, 1978.

[IV] **Moler C.B., Little J.N., Bangert S.** PC-MATLAB User's Guide, The Math Works Inc., 20 N. Main St., Sherborn, Mass., 1987.

- *Pentru algoritmi de calcul matricial:*

[V] **Stewart G. W.** Introduction to Matrix Computations, Academic Press, New York and London, 1973.

[VI] **Golub G. H., Van Loan Ch. F.** Matrix Computations, Second edition, The Johns Hopkins University Press, Baltimore and London, 1988.

- *Pentru chestiuni teoretice de calcul matricial:*

[VII] **Gantmaher F.R.** Teoriia matriţ (ediţia a 2-a), Ed. Nauka, Moscova, 1966. (The Theory of Matrices, vols. 1-2, Chelsea, New York, 1959).

- *Lucrări în limba română:*

[VIII] **Ionescu V., Lupaş L.** Tehnici de calcul în teoria sistemelor, vol.1. Sisteme liniare, vol.2. Sisteme optimale, E.T., Bucureşti, 1973.

[IX] **Bucur C.M., Popeea C.A., Simion Gh.Gh.** Matematici speciale. Calcul numeric, E.D.P., Bucureşti, 1983.

[X] **Ionescu V., Varga A.** Teoria sistemelor. Sinteza robustă. Metode numerice de calcul., Ed. ALL, Bucureşti, 1994.

[XI] **Iorga V., Jora B., Nicolescu C., Lopătan I., Fătu I.,** Programare numerică, Ed. Teora, Bucureşti, 1996.

- *Alte titluri de uz general:*

[XII] **Åström K.J., Wittenmark B.** Computer Controlled Systems, Prentice Hall Inc., Englewood Cliffs, NJ, 1984.

[XIII] **Jamshidi M., Herget C.J. (ed.),** Computer-Aided Control Systems Engineering, North-Holland, Amsterdam, 1985. (Maşinostroienie, Moscova, 1989.)

[XIV] **Solodovnicov V.V. (ed.),** Avtomatizirovannoe proektirovanie sistem upravleniia, Maşinostroienie, Moscova, 1990.

[XV] **Jamshidi M., Tarokh M., Shafai B.** Computer-Aided Analysis and Design of Linear Control Systems, Prentice Hall Inc., Englewood Cliffs, NJ, 1992.

Lista de notaţii

• Notaţii generale

$\mathcal{N}$ – mulţimea numerelor naturale.

$\mathcal{Z}$ – mulţimea numerelor întregi.

$\mathcal{R}$ – mulţimea numerelor reale.

$\mathcal{C}$ – mulţimea numerelor complexe.

$\mathcal{C}^-$ – semiplanul stâng deschis al planului variabilei complexe s.

$\mathcal{D}_1(0)$ – discul unitate deschis al planului variabilei complexe z.

$\mathcal{R}^n$ – spaţiul liniar n-dimensional al vectorilor (coloană) x cu n componente reale $x_i \in \mathcal{R}$, $i = 1 : n$.

$\mathcal{C}^n$ – spaţiul liniar n-dimensional al vectorilor (coloană) x cu n componente complexe $x_i \in \mathcal{C}$, $i = 1 : n$.

$e_k,\ k = 1 : n$ – baza standard a spaţiului liniar $\mathcal{R}^n$, respectiv $\mathcal{C}^n$.

$\mathcal{R}^{m\times n}$ – spaţiul liniar al matricilor cu m linii şi n coloane cu elemente reale $a_{ij} \in \mathcal{R}$, $i = 1 : m$, $j = 1 : n$.

$\mathcal{C}^{m\times n}$ – spaţiul liniar al matricilor cu m linii şi n coloane cu elemente complexe $a_{ij} \in \mathcal{C}$, $i = 1 : m$, $j = 1 : n$. [1]

[1] În calcule, vectorii se identifică cu matricile cu o singură coloană iar scalarii se identifică cu matricile (sau vectorii) cu un singur element.

A^T – transpusa matricii (reale sau complexe) A.

A^H – conjugata hermitică a matricii (complexe) A, i.e. $A^H = \bar{A}^T$, unde $\bar{A}$ este conjugata complexă a lui A.

A^+ – pseudoinversa normală (Moore-Penrose) a matricii A; dacă A este monică $A^+ = (A^T A)^{-1} A^T$, dacă A este epică atunci $A^+ = A^T (A A^T)^{-1}$.

$\sigma_i(A),\ i = 1 : p,\ p = \min(m, n)$ – valorile singulare ale matricii A ordonate astfel încât $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p$.

$\sigma(A)$ – mulţimea $\{\sigma_1(A), \sigma_2(A), \ldots, \sigma_p(A)\}$ a valorilor singulare ale matricii A.

$r = \text{rang} A$ – rangul matricii A, i.e. numărul valorilor singulare nenule.

I_n – matricea unitate de ordinul n.

A^{-1} – inversa matricii patrate nesingulare A, i.e. $AA^{-1} = A^{-1}A = I_n$.

$A^{-T} = (A^{-1})^T = (A^T)^{-1}$

$A^{-H} = (A^{-1})^H = (A^H)^{-1}$

$\text{tr} A$ – urma matricii patrate A, i.e. suma elementelor diagonale.

$\det A$ – determinantul matricii patrate A.

$\lambda_i(A),\ i = 1 : n$ – valorile proprii ale matricii patrate A de ordin n.

$\lambda(A)$ – spectrul (de valori proprii) $\{\lambda_1(A), \lambda_2(A), \ldots, \lambda_n(A)\}$ al matricii A.

$\rho(A) = \max_{i=1:n} |\lambda_i(A)|$ – raza spectrală a matricii A.

$\text{cond}(A) = \|A\| \, \|A^{-1}\|$ – numărul de condiţie la inversare al matricii A ($\|\cdot\|$ este o normă matricială consistentă, vezi mai jos).

$(x, y) = y^T x$ – produsul scalar standard a doi vectori reali; în cazul complex produsul scalar este $(x, y) = y^H x$.

$\|x\| = (x, x)^{1/2}$ – norma euclidiană a vectorului x; se notează $\|x\|_Q^2 = x^T Q x$ unde Q este o matrice simetrică ($Q = Q^T$).

$\|x\|_p = \left(\sum_{i=1}^n |x|^p\right)^{1/p}$ – p-normele vectorului n-dimensional x, $p \geq 1$; în calcule se utilizează în special $\|x\|_1, \|x\|_2 = \|x\|$ şi $\|x\|_\infty = \max_{i=1:n} |x_i|$.

$(A, B) = \text{tr}(B^T A)$ $(\text{tr}(B^H A))$ – produsul scalar a două matrici reale (complexe).

$\|A\|_F = (A,B)^{1/2}$ – norma Frobenius a matricii A,
$\|A\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2$ sau $\|A\|_F^2 = \sum_{i=1}^r {\sigma_i}^2$.

$|A|_p = (\sum_{i=1}^r {\sigma_i}^p)^{1/p}$ – p-normele Schatten, $p \geq 1$; în calcule se utilizează în special norma-urmă $|A|_1 = \sum_{i=1}^r \sigma_i$, norma Frobenius $|A|_2 = \|A\|_F$ şi norma spectrală $|A|_\infty = \sigma_1(A)$.

$\|A\|_p = \max_{\|x\|_p=1} \|Ax\|_p$ – p-normele induse; în calcule se utilizează în special norma $\|A\|_1 = \max_{j=1:n} \sum_{i=1}^m |a_{ij}|$, norma spectrală $\|A\|_2 = \sigma_1(A)$ şi norma $\|A\|_\infty = \max_{i=1:m} \sum_{j=1}^n |a_{ij}|$.

• Notaţii sistemice generale

$x \in \mathcal{R}^n$ – vectorul (mărimilor) de stare.

$u \in \mathcal{R}^m$ – vectorul (mărimilor) de intrare.

$y \in \mathcal{R}^l$ – vectorul (mărimilor) de ieşire.

$S = (A,B,C,D)$ – reprezentarea de stare a unui sistem liniar ($A \in \mathcal{R}^{n\times n}$, $B \in \mathcal{R}^{n\times m}$, $C \in \mathcal{R}^{l\times n}$ şi $D \in \mathcal{R}^{l\times m}$).

$T = (N,p)$ – reprezentarea de transfer a unui sistem liniar (N este matricea coeficienţilor polinoamelor ce definesc numărătorul matricii de transfer iar p este vectorul coeficienţilor polinomului numitor comun i.e. $T(s) = C(sI-A)^{-1}B + D = N(s)/p(s)$).

$\delta(t)$ $(\delta(k))$ – impulsul unitate continuu (discret).

$1(t)$ $(1(k))$ – funcţia treaptă unitară continuă (discretă).

• Transformări

(1) MAN (MAN^{-1} sau MAN^T) – transformare de *echivalenţă* (bilaterală) a matricii $A \in \mathcal{R}^{m\times n}$ (M şi N sunt matrici patrate nesingulare; transformarea de echivalenţă conservă rangul iar dacă M, N sunt ortogonale atunci conservă şi valorile singulare).

(2) NAN^{-1} – transformare de *asemănare* a matricii $A \in \mathcal{R}^{n\times n}$ (transformarea de asemănare conservă valorile proprii).

(3) NAN^T – transformare de *congruenţă* a matricii $A \in \mathcal{R}^{n\times n}$ (N este nesingulară; aplicată unei matrici A simetrice, transformarea de congruenţă conservă rangul şi inerţia i.e. numerele de valori proprii negative, nule şi, respectiv, pozitive).

Dacă N este ortogonală atunci transformările (2) şi (3) coincid şi definesc transformarea de *asemănare ortogonală*.

- **Prescurtări**

SISO – siglă pentru sisteme simple având o singură intrare şi o singură ieşire (**S**ingle-**I**nput **S**ingle-**O**utput).

SIMO, (MISO, MIMO) – siglă pentru sistemele cu o singură intrare şi mai multe ieşiri. Celelalte prescurtări au semnificaţii evidente.

FSR(G) – forma Schur reală (generalizată).

FSC(G) – forma Schur complexă (generalizată).

DVS – descompunerea valorilor singulare.

FSH – forma (bloc-)superior Hessenberg.

EM(A)L – ecuaţie matricială (algebrică) Liapunov continuă.

DEM(A)L – ecuaţie matricială (algebrică) Liapunov discretă.

EM(A)R – ecuaţie matricială (algebrică) Riccati continuă.

DEM(A)R – ecuaţie matricială (algebrică) Riccati discretă.

Capitolul 1

Rezolvarea ecuaţiilor matriceale liniare

Acest capitol este consacrat tehnicilor de rezolvare a unor sisteme liniare, în general de mari dimensiuni, structurate în exprimări matriceale care permit dezvoltarea unor metode specifice de calcul.

1.1 Ecuaţii matriceale liniare

Ecuaţiile matriceale liniare sunt sisteme de ecuaţii liniare care pot fi scrise compact într-o formă matriceală de tipul

$$f(X) = C \tag{1.1}$$

unde $X \in \mathcal{R}^{m\times n}$ este matricea necunoscutelor, $C \in \mathcal{R}^{p\times q}$ o matrice dată, iar $f : \mathcal{R}^{m\times n} \to \mathcal{R}^{p\times q}$ o aplicaţie liniară (i.e. f satisface $f(X_1 + X_2) = = f(X_1) + f(X_2)$ şi $f(\alpha X_1) = \alpha f(X_1)$ pentru orice matrice $X_1, X_2 \in \mathcal{R}^{m\times n}$ şi orice scalar α). Se poate arăta că orice aplicaţie liniară de argument matriceal poate fi scrisă sub forma

$$f(X) = \sum_{i=1}^{k} A_i X B_i, \quad A_i \in \mathcal{R}^{p\times m},\ B_i \in \mathcal{R}^{n\times q} \tag{1.2}$$

pentru un anumit k şi, în consecinţă, (1.1) devine

$$\sum_{i=1}^{k} A_i X B_i = C. \tag{1.3}$$

Evident, ecuaţia (1.3) poate fi scrisă într-o formă "desfăşurată" ca sistem de pq ecuaţii liniare cu mn necunoscute care poate servi ca bază pentru exprimarea condiţiilor de existenţă/unicitate ca şi pentru elaborarea procedurilor de rezolvare într-un sens sau altul. O astfel de abordare eludează "structura matriceală" a sistemului iar aplicarea tehnicilor clasice de rezolvare pe sistemul desfăşurat, neexploatând structura internă a datelor de intrare, este, de cele mai multe ori, ineficientă (vezi exerciţiul 1.9).

Din acest motiv, pentru diferite cazuri particulare ale ecuaţiei matriceale (1.3), condiţiile de existenţă/unicitate ale soluţiilor, într-un sens precizat, se exprimă în raport cu matricele A_i, B_i, $i = 1 : k$, iar metodele de rezolvare fac apel la tehnici speciale.

Ecuaţiile matriceale liniare (1.3) cele mai întâlnite se obţin pentru $k = 1$

$$AXB = C \tag{1.4}$$

a cărei rezolvare se reduce imediat la rezolvarea unor ecuaţii matriceale având forma particulară

$$AX = C \tag{1.5}$$

$$XB = C \tag{1.6}$$

respectiv, pentru $k = 2$,

$$A_1XB_1 + A_2XB_2 = C \tag{1.7}$$

cu particularizările

$$AX + XB = C, \tag{1.8}$$

$$AXB + X = C. \tag{1.9}$$

Ecuaţiile (1.8), (1.9) sunt cunoscute sub denumirea de *ecuaţii matriceale tip Sylvester*. Datorită frecventei utilizări a ecuaţiilor matriceale de forma (1.8) în teoria sistemelor dinamice continue, respectiv a ecuaţiilor de forma (1.9) în domeniul sistemelor dinamice discrete, în continuare ecuaţia (1.8) va fi referită ca *ecuaţie Sylvester continuă*, respectiv (1.9) ca *ecuaţie Sylvester discretă*.

În sfârşit, considerând în (1.8) $B \leftarrow A$, $A \leftarrow A^T$ şi în (1.9) $B \leftarrow A$, $A \leftarrow -A^T$, $C \leftarrow -C$, obţinem ecuaţiile matriceale liniare cunoscute sub denumirile de *ecuaţie Liapunov continuă* pentru

$$A^TX + XA = C \tag{1.10}$$

respectiv *ecuaţie Liapunov discretă* pentru

$$A^TXA - X = C. \tag{1.11}$$

În cele mai multe aplicaţii matricele de intrare A, B, C sunt reale, matricea soluţie X rezultând, la rândul său, reală. Totuşi, algoritmii prezentaţi rămân aplicabili şi în cazul matricelor de intrare complexe. Mai mult, chiar în cazul unor date iniţiale reale, unele din metode de calcul, apelând la determinarea valorilor şi vectorilor proprii, necesită, tranzitoriu, utilizarea unei aritmetici complexe. Atunci când pentru date de intrare reale se poate utiliza exclusiv o aritmetică reală (vezi exerciţiul 1.14) vor fi făcute precizările cuvenite.

În cele ce urmează vom fi interesaţi în calculul soluţiilor ecuaţiilor matriceale liniare cu semnificaţie pentru domeniul teoriei sistemelor în general, şi al sistemelor automate în special, respectiv al sistemelor de forma (1.5), (1.6), (1.8), (1.9) şi (1.10), (1.11). Condiţiile de existenţă şi unicitate ale soluţiilor se exprimă în mod specific pentru fiecare tip de ecuaţie, singurul rezultat comun fiind dat de

Propoziţia 1.1 *Dacă ecuaţia matriceală liniară (1.3) admite o soluţie $X \in \mathcal{R}^{m\times n}$, atunci această soluţie este unică dacă şi numai dacă ecuaţia matriceală omogenă (obţinută din ecuaţia iniţială pentru $C = 0$) admite drept unică soluţie $X = 0$.*

Demonstraţia este propusă ca exerciţiu pentru cititor sau poate fi găsită în [VII]. $\diamond$

În cazul ecuaţiilor matriceale liniare (1.8), (1.9) şi (1.10), (1.11) o analiză simplă arată că matricea necunoscută X şi membrul drept C au aceleaşi dimensiuni $(m \times n)$. În consecinţă, enunţul de mai sus capătă următoarea formă mai precisă.

Propoziţia 1.2 *Ecuaţiile matriceale liniare (1.8), (1.9) şi (1.10), (1.11) sunt global solubile, i.e. admit o soluţie $X \in \mathcal{R}^{m\times n}$ oricare ar fi membrul drept C, dacă şi numai dacă ecuaţiile matriceale omogene corespunzătoare admit drept unică soluţie $X = 0$.*

Altfel spus, are loc următoarea alternativă (a lui Fredholm): fie ecuaţiile considerate sunt global solubile, fie ecuaţiile omogene admit o soluţie netrivială $X \neq 0$.

1.2 Rezolvarea ecuaţiilor matriceale de tip $AX = C$

Ecuaţiile matriceale de tipul $AX = C$, unde $A \in \mathcal{R}^{p\times m}$, $X \in \mathcal{R}^{m\times n}$, $C \in \mathcal{R}^{p\times n}$,reprezintă o colecţie de ecuaţii vectoriale obişnuite, toate având

aceeaşi matrice A a coeficienţilor. Într-adevăr, partiţionând matricele X şi C pe coloane

$$X = [\, x_1 \; x_2 \; \ldots \; x_n \,], \quad C = [\, c_1 \; c_2 \; \ldots \; c_n \,] \tag{1.12}$$

cu $x_j = Xe_j,\ c_j = Ce_j,\ j = 1 : n$, ecuaţia matriceală $AX = C$ devine echivalentă cu setul de sisteme liniare

$$Ax_j = c_j, \quad j = 1 : n \tag{1.13}$$

care se rezolvă cu mijloacele clasice, cu precizarea că transformările necesare asupra matricei A a sistemelor (1.13) (cum ar fi, de exemplu, în cazul $p = m$ şi A inversabilă, triangularizarea prin eliminare gaussiană cu pivotare parţială) se efectuează o singură dată. Având în vedere că rezolvarea sistemelor (1.13) nu ridică probleme deosebite, scrierea algoritmilor face obiectul exerciţiilor 1.1 şi 1.3.

Observaţia 1.1 Ecuaţiile de tipul $XB = C$ se reduc la ecuaţii de tipul $AX = C$ prin transpunere. $\diamond$

Observaţia 1.2 Dacă matricea $A \in \mathcal{R}^{p\times m}$ este *monică* (i.e. are coloanele liniar independente) iar sistemele (1.13) sunt rezolvate în sensul celor mai mici pătrate (CMMP), atunci ansamblul *pseudosoluţiilor*

$$x_j^* = A^+ c_j = (A^T A)^{-1} A^T c_j, \quad j = 1 : n \tag{1.14}$$

respectiv

$$X^* = A^+ C, \qquad A^+ = (A^T A)^{-1} A^T \tag{1.15}$$

are o semnificaţie asemănătoare şi anume aceea că matricea $X^* \in \mathcal{R}^{m\times n}$ minimizează norma Frobenius a reziduului matriceal

$$R = C - AX \tag{1.16}$$

adică

$$\|R^*\|_F = \|C - AX^*\|_F = \min_{X\in\mathcal{R}^{m\times n}} \|C - AX\|_F. \tag{1.17}$$

Într-adevăr,

$$\|R\|_F = \sqrt{\sum_{j=1}^{n}\sum_{i=1}^{p} r_{ij}^2} = \sqrt{\sum_{j=1}^{n} \|c_j - Ax_j\|^2}$$

este minimă dacă şi numai dacă $\|c_j - Ax_j\|$ sunt minime pentru toţi $j = 1 : n$. $\diamond$

Observaţia 1.3 Dacă matricea $A \in \mathcal{R}^{p\times m}$ este *epică* (adică are liniile liniar independente), atunci ansamblul soluţiilor *normale* ale sistemelor (1.13)

$$x_j^* = A^+ c_j = A^T (AA^T)^{-1} c_j, \quad j = 1:n \tag{1.18}$$

respectiv

$$X^* = A^+ C, \qquad A^+ = A^T (AA^T)^{-1} \tag{1.19}$$

reprezintă soluţia de normă Frobenius minimă a sistemului $AX = C$:

$$\|X^*\|_F = \min_{\substack{AX=C \\ X\in\mathcal{R}^{m\times n}}} \|X\|_F. \tag{1.20}$$

Într-adevăr, este evident că X^* dat de (1.19) este o soluţie a ecuaţiei matriceale $AX = C$. În plus, dacă $A = LQ$ este o factorizare a matricei A, unde $Q \in \mathcal{R}^{m\times m}$ este ortogonală iar $L \in \mathcal{R}^{p\times m}$ are structura

$$L = [\, L' \;\; 0 \,] \tag{1.21}$$

cu $L' \in \mathcal{R}^{p\times p}$ inferior triunghiulară nesingulară, atunci ecuaţia matriceală $AX = C$ se scrie

$$LQX = C. \tag{1.22}$$

Cu notaţia

$$QX = Y = \begin{bmatrix} Y' \\ Y'' \end{bmatrix}, \tag{1.23}$$

cu $Y' \in \mathcal{R}^{p\times n}$, $Y'' \in \mathcal{R}^{(m-p)\times n}$, având în vedere structura (1.21) a matricei L, rezultă

$$Y' = (L')^{-1} C.$$

În consecinţă, toate soluţiile sistemului $AX = C$ se scriu sub forma

$$X = Q^T \begin{bmatrix} (L')^{-1} C \\ Y'' \end{bmatrix} \tag{1.24}$$

cu $Y'' \in \mathcal{R}^{(m-p)\times n}$ arbitrar. Dar

$$\|X\|_F^2 = \|Q^T \begin{bmatrix} (L')^{-1} C \\ Y'' \end{bmatrix}\|_F^2 = \| \begin{bmatrix} (L')^{-1} C \\ Y'' \end{bmatrix}\|_F^2 =$$

$$= \|(L')^{-1} C\|_F^2 + \|Y''\|_F^2$$

minimul obţinându-se pentru $Y'' = 0$. Prin urmare soluţia de normă Frobenius minimă este

$$X^* = Q^T \begin{bmatrix} (L')^{-1}C \\ 0 \end{bmatrix} \tag{1.25}$$

verificându-se imediat, pe baza factorizării LQ, că aceasta coincide cu soluţia (1.19). $\diamond$

Observaţia 1.4 În cazul în care matricea A este de rang nemaximal

$$rang\, A < \min(p, m)\ ,$$

notând cu $x_j^*,\ j = 1 : n$, *pseudosoluţiile normale*, în sensul celor mai mici pătrate, ale sistemelor (1.13), matricea $X^* = (x_j)_{j=1:n}$ poate fi interpretată drept pseudosoluţia normală în sensul normei Frobenius a ecuaţiei matriceale $AX = C$. Cu alte cuvinte $X^* \in \mathcal{R}^{m\times n}$ este matricea de normă Frobenius minimă dintre toate matricele $X \in \mathcal{R}^{m\times n}$ care minimizează norma Frobenius a reziduului matriceal $R = C - AX$:

$$\|X^*\|_F = \min_{\substack{X\in\mathcal{R}^{m\times n} \\ \|C-AX\|_F=\text{minim}}} \|X\|_F. \tag{1.26}$$

Justificarea afirmaţiilor de mai sus face obiectul exerciţiului 1.2. $\diamond$

1.3 Rezolvarea ecuaţiilor matriceale Sylvester

I. Vom începe cu prezentarea modalităţilor de rezolvare a *ecuaţiei matriceale Sylvester continue* (1.8)

$$AX + XB = C$$

unde în general $A \in \mathcal{C}^{m\times m}$, $B \in \mathcal{C}^{n\times n}$, $C \in \mathcal{C}^{m\times n}$. Condiţiile de existenţă şi unicitate ale soluţiei acestei ecuaţii sunt date de

Propoziţia 1.3 *Ecuaţia matriceală (1.8) admite o soluţie $X \in \mathcal{C}^{m\times n}$ unică dacă şi numai dacă*

$$\lambda_i + \mu_j \neq 0 \tag{1.27}$$

oricare ar fi $\lambda_i \in \lambda(A)$ şi oricare ar fi $\mu_j \in \lambda(B)$ sau, altfel spus, dacă şi numai dacă

$$\lambda(A) \cap \lambda(-B) = \emptyset. \tag{1.28}$$

Dacă matricele A, B, C sunt reale iar condiţiile (1.27) sunt satisfăcute atunci soluţia X rezultă şi ea reală.

Demonstraţie. Există matricele unitare $U \in \mathcal{C}^{m\times m}$ şi $V \in \mathcal{C}^{n\times n}$ (i.e. $U^HU = UU^H = I_m$, $V^HV = VV^H = I_n$, indicele superior H semnificând dubla operaţie de transpunere şi conjugare) astfel încât

$$U^HAU = S \in \mathcal{C}^{m\times m}, \tag{1.29}$$

$$V^HBV = T \in \mathcal{C}^{n\times n} \tag{1.30}$$

sunt *formele Schur complexe* ale matricelor A respectiv B şi, deci, au o structură superior triunghiulară. Din (1.29), (1.30) avem relaţiile

$$A = USU^H \tag{1.31}$$

$$B = VTV^H \tag{1.32}$$

cu care ecuaţia (1.8) devine

$$USU^HX + XVTV^H = C$$

de unde

$$SU^HXV + U^HXVT = U^HCV. \tag{1.33}$$

Notând cu

$$Y = U^HXV, \quad \tilde{C} = U^HCV \tag{1.34}$$

(1.33) se scrie

$$SY + YT = \tilde{C}. \tag{1.35}$$

Având în vedere natura nesingulară a transformărilor efectuate, ecuaţia matriceală (1.8) admite o soluţie unică $X \in \mathcal{C}^{m\times n}$ dacă şi numai dacă ecuaţia (1.35) admite o soluţie unică $Y \in \mathcal{C}^{m\times n}$. Condiţiile de existenţă şi unicitate ale soluţiei ecuaţiei (1.35) rezultă din aplicarea unei tehnici de rezolvare directă. Într-adevăr, ţinând seama de structura superior triunghiulară a matricelor S şi T rezultă că, scriind pe coloane, (1.35) devine

$$Sy_j + Yt_j = \tilde{c}_j, \quad j = 1:n \tag{1.36}$$

unde $y_j = Ye_j$, $t_j = Te_j$, $\tilde{c}_j = Ce_j$. Dar

$$t_j = [\, t_{1j}\ t_{2j}\ \ldots\ t_{jj}\ 0\ \ldots\ 0\,]^T$$

şi, prin urmare, (1.36) devine

$$Sy_j + \sum_{k=1}^{j} t_{kj}y_k = \tilde{c}_j, \quad j = 1:n \tag{1.37}$$

sau

$$(S + t_{jj} I_m) y_j = \tilde{c}_j - \sum_{k=1}^{j-1} t_{kj} y_k, \quad j = 1 : n. \tag{1.38}$$

Se vede acum limpede că ecuaţiile (1.38) sunt sisteme liniare triunghiulare având membrul drept calculabil dacă ordinea de rezolvare a acestor sisteme este $j = 1, 2, 3, ..., n$. Mai mult, existenţa şi unicitatea soluţiei este condiţionată de nesingularitatea matricelor $(S + t_{jj} I_m),\ j = 1 : n$. Matricea S fiind superior triunghiulară, această condiţie este satisfăcută dacă şi numai dacă

$$s_{ii} + t_{jj} \neq 0, \quad \forall\, i = 1 : m,\ \forall\, j = 1 : n,$$

condiţie identică cu (1.27) întrucât $\lambda(A) = \lambda(S)$ şi $\lambda(B) = \lambda(T)$, observaţie care încheie demonstraţia propoziţiei. $\diamond$

Tehnicile de rezolvare ale ecuaţiei Sylvester continue fac apel la transformări care conduc la ecuaţii matriceale în care, în locul matricelor A şi B, apar matrice cu o structură simplificată (Hessenberg, triunghiulară, diagonală) fapt care permite aplicarea metodelor consacrate de la sistemele clasice. O primă alternativă, numită ad-hoc *varianta Schur-Schur*, are la bază procedura evidenţiată de demonstraţia propoziţiei 1.3. În cazul uzual, în care datele de intrare precum şi soluţia X sunt reale, redactarea algoritmului de calcul este următoarea.

Algoritmul 1.1 (Date matricele $A \in \mathcal{R}^{m\times m}$, $B \in \mathcal{R}^{n\times n}$, $C \in \mathcal{R}^{m\times n}$ cu $\lambda(A) \cap \lambda(-B) = \emptyset$, algoritmul calculează soluţia $X \in \mathcal{R}^{m\times n}$ a ecuaţiei Sylvester continue $AX + XB = C$ prin aducerea matricelor A şi B la forma Schur complexă prin transformări unitare de asemănare. Algoritmul utilizează funcţia **fsc** care calculează forma Schur complexă şi matricea unitară de transformare corespunzătoare pentru o matrice de intrare dată.)

1. $[U, S] = \mathbf{fsc}(A)$
2. $[V, T] = \mathbf{fsc}(B)$
3. $C \leftarrow \tilde{C} = U^H C V$
4. `Pentru` $j = 1 : n$
   1. `Dacă` $j > 1$ `atunci`
      1. $c_j = c_j - \sum_{k=1}^{j-1} t_{kj} y_k$.
   2. Se rezolvă sistemul superior triunghiular $(S + t_{jj} I_m) y_j = c_j$.

5. $X = \mathrm{Re}(UYV^H)$

Comentarii. Referitor la algoritmul de mai sus se cuvin următoarele precizări:

1. Aducerea unei matrice reale la forma Schur complexă prin transformări de asemănare se face în două etape: într-o primă etapă cu algoritmul **QR** se obţine forma Schur reală după care, într-o a doua etapă, utilizând rotaţii complexe sau reflectori complecşi (exerciţiile 1.4 : 1.7) blocurile 2×2 sunt reduse la forma superior triunghiulară complexă.

2. Efectuarea calculelor în format virgulă mobilă cu numere complexe conduce la un rezultat afectat de erori de rotunjire complexe deşi rezultatul teoretic este o matrice reală. De aceea, în ultima instrucţiune, este inclusă eliminarea părţilor imaginare.

3. Evident, efortul de calcul cel mai important se consumă în execuţia instrucţiunilor 1 şi 2 de aducere la forma Schur complexă a matricelor A şi B şi de acumulare a transformărilor. $\diamond$

Din motive de eficienţă se impune analiza alternativelor în care se renunţă la aducerea la forma Schur a ambelor matrice A şi B.

• Astfel, în aşa numita *variantă Hessenberg-Schur* numai matricea B este adusă la forma Schur complexă apărând următoarele diferenţe în raport cu algoritmul de mai sus:

– În instrucţiunea 1 matricea A este adusă printr-un algoritm de calcul direct (neiterativ) – algoritmul **HQ** – la forma superior Hessenberg. În acest fel se evită faza iterativă a algoritmului **QR** şi trecerea de la forma Schur reală la forma Schur complexă.

– În compensaţie, la instrucţiunea 4.2, se rezolvă, în loc de n sisteme triunghiulare, n sisteme de tip Hessenberg incluzând eliminare gaussiană cu eventuală pivotare.

Scrierea explicită a algoritmului face obiectul exerciţiului 1.10.

• În cazul matricelor A şi B *simple* (adică al matricelor pentru care există un set complet de vectori proprii liniar independenţi, în particular, al matricelor cu valori proprii distincte) metoda bazată pe calculul valorilor şi vectorilor proprii poate conduce la o precizie superioară a rezultatelor simultan cu un timp de execuţie rezonabil. Fie

$$\begin{array}{l} \lambda(A) = \{\lambda_1,\, \lambda_2, \ldots, \lambda_m\} \subset \mathcal{C} \\ \lambda(B) = \{\mu_1,\, \mu_2, \ldots, \mu_n\} \subset \mathcal{C} \end{array} \tag{1.39}$$

spectrele de valori proprii ale matricelor A respectiv B şi

$$\begin{array}{l} U = [\, u_1 \; u_2 \; \ldots \; u_m \,], \\ V = [\, v_1 \; v_2 \; \ldots \; v_n \,] \end{array} \tag{1.40}$$

matricele corespunzătoare de vectori proprii (adică $u_j = Ue_j$ este un vector propriu al matricei A asociat valorii proprii $\lambda_j \in \lambda(A)$, etc.). Atunci

$$\begin{array}{ll} A = ULU^{-1}, & L = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m) \\ B = VMV^{-1}, & M = \mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n) \end{array} \tag{1.41}$$

şi ecuaţia Sylvester (1.8) se scrie sub forma

$$ULU^{-1}X + XVMV^{-1} = C \tag{1.42}$$

echivalentă cu

$$LU^{-1}XV + U^{-1}XVM = U^{-1}CV. \tag{1.43}$$

Notând

$$\begin{array}{l} U^{-1}XV = Y \\ U^{-1}CV = Z \end{array} \tag{1.44}$$

din (1.43) rezultă ecuaţia

$$LY + YM = Z \tag{1.45}$$

cu necunoscuta matriceală Y ale cărei elemente, în condiţiile propoziţiei 1.1, se calculează imediat cu relaţiile

$$y_{ij} = z_{ij}/(\lambda_i + \mu_j)\,, \quad i = 1:m,\ j = 1:n. \tag{1.46}$$

Matricea necunoscută X din ecuaţia iniţială rezultă din (1.44)

$$X = UYV^{-1}. \tag{1.47}$$

Presupunând că dispunem de o funcţie numită **vvp** care, pentru o matrice de intrare simplă T, furnizează o matrice V ale cărui coloane sunt vectorii proprii ai matricei T şi un vector p al valorilor proprii corespunzători cu sintaxa

$$[\, V, p \,] = \mathbf{vvp}(T) \tag{1.48}$$

putem prezenta următorul algoritm.

Algoritmul 1.2 (Date matricele simple $A \in \mathcal{R}^{m\times m}$, $B \in \mathcal{R}^{n\times n}$ cu $\lambda(A) \cap \lambda(-B) = \emptyset$ şi matricea $C \in \mathcal{R}^{m\times n}$, algoritmul calculează soluţia $X \in \mathcal{R}^{m\times n}$ a ecuaţiei matriceale $AX + XB = C$ pe baza calculului valorilor şi vectorilor proprii cu ajutorul funcţiei **vvp**.)

1. $[\,U,\,l\,] = \mathbf{vvp}(A)$
2. $[\,V,\,m\,] = \mathbf{vvp}(B)$
3. Se rezolvă sistemul matriceal $UZ = CV$ cu necunoscuta Z.
4. `Pentru` $i = 1 : m$
    1. `Pentru` $j = 1 : n$
        1. $y_{ij} = z_{ij}/(l_i + m_j)$.
5. Se rezolvă sistemul matriceal $XV = UY$ în raport cu X.
6. $X = \mathrm{Re}(X)$.

Rezolvarea sistemelor matriceale de la instrucţiunile 3 şi 5 ale algoritmului se realizează conform recomandărilor secţiunea 1.2.

II. Modalităţile de calcul ale soluţiei *ecuaţiei matriceale Sylvester discrete* (1.9)

$$AXB + X = C$$

unde $A \in \mathcal{C}^{m\times m}$, $B \in \mathcal{C}^{n\times n}$, $C \in \mathcal{C}^{m\times n}$ sunt asemănătoare. Condiţiile de existenţă şi unicitate ale soluţiei sunt date de

Propoziţia 1.4 *Ecuaţia matriceală (1.9) are o soluţie $X \in \mathcal{C}^{m\times n}$ unică dacă şi numai dacă*

$$\lambda_i\mu_j \neq -1 \tag{1.49}$$

oricare ar fi $\lambda_i \in \lambda(A)$ şi oricare ar fi $\mu_j \in \lambda(B)$. Dacă matricele A, B, C sunt reale iar condiţiile (1.49) sunt satisfăcute atunci soluţia X rezultă şi ea reală.

Demonstraţie. Fie $U \in \mathcal{C}^{m\times m}$ şi $V \in \mathcal{C}^{n\times n}$ matrice unitare care definesc reducerea matricelor A şi B la *formele Schur complexe* S, respectiv T, ca în (1.29) şi (1.30). Cu (1.31) şi (1.32), ecuaţia (1.9) devine

$$USU^HXVTV^H + X = C \tag{1.50}$$

care, la rândul ei, este echivalentă cu

$$SYT + Y = \tilde{C} \tag{1.51}$$

unde Y şi $\tilde{C}$ sunt date de (1.34). Ecuaţia (1.9) are o soluţie unică X dacă şi numai dacă ecuaţia (1.51) are o soluţie unică Y. Dacă $y_j = Ye_j$, $t_j =$

$Te_j,\ \tilde{c}_j = \tilde{C}e_j,\ j = 1 : n$ atunci, ţinând seama de structura superior triunghiulară a matricelor T şi S, din (1.51) obţinem

$$\begin{aligned} &(St_{11} + I_m)y_1 = \tilde{c}_1, \\ &(St_{jj} + I_m)y_j = \tilde{c}_j - S\sum_{k=1}^{j-1} t_{kj}y_k, \quad j = 2 : n. \end{aligned} \tag{1.52}$$

Ecuaţiile (1.52) sunt sisteme liniare superior triunghiulare care admit soluţii unice dacă şi numai dacă matricele $St_{jj} + I_m,\ j = 1 : n$ sunt nesingulare, respectiv

$$s_{ii}t_{jj} + 1 \neq 0, \quad \forall\, i \in 1 : m,\ \ \forall\, j \in 1 : n. \tag{1.53}$$

Cum însă $\{s_{ii} \,|\, i = 1 : m\} = \lambda(A)$ şi $\{t_{jj} \,|\, j = 1 : n\} = \lambda(B)$ şi, în plus, rezolvarea sistemelor (1.52) în condiţiile (1.53) este posibilă în ordinea $j = 1, 2, ..., n$ în care termenul liber este calculabil, rezultă că propoziţia este demonstrată. $\diamond$

Demonstraţia de mai sus conduce imediat la algoritmul de rezolvare a ecuaţiei matriceale Sylvester discrete în *varianta Schur-Schur*.

Algoritmul 1.3 (Fiind date matricele $A \in \mathcal{R}^{m\times m},\ B \in \mathcal{R}^{n\times n}$ astfel încât $\lambda_i\mu_j \neq -1$ pentru toţi $\lambda_i \in \lambda(A)$ şi $\mu_j \in \lambda(B)$ şi matricea $C \in \mathcal{R}^{m\times n}$ algoritmul calculează soluţia X a ecuaţiei Sylvester discrete $AXB + X = C$ prin reducerea matricelor A şi B la forma Schur complexă. Algoritmul utilizează funcţia **fsc** pentru calculul formei Schur complexe şi a matricei de transformare corespunzătoare.)

1. $[\,U,\, S\,] = \mathbf{fsc}(A)$
2. $[\,V,\, T\,] = \mathbf{fsc}(B)$
3. $C \leftarrow \tilde{C} = U^H C V$
4. `Pentru` $j = 1 : n$
   1. `Dacă` $j > 1$ `atunci`
      1. $c_j \leftarrow c_j - S\,(\,\sum_{k=1}^{j-1} t_{kj}y_k\,)$
   2. Se rezolvă sistemul superior triunghiular $(St_{jj} + I_m)y_j = c_j$.
5. $X = \mathrm{Re}(UYV^H)$.

Având în vedere similitudinea algoritmilor 1.1 şi 1.2, similitudine care se extinde la celelalte variante, lăsăm în sarcina cititorului elaborarea *variantei Hessenberg–Schur* ca şi justificarea faptului că următorul algoritm, cu notaţiile din algoritmul 1.2, rezolvă problema în cazul matricelor de intrare A şi B *simple* (diagonalizabile).

Algoritmul 1.4 (Date matricele simple $A \in \mathcal{R}^{m\times m}$, $B \in \mathcal{R}^{n\times n}$ cu $\lambda_i\mu_j \neq -1$ pentru toţi $\lambda_i \in \lambda(A), \mu_j \in \lambda(B)$ şi matricea $C \in \mathcal{R}^{m\times n}$, algoritmul calculează soluţia X a ecuaţiei $AXB + X = C$ pe baza determinării valorilor şi vectorilor proprii ale matricelor A şi B cu ajutorul funcţiei **vvp**.)

1. $[U, l\,] = \mathbf{vvp}(A)$
2. $[V, m\,] = \mathbf{vvp}(B)$
3. Se rezolvă sistemul matriceal $UZ = CV$ în raport cu Z.
4. `Pentru` $i = 1 : m$
    1. `Pentru` $j = 1 : n$
        1. $y_{ij} = z_{ij}/(l_i m_j + 1)$
5. Se rezolvă sistemul matriceal $XV = UY$ în raport cu X.
6. $X = \mathrm{Re}X$.

1.4 Rezolvarea ecuaţiilor matriceale Liapunov

Ecuaţiile matriceale Liapunov continuă $A^TX+XA = C$ şi discretă $A^TXA-X = C$ sunt cazuri particulare ale ecuaţiilor Sylvester corespunzătoare şi, prin urmare, rezultatele paragrafului precedent sunt aplicabile în întregime. Concret, având în vedere că $\lambda(A^T) = \lambda(A)$, obţinem următoarele consecinţe ale propoziţiilor 1.3 şi 1.4.

Corolar 1.1 *Ecuaţia matriceală Liapunov continuă (1.10) admite o soluţie unică dacă şi numai dacă matricea A nu are nici o pereche de valori proprii opuse, i.e.*

$$\lambda_i + \lambda_j \neq 0, \quad \forall\, \lambda_i, \lambda_j \in \lambda(A), \tag{1.54}$$

În particular, $0 \notin \lambda(A)$ i.e. matricea A este nesingulară.

Corolar 1.2 *Ecuaţia matriceală Liapunov discretă (1.11) admite o soluţie unică dacă şi numai dacă matricea A nu are nici o pereche de valori proprii inverse, i.e.*

$$\lambda_i\lambda_j \neq 1, \quad \forall\, \lambda_i, \lambda_j \in \lambda(A), \tag{1.55}$$

În particular, $\pm 1 \notin \lambda(A)$.

Algoritmii 1.1, 1.2, 1.3 şi 1.4 se simplifică în sensul că este necesară o singură reducere la forma Schur, respectiv un singur calcul de valori şi vectori proprii.

I. Într-adevăr, să considerăm mai întâi cazul *ecuaţiei Liapunov continue* (1.10) şi fie reducerea (1.31) a matricei $A \in \mathcal{R}^{n\times n}$ la *forma Schur complexă*. Rezultă

$$A^T = A^H = US^HU^H. \tag{1.56}$$

Cu (1.31), (1.56) ecuaţia Liapunov (1.10) devine

$$US^HU^HX + XUSU^H = C \tag{1.57}$$

respectiv

$$S^HY + YS = \tilde{C} \tag{1.58}$$

unde

$$Y = U^HXU, \quad \tilde{C} = U^HCU. \tag{1.59}$$

Întrucât S^H are o structură inferior triunghiulară, scriind ecuaţia matriceală (1.58) pe coloane obţinem sistemele inferior triunghiulare

$$(S^H + s_{jj}I_n)y_j = \tilde{c}_j - \sum_{k=1}^{j-1} s_{kj}y_k, \quad j = 1:n. \tag{1.60}$$

În condiţiile corolarului 1.1, aceste sisteme sunt nesingulare şi pot fi rezolvate în ordinea $j = 1, 2 ..., n$. Cu observaţia că soluţia X a ecuaţiei iniţiale se calculează apoi din prima relaţie (1.59) rezultă următorul algoritm.

Algoritmul 1.5 (Date matricele $A \in \mathcal{R}^{n\times n}$, satisfăcând (1.54), şi $C \in \mathcal{R}^{n\times n}$ oarecare, algoritmul calculează soluţia X a ecuaţiei Liapunov $A^TX + XA = C$ pe baza reducerii matricei A la forma Schur complexă cu ajutorul funcţiei **fsc**.)

1. $[U, S] = \mathbf{fsc}(A)$
2. $C \leftarrow U^HCU$
3. `Pentru` $j = 1:n$
   1. `Dacă` $j > 1$ `atunci`
      1. $c_j = c_j - \sum_{k=1}^{j-1} s_{kj}y_k$
   2. Se rezolvă sistemul inferior triunghiular $(S^H + s_{jj}I_n)y_j = c_j$.
4. $X = \mathrm{Re}(UYU^H)$.

În multe aplicaţii matricea C din (1.10) este *simetrică*. Într-un astfel de caz apar simplificări suplimentare datorită faptului că, în condiţiile de existenţă şi unicitate menţionate, şi matricea soluţie este *simetrică*. Într-adevăr, dacă X este soluţia, se verifică imediat că X^T satisface ecuaţia

matriceală (1.10) şi, cum soluţia este unică, rezultă $X = X^T$. Acest fapt poate fi exploatat observând că, datorită simetriei matricelor C şi X, matricele Y şi $\tilde{C}$ din (1.58) sunt hermitice şi în consecinţă este suficient să se calculeze, de exemplu, numai triunghiul superior al matricelor $\tilde{C}, Y$ şi, în final, X (exerciţiul 1.11).

II. Algoritmul de rezolvare a *ecuaţiei Liapunov discrete* (1.11), bazat pe reducerea matricei A la *forma Schur complexă*, urmăreşte îndeaproape schema de mai sus.

Algoritmul 1.6 (Date matricele $A \in \mathcal{R}^{n\times n}$, satisfăcând (1.55), şi $C \in \mathcal{R}^{n\times n}$ oarecare, algoritmul calculează soluţia X a ecuaţiei Liapunov $A^T X A - X = C$ pe baza reducerii matricei A la forma Schur complexă cu ajutorul funcţiei **fsc**.)

1. $[\,U,\, S\,] = \mathbf{fsc}(A)$
2. $C \leftarrow U^H C U$
3. `Pentru` $j = 1 : n$
   1. `Dacă` $j > 1$ `atunci`
      1. $c_j = c_j - S^H(\sum_{k=1}^{j-1} s_{kj} y_k)$
   2. Se rezolvă sistemul inferior triunghiular $(S^H s_{jj} - I_n) y_j = c_j$.
4. $X = \mathrm{Re}(UYU^H)$.

Tratarea cazului în care matricele C şi, drept consecinţă, X sunt simetrice face obiectul exerciţiului 1.11.

Variantele bazate pe calculul valorilor şi vectorilor proprii ale matricei A sunt, de asemenea, propuse cititorului (exerciţiul 1.12).

Cazul important, în care matricea A este *stabilă* (vezi cap. 5) iar matricea C este *simetrică şi negativ (semi)definită* este abordat în capitolul 7 (se pot consulta [X] şi [6], [7]).

Încheiem prezentarea metodelor de rezolvare a ecuaţiilor matriceale liniare prin evidenţierea posibilităţii de a reduce rezolvarea unei ecuaţii Liapunov continue la rezolvarea unei ecuaţii corespondente discrete şi reciproc. Pentru început să observăm că dacă ecuaţia Liapunov discretă (1.11) admite o soluţie unică atunci, conform (1.55), $\pm 1 \notin \lambda(A)$ respectiv matricea $A - I_n$ este nesingulară şi, deci, putem defini transformarea (omografică)

$$B = (A - I_n)^{-1}(A + I_n). \tag{1.61}$$

Apoi, din (1.61) rezultă că $1 \notin \lambda(B)$, i.e $B - I_n$ este nesingulară (exerciţiul 1.13), deci

$$A = (B + I_n)(B - I_n)^{-1}, \tag{1.62}$$

respectiv,

$$A^T = (B^T - I_n)^{-1}(B^T + I_n) \tag{1.63}$$

expresii care, introduse în ecuaţia Liapunov discretă (1.11), conduc, după câteva prelucrări elementare care conservă soluţia, la

$$B^T X + XB = D \tag{1.64}$$

unde

$$D = \frac{1}{2}(B^T - I_n)C(B - I_n). \tag{1.65}$$

În consecinţă, ecuaţia Liapunov continuă definită de (1.64), (1.61) şi (1.65) are aceeaşi soluţie cu ecuaţia Liapunov discretă (1.11) şi poate servi ca suport pentru o modalitate de rezolvare a acesteia din urmă. Reciproc, dată ecuaţia Liapunov continuă (1.64), respectiv date matricele $n \times n$ B şi D cu valorile proprii ale matricei B satisfăcând condiţia $\mu_i + \mu_j \neq 0$ oricare ar fi $\mu_i, \mu_j \in \lambda(B)$ (în particular $0 \notin \lambda(B)$) putem găsi soluţia X a acestei ecuaţii rezolvând ecuaţia Liapunov discretă (1.11) cu A dat de (1.62) şi

$$C = 2(B^T - I_n)^{-1} D (B - I_n)^{-1} \tag{1.66}$$

obţinut din (1.65).

Proprietăţi numerice

Algoritmii prezentaţi utilizează în exclusivitate transformări ortogonale (unitare) ceea ce le conferă remarcabile proprietăţi numerice. Toţi algoritmii includ cel puţin o execuţie a algoritmului **QR** de aducere iterativă a uneia din matricele de intrare la forma Schur reală sau complexă urmată de rezolvarea unor sisteme triunghiulare sau de tip Hessenberg. Algoritmul **QR** este admis a fi un algoritm numeric stabil ca şi algoritmii de rezolvare prin substituţie a sistemelor triunghiulare. Mai mult, se poate arăta că algoritmii din prezentul capitol sunt numeric stabili în întregul lor, i.e. soluţia calculată reprezintă soluţia exactă a problemei respective cu datele de intrare perturbate nesemnificativ. În consecinţă, preciziile efective ce se obţin depind de condiţionările numerice ale matricelor iniţiale şi de nivelul practicat al toleranţelor.

Programe MATLAB disponibile

Pentru rezolvarea ecuaţiilor matriceale Sylvester şi Liapunov continue este disponibilă funcţia **lyap**, care implementează algoritmii 1.1 şi 1.5 (se apelează funcţiile **schur** pentru aducerea la forma Schur reală şi **rsf2csf** pentru obţinerea formei Schur complexe). În cazul discret se poate utiliza funcţia **dlyap**, care impementează metoda transformării omografice (1.61).

Exerciţii

E 1.1 Fie date matricele $A \in \mathcal{R}^{m\times m}$ nesingulară şi $C \in \mathcal{C}^{m\times n}$. Scrieţi algoritmii de rezolvare a sistemului matriceal $AX = C$ folosind eliminarea gaussiană cu pivotare parţială şi completă. Evaluaţi numărul asimptotic de operaţii aritmetice. Cum procedaţi dacă matricea A este simetrică, eventual pozitiv definită ?

E 1.2 Se consideră date matricele $A \in \mathcal{R}^{p\times m}$, $C \in \mathcal{R}^{p\times n}$ şi fie

$$A = U_1\Sigma_1 {V_1}^T, \quad \Sigma_1 = diag(\sigma_1, \sigma_2, \ldots, \sigma_r), \;\; \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$$

descompunerea valorilor singulare a lui A. Arătaţi că $X^* = V_1\Sigma_1^{-1}U_1^T C$ este matricea de normă Frobenius minimă dintre toate matricele $X \in \mathcal{R}^{m\times n}$ care minimizează norma Frobenius a reziduului matriceal $R = C - AX$.

E 1.3 Admiţând că dispuneţi de o funcţie care calculează valorile singulare şi matricele de transformare corespunzătoare pentru o matrice de intrare $A \in \mathcal{R}^{p\times m}$ dată, scrieţi algoritmul de rezolvare, în sensul celor mai mici pătrate, a ecuaţiei matriceale $AX = C$.

E 1.4 Dată o matrice $A \in \mathcal{R}^{2\times 2}$ având $\lambda(A) \subset \mathcal{C}\backslash\mathcal{R}$, determinaţi o matrice de rotaţie bidimensională complexă $P \in \mathcal{C}^{2\times 2}$, definită de

$$P = \begin{bmatrix} c & s \\ -\overline{s} & c \end{bmatrix}, \quad c^2 + |s|^2 = 1, \; c \in \mathcal{R}$$

astfel încât matricea $S = P^H AP \in \mathcal{C}^{2\times 2}$, $P^H \stackrel{\text{def}}{=} \overline{P}^T$ să fie superior triunghiulară.

E 1.5 Se dă o matrice $A \in \mathcal{R}^{n\times n}$ în formă Schur reală. Se cere un algoritm de calcul al unei matrice unitare [1] $Q \in \mathcal{C}^{n\times n}$ şi al matricei supe-

[1] O matrice $P \in \mathcal{C}^{n\times n}$ se numeşte *unitară* dacă $P^H P = PP^H = I_n$ şi joacă, în calculul matriceal complex, acelaşi rol cu cel jucat de matricele ortogonale în calculul matriceal real.

rior triunghiulare $S \in \mathcal{C}^{n\times n}$ astfel încât $Q^H AQ = S$. (Matricea S astfel obţinută se numeşte *formă Schur complexă* a matricei A).

Indicaţie. Matricea Q poate fi un produs de matrice unitare de forma

$$Q_i = \begin{bmatrix} I & 0 & 0 \\ 0 & P_i & 0 \\ 0 & 0 & I \end{bmatrix} \in \mathcal{C}^{n\times n}$$

cu $P_i \in \mathcal{C}^{2\times 2}$ o rotaţie complexă bidimensională.

E 1.6 Fie $A \in \mathcal{C}^{2\times 2}$ superior triunghiulară. Să se calculeze o matrice unitară $U \in \mathcal{C}^{2\times 2}$ astfel încât

$$U^H AU = \begin{bmatrix} a_{22} & * \\ 0 & a_{11} \end{bmatrix},$$

i.e. să se obţină permutarea elementelor diagonale.

E 1.7 Dacă $B \in \mathcal{C}^{n\times n}$ este o matrice superior triunghiulară dată, se cere un algoritm de calcul al unei matrice unitare $V \in \mathcal{C}^{n\times n}$ şi a matricei superior triunghiulare $S \in \mathcal{C}^{n\times n}$ astfel încât $V^H BV = S$ şi $|s_{11}| \leq |s_{22}| \leq \ldots \leq |s_{nn}|$.

Observaţie. Dacă B este forma Schur complexă a unei matrice $A \in \mathcal{R}^{n\times n}$, atunci matricea S de mai sus poartă numele de *formă Schur complexă ordonată* a matricei A. Evident, se poate utiliza orice alt criteriu de ordonare a elementelor diagonale.

Indicaţie. Se foloseşte o secvenţă de transformări de forma

$$V_i = \begin{bmatrix} I & 0 & 0 \\ 0 & U_i & 0 \\ 0 & 0 & I \end{bmatrix}$$

unde U_i sunt matrice 2×2 determinate în exerciţiul 1.6.

E 1.8 Admiţând că numărul asimptotic de operaţii aritmetice în format virgulă mobilă necesar pentru execuţia algoritmului **QR** de aducere a unei matrice reale $n \times n$ la forma Schur reală, (inclusiv calculul matricei de transformare) este, în medie statistică, conform [VI], $N_1 = 12n^3$, evaluaţi

a) numărul asimptotic de operaţii impus de execuţia algoritmului de la exerciţiul 1.7;

b) numărul asimptotic de operaţii necesar pentru rezolvarea, cu ajutorul algoritmului 1.2 (varianta Schur–Schur), a ecuaţiei Sylvester continue (1.8);

c) numărul asimptotic de operaţii necesar pentru rezolvarea ecuaţiei Liapunov continue (1.10) cu ajutorul algoritmului 1.5, respectiv pentru rezolvarea ecuaţiei Liapunov discrete (1.11) cu ajutorul algoritmului 1.6.

E 1.9 Admiţând că sunt îndeplinite condiţiile de existenţă şi unicitate a soluţiei, scrieţi un algoritm eficient de rezolvare a ecuaţiei matriciale Sylvester continue (1.8) prin interpretarea "desfăşurată" a acesteia ca un sistem de mn ecuaţii cu mn necunoscute. Algoritmul nu trebuie să includă nici o procedură iterativă. Idem pentru ecuaţia Sylvester discretă (1.9).

Evaluaţi numărul de operaţii aritmetice necesar şi comparaţi cu cel determinat la exerciţiul 1.8. Ce concluzii se impun ?

Indicaţie. Dacă $\bar{x} \in \mathcal{R}^{mn}$ şi $\bar{c} \in \mathcal{R}^{mn}$ sunt vectorii definiţi, de exemplu, prin concatenarea coloanelor matricelor X respectiv C arătaţi că ecuaţia (1.8) se poate scrie în forma $(I_n \otimes A + B^T \otimes I_m)\bar{x} = \bar{c}$ iar (1.9) în forma $(B^T \otimes A + I_{mn})\bar{x} = \bar{c}$. În relaţiile de mai sus $\otimes$ semnifică produsul Kronecker definit în felul următor: dacă $U \in \mathcal{R}^{p\times q}$, $V \in \mathcal{R}^{r\times s}$ atunci $W \stackrel{\text{def}}{=} U \otimes V \in \mathcal{R}^{pr\times qs}$ este matricea având structura bloc $W = [\, W_{ij} \,]_{i=1:p,\, j=1:q}$ cu $W_{ij} = u_{ij}V$.

E 1.10 Fiind date matricele $A \in \mathcal{R}^{m\times m}, B \in \mathcal{R}^{n\times n}$ cu $\lambda(A) \cap \lambda(-B) = \emptyset$ şi $C \in \mathcal{R}^{m\times n}$, să se scrie algoritmii Hessenberg – Schur de rezolvare a ecuaţiei matriceale Sylvester continue $AX + XB = C$, respectiv a celei discrete $AXB + X = C$, bazate pe aducerea, prin transformări ortogonale, a matricei A la forma superior Hessenberg şi a matricei B la forma Schur complexă. Presupuneţi că se dispune de funcţia **fsc** de calcul a formei Schur complexe, toate celelalte prelucrări urmând a fi scrise explicit.

E 1.11 Scrieţi corespondenţii algoritmilor 1.5 şi 1.6 (varianta Schur) pentru rezolvarea ecuaţiilor matriceale Liapunov $A^TX + XA = C$ şi $A^TXA - X = C$ în ipoteza că matricea C este simetrică, minimizând numărul de operaţii aritmetice şi memoria ocupată.

E 1.12 Elaboraţi algoritmii de rezolvare a ecuaţiilor Liapunov continuă şi discretă (1.10) şi (1.11) având matricele A simple, folosind o procedură de calcul a valorilor şi vectorilor proprii. Exploataţi cât mai eficient ipoteza simetriei matricei termenilor liberi. Admiţând că efortul de calcul al valorilor şi vectorilor proprii este $N_1 = kn^3$, determinaţi efortul total solicitat de rezolvarea ecuaţiilor matriceale date.

E 1.13 Fie o matrice $A \in R^{n\times n}$ cu proprietatea $\lambda_i\lambda_j \neq 1,\ \forall\, \lambda_i, \lambda_j \in \lambda(A)$. Se cere să se arate:

a) Dacă $B = (A - I_n)^{-1}(A + I_n)$ atunci pentru orice $\lambda_i \in \lambda(A)$ avem $\mu_i = (\lambda_i + 1)/(\lambda_i - 1) \in \lambda(B)$;

b) $\{0, 1\} \cap \lambda(B) = \emptyset$;

c) $\lambda(B) \cap \lambda(-B) = \emptyset$.

E 1.14 Elaboraţi algoritmii de tip Schur-Schur pentru rezolvarea ecuaţiilor Sylvester $AX + XB = C$ şi $AXB + X = C$ care să utilizeze formele Schur reale ale matricelor A şi B şi exclusiv o aritmetică reală [2].

Indicaţie. Utilizând reducerea matricelor A şi B la forma Schur reală ecuaţiile (1.35), respectiv (1.51), au matricele S şi T cvasisuperior triunghiulare. Partiţionând matricea Y în blocuri conform dimensiunilor blocurilor diagonale ale matricelor S şi T, scriind (1.35), respectiv (1.51), pe bloc-coloane şi utilizând o tehnică de bloc-substituţie înapoi se obţine că ecuaţiile (1.35), respectiv (1.51), se reduc la un set de ecuaţii Sylvester continue, respectiv discrete, definite de blocurile diagonale ale matricelor S şi T. Aceste ecuaţii se pot rezolva, în ordinea sugerată mai sus, într-o aritmetică reală, prin desfăşurarea lor în sisteme uzuale de ordin cel mult patru (vezi exerciţiul 1.9).

E 1.15 Elaboraţi algoritmii de tip Schur pentru rezolvarea ecuaţiilor Liapunov $A^T X + XA = C$ şi $A^T XA - X = C$ care să utilizeze exclusiv forma Schur reală a matricei A şi o aritmetică reală [3].

Indicaţie. Adaptaţi indicaţiile de la exerciţiul precedent. Pentru detalii puteţi consulta [4-6] şi [X].

Bibliografie

[1] **Davison E.J., Man F.T.** *The Numerical Solution of $A^TQ + QA = -C$*, IEEE Trans. Automat. Contr., vol. AC-13, pp. 448–449, 1968.

[2] **Berger C.S.** *A Numerical Solution of the Matrix Equation $P = \Phi P \Phi^T + S$*, IEEE Trans. Automat. Contr., vol.AC-16, pp. 381-382, 1971.

[3] **Bartell R.H., Stewart G.W.** *Solution of the Matrix Equation $AX + XB = C$*, Commun. Ass. Comput. Mach., vol 15, pp. 821-826, 1972.

[4] **Barraud A.Y.** *A Numerical Algorithm to Solve $A^TXA - X = Q$*, IEEE Trans. Automat. Contr., vol.AC-22, pp. 883–885, 1977.

[2] Varianta Schur-Schur reală pentru rezolvarea ecuaţiei Sylvester continue este referită în literatura de specialitate cu denumirea de *algoritmul Bartels-Stewart* [3].

[3] Varianta Schur reală pentru rezolvarea ecuaţiei Liapunov discrete este referită în literatura de specialitate cu denumirea de *algoritmul Kitagawa-Barraud* [4, 5].

[5] **Kitagawa G.** *An Algorithm for Solving the Matrix Equation* $X = F^T X F + S$, Int. J. Control, vol. 25, pp. 745–753, 1977.

[6] **Sima V.** *Comparison of some Algorithms for Solving the Lyapunov-type Equations*, Rev. Roum. Sci.Techn. – Electrotechn. Energ., vol. 25, pp. 625–632, 1980.

[7] **Hammarling S.J.** *Numerical Solution of the Stable, Non-negative Definite Lyapunov Equation*, IMA J. Numer. Anal., vol. 2, pp. 303–323, 1982.

Capitolul 2

Calculul funcţiilor de matrice
Exponenţiala matriceală

Calculul funcţiilor de matrice este necesar în multe aplicaţii. În particular, analiza comportării dinamice a sistemelor liniare continue face apel la exponenţiala matriceală, pentru calculul căreia a fost elaborată o întreagă serie de tehnici numerice mai mult sau mai puţin performante. În consecinţă, prezentăm în continuare unele metode generale de calcul pentru funcţiile de matrice, precum şi metodele specifice, cele mai apreciate, de calcul pentru funcţia exponenţială de argument matriceal.

2.1 Funcţii de matrice

Considerăm o funcţie $f : D \subset \mathcal{C} \to \mathcal{C}$ definită pe o mulţime D din planul complex 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

$$F = f(A). \tag{2.1}$$

Pentru început observăm că dacă f este o funcţie polinomială

$$f(z) = \sum_{i=0}^{N} c_i z^i, \quad c_i \in \mathcal{C},\ i = 0 : N \tag{2.2}$$

atunci matricea

$$F = \sum_{i=0}^{N} c_i A^i \tag{2.3}$$

este bine definită şi poate fi numită *valoarea polinomului f în punctul* (sau *pe matricea*) A.

Fie acum $\mu_A(z)$ polinomul minimal al matricei A [1] şi $\lambda_i \in \mathcal{C},\ i = 1 : l$, zerourile acestuia, zeroul λ_i având ordinul de multiplicitate m_i. Avem

$$\mu_A(z) = z^m + \sum_{k=0}^{m-1} \alpha_i z^i = \prod_{i=1}^{l} (z - \lambda_i)^{m_i} \tag{2.4}$$

unde

$$m \stackrel{\text{def}}{=} \text{grad}(\mu_A) = \sum_{i=1}^{l} m_i. \tag{2.5}$$

Conform teoremei împărţirii cu rest, există unic două polinoame q, r cu $\text{grad}(r) < m$ astfel încât

$$f = \mu_A\, q + r. \tag{2.6}$$

Polinomul minimal μ_A fiind un polinom anulator pentru matricea A, i.e. $\mu_A(A) = 0$, din (2.6) rezultă

$$F \stackrel{\text{def}}{=} f(A) = r(A). \tag{2.7}$$

În consecinţă, valoarea polinomului f în punctul A este aceeaşi cu cea a polinomului r şi acelaşi lucru se poate spune despre orice alt polinom al cărui rest la împărţirea prin μ_A este r. Polinomul r poate fi determinat prin aplicarea algoritmului de împărţire cu rest a polinoamelor.

O altă modalitate de calcul a polinomului r se bazează pe faptul că din (2.4) rezultă

$$\mu_A^{(k)}(\lambda_i) = 0, \quad k = 0 : m_i - 1, \tag{2.8}$$

şi, în consecinţă, cei (cel mult) m coeficienţi ai polinomului r se pot determina, conform (2.6), prin rezolvarea sistemului de m ecuaţii liniare

$$r^{(k)}(\lambda_i) = f^{(k)}(\lambda_i), \quad i = 1 : l, \ \ k = 0 : m_i - 1, \tag{2.9}$$

unde indicele superior indică ordinul derivatei.

[1] Amintim că μ_A este polinomul monic de grad minim cu proprietatea $\mu_A(A) = 0$. Dacă $p(z) = \det(zI - A)$ este polinomul caracteristic al matricei A iar $d(z)$ este cel mai mare divizor comun (monic) al tuturor minorilor de ordinul $n-1$ ai matricei caracteristice $zI - A$ atunci $\mu_A(z) = p(z)/d(z)$.

Relaţiile (2.7) şi (2.9) servesc ca bază pentru definirea valorii în punctul A a oricărei funcţii f care admite derivatele cerute în (2.9). Pentru o abordare formală introducem

Definiţia 2.1 *Fie* $A \in \mathcal{C}^{n\times n}$ *şi*

$$\Lambda = \{(\lambda_i, m_i) \mid i = 1 : l,\ \lambda_i \in \mathcal{C},\ m_i \in \mathcal{N}\} \tag{2.10}$$

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*

$$f(\Lambda) = \left\{ f^{(k)}(\lambda_i) \mid i = 1 : l,\ k = 0 : m_i - 1 \right\}. \tag{2.11}$$

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

În condiţiile definiţiei 2.1 introducem

Definiţia 2.2 *Fie date o matrice* $A \subset \mathcal{C}^{n\times n}$ *şi o funcţie* f *definită pe spectrul lui* A*. Dacă polinomul minimal* μ_A *al matricei* A *are gradul* m *atunci polinomul* r *de grad cel mult* $m-1$*, unic determinat de sistemul liniar (2.9), se numeşte* polinomul de interpolare Lagrange-Sylvester al funcţiei f pe spectrul matricei A.

Putem acum să introducem noţiunea de funcţie de matrice extinzând relaţia (2.7) de la polinoame la orice funcţie definită pe spectrul matricei argument.

Definiţia 2.3 *Dacă* f *este o funcţie definită pe spectrul unei matrice* A *atunci* valoarea funcţiei f în punctul A *este*

$$f(A) = r(A), \tag{2.12}$$

unde r *este polinomul de interpolare Lagrange-Sylvester al funcţiei* f *pe spectrul lui* A*.*

În scopul evidenţierii cazurilor când o funcţie de matrice este reală, în continuare vom utiliza următoarea terminologie

Definiţia 2.4 *O funcţie* $f : D \subset \mathcal{C} \to \mathcal{C}$ *este* reală pe spectrul matricei $A \in \mathcal{R}^{n\times n}$ *dacă*

$$f(\overline{\lambda}_i) = \overline{f(\lambda_i)},\ \ \forall\, \lambda_i\ ,\ \overline{\lambda}_i \in \lambda(A),$$

unde $\overline{z}$ este notaţia pentru conjugatul numărului complex z.

Î̂n particular $f(\lambda_i) \in \mathcal{R}$ pentru orice $\lambda_i \in \lambda(A) \cap \mathcal{R}$.

În acest context avem

Propoziţia 2.1 *Dacă funcţia f este reală pe spectrul matricei reale $A \in \mathcal{R}^{n\times n}$ atunci polinomul de interpolare Lagrange-Sylvester are coeficienţii reali şi, în consecinţă, matricea $F = f(A)$ este reală.*

Exemplul 2.1 Fie $f(z) = e^z$ şi

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

Avem $\mu_A(z) = z^2 - 3z + 2$ şi $\Lambda = \{(1,1)\,,\,(2,1)\}$. Dacă $r(z) = r_1 z + r_0$, atunci sistemul (2.9) se scrie

$$\begin{cases} r_1 + r_0 = e \\ 2r_1 + r_0 = e^2 \end{cases}$$

de unde rezultă $r_0 = 2e - e^2$, $r_1 = e^2 - e$ şi, în consecinţă,

$$F = e^A = (e^2 - e)A + (2e - e^2)I_3 = \begin{bmatrix} e & e^2 - e & 0 \\ 0 & e^2 & 0 \\ 0 & 0 & e^2 \end{bmatrix}. \diamond$$

Definiţia 2.3 a funcţiilor de matrice pe baza polinomului de interpolare Lagrange-Sylvester pune în evidenţă următoarele aspecte.

a) Evaluarea oricărei funcţii de matrice se reduce la evaluarea unui polinom matriceal. Totuşi, deşi sunt stabilite expresii analitice pentru coeficienţii polinomului de interpolare Lagrange-Sylvester, această cale nu este recomandată pentru calculul numeric al funcţiilor de matrice din considerente de eficienţă şi stabilitate numerică.

b) Valoarea funcţiei f în punctul A este determinată exclusiv de mulţimea valorilor funcţiei f pe spectrul matricei A.

c) Dacă matricea A nu are valori proprii multiple, atunci polinomul minimal coincide cu polinomul caracteristic iar sistemul liniar (2.9) a cărui soluţie furnizează coeficienţii polinomului de interpolare Lagrange-Sylvester devine

$$r(\lambda_i) = f(\lambda_i), \quad i = 1 : n. \tag{2.13}$$

Următoarele rezultate (ale căror demonstraţii pot fi găsite, de exemplu, în [VII]), permit evidenţierea unor proprietăţi utile în elaborarea unor proceduri de calcul efectiv.

Teorema 2.1 *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*

$$f(A) = \frac{1}{2\pi i} \oint_\Gamma (zI - A)^{-1} f(z) dz \; . \tag{2.14}$$

Expresia (2.14) poate servi ca definiţie pentru funcţiile analitice (pe un domeniu) iar calculul integralei Cauchy

$$f_{ij} = \frac{1}{2\pi i} \oint_\Gamma e_i^T (zI - A)^{-1} e_j f(z) dz \tag{2.15}$$

poate fi efectuat cu ajutorul teoremei reziduurilor.

Exemplul 2.2 Considerăm funcţia f şi matricea A din exemplul 2.1. Avem $\lambda(A) = \{1, 2, 2\}$ şi

$$(zI - A)^{-1} = \begin{bmatrix} \frac{1}{z-1} & \frac{1}{(z-1)(z-2)} & 0 \\ 0 & \frac{1}{z-2} & 0 \\ 0 & 0 & \frac{1}{z-2} \end{bmatrix}.$$

Prin urmare, f fiind analitică în tot planul complex, putem alege un domeniu simplu conex oarecare ce satisface $\lambda(A) \subset D$. Obţinem

$$f_{11} = \frac{1}{2\pi i} \oint_\Gamma \frac{e^z}{z-1} dz = \text{Rez} \left[\frac{e^z}{z-1} \right] \Big|_{z=1} = e,$$

$$f_{22} = f_{33} = \frac{1}{2\pi i} \oint_\Gamma \frac{e^z}{z-2} dz = \text{Rez} \left[\frac{e^z}{z-2} \right] \Big|_{z=2} = e^2,$$

$$f_{13} = \frac{1}{2\pi i} \oint_\Gamma \frac{e^z}{(z-1)(z-2)} dz = \text{Rez} \left[\frac{e^z}{(z-1)(z-2)} \right] \Big|_{z=1} +$$

$$+\text{Rez} \left[\frac{e^z}{(z-1)(z-2)} \right] \Big|_{z=2} = -e + e^2,$$

celelalte elemente ale matricei $F = e^A$ fiind, evident, nule. $\diamond$

În cele ce urmează vom aborda în exclusivitate cazul funcţiilor analitice şi vom putea utiliza, în consecinţă, relaţia (2.14) ca o relaţie definitorie a funcţiilor de matrice.

Menţionăm, de asemenea, posibilitatea exprimării funcţiilor de matrice prin serii matriceale de puteri. În acest sens avem următoarea

Teorema 2.2 *Dacă funcţia $f(z)$ se poate dezvolta în serie de puteri în jurul punctului $z = z_0$*

$$f(z) = \sum_{k=0}^{\infty} \alpha_k (z - z_0)^k \tag{2.16}$$

şi seria este convergentă în discul $|z - z_0| < r$ atunci această dezvoltare rămâne valabilă dacă argumentul scalar este înlocuit cu argumentul matriceal A

$$f(A) = \sum_{k=0}^{\infty} \alpha_k (A - z_0 I)^k \tag{2.17}$$

oricare ar fi matricea A al cărei spectru se află în interiorul discului de convergenţă.

Din această teoremă rezultă, printre altele, că dezvoltările

$$e^A = \sum_{k=0}^{\infty} \frac{1}{k!} A^k,$$

$$\sin A = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} A^{2k+1}, \quad \cos A = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!} A^{2k}$$

sunt valabile oricare ar fi matricea $A \in \mathcal{C}^{n \times n}$.

Exemplul 2.3 Reluăm matricea A şi funcţia f din exemplul 2.1. Prin inducţie rezultă imediat

$$A^k = \begin{bmatrix} 1 & 2^k - 1 & 0 \\ 0 & 2^k & 0 \\ 0 & 0 & 2^k \end{bmatrix}$$

şi deci

$$F = e^A = \sum_{k=0}^{\infty} \frac{1}{k!} A^k = \begin{bmatrix} \sum_{k=0}^{\infty} \frac{1}{k!} & \sum_{k=0}^{\infty} \frac{2^k - 1}{k!} & 0 \\ 0 & \sum_{k=0}^{\infty} \frac{2^k}{k!} & 0 \\ 0 & 0 & \sum_{k=0}^{\infty} \frac{2^k}{k!} \end{bmatrix} =$$

$$= \begin{bmatrix} e & e^2 - e & 0 \\ 0 & e^2 & 0 \\ 0 & 0 & e^2 \end{bmatrix}. \diamond$$

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 2.2 *Dacă funcţia f este definită pe spectrul matricei A şi*

$$B = TAT^{-1} \tag{2.18}$$

unde T este o matrice nesingulară, atunci

$$f(B) = Tf(A)T^{-1}. \tag{2.19}$$

Demonstraţia este imediată pentru funcţiile analitice pe baza relaţiei definitorii (2.14) întrucât din (2.18) rezultă

$$(zI - B)^{-1} = (T(zI - A)T^{-1})^{-1} = T(zI - A)^{-1}T^{-1}. \diamond$$

Transferul transformărilor de asemănare (2.18) de la nivelul argumentelor matriceale la nivelul funcţiilor are o importanţă decisivă în elaborarea unor tehnici de calcul adecvate care se pot focaliza asupra unor structuri matriceale remarcabile.

Propoziţia 2.3 *Dacă matricea A este (bloc-) diagonală*

$$A = diag(A_{11}, A_{22}, \ldots, A_{pp}) \tag{2.20}$$

atunci $f(A)$ este (bloc-) diagonală şi

$$F \stackrel{\text{def}}{=} f(A) = diag\left(f(A_{11}), f(A_{22}), \ldots, f(A_{pp})\right). \tag{2.21}$$

Demonstraţie. Din faptul că

$$(zI - A)^{-1} = diag\left((zI - A_{11})^{-1}, (zI - A_{22})^{-1}, \ldots, (zI - A_{pp})^{-1}\right)$$

(2.21) rezultă direct din (2.14). $\diamond$

Observaţia 2.1 Conform propoziţiilor 2.2 şi 2.3, dacă

$$T^{-1}AT = diag(J_1, \ldots, J_p)$$

este *forma Jordan* a lui A, atunci avem

$$f(A) = T\, diag(f(J_1), \ldots, f(J_p))T^{-1},$$

şi calculul lui $f(A)$ se reduce la calculul matricelor $f(J_k)$, unde J_k sunt blocurile Jordan. Considerând un bloc Jordan generic

$$J = \begin{bmatrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & \cdots & 0 \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ 0 & 0 & 0 & \cdots & \lambda \end{bmatrix}$$

şi utilizând definiţia 2.3 obţinem

$$f(J) = \begin{bmatrix} f(\lambda) & f'(\lambda) & \ldots & \frac{f^{(m-1)}(\lambda)}{(m-1)!} \\ 0 & f(\lambda) & \ldots & \frac{f^{(m-2)}(\lambda)}{(m-2)!} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & f(\lambda) \end{bmatrix},$$

(exerciţiul 2.4). $\diamond$

Propoziţia 2.4 *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) $f_{ii} = f(a_{ii}), \quad i = 1 : n.$

Demonstraţie.

a) Se ştie că inversa unei matrice superior (inferior) triunghiulare nesingulare este superior (inferior) triunghiulară. În consecinţă $(zI - A)^{-1}$ este superior (inferior) triunghiulară şi, din (2.14), rezultă că aceeaşi structură o are şi matricea $F = f(A)$.

b) Dacă A este o matrice triunghiulară atunci elementele diagonale ale matricei $(zI - A)^{-1}$ sunt

$$\frac{1}{z - a_{ii}}, \quad i = 1 : n.$$

Funcţia f fiind definită pe spectrul matricei A, rezultă că $\lambda_i = a_{ii}$, $i = 1 : n$ nu sunt puncte singulare pentru f şi, în consecinţă, utilizând formula

integrală a lui Cauchy pentru elementele diagonale din (2.14), rezultă

$$f_{ii} = \frac{1}{2\pi i} \oint_{\Gamma} \frac{f(z)}{z - a_{ii}} dz = f(a_{ii}), \quad i = 1 : n.$$

q.e.d. $\diamond$

Propoziţia 2.5 *Fie $\lambda(A) = \{\lambda_1, \ldots, \lambda_n\}$ spectrul matricei $A \in \mathcal{C}^{n\times n}$. Atunci pentru orice funcţie f definită pe spectrul lui A avem*

$$\lambda(f(A)) = \{f(\lambda_1), \ldots, f(\lambda_n)\}. \tag{2.22}$$

Demonstraţie. Fie $B = Q^H A Q$ forma Schur complexă a matricei A. Atunci B este superior triunghiulară şi $b_{ii} = \lambda_i,\ i = 1 : n$. Aplicând propoziţiile 2.2 şi 2.4 rezultă

$$f(A) = Q f(B) Q^H$$

şi, deci, valorile proprii ale matricei $f(A)$ sunt $f(b_{ii}) = f(\lambda_i),\ i = 1 : n$. $\diamond$

Propoziţia 2.6 *Matricele $A \in \mathcal{C}^{n\times n}$ şi $F = f(A)$ comută i.e.*

$$Af(A) = f(A)A. \tag{2.23}$$

Demonstraţie. Existenţa matricei $F = f(A)$ presupune că funcţia f este definită pe spectrul matricei A. În acest caz însă şi funcţia h definită de $h(z) = zf(z) = f(z)z$ este definită pe spectrul matricei A. Din faptul că $h(A)$ are semnificaţie rezultă (2.23). $\diamond$

Teorema 2.1 împreună cu proprietatea (2.23) stabileşte un izomorfism $f(z) \rightsquigarrow f(A)$ între algebra comutativă a funcţiilor analitice $f(z)$ pe spectrul lui A şi algebra (de asemenea comutativă) a funcţiilor de matrice $f(A)$.

Observaţia 2.2 Din propoziţiile 2.2 şi 2.4 rezultă că dacă $Q^H A Q = B$ este *forma Schur* a lui A atunci $f(A) = Q f(B) Q^H$, unde $G = f(B)$ este superior triunghiulară cu $g_{ii} = f(b_{ii})$. Ţinând seama că forma Schur prezintă proprietăţi numerice net superioare formei canonice Jordan şi că propoziţia 2.6 stă la baza unui algoritm fiabil de calcul al funcţiilor de matrice triunghiulare (vezi secţiunea următoare) deducem că rezultatele de mai sus fundamentează o categorie importantă de proceduri de calcul al funcţiilor de argument matriceal. $\diamond$

Proprietăţile următoare, care ilustrează izomorfismul de algebre comutative amintit anterior, permit extinderea unor identităţi care pun în relaţie funcţii de argument scalar cu funcţii de matrice.

Propoziţia 2.7 *Fie $g(z_1, z_2, \ldots, z_p)$ un polinom în nedeterminatele z_1, $z_2, \ldots, z_p$ şi $f_i,\ i = 1 : p$ funcţii definite pe spectrul matricei $A \in \mathcal{C}^{n\times n}$. Definim funcţia*

$$h(z) \equiv g(f_1(z), \ldots, f_p(z)). \tag{2.24}$$

Atunci h este o funcţie definită pe spectrul matricei A şi dacă

$$h^{(k)}(\lambda_i) = 0, \quad i = 1 : l,\ k = 0 : m_i - 1 \tag{2.25}$$

unde $(\lambda_i, m_i),\ i = 1 : l$ sunt zerourile polinomului minimal μ_A cu multiplicităţile respective, atunci

$$H \stackrel{\text{def}}{=} h(A) = g(f_1(A), \ldots, f_p(A)) = 0. \tag{2.26}$$

Demonstraţie. Dacă $r_i,\ i = 1 : p$ sunt polinoamele de interpolare Lagrange-Sylvester ale funcţiilor f_i pe spectrul matricei A, atunci considerăm funcţia u definită de

$$u(z) = g(r_1(z), \ldots, r_p(z))$$

care este, evident, o funcţie polinomială de argument scalar. În aceste condiţii, din (2.25), (2.26) şi modul de definire a polinoamelor de interpolare Lagrange-Sylvester (2.9) rezultă

$$u^{(k)}(\lambda_i) = 0, \quad i = 1 : l,\ k = 0 : m_i - 1,$$

respectiv polinomul de interpolare al funcţiei u este identic nul, de unde obţinem

$$u(A) = g(r_1(A), \ldots, r_p(A)) = g(f_1(A), \ldots, f_p(A)) = 0$$

q.e.d. $\diamond$

Câteva consecinţe imediate ale propoziţiei 2.7, care evidenţiază transferul unor formule scalare în cazul matriceal sunt următoarele relaţii utile.

a) Fie polinomul $g(z_1, z_2) = z_1^2 + z_2^2 - 1$ şi funcţiile $f_1(z) = \sin z,\ f_2(z) = \cos z$. Atunci funcţia definită în (2.25)

$$h(z) \equiv g(\sin z, \cos z) = \sin^2 z + \cos^2 z - 1$$

este identic nulă şi prin urmare (2.26) este satisfăcută pentru orice matrice A. De aici rezultă că, pentru toate matricele pătrate A, avem

$$\sin^2 A + \cos^2 A = I.$$

b) Similar, considerând polinomul $g(z_1, z_2) = z_1 z_2 - 1$ şi funcţiile $f_1(z) = e^z,\ f_2(z) = e^{-z}$ obţinem $e^A e^{-A} = I$, adică

$$e^{-A} = \left(e^A\right)^{-1}.$$

c) De asemenea, cu $g(z_1, z_2, z_3) = z_1 + z_2 + z_3$ şi $f_1(z) = e^{iz},\ f_2(z) = -\cos z,\ f_3(z) = -i \sin z$ rezultă că pentru orice matrice patrată A este adevărată formula

$$e^{iA} = \cos A + i \sin A.$$

Propoziţia 2.8 *Dacă*

$$\rho(z) = \frac{g(z)}{h(z)} \tag{2.27}$$

este o funcţie raţională ireductibilă, definită pe un domeniu care nu conţine rădăcinile polinomului $h(z)$ iar $A \in \mathcal{C}^{n\times n}$ atunci funcţia ρ este definită pe spectrul matricei A dacă şi numai dacă

$$h(\lambda_i) \neq 0 \quad \forall\, \lambda_i \in \lambda(A). \tag{2.28}$$

Dacă (2.28) sunt satisfăcute atunci

$$\rho(A) = g(A)[h(A)]^{-1} = [h(A)]^{-1} g(A). \tag{2.29}$$

Demonstraţie. Funcţia ρ este indefinit derivabilă în toate punctele în care $h(z)$ nu se anulează. În consecinţa $\rho^{(k)}(\lambda_i)$ sunt definite dacă şi numai dacă (2.28) sunt adevărate. Pe de altă parte din (2.28) şi propoziţia 2.5 rezultă că $0 \notin \lambda(h(A))$ i.e. matricea $h(A)$ este nesingulară. Aplicând acum propoziţia 2.7, din identitatea

$$h(z)\rho(z) = \rho(z)h(z) = g(z)$$

rezultă

$$h(A)\rho(A) = \rho(A)h(A) = g(A)$$

care împreună cu nesingularitatea matricei $h(A)$ conduc la (2.29). $\diamond$

Propoziţia 2.9 *Dacă funcţia compusă $h = g \circ f$ este definită pe spectrul matricei $A \in \mathcal{C}^{n\times n}$ atunci*

$$h(A) = g(f(A) \tag{2.30}$$

i.e. $h(A) = g(B)$ unde $B = f(A)$.

Demonstraţia este propusă cititorului. $\diamond$

2.2 Calculul funcţiilor de matrice

Tehnicile numerice de evaluare a funcţiilor de matrice, recomandate de experienţa numerică acumulată, evită determinarea efectivă a polinomului de interpolare Lagrange-Sylvester, în primul rând din motive de eficienţă [1]. Metodele care s-au impus – şi pentru care există şi unele rezultate teoretice privind problemele legate de stabilitatea numerică – 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 expuse 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 2.2 şi 2.3. 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 $\Lambda = diag(\lambda_1,\, \lambda_2,\, \ldots,\, \lambda_n)$.

Prin urmare, pentru orice funcţie definită pe spectrul matricei A, aplicând (2.19) şi (2.21), rezultă

$$F = f(A) = Vf(\Lambda)V^{-1} = Vdiag(f(\lambda_1),\; f(\lambda_2),\; \ldots,\; f(\lambda_n))V^{-1}. \quad (2.31)$$

Introducând acum funcţia **vvp** de calcul a valorilor şi vectorilor proprii ai unei matrice (desigur, prin aplicarea algoritmului QR) cu sintaxa

$$[\, V,\, p \,] = \mathbf{vvp}(A)$$

unde p este vectorul valorilor proprii pentru matricea A, relaţia (2.31) stă la baza următorului algoritm de calcul al funcţiilor de matrice diagonalizabile.

Algoritmul 2.1 (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. $[\, V,\, p \,] = \mathbf{vvp}(A)$
2. `Pentru` $i = 1 : n$
    1. $\beta_i = f(p_i)$
3. $D = V\mathrm{diag}(\beta_1,\; \beta_2,\; \ldots,\; \beta_n)$

4. Se rezolvă sistemul matriceal liniar nesingular $FV = D$ în raport cu F.

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

În cazul funcţiilor f reale pe spectrul matricei reale $A \in \mathcal{R}^{n\times n}$ (vezi definiţia 2.4), conform propoziţiei 2.1 matricea rezultat F este reală. Întrucât matricele V, D şi vectorul p din algoritmul 2.1 nu sunt în mod necesar reale, calculul (aproximativ) în aritmetică complexă conduce la apariţia unor componente parazite imaginare în soluţia calculată . De aceea, într-un astfel de caz, algoritmul 2.1 se completează cu o instrucţiune de eliminare a acestor componente:

5. $F = \mathrm{Re}(F)$.

Dacă matricea A nu este diagonalizabilă, o extindere a algoritmului 2.1 apelează la calculul formei Jordan a matricei A, iar aplicarea propoziţiilor 2.1 şi 2.2 reduce calculul funcţiei de matrice la calculul funcţiilor având ca argumente blocurile Jordan. Această tehnică nu este însă recomandată în primul rând datorită unei sensibilităţi ridicate a structurii Jordan în raport cu perturbaţiile ce apar la nivelul datelor de intrare.

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 2.3 şi proprietatea de comutativitate formulată în propoziţia 2.6. 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, conform (2.23), avem

$$FT = TF. \tag{2.32}$$

Ţinând cont, conform propoziţiei 2.4, că matricea F este, de asemenea, superior triunghiulară şi scriind (2.32) pe elemente obţinem

$$\sum_{k=i}^{j} t_{ik} f_{kj} = \sum_{k=i}^{j} f_{ik} t_{kj}, \quad j = 1:n,\ i = 1:j, \tag{2.33}$$

de unde, în ipotezele menţionate, avem

$$f_{ij} = \frac{1}{t_{jj} - t_{ii}}[t_{ij}(f_{jj} - f_{ii}) + \sum_{k=i+1}^{j-1} (t_{ik}f_{kj} - f_{ik}t_{kj})], \quad j = 1 : n,\ i = 1 : j. \tag{2.34}$$

Relaţia (2.34) 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 (2.34) să apară numai elemente deja calculate. O astfel de ordine există şi ea poate fi evidenţiată observând că elementele diagonale sunt calculabile, conform (2.22), cu formula

$$f_{ii} = f(t_{ii}) \tag{2.35}$$

şi că în membrul drept al relaţiei (2.34) 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 2.1).

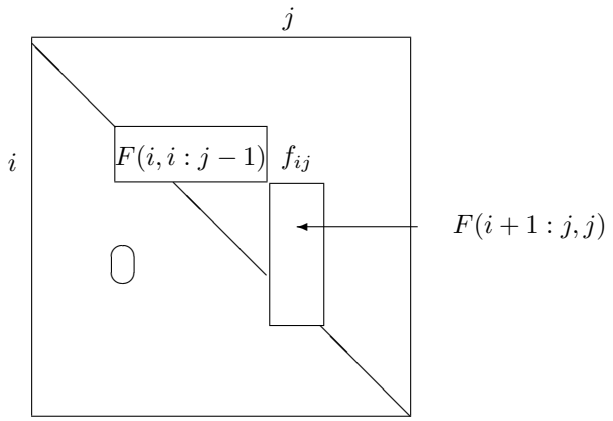

Figura 2.1: Matricea $F = f(T)$.

De exemplu, se poate adopta o ordine diagonală de calcul al elementelor triunghiului superior, după cum este indicat în diagrama din figura 2.2 pentru $n = 5$, numărul înscris în matrice marcând numărul de ordine pentru calculul elementului de pe poziţia respectivă.

Aceasta nu este singura ordine posibilă. Într-adevăr, dacă efectuăm calculele pe coloane în ordinea $j = 1, 2, \ldots, n$, pe fiecare coloană calculele

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

Figura 2.2: Ordinea diagonală de calcul a elementelor matricei $F = f(T)$.

efectuându-se de jos în sus (vezi diagrama a) din figura 2.3) 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 2.3), 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 2.3: 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 $f_{ii},\ i = 1 : n$ cu relaţia (2.35).
2. `Pentru` $q = 2 : n$
    1. `Pentru` $i = 1 : n - q + 1$
        1. Se calculează $f_{i,i+q-1}$ cu relaţia (2.34).

Pentru a nu modifica indexarea folosită în expresia (2.34) a elementului f_{ij} facem următoarele schimbări de indici:

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

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

Algoritmul 2.2 (**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. $f_{ij} = \dfrac{s}{t_{jj} - t_{ii}}$

Schema recurentă care stă la baza algoritmului 2.2 este deosebit de eficientă, în afara celor n evaluări de funcţii scalare de la instructiunea 1, algoritmul necesitând un număr asimptotic de $N_{op} \approx \frac{2n^3}{3}$ operaţii în virgulă mobilă.

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. Introducem funcţia **fsc** de calcul a formei Schur complexe având sintaxa

$$[\, S, U \,] = \mathbf{fsc}(A)$$

unde datele de ieşire sunt matricea unitară de transformare U şi forma Schur complexă S a matricei A legate prin relaţiile

$$S = U^H AU, \quad A = USU^H, \tag{2.36}$$

indicele superior H semnificând dubla operatie de transpunere şi conjugare. Notăm cu **funmt** ("**fun**cţie de **m**atrice **t**riunghiulară") funcţia de calcul realizată de algoritmul 2.2 de mai sus, cu sintaxa (neformală)

$$F = \mathbf{funmt}(f, T)$$

având ca parametrii de intrare funcţia f şi matricea superior triunghiulară T, iar ca parametru de ieşire matricea $F = f(T)$. Cu aceste notaţii prezentăm

> **Algoritmul 2.3** (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. $[\, S, U \,] = \mathbf{fsc}(A)$
> 2. $T = \mathbf{funmt}(f, S)$
> 3. $F = UTU^H$

În cazul în care matricea A are valori proprii multiple, algoritmul 2.3 nu mai este funcţional pentru că formula (2.34) nu mai este aplicabilă. De asemenea, în situaţia existenţei unor valori proprii foarte apropiate, aplicarea formulei (2.34) conduce la fenomene de instabilitate numerică. Soluţia problemei într-un astfel de caz poate consta în utilizarea unei variante "bloc" a algoritmului Parlett după o prealabilă grupare a valorilor proprii apropiate în cadrul aşa numitei forme Schur complexe ordonate. Pentru detalii recomandăm [1], [VII].

2.3 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

$$\Phi(t) = e^{tA}, \tag{2.37}$$

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 $\Phi(0) = I$.

Având în vedere importanţa unui calcul fiabil al exponenţialei matriceale pentru studiul sistemelor liniare, o discuţie prealabilă a condiţionării numerice a problemei calculului lui $\Phi(t) = e^{tA}$, adică a sensibilităţii lui $\Phi(t)$ în raport cu variaţiile lui A, este necesară, întrucât erorile de calcul ale lui $\Phi(t)$ se traduc, prin mecanismul de propagare inversă, în perturbaţii la nivelul datelor de intrare, adică al matricei A.

Notând cu $\hat{\Phi}(t)$ exponenţiala asociată matricei peturbate $A + E$ avem $\dot{\hat{\Phi}} = (A + E)\hat{\Phi}$ cu $\hat{\Phi}(0) = I$. Punând $\hat{\Phi} = \Phi + \Delta\Phi$, obţinem

$$\dot{\Phi} + \Delta\dot{\Phi} = A\Phi + A\Delta\Phi + E\Phi + E\Delta\Phi\,,$$

unde $\Phi(0) = I$, deci $\Delta\Phi(0) = 0$. Pentru simplitate, neglijăm termenul $E\Delta\Phi$ şi, prin integrare, deducem

$$\Delta\Phi(t) = \int_0^t e^{(t-\tau)A} E e^{\tau A}\, d\tau = \Phi(t) \int_0^t e^{-\tau A} E e^{\tau A}\, d\tau\,.$$

Utilizând norma matriceală $\|\cdot\| = \|\cdot\|_2$ putem scrie

$$\|\Delta\Phi(t)\| \le \|\Phi(t)\| \cdot \| \int_0^t e^{-\tau A} E e^{\tau A}\, d\tau\|$$

relaţie din care rezultă

$$\frac{\|\Delta\Phi(t)\|}{\|\Phi(t)\|} \le \mathrm{cond}(\Phi(t)) \frac{\|E\|}{\|A\|}, \tag{2.38}$$

unde

$$\mathrm{cond}(\Phi(t)) \stackrel{\mathrm{def}}{=} \max_{\|E\| = 1} \| \int_0^t e^{-\tau A} E e^{\tau A}\ d\tau\| \cdot \|A\| \tag{2.39}$$

este *numărul de condiţie* al lui A relativ la calculul lui $\Phi(t) = e^{tA}$.

Obţinem o evaluare grosieră a lui $\mathrm{cond}(\Phi(t))$ observând că

$$\| \int_o^t e^{-\tau A} E e^{\tau A}\, d\tau\| \le \int_0^t \|e^{-\tau A}\| \cdot \|E\| \cdot \|e^{\tau A}\|\, d\tau$$

unde

$$\|e^{\pm\tau A}\| \le e^{\tau\|A\|}\,, \quad \tau > 0.$$

De asemenea, avem

$$\int_0^t e^{2\tau\|A\|}\ dt = \frac{1}{2\|A\|}(e^{2t\|A\|} - 1)\,,$$

deci

$$\mathrm{cond}\Phi(t) \leq \frac{1}{2}(e^{2t\|A\|} - 1). \tag{2.40}$$

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

$$t\|A\| \leq 1, \tag{2.41}$$

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

$$e^{tA} = \left(e^{\frac{t}{2^m}A}\right)^{2^m}, \quad m \geq 1. \tag{2.42}$$

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 (2.42), 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

$$m = 1 + [\log_2(t\|A\|)], \tag{2.43}$$

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. $m \leftarrow m - 1$

Observaţia 2.3 Având în vedere necesitatea limitării normei lui tA la calculul exponenţialei, schema $(1/2)^2$ se aplică întotdeauna indiferent de metoda utilizată pentru calculul efectiv (Taylor, Padé, etc.). Trebuie însă

menţionat că ridicarea la pătrat (adică înmulţirea a două matrice) introduce erori de rotunjire suplimentare, mai precis calculul lui F^2 se face cu erori mici faţă de $\|F\|^2$, dar nu neapărat faţă de $\|F^2\|$. Din această cauză utilizarea schemei $(1/2)^2$ ridică probleme de precizie dacă norma $\|e^{tA}\|$ e mică [2]. Pe de altă parte, dacă exponenţiala e utilizată pentru simularea evoluţiei unui sistem liniar, o condiţie de tipul $t\|A\| < 1$ orientează alegerea pasului de discretizare şi, în consecinţă, poate fi asigurată de la început. $\diamond$

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 [3]. 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

$$T_p(z) = \sum_{k=0}^{p} c_k z^k \tag{2.44}$$

[2] Acest lucru e posibil chiar dacă $t\|A\| \gg 1$ deoarece marginea $\|e^{tA}\| \leq e^{t\|A\|}$ utilizată mai sus este în general supraestimată (de exemplu $A = -5$, $t = 1$). De aceea este util să dispunem pentru evaluarea lui $\mathrm{cond}(\Phi(t))$ de estimări mai precise pentru $\|e^{tA}\|$, eventual capabile să ţină seama de repartiţia spectrului lui A şi în special de "dispersia" acestuia ("stiffness"). Se ştie că dacă A are valorile proprii λ_i, atunci e^{tA} are valorile proprii $e^{\lambda_i t}$ (propoziţia 2.5); pe de altă parte pentru orice matrice A şi orice normă matriceală $\|\cdot\|$ avem $\|A\| \geq \rho(A)$, unde $\rho(A) \stackrel{\mathrm{def}}{=} max \mid \lambda_i \mid$, deci $\|e^{tA}\| \geq max \mid e^{\lambda_i t} \mid$. Se poate arăta că dacă $\mathrm{Re}\,\lambda_i < \alpha$ atunci există M astfel încât $\|e^{tA}\| \leq Me^{\alpha t}$, $t \geq 0$. În mod analog, dacă $\beta < \mathrm{Re}\,\lambda_i$ atunci există N astfel încât $\|e^{-tA}\| \leq Ne^{-\beta t}$, $t \geq 0$. Prin urmare dacă spectrul lui A e cuprins în banda $\beta < \mathrm{Re}\,\lambda_i < \alpha$ atunci

$$\|\int_0^t e^{-tA}Ee^{tA}\,dt\| \leq MN\|E\|\int_0^t e^{(\alpha-\beta)\tau}\,d\tau = MN\|E\|\frac{e^{(\alpha-\beta)t}-1}{\alpha-\beta}$$

deci

$$\mathrm{cond}\,\Phi(t) \leq MN\frac{e^{\delta t}-1}{\delta}\|A\|$$

unde

$$\delta \stackrel{\mathrm{def}}{=} \alpha - \beta > 0$$

De aici se vede influenţa nefavorabilă a dispersiei δ a spectrului (cu cât δ e mai mare cu atât $\mathrm{cond}\Phi(t)$ e mai mare) precum şi faptul că numai pentru δ mic (adică $\frac{e^{\delta t}-1}{\delta} \approx t$) $\mathrm{cond}\Phi(t)$ e realmente limitat de $t\|A\|$.

[3] În cazuri speciale (matrice simetrice cu valori proprii reale şi negative, cum sunt cele rezultate prin discretizarea spaţială a problemelor la limite eliptice) se pot utiliza aproximări *globale* (uniforme în sens Cebâşev sau în medie pătratică) valabile pe domeniul ce conţine spectrul lui A.

unde coeficienţii c_k se determină din condiţiile

$$\frac{d^k}{dz^k}(e^z - T_p(z))\bigg|_{z=0} = 0\,, \quad k = 0:p\,. \tag{2.45}$$

Obţinem

$$c_k = \frac{1}{k!}\,, \tag{2.46}$$

adică cea mai bună aproximare polinomială locală de ordin p coincide cu polinomul Taylor corespunzător (fapt binecunoscut, de altfel).

Exemplul 2.4 Pentru $p = 1$ din (2.44) obţinem $T_1(z) = 1 + z$ iar aproximaţia $T_1(tA) = I + tA$ a lui e^{tA} corespunde integrării numerice a ecuaţiei diferenţiale $\dot{x}(t) = Ax(t)$ prin metoda Euler explicită de ordin 1. În mod similar se poate arăta că metoda Runge-Kutta de ordin 4 corespunde alegerii $p = 4$ în (2.44). $\diamond$

Putem evalua eroarea de trunchiere comisă prin utilizarea aproximaţiei polinomiale (2.44) observând că

$$e^z - T_p(z) \stackrel{\text{def}}{=} \sum_{k\geq p+1} \frac{z^k}{k!} = \frac{z^{p+1}}{(p+1)!}\left[1 + \frac{z}{p+2} + \frac{z^2}{(p+2)(p+3)} + \cdots\right]$$

unde paranteza dreaptă e majorată prin

$$1 + \frac{|z|}{1} + \frac{|z|^2}{1\cdot 2} + \cdots = e^{|z|}$$

sau (dacă $|z| < p+2$) prin

$$1 + \frac{|z|}{p+2} + \frac{|z|^2}{(p+2)^2} + \cdots = \frac{1}{1 - \frac{|z|}{p+2}}\,.$$

Punând $z = tA$ şi utilizând prima evaluare (evident, acoperitoare) obţinem eroarea relativă de trunchiere sub forma

$$\frac{\|e^{tA} - T_p(tA)\|}{\|e^{tA}\|} \leq \frac{(t\|A\|)^{p+1}}{(p+1)!}\,. \tag{2.47}$$

Altfel spus, avem

$$\frac{\|e^{tA} - T_p(tA)\|}{\|e^{tA}\|} \leq tol\,, \tag{2.48}$$

unde *tol* este precizia de calcul impusă, dacă alegem p astfel încât

$$\frac{(t\|A\|)^{p+1}}{(p+1)!} < tol\,. \tag{2.49}$$

Pentru orientare, menţionăm că dacă $t\|A\| < \frac{1}{2}$ şi *tol* este de ordinul 10^{-5}, atunci condiţia (2.49) impune $p = 5 \div 6$.

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

$$T_{k+1}(tA) = T_k(tA) + X_k\,, \quad X_k = \frac{1}{k!}t^k A^k\,, \tag{2.50}$$

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

$$X_k = \frac{1}{k}tAX_{k-1}\,, \tag{2.51}$$

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 2.4 (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 (2.49).
2. $X = I$
3. $F = I$
4. `Pentru` $k = 1 : p$
   1. $X \leftarrow \frac{1}{k}tAX$
   2. $F \leftarrow F + X$

Numărul de operaţii necesare este de ordinul $N_{op} \sim pn^3$ (pentru efectuarea produsului matriceal de la pasul 4.1).

Observaţia 2.4 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

$$R_{pq}(z) = \frac{N_p(z)}{D_q(z)} = \frac{c_0 + c_1 z + \cdots + c_p z^p}{1 + d_1 z + \cdots + d_q z^q}\,, \qquad (d_0 \stackrel{\text{def}}{=} 1)\,, \tag{2.52}$$

unde coeficienţii $c_k\,,\, d_k$ se determină din condiţiile

$$\frac{d^k}{dz^k}(e^z - R_{pq}(z)) = 0\,, \quad k = 0 : p + q\,. \tag{2.53}$$

Echivalent, se poate utiliza metoda coeficienţilor nedeterminaţi, scriind formal

$$D_q(z)(\sum_{k\geq 0}\frac{z^k}{k!}) = N_p(z)$$

şi egalând coeficienţii gradelor egale din cei doi membri până la gradul $p+q$ necesar. Rezultă

$$c_k = \frac{(p+q-k)!p!}{(p+q)!k!(p-k)!}\,, \quad d_k = \frac{(p+q-k)!q!}{(p+q)!k!(q-k)!}(-1)^k\,. \tag{2.54}$$

Exemplul 2.5 Pentru $q = 0$ obţinem aproximaţia polinomială (Taylor). Primele aproximaţii raţionale "diagonale", i.e. cu $p = q$, sunt

$$R_{11}(z) = \frac{1+\frac{z}{2}}{1-\frac{z}{2}}$$

cu polul $z = 2$, respectiv

$$R_{22}(z) = \frac{1+\frac{z}{2}+\frac{z^2}{12}}{1-\frac{z}{2}+\frac{z^2}{12}}$$

cu polii $z_{1,2} = 3 \pm i\sqrt{3}$ având $\mathrm{Re}\, z_{1,2} = 3$, plasaţi pe cercul de rază $2\sqrt{3}$. $\diamond$

Prin urmare, trecând la cazul matriceal, constatăm că în general existenţa aproximaţiei raţionale $R_{pq}(tA)$ nu este asigurată necondiţionat (vezi propoziţia 2.8), întrucât matricea $D_q(tA)$ poate fi singulară; de exemplu,

$$D_1(tA) = I - \frac{1}{2}tA$$

este singulară dacă $t = 1$ şi A are o valoare proprie $\lambda = 2$. (În legătură cu aceasta, observăm că dacă este îndeplinită condiţia $t\|A\| < 1$, adică $\|A\| < 1$, atunci cu atît mai mult $\mid \lambda_i \mid < 1$, deci această situaţie este imposibilă). În general, se poate arăta că toate zerourile lui $D_q(z)$ sunt situate în $\mathrm{Re}\, z > 0$ şi se acumulează la ∞ pentru $q \to \infty$ astfel încât $D_q(tA)$ rezultă sigur nesingulară dacă este satisfăcută cel puţin una din următoarele condiţii

i) $\mathrm{Re}\,\lambda_i < 0$, i.e. matricea A este stabilă;

ii) q este suficient de mare;

iii) $t\|A\| < 2$.

În consecinţă, condiţia de limitare a normei lui tA este utilă pentru a asigura nu numai buna condiţionare a problemei calculului lui e^{tA} ci şi

existenţa aproximaţiei raţionale considerate mai sus, independent de caracteristicile de stabilitate ale matricei date A sau de ordinul q adoptat în calcule.

Pe de altă parte, pentru a conecta exemplele 2.4 şi 2.5, este instructiv să observăm că aproximaţia raţională $R_{11}(tA)$ corespunde integrării numerice a ecuaţiei diferenţiale $\dot{x}(t) = Ax(t)$ prin metoda Euler implicită de ordin 1.

Eroarea de trunchiere poate fi limitată alegînd convenabil gradele p, q ale aproximării (2.51). Se poate arăta că (vezi [4]) dacă $t\|A\| < \frac{1}{2}$ atunci

$$R_{pq}(tA) = e^{t(A+E)} \tag{2.55}$$

unde $AE = EA$ iar

$$\|tE\| \leq 8\frac{p!q!}{(p+q)!} \cdot \frac{(t\|A\|)^{p+q+1}}{(p+q+1)!}\,. \tag{2.56}$$

Mai mult, deoarece $AE = EA$ implică $e^{t(A+E)} = e^{tA} \cdot e^{tE}$ (exerciţiul 2.12), eroarea relativă de trunchiere comisă prin utilizarea aproximaţiei Padé $R_{pq}(tA)$ poate fi evaluată acoperitor prin

$$\frac{\|e^{t(A+E)} - e^{tA}\|}{\|e^{tA}\|} \leq \|e^{tE} - I\| = \|tE + \frac{t^2E^2}{2!} + \cdots\| \leq t\,\|E\|\,e^{t\|E\|}\,. \tag{2.57}$$

De exemplu, în cazul aproximaţiei polinomiale ($q = 0$) avem

$$\|tE\| \leq 8\frac{(t\|A\|)^{p+1}}{(p+1)!} \tag{2.58}$$

şi această inegalitate poate fi utilizată în locul lui (2.47). De menţionat că, în orice caz, evaluarea (2.56), valabilă în cazul general $p \neq q$, este simetrică în raport cu p şi q, deci nu permite alegerea lor univocă.

Avem nevoie de un criteriu suplimentar care în cazul de faţă este efortul de calcul necesar pentru realizarea unei aproximaţii de ordin $p+q$ dat. Într-adevăr, dacă $p+q$ e fixat, atunci trebuie să calculăm polinoamele $N_p(tA)$ şi $D_q(tA)$ (printr-o schemă similară cu cea utilizată mai sus pentru $T_p(tA)$), iar în acest scop sunt necesare pn^3 şi, respectiv, qn^3 operaţii. Având în vedere expresiile coeficienţiilor c_k , d_k, cazul diagonal $p = q$ este evident optimal deoarece corespunde la $d_k = (-1)^k c_k$ adică $N_p(tA)$ şi $D_p(tA)$ au (până la semn) aceiaşi termeni care se calculează – o singură dată pentru ambele polinoame – în pn^3 operaţii. Odată fixată relaţia $p = q$ valoarea concretă se alege din condiţia $\|tE\| < eps$, mai precis

$$8\frac{(p!)^2}{(2p)!} \cdot \frac{(t\|A\|)^{2p+1}}{(2p+1)!} \leq eps\,, \tag{2.59}$$

adică, deoarece exponentul din (2.59) este $2p+1$ în loc de $p+1$ din (2.49), p corespunzător aproximaţiei raţionale e aproximativ jumătate din p corespunzător aproximaţiei polinomiale. Altfel spus, cu acelaşi p aproximaţia raţională asigură un ordin de precizie dublu.

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

$$N_{k+1}(tA) = N_k(tA) + X_k \ , \ X_k = c_k t^k A^k \tag{2.60}$$

unde

$$X_k = \frac{p-k+1}{k(2p-k+1)} tAX_{k-1}. \tag{2.61}$$

Rezulta următorul algoritm.

Algoritmul 2.5 (**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 (2.59).
2. $X = I$
3. $N = I$
4. $D = I$
5. `Pentru` $k = 1 : p$
   1. $X \leftarrow \frac{p-k+1}{k(2p-k+1)} tAX$
   2. $N \leftarrow N + X$
   3. $D \leftarrow D + (-1)^k X$
6. Se rezolvă ecuaţia matriceală $DF = N$ în raport cu F.

Numărul de operaţii necesar este de ordinul $N_1 = pn^3$ (pentru efectuarea produsului matriceal de la pasul 5.1.), $N_2 = \frac{n^3}{3}$ (pentru triangularizarea lui D) şi $N_3 = n^3$ (pentru calculul celor n coloane ale lui F), adică, în total, $N_{op} \approx (p + \frac{4}{3})n^3$, comparabil cu cel obţinut anterior.

Observaţia 2.5 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**, care implementează algoritmul 2.3. Pentru calculul exponenţialei matriceale este disponibilă funcţia **expm**, care implementează algoritmul 2.5, bazat pe aproximarea Padé. (Opţional, pentru

compararea rezultatelor, se poate recurge la utilizarea unor versiuni ale funcţiei **expm** ce implementează algoritmii 2.1 şi 2.3).

Alte proceduri de discretizare aproximativă a unor sisteme de ecuaţii diferenţiale liniare, utilizate tradiţional în analiză şi simulare (aproximaţiile de ordin zero şi unu, aproximaţia Tustin, etc.) fac obiectul funcţiei MATLAB **c2dm**.

Exerciţii

E 2.1 Se consideră dată matricea

$$A = \begin{bmatrix} 3 & -3 & 2 \\ -1 & 5 & -2 \\ -1 & 3 & 0 \end{bmatrix}.$$

Să se calculeze următoarele funcţii de matrice folosind definiţia 2.3 şi verificând în prealabil că funcţiile respective sunt definite pe spectrul matricei argument: a) $F = A^{-1}$, b) $G = \sqrt{A}$, c) $H = e^{Aln2}$, d) $S_1 = \sin(\frac{\pi}{4}A)$, e) $C_1 = \cos(\frac{\pi}{4}A)$, f) $S_2 = \sin(\frac{\pi}{8}A)$, g) $C_2 = \cos(\frac{\pi}{8}A)$.

E 2.2 Verificaţi că matricele calculate în exerciţiul precedent satisfac relaţiile: a) $G^2 = A$ (există matrice $X \neq G$ care satisfac $X^2 = A$?), b) $S_1 = 2S_2C_2 = 2C_2S_2$, c) $C_1^2 = C_2^2 - S_2^2$, etc. Puteţi generaliza aceste rezultate arătând, e.g., că $(\sqrt{A})^2 = A$, $\sin 2A = 2\sin A \cos A$, $\cos 2A = 2(\cos A)^2 - I$ etc, oricare ar fi matricea patrată A pe spectrul căreia funcţiile respective sunt definite ?

E 2.3 Reluaţi matricea A din exerciţiul 2.1 şi calculaţi funcţiile de matrice cerute pe baza teoremei 2.1 (relaţiile (2.15)).

E 2.4 Să se deducă expresia analitică a matricei $F = f(J)$ unde $J \in \mathcal{C}^{m\times m}$ este un bloc Jordan iar f este o funcţie definită pe spectrul lui J (vezi observaţia 2.1).

E 2.5 Verificaţi că matricea A din exerciţiul 2.1 este simplă şi determinaţi un set complet de vectori proprii al acesteia. Pe această bază calculaţi funcţiile de matrice cerute utilizând algoritmul 2.1.

E 2.6 Fie $A \in \mathcal{C}^{n\times n}$ o matrice simplă şi v_j, $j = 1 : n$ un set complet de vectori proprii al acestei matrice. Dacă $u_j = V^{-1}(:, j)$, unde V este matricea vectorilor proprii menţionaţi, iar f este o funcţie definită pe spectrul

$\lambda(A) = \{ \lambda_1, \lambda_2, \ldots, \lambda_n \}$ al matricei date să se arate că

$$F \stackrel{\text{def}}{=} f(A) = \sum_{j=1}^{n} f(\lambda_j)\, v_j u_j^H.$$

Folosind setul de vectori proprii al matricei A din exerciţiul 2.1, calculat în exerciţiul precedent, aplicaţi formula de mai sus pentru calculul funcţiilor de matrice cerute.

E 2.7 Rescrieţi algoritmul Parlett de calcul al funcţiilor de matrice superior triunghiulare cu valori proprii distincte calculând elementele matricei $F = f(T)$ în ordinea "pe coloane", respectiv "pe linii" (vezi figura 2.3). Adaptaţi acest algoritm pentru calculul funcţiilor de matrice inferior triunghiulare cu elementele diagonale distincte.

E 2.8 Scrieţi un algoritm eficient pentru calculul funcţiei $F = f(A)$, unde A este superior bidiagonală cu elemente diagonale distincte.

E 2.9 Scrieţi un algoritm eficient pentru calculul unei funcţii $F = f(A)$, unde $A \in \mathcal{R}^{n \times n}$ este simetrică. Consideraţi cazurile $f(z) = e^z$, $f(z) = ln\, z$, $f(z) = \sqrt{z}$.

Indicaţie. Utilizând algoritmul **QR** simetric se obţine $A = Q^T \Lambda Q$, unde Q este ortogonală iar $\Lambda = \text{diag}\,(\lambda_1, \lambda_2, \ldots, \lambda_n)$ conţine valorile proprii (reale) ale lui A. Apoi se aplică (2.19).

E 2.10 Scrieţi un algoritm eficient pentru calculul matricei de tranziţie $\Phi(k) = A^k$ a sistemului liniar discret $x(k+1) = Ax(k)$.

Indicaţie. Executând $A \leftarrow A * A$ de zece ori se obţine $A \leftarrow A^{1024}$. Cum procedaţi dacă, de exemplu, $k = 1453$, $k = 1789$ sau $k = 1917$?

E 2.11 Discutaţi metodele de calcul ale exponenţialei matriceale e^{tA}, unde A este o matrice companion, e.g. are forma

$$A = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -a_1 & -a_2 & \cdots & -a_n \end{bmatrix},$$

şi scrieţi algoritmul de calcul pe care îl recomandaţi.

Indicaţie. Polinomul caracteristic al lui A este $p(s) \stackrel{\text{def}}{=} \det(sI_n - A) = s^n + a_n s^{n-1} + \cdots + a_1$. Utilizând teorema Cayley – Hamilton, conform

căreia $p(A) = 0$, precum şi dezvoltarea în serie a lui e^z, găsim

$$e^{tA} = \sum_{i=1}^{n} \alpha_i(t) A^{i-1} \,,$$

unde coeficienţii funcţiilor analitice $\alpha_i(t)$ pot fi calculaţi recurent. Alternativ, utilizând transformata Laplace a exponenţialei, putem scrie

$$(sI - A)^{-1} = \sum_{k=1}^{n} \frac{s^{k-1}}{p(s)} A_k \,,$$

unde atât matricele A_k cât şi transformatele inverse ale funcţilor $s^{k-1}/p(s)$ pot fi, în principiu, evaluate numeric. Experienţa numerică arată că, în ambele situaţii, efectul erorilor de rotunjire şi trunchiere poate fi catastrofal, ceea ce confirmă încă o dată faptul că o metodă aplicabilă în regim de lucru "cu hârtia şi creionul" este aproape sigur contraindicată în calitate de algoritm de uz curent (pe calculator). În cazul de faţă se recomandă utilizarea algoritmului 2.5, în care structura rară a lui A reduce considerabil numărul de operaţii la pasul 5.1.

E 2.12 Arătaţi că $e^{A+B} = e^A e^B$ este adevărată dacă şi numai dacă $AB = BA$. Puteţi da o expresie (aproximativă) pentru e^{A+B} dacă $AB \neq BA$?

E 2.13 Scrieţi *schema* $(1/2)^2$ pentru calculul funcţiilor $\cos(tA)$ şi $\sin(tA)$ şi argumentaţi utilitatea ei. Idem pentru funcţiile $\cosh(tA)$ şi $\sinh(tA)$.

E 2.14 Scrieţi analogul algoritmilor 2.4 şi 2.5 pentru calculul funcţiilor din exerciţiul 2.13.

Observaţie. Soluţia ecuaţiei diferenţiale $\ddot{x}(t) + Ax(t) = 0$ cu condiţiile iniţiale $x(0) = x$, $\dot{x}(0) = y$ se scrie sub forma

$$x(t) = \left(\cos A^{\frac{1}{2}} t \right) x + \left(A^{-\frac{1}{2}} \sin A^{\frac{1}{2}} t \right) y \,.$$

Examinând seriile corespunzătoare, explicaţi cum se aplică rezultatele de mai sus în această situaţie. Cum procedaţi dacă A este simetrică şi pozitiv definită ?

E 2.15 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?

E 2.16 Fie $A \in \mathcal{R}^{n\times n}$ şi $B \in \mathcal{R}^{n\times m}$ două matrice date. Utilizând definiţia matricei de tranziţie, precizaţi structura exponenţialei matriceale e^{tA^0}, unde

$$A^0 = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}.$$

Indicaţie. Pe baza rezultatelor din § 2.1, se constată imediat că e^{tA^0} are structura

$$e^{tA^0} = \begin{bmatrix} F(t) & G(t) \\ 0 & I_m \end{bmatrix}.$$

Cine sunt $F(t)$ şi $G(t)$?

Observaţie. Aceste rezultate sunt utilizate în secţiunea 4.3 în legătură cu problema discretizării sistemelor liniare.

E 2.17 Generalizaţi constatarea făcută în exerciţiul 2.16 pentru a obţine o procedură eficientă de calcul a integralelor

$$G_k(t) = \int_0^t e^{\eta A} \frac{(t-\eta)^{k-1}}{(k-1)!}\, d\eta, \quad k = 1:p.$$

Indicaţie. Consultaţi [6].

E 2.18 Utilizând seria lui $e^{\eta A}$ şi efectuând integrarea, obţineţi dezvoltările în serie ale funcţiilor $G_k(t)$ şi scrieţi un algoritm de tip 2.4 pentru evaluarea lor. Estimaţi eroarea de trunchiere.

Bibliografie

[1] **Parlett B. N.** *A Recurrence among the Elements of Functions of Triangular Matrices*, Lin. Alg. and Its Applic., vol. 14, pp. 117–121, 1976.

[2] **Ward R. C.** *Numerical Computation of the Matrix Exponential with Accuracy Estimate*, SIAM J. Numer. Anal., vol. 14, pp. 600–610, 1977.

[3] **Van Loan C. F.** *The Sensitivity of the Matrix Exponential*, SIAM J. Numer. Anal., vol. 6, pp. 971–981, 1977.

[4] **Moler C. B., Van Loan C. F.** *Nineteen Dubious Ways to Compute the Exponential of a Matrix*, SIAM Review, vol. 20, pp. 801–836, 1978.

[5] **Van Loan C. F.** *A Note on the Evaluation of Matrix Polynomials*, IEEE Trans. Automat. Contr., vol. AC-24, pp. 320–321, 1978.

[6] **Van Loan C. F.** *Computing Integrals Involving the Matrix Exponential*, IEEE Trans. Automat. Contr., vol. AC-24, pp. 395–404, 1978.

Capitolul 3

Tehnici de procesare a modelelor sistemice liniare

În acest capitol ne vom ocupa de reprezentările numerice ale modelelor matematice ale sistemelor liniare şi vom prezenta principalele proceduri de calcul privitoare la utilizarea şi manipularea acestora în rezolvarea problemelor de analiză şi sinteză sistemică.

3.1 Modele sistemice liniare

În general, un sistem liniar este definit printr-un model de stare de forma

$$(\mathbf{S}) \qquad \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{3.1}$$

unde vectorii $x(t) \in \mathcal{R}^n$, $u(t) \in \mathcal{R}^m$ şi $y(t) \in \mathcal{R}^l$ sunt *starea*, *intrarea* şi, respectiv, *ieşirea* sistemului la momentul curent $t \in \mathcal{R}$ iar A, B, C, D sunt matrice constante de dimensiuni corespunzătoare. Dimensiunea n a spaţiului stărilor $\mathcal{X} = \mathcal{R}^n$ se numeşte ordinul (sau dimensiunea) lui $(\mathbf{S})$.

Pe scurt, un model de stare al unui sistem liniar este definit de un cvartet de matrice şi în consecinţă se notează $S = (A, B, C, D)$.

Semnificaţia dinamică a modelului de stare $(\mathbf{S})$ poate fi rezumată astfel. Dacă starea iniţială $x(0) = x$ şi funcţia de intrare $u(t)$, $t \geq 0$ sunt precizate, atunci traiectoria de stare $x(t)$, $t \geq 0$ este soluţia ecuaţiei diferenţiale din (3.1) iar funcţia de ieşire $y(t)$, $t \geq 0$ rezultă din a doua relaţie (3.1). (Metodele de calcul corespunzătoare sunt expuse în capitolul 4). Din acest

punct de vedere, matricea $A \in \mathcal{R}^{n\times n}$ caracterizează dinamica internă a sistemului (adică evoluţia acestuia la nivelul spaţiului stărilor), coloanele matricei $B \in \mathcal{R}^{n\times m}$ definesc canalele de intrare, liniile matricei $C \in \mathcal{R}^{l\times n}$ definesc canalele de ieşire iar elementele matricei $D \in \mathcal{R}^{l\times m}$ reprezintă căile de transfer direct intrare-ieşire. Dacă $D = 0$ atunci sistemul S, notat pe scurt $S = (A, B, C)$, este pur dinamic, în sensul că orice transfer intrare-ieşire este posibil numai prin intermediul modificării (dinamice) a stării $x(t),\ t \geq 0$. Aceste idei sunt evidenţiate de schema bloc asociată lui (**S**) (figura 3.1).

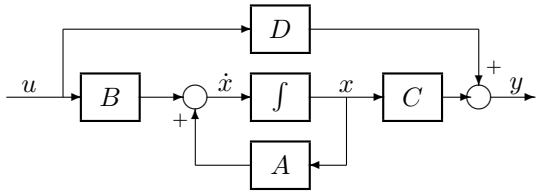

Figura 3.1: Structura modelului de stare al unui sistem liniar.

Modelul de stare (**S**) este invariant la o transformare liniară de stare

$$\tilde{x} = Tx, \tag{3.2}$$

unde $T \in \mathcal{R}^{n\times n}$ este o matrice *nesingulară* arbitrară, în sensul că noul vector de stare $\tilde{x}$ satisface ecuaţii ($\tilde{\mathbf{S}}$) de acelaşi tip cu (**S**), în care matricele corespunzătoare sunt

$$\begin{array}{ll} \tilde{A} = TAT^{-1}\,, & \tilde{B} = TB\,, \\ \tilde{C} = \ CT^{-1}\,, & \tilde{D} = D\,. \end{array} \tag{3.3}$$

Modelele (**S**) şi ($\tilde{\mathbf{S}}$) legate prin relaţiile (3.3) se numesc *echivalente* (sau *asemenea*) şi sunt indiscernabile prin experimente intrare-ieşire, deci reprezintă un acelaşi sistem liniar considerat modulo relaţia de echivalenţă (3.3). Pe scurt, au sens sistemic numai acele proprietăţi ale sistemului (**S**) care sunt *invarianţi* ai lui (**S**) în raport cu transformările (3.3).

Din motive de eficienţă şi siguranţă a calculului, deseori vom restrânge clasa transformărilor utilizând în (3.3) numai matrice $T = U$ *ortogonale*. În acest caz, vom spune că modelele (**S**) şi ($\tilde{\mathbf{S}}$), legate prin relaţiile

$$\begin{array}{ll} \tilde{A} = UAU^T\,, & \tilde{B} = UB\,, \\ \tilde{C} = \ CU^T\,, & \tilde{D} = D\,, \end{array} \tag{3.4}$$

sunt *ortogonal echivalente* şi, în mod corespunzător, vom acorda o atenţie specială *invarianţilor ortogonali* ai lui (**S**). În legătură cu aceste noţiuni, observăm că orice echivalenţă ortogonală este o echivalenţă, deci clasa invarianţilor ortogonali este mai largă decât clasa invarianţilor (orice invariant este invariant ortogonal dar reciproca este falsă!). De aceea, invarianţii ortogonali au deseori proprietăţi de robusteţe şi permit evaluări cantitative, ceea ce le conferă avantaje de calcul şi interpretare sistemică importante. Pentru concretizarea acestor idei în legătură cu noţiunile fundamentale de controlabilitate şi observabilitate, cititorul poate consulta capitolul 5.

În practică se consideră deseori modele de stare generalizate (de tip ,,descriptor") de forma

$$(\mathbf{S}') \qquad \begin{cases} E\dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{3.5}$$

unde în plus faţă de (**S**), aici apare şi matricea *nesingulară* $E \in \mathcal{R}^{n\times n}$.

Interesant este cazul în care E este aproape singulară astfel încât (la fel ca în cazul fascicolelor) formarea produselor $E^{-1}A$, $E^{-1}B$ (respectiv AE^{-1}, CE^{-1}) este nedorită. În acest caz transformările (3.3), (3.4) se scriu

$$\begin{array}{lll} \tilde{E} = TES\,, & \tilde{A} = TAS\,, & \tilde{B} = TB\,, \\ & \tilde{C} = \ CS\,, & \tilde{D} = D\,, \end{array} \tag{3.6}$$

în care T, S sunt matrice nesingulare, respectiv

$$\begin{array}{lll} \tilde{E} = UEV\,, & \tilde{A} = UAV\,, & \tilde{B} = UB\,, \\ & \tilde{C} = \ CV\,, & \tilde{D} = D\,, \end{array} \tag{3.7}$$

în care U, V sunt matrice ortogonale.

Observăm că dacă $E = I$, adică (**S**′) este de forma (**S**), atunci obţinem $\tilde{E} = I$ dacă şi numai dacă $S = T^{-1}$ şi în acest caz (3.6), (3.7) se reduc respectiv la (3.3), (3.4). Pe de altă parte, dacă $E \neq I$ atunci obţinem $\tilde{E} = I$ dacă şi numai dacă $TES = I$, de exemplu $S = I$ şi $T = E^{-1}$ sau $S = E^{-1}$ şi $T = I$, în ambele cazuri transformarea (care implică calculul lui E^{-1}) fiind rău condiţionată.

Din punct de vedere sistemic, modelele de tip (**S**′) cu E aproape singulară corespund sistemelor liniare cu constante de timp mici (sau dinamică internă parazită) şi tehnicile de aproximare a comportării acestui tip de sisteme pe baza unor modele de ordin redus se înscriu în clasa aşa numitelor metode de perturbaţii singulare.

În ultimul timp se studiază şi modele de tip (**S**′) în care matricea E este

singulară. În cazul în care fascicolul (E, A) este regulat [1] aceste modele corespund unor sisteme ,,singulare" care, de exemplu, conţin elemente de derivare pură.

În practica inginerească proprietăţile de transfer intrare-ieşire ale unui sistem liniar în starea iniţială nulă, adică pentru $x(0) = 0$, sunt caracterizate prin intermediul matricei de transfer $T(s)$.

De exemplu, dacă sistemul este definit printr-un model de tip (**S**) atunci avem

$$T(s) = C(sI - A)^{-1}B + D \tag{3.8}$$

şi se constată uşor (pe baza expresiei lui $(sI - A)^{-1}$) că $T(s)$ este o matrice cu l linii şi m coloane ale cărei elemente sunt funcţii raţionale de variabilă complexă s, pe scurt $T(s) \in \mathcal{R}^{l\times m}(s)$ [2]. De asemenea avem

$$\lim_{s\to\infty} T(s) = D\,, \tag{3.9}$$

deci $T(s)$ este proprie, adică fiecare element are gradul numitorului mai mare decât sau cel mult egal cu gradul numărătorului; în particular $T(s)$ este strict proprie dacă şi numai dacă $D = 0$. În sfârşit, $T(s)$ este evident invariantă în raport cu transformările de echivalenţă (3.3) astfel încât ea are într-adevăr caracter sistemic (caracterizează efectiv sistemul, şi nu doar modelul (**S**) considerat).

În teoria realizării sistemelor liniare se arată că, reciproc, orice matrice de transfer $T(s)$ admite o realizare de stare, adică există un model de tip (**S**) ale cărui matrice A, B, C, D satisfac (3.8). Mai mult, acest model este în esenţă unic determinat (modulo echivalenţa (3.3)) şi de ordin minim dacă şi numai dacă el este controlabil şi observabil. (Altfel spus, matricea de transfer este un invariant complet pentru sistemele controlabile şi observabile). Prin urmare, în toate chestiunile care privesc exclusiv proprietăţile de transfer intrare-ieşire ale sistemelor liniare, modelele de stare pot fi suplinite de matricele de transfer corespunzătoare [3].

---

[1] Un fascicol matriceal (E, A) se numeşte *regulat* dacă polinomul $\det(\lambda E - A)$ nu este identic nul.

[2] Notăm cu $\mathcal{R}[s]$ inelul polinoamelor şi cu $\mathcal{R}(s)$ corpul său de fracţii, i.e. corpul funcţiilor raţionale, ambele cu coeficienţi reali. Indicăm, ca de obicei, prin indici superiori dimensiunea vectorilor şi matricelor formate cu aceste elemente.

[3] Afirmaţia nu e valabilă pentru modelele de stare necontrolabile sau/şi neobservabile (utilizate în multe aplicaţii), pentru care nu există (şi nu poate exista) o descriere adecvată în termenii matricelor de transfer. Altfel spus, în general o matrice de transfer identifică nu un sistem ci o întreagă clasă de sisteme liniare *echivalente intrare-iesire* cu un sistem dat.

În acest sens în continuare vom considera şi modele sistemice ,,de transfer", definite prin matrice de forma

$$(\mathbf{T}) \qquad T(s) = [\, T_{ij}(s) \,] \,, \quad T_{ij}(s) = \frac{N_{ij}(s)}{p_{ij}(s)} \tag{3.10}$$

în care $p_{ij},\ N_{ij} \in \mathcal{R}[s]$ şi $\partial p_{ij} \geq \partial N_{ij},\ i = 1:l,\ j = 1:m$ iar ∂ denotă gradul.

Alte forme sub care se poate prezenta modelul ($\mathbf{T}$) precum şi procedurile de conversie corespunzătoare vor fi discutate în continuare.

Similar cu (3.1), un sistem liniar discret este definit tot printr-un cvartet de matrice $S = (A, B, C, D)$, iar matricea de transfer corespunzătoare $T(z)$ are aceeaşi expresie (3.8). Deoarece toate rezultatele procedurale (pur algebrice) din acest capitol se aplică identic şi în cazul discret, orice referire explicită la acesta este lipsită de interes.

3.2 Conexiuni

În acest paragraf arătăm cum se construiesc modelele sistemice de tip ($\mathbf{S}$) şi ($\mathbf{T}$) pentru conexiunile uzuale a două sisteme $S_i,\ i = 1, 2$ definite prin modele de stare

$$(\mathbf{S}_i) \qquad \left\{ \begin{array}{l} \dot{x_i}(t) = A_i x_i(t) + B_i u_i(t) \\ y_i(t) = C_i x_i(t) + D_i u_i(t) \,, \end{array} \right. \tag{3.11}$$

respectiv prin matrice de transfer

$$(\mathbf{T}_i) \qquad y_i(s) = T_i(s) u_i(s) \,. \tag{3.12}$$

Construcţia modelelor sistemice pentru structuri oricât de complexe se reduce, în ultimă instanţă, la aplicarea repetată a procedurilor prezentate mai jos.

În toate cazurile considerate vectorul de stare al sistemului realizat prin conexiune este

$$x = \left[\begin{array}{c} x_1 \\ x_2 \end{array} \right] \in \mathcal{R}^n \,, \quad n = n_1 + n_2 \,, \tag{3.13}$$

adică întotdeauna vectorul de stare rezultă prin agregarea vectorilor de stare ai sistemelor componente iar ordinul conexiunii este egal cu suma ordinelor sistemelor conectate.

Conexiunea paralel

Pentru a putea fi conectate în paralel, sistemele S_1 şi S_2 trebuie să satisfacă următoarele condiţii structurale:

$$m_1 = m_2\,, \quad l_1 = l_2\,, \tag{3.14}$$

adică să aibă acelaşi număr de intrări şi acelaşi număr de ieşiri. Relaţiile de interconexiune (vezi figura 3.2 (b)) sunt

$$u_1 = u_2 = u\,, \quad y = y_1 + y_2\,. \tag{3.15}$$

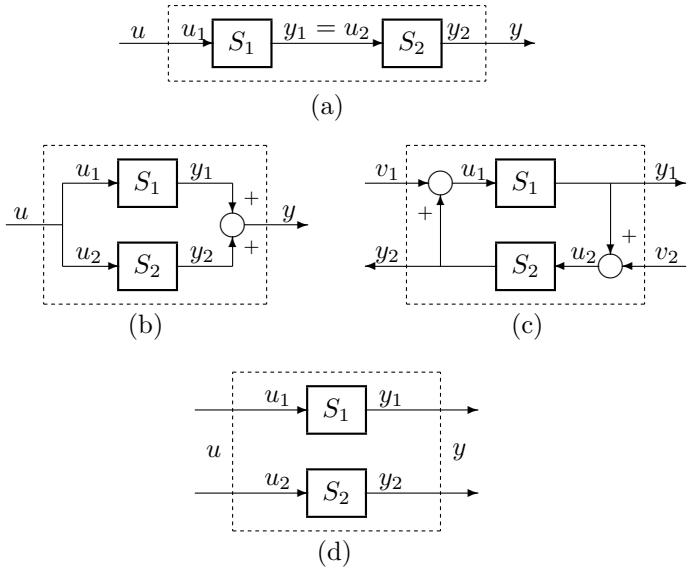

Figura 3.2: Conexiunile sistemice fundamentale: serie (a), paralel (b), reacţie (c) şi produs direct (d).

În consecinţă, ecuaţiile de stare se scriu

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \\ \\ y = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [D_1 + D_2]\, u, \end{cases} \tag{3.16}$$

unde matricele lui $S = (A, B, C, D)$ au expresii evidente, în particular A este bloc - diagonală. Matricea de transfer este

$$T(s) = T_1(s) + T_2(s), \tag{3.17}$$

deci, în cazul SISO, cu notaţii evidente, avem

$$T(s) \stackrel{\text{def}}{=} \frac{N(s)}{p(s)} = \frac{N_1(s)p_2(s) + N_2(s)p_1(s)}{p_1(s)p_2(s)}. \tag{3.18}$$

Observăm că $T(s)$ nu rezultă neapărat ireductibilă chiar dacă $T_i(s)$, $i = 1 : 2$, au această proprietate.

Conexiunea serie (cascadă sau tandem)

Pentru a putea fi conectate în serie, sistemele trebuie să satisfacă următoarea condiţie structurală:

$$l_1 = m_2\,. \tag{3.19}$$

Relaţiile de interconexiune (vezi figura 3.2 (a)) sunt

$$u_1 = u, \quad u_2 = y_1, \quad y = y_2\,. \tag{3.20}$$

În consecinţă, ecuaţiile de stare se scriu

$$\left\{ \begin{array}{rcl} \left[\begin{array}{c} \dot{x}_1 \\ \dot{x}_2 \end{array} \right] & = & \left[\begin{array}{cc} A_1 & 0 \\ B_2C_1 & A_2 \end{array} \right] \left[\begin{array}{c} x_1 \\ x_2 \end{array} \right] + \left[\begin{array}{c} B_1 \\ B_2D_1 \end{array} \right] u \\ y & = & \left[\begin{array}{cc} D_2C_1 & C_2 \end{array} \right] \left[\begin{array}{c} x_1 \\ x_2 \end{array} \right] + \left[\begin{array}{c} D_2D_1 \end{array} \right] u, \end{array} \right. \tag{3.21}$$

unde, din nou, matricele lui S au expresii evidente, în particular A este bloc - inferior - triunghiulară. Matricea de transfer este

$$T(s) = T_2(s)T_1(s)\,, \tag{3.22}$$

(în această ordine a matricelor factor!).

Conexiunea în circuit închis (în buclă sau cu reacţie)

Pentru a putea fi conectate în circuit închis, sistemele trebuie să satisfacă următoarele condiţii structurale:

$$l_2 = m_1, \quad l_1 = m_2\,, \tag{3.23}$$

precum şi condiţia ,,de bună formulare" a conexiunii,

$$\det(I - D_1 D_2) \neq 0\,. \tag{3.24}$$

Această condiţie, necesară pentru a putea exprima y în mod unic funcţie de x_1, x_2 precum şi de intrările externe $v_1,\ v_2$ (vezi figura 3.2 (c)), este generic satisfăcută în raport cu D_1, D_2 şi este satisfăcută în mod sigur dacă fie $D_1 = 0$, fie $D_2 = 0$, adică cel puţin unul dintre sistemele conectate în circuit închis este strict propriu (pur dinamic).

Relaţiile de interconexiune sunt

$$u_1 = v_1 + y_2\,, \quad u_2 = v_2 + y_1\,. \tag{3.25}$$

În consecinţă, ecuaţiile de stare ale conexiunii

$$\begin{cases} \dot{x_1} = A_1 x_1 + B_1(v_1 + y_2) \\ \dot{x_2} = A_2 x_2 + B_2(v_2 + y_1) \end{cases}$$

$$\begin{cases} y_1 = C_1 x_1 + D_1(v_1 + y_2) \\ y_2 = C_2 x_2 + D_2(v_2 + y_1) \end{cases}$$

se scriu sub forma evidentă

$$\left[\begin{array}{c} \dot{x_1} \\ \dot{x_2} \end{array}\right] = \left[\begin{array}{cc|cc} A_1 & 0 & B_1 & 0 \\ 0 & A_2 & 0 & B_2 \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ -- \\ v_1 \\ v_2 \end{array}\right] + \left[\begin{array}{cc} 0 & B_1 \\ B_2 & 0 \end{array}\right] \left[\begin{array}{c} y_1 \\ y_2 \end{array}\right] \tag{3.26}$$

$$\left[\begin{array}{cc} I & -D_1 \\ -D_2 & I \end{array}\right] \left[\begin{array}{c} y_1 \\ y_2 \end{array}\right] = \left[\begin{array}{cc|cc} C_1 & 0 & D_1 & 0 \\ 0 & C_2 & 0 & D_2 \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ -- \\ v_1 \\ v_2 \end{array}\right]. \tag{3.27}$$

Introducând matricele produsului direct $S_d = (A_d, B_d, C_d, D_d)$ al celor două sisteme, vezi fig. 3.2 (d), unde

$$A_d = \left[\begin{array}{cc} A_1 & 0 \\ 0 & A_2 \end{array}\right], \quad B_d = \left[\begin{array}{cc} B_1 & 0 \\ 0 & B_2 \end{array}\right],$$

$$C_d = \left[\begin{array}{cc} C_1 & 0 \\ 0 & C_2 \end{array}\right], \quad D_d = \left[\begin{array}{cc} D_1 & 0 \\ 0 & D_2 \end{array}\right],$$

precum şi matricele de interconexiune

$$B^* = \begin{bmatrix} 0 & B_1 \\ B_2 & 0 \end{bmatrix}, \quad D^* = \begin{bmatrix} 0 & D_1 \\ D_2 & 0 \end{bmatrix}$$

ecuaţiile (3.26) şi (3.27) devin

$$\dot{x} = \begin{bmatrix} A_d & B_d \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + B^* y$$

$$(I - D^*)y = \begin{bmatrix} C_d & D_d \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix},$$

unde, în virtutea condiţiei de bună formulare, matricea $I - D^*$ este inversabilă. Prin urmare avem

$$\begin{cases} \dot{x} = \left(\begin{bmatrix} A_d & B_d \end{bmatrix} + B^*(I - D^*)^{-1} \begin{bmatrix} C_d & D_d \end{bmatrix} \right) \begin{bmatrix} x \\ v \end{bmatrix}, \\ y = (I - D^*)^{-1} \begin{bmatrix} C_d & D_d \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix}. \end{cases} \tag{3.28}$$

Matricele modelului de stare agregat $S = (A, B, C, D)$ al conexiunii în circuit închis au expresiile

$$\begin{array}{ll} A = A_d + B^*(I - D^*)^{-1} C_d, & B = B_d + B^*(I - D^*)^{-1} D_d\,, \\ C = (I - D^*)^{-1} C_d, & D = (I - D^*)^{-1} C_d\,. \end{array} \tag{3.29}$$

iar calculul se face în ordinea

$$\begin{bmatrix} C & D \end{bmatrix} = (I - D^*)^{-1} \begin{bmatrix} C_d & D_d \end{bmatrix},$$

$$\begin{bmatrix} A & B \end{bmatrix} = \begin{bmatrix} A_d & B_d \end{bmatrix} + B^* \begin{bmatrix} C & D \end{bmatrix}.$$

Matricea de transfer este

$$T(s) = (I - T^*(s))^{-1} T_d(s)\,, \tag{3.30}$$

unde

$$T_d(s) = \begin{bmatrix} T_1(s) & 0 \\ 0 & T_2(s) \end{bmatrix}, \qquad T^*(s) = \begin{bmatrix} 0 & T_1(s) \\ T_2(s) & 0 \end{bmatrix}. \tag{3.31}$$

Relaţiile de mai sus pentru construcţia modelelor de stare ale sistemelor agregat evidenţiază faptul că în toate cazurile se face apel exclusiv la calcule matriceale elementare precum şi la manipularea unor structuri matriceale organizate pe blocuri (ceea ce, de exemplu în MATLAB, este, de asemenea, elementar). În consecinţă, scrierea algoritmilor corespunzători este lăsată în sarcina cititorului (vezi exerciţiile 3.1 şi 3.2).

3.3 Realizări

În acest paragraf prezentăm modalităţile de construcţie a realizărilor de stare $S = (A, B, C, D)$, nu nepărat de ordin minim, pentru sisteme definite de matrice de transfer $T(s)$ raţionale *proprii*. Aceste proceduri sunt importante în aplicaţii, deoarece deseori sistemele liniare se descriu utilizând limbajul intrare - ieşire al matricelor de transfer dar se tratează numeric utilizând, aproape exclusiv, realizările de stare corespunzătoare.

În toate cazurile considerate, matricea D a realizării de stare se obţine *"extrăgând partea întreagă"* a lui $T(s)$, adică scriind

$$T(s) = T'(s) + D, \tag{3.32}$$

unde $T'(s)$ este strict proprie. Această operaţie se poate efectua separat pentru fiecare element al lui T, mai precis, dacă

$$T_{ij}(s) \stackrel{\text{def}}{=} \frac{N_{ij}(s)}{p_{ij}(s)}\,, \tag{3.33}$$

unde polinomul p_{ij} este monic [4] şi de grad n, atunci d_{ij} este coeficientul termenului de grad n al lui N_{ij} astfel încât

$$\frac{N_{ij}(s)}{p_{ij}(s)} = d_{ij} + \frac{N'_{ij}(s)}{p_{ij}(s)}\,, \quad N'_{ij}(s) = N_{ij}(s) - d_{ij}p_{ij}(s)\,. \tag{3.34}$$

Altfel spus, dacă notăm $d_{ij} \stackrel{\text{def}}{=} \nu_0$, atunci dispunând coeficienţii $a_0 = 1$, $a_k,\ k = 1 : n$ ai polinomului p_{ij} şi $\nu_k,\ k = 0 : n$ ai polinomului N_{ij} în ordinea descrescătoare a puterilor

$$\begin{array}{cccc} 1 & a_1 & \dots & a_n \\ \nu_0 & \nu_1 & \dots & \nu_n \end{array}$$

şi făcând eliminare gaussiană pentru a anula ν_0, obţinem coeficienţii polinomului N'_{ij}, adică

$$\nu_k \leftarrow \nu'_k = \nu_k - \nu_0 a_k\,, \quad k = 1 : n. \tag{3.35}$$

Operaţia de mai sus se repetă pentru $i = 1 : l,\ j = 1 : m$.

În consecinţă, rămâne să construim realizarea de stare a matricei $T'(s)$ *strict proprie* cu elementele

$$T'_{ij}(s) = \frac{N'_{ij}(s)}{p_{ij}(s)}\,, \tag{3.36}$$

[4] Un polinom se numeşte *monic* dacă are coeficientul termenului de grad maxim egal cu 1.

adică să construim $S' = (A, B, C)$ astfel încât

$$T'(s) = C(sI - A)^{-1}B\,. \tag{3.37}$$

Pentru simplificarea notaţiilor, mai departe eliminăm indicele superior $'$ şi considerăm succesiv câteva cazuri reprezentative de complexitate crescândă. (Subliniem din nou că, deşi procedurile de realizare obţinute au un caracter elementar şi, ca atare, ar justifica o tratare mai expeditivă, ele sunt esenţiale pentru aplicarea metodelor de calcul numeric în analiza şi sinteza sistemică).

A. Sisteme simple, cu o singură intrare şi o singură ieşire (SISO)

În cazul sistemelor cu o singură intrare şi o singură ieşire sau, pe scurt, SISO (**S**ingle-**I**nput, **S**ingle-**O**utput) se dă funcţia de transfer (scalară) strict proprie

$$T(s) = \frac{N(s)}{p(s)}\,, \tag{3.38}$$

cu

$$\begin{aligned} p(s) &= s^n + a_1 s^{n-1} + \cdots + a_n, \\ N(s) &= \qquad \nu_1 s^{n-1} + \cdots + \nu_n. \end{aligned} \tag{3.39}$$

O realizare de stare a lui $T(s)$ se scrie, prin inspecţie, într-una din cele două forme duale următoare

1^0. *Forma standard controlabilă*

$$(\mathbf{C}) \quad \left\{ \begin{array}{l} A_c = \begin{bmatrix} -a_1 & \cdots & -a_{n-1} & -a_n \\ 1 & \cdots & 0 & 0 \\ & \ddots & & \\ & & 1 & 0 \end{bmatrix}, \quad B_c = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \\ \\ C_c = \begin{bmatrix} \nu_1 & \ldots & \nu_{n-1} & \nu_n \end{bmatrix}, \end{array} \right. \tag{3.40}$$

sau, echivalent

$$(\mathbf{C}') \quad \left\{ \begin{array}{l} A_c' = \begin{bmatrix} 0 & 1 & & \\ & & \ddots & \\ & & & 1 \\ -a_n & -a_{n-1} & \cdots & -a_1 \end{bmatrix}, \quad B_c' = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \\ \\ C_c' = \begin{bmatrix} \nu_n & \nu_{n-1} & \cdots & \nu_1 \end{bmatrix}. \end{array} \right. \tag{3.41}$$

2^0. *Forma standard observabilă*

$$(\mathbf{O}) \quad \left\{ \begin{array}{l} A_o = \begin{bmatrix} -a_1 & 1 & & \\ \vdots & & \ddots & \\ -a_{n-1} & & & 1 \\ -a_n & 0 & \cdots & 0 \end{bmatrix}, \quad B_o = \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_{n-1} \\ \nu_n \end{bmatrix}, \\ \\ C_o = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}, \end{array} \right. \tag{3.42}$$

sau, echivalent

$$(\mathbf{O}') \quad \left\{ \begin{array}{l} A_o' = \begin{bmatrix} 0 & 0 & -a_n \\ 1 & & -a_{n-1} \\ & \ddots & \vdots \\ & 1 & -a_1 \end{bmatrix}, \quad B_o' = \begin{bmatrix} \nu_n \\ \nu_{n-1} \\ \vdots \\ \nu_1 \end{bmatrix}, \\ \\ C_o' = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}. \end{array} \right. \tag{3.43}$$

Formele $(\mathbf{C})$ şi $(\mathbf{C}')$ (respectiv $(\mathbf{O})$ şi $(\mathbf{O}')$) diferă neesenţial între ele printr-o permutare evidentă a variabilelor de stare. Formele $(\mathbf{C})$ şi $(\mathbf{O})$ (respectiv $(\mathbf{C}')$ şi $(\mathbf{O}')$) se numesc *duale* deoarece au loc relaţiile

$$A_o = A_c^T, \quad B_o = C_c^T, \quad C_o = B_c^T. \tag{3.44}$$

(Implicaţiile calculatorii ale acestui tip de relaţii vor fi exploatate intensiv în capitolele următoare).

Matricele A_c şi A_o' (respectiv A_o şi A_c') sunt într-o formă superior (inferior) Hessenberg particulară, numită *forma superior (inferior) Frobenius*. Sugestiv, aceste matrice se mai numesc *matrice companion* ale polinomului $p(s)$, cu ai cărui coeficienţi sunt formate.

Prin calcul direct se poate arăta că forma standard controlabilă $(\mathbf{C})$ se bucură de următoarele proprietăţi:

I. Are loc egalitatea

$$(sI - A_c)^{-1}B_c = \frac{L(s)}{p(s)}\,, \tag{3.45}$$

unde

$$L(s) = \begin{bmatrix} s^{n-1} \\ \vdots \\ s \\ 1 \end{bmatrix}.$$

În consecinţă

a) $p(s)$ este polinomul caracteristic al lui A_c.

b) (A_c, B_c, C_c) e o realizare a lui $T(s)$.

II. Matricea de controlabilitate

$$R_c \stackrel{\text{def}}{=} \begin{bmatrix} B_c & A_cB_c & \cdots & A_c^{n-1}B_c \end{bmatrix}$$

este superior triunghiulară nesingulară, deci perechea (A_c, B_c) este controlabilă.

III. Perechea (C_c, A_c) este observabilă (rang $Q_c = n$), deci matricea de observabilitate

$$Q_c = \begin{bmatrix} C_c \\ C_cA_c \\ \vdots \\ C_cA_c^{n-1} \end{bmatrix}$$

este nesingulară, dacă şi numai dacă $T(s)$ este ireductibilă, adică polinoamele $N(s)$ şi $p(s)$ sunt coprime.

Proprietăţile I – II de mai sus justifică denumirea de *formă standard controlabilă* atribuită realizărilor (**C**) (sau (**C**′)), iar proprietatea III arată că ordinul minim al realizării coincide cu gradul numitorului funcţiei de transfer $T(s)$, presupusă ireductibilă.

Proprietăţile formei standard observabile (**O**) se formulează prin dualitate. În particular avem

$$R_o = Q_c^T, \quad Q_o = R_c^T, \tag{3.46}$$

deci, la fel ca mai sus, ordinul minim coincide cu gradul numitorului funcţiei de transfer, presupusă ireductibilă.

B. Sisteme cu o singură intrare şi mai multe ieşiri (SIMO)

În cazul sistemelor SIMO (**S**ingle-**I**nput, **M**ulti-**O**utput) matricea de transfer, considerată dată, are o structură tip coloană

$$T(s) = \begin{bmatrix} T_1(s) \\ \vdots \\ T_l(s) \end{bmatrix} \in \mathcal{R}^l(s)\,, \tag{3.47}$$

unde fiecare $T_i(s)$ este o funcţie raţională strict proprie de forma

$$T_i(s) = \frac{N_i(s)}{p_i(s)}\,, \quad i = 1:l \tag{3.48}$$

cu

$$\begin{aligned} p_i(s) = \;& s^{n(i)} + a_1(i)s^{n(i)-1} + \cdots + a_{n(i)}(i), \\ N_i(s) = \;& \nu_1(i)s^{n(i)-1} + \cdots + \nu_{n(i)}(i). \end{aligned} \tag{3.49}$$

Există două posibilităţi de scriere a unei realizări, una *"naivă"*, care conduce la o realizare de ordin

$$n_I = \sum_{i=1}^{l} n(i)\,, \tag{3.50}$$

alta *"eficientă"*, care conduce la o realizare de ordin n_{II} egal cu gradul celui mai mic multiplu comun (cmmmc) al numitorilor $p_i(s),\ i = 1:l$ (evident, $n_{II} \leq n_I$).

B1. Realizarea "naivă"

Scriem relaţia $y(s) = T(s)u(s)$ pe componente sub forma

$$y_i(s) = T_i(s)u(s), \quad i = 1:l \tag{3.51}$$

şi realizăm fiecare funcţie de transfer $T_i(s)$ printr-un sistem $S_i = (A_i, B_i, C_i)$ utilizând metodele expuse la punctul A, de exemplu putem considera $S_i =$

(A_i, B_i, C_i) în forma standard controlabilă (**C**), i.e.

$$A_i \stackrel{\text{def}}{=} \begin{bmatrix} -a_1(i) & \cdots & -a_{n(i)-1}(i) & -a_{n(i)}(i) \\ 1 & \cdots & 0 & 0 \\ & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}, \quad B_i \stackrel{\text{def}}{=} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$$C_i \stackrel{\text{def}}{=} \begin{bmatrix} \nu_1(i) & \dots & \nu_{n-1}(i) & \nu_n(i) \end{bmatrix}. \tag{3.52}$$

Apoi realizăm conexiunea "paralel la intrare" (vezi figura 3.3 (a)), echivalentă cu (3.51) prin

$$\begin{cases} A = \begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_l \end{bmatrix}, \qquad B = \begin{bmatrix} B_1 \\ \vdots \\ B_l \end{bmatrix}, \\ C = \begin{bmatrix} C_1 & & \\ & \ddots & \\ & & C_l \end{bmatrix}, \end{cases} \tag{3.53}$$

blocurile nediagonale ale matricelor A şi C fiind nule.

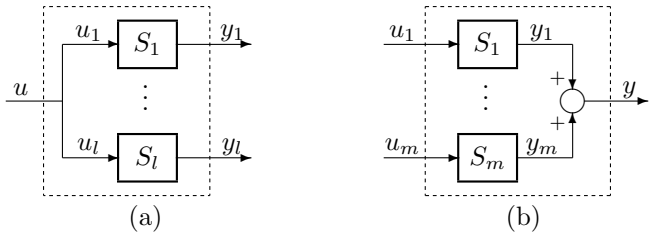

Figura 3.3: Conexiunile "paralel la intrare" (a) şi "paralel la ieşire" (b).

Realizarea (3.53) se numeşte *cu ieşiri decuplate* (sau *decuplată la ieşire*) datorită formei bloc diagonale a perechii (C, A).

B2. Realizarea "eficientă"

Aducând la acelaşi numitor toate elementele lui $T(s)$, scriem

$$T(s) = \begin{bmatrix} T_1(s) \\ \vdots \\ T_l(s) \end{bmatrix} = \begin{bmatrix} N_1^*(s) \\ \vdots \\ N_l^*(s) \end{bmatrix} p^{-1}(s) \stackrel{\text{def}}{=} N(s)p^{-1}(s)\,, \tag{3.54}$$

unde

$$p(s) = s^n + a_1 s^{n-1} + \cdots + a_n \stackrel{\text{def}}{=} \text{cmmmc}(p_1(s), \cdots, p_l(s))\,, \tag{3.55}$$

$$N_i^*(s) = \nu_1^*(i)s^{n-1} + \cdots + \nu_n^*(i) \tag{3.56}$$

deci numărătorul $N \in \mathcal{R}^l[s]$ se scrie sub forma:

$$N(s) = \nu_1 s^{n-1} + \cdots + \nu_n \tag{3.57}$$

unde vectorii $\nu_k \in \mathcal{R}^l$ au componentele $\nu_k{}^*(i),\ i = 1:l,\ k = 1:n$.

Matricea de transfer factorizată $T(s) = N(s)p^{-1}(s)$ se realizează direct în forma *standard controlabilă* (**C**), unde acum polinomul caracteristic al matricei A_c coincide cu $p(s)$ iar C_c conţine, pe fiecare coloană, coeficienţii ν_k ai lui N, adică $C_c \in \mathcal{R}^{l\times n}$.

Observaţia 3.1 Realizarea obţinută este controlabilă. Dacă factorizarea Np^{-1} este coprimă la dreapta, adică nu există un polinom D (divizor) care divide simultan p şi fiecare N_i^*, atunci realizarea astfel obţinută este şi observabilă, deci de ordin minim. $\diamond$

C. Sisteme cu mai multe intrări şi o singură ieşire (MISO)

În acest caz matricea de transfer are o structură linie

$$T(s) = \begin{bmatrix} T_1(s) & \cdots & T_m(s) \end{bmatrix} \in \mathcal{R}^{1\times m}(s)\,. \tag{3.58}$$

unde fiecare $T_i(s)$ este o funcţie raţională strict proprie de forma

$$T_i(s) = \frac{N_i(s)}{p_i(s)},\ i = 1:m.$$

Ca şi în cazul precedent, prezentăm pentru problema realizării o soluţie "naivă" şi una "eficientă".

C1. Realizarea "naivă"

Scriem relaţia $y(s) = T(s)u(s)$ pe componente sub forma

$$y(s) = \sum_{i=1}^{m} T_i(s)u_i(s) \tag{3.59}$$

şi realizăm fiecare funcţie de transfer $T_i(s)$ printr-un model de stare $S_i = (A_i, B_i, C_i)$, utilizând metodele expuse la punctul A. Apoi realizăm conexiunea "paralel la ieşire" (vezi figura 3.3 (b)), echivalentă cu (3.59) prin

$$\left\{ \begin{array}{l} A = \left[\begin{array}{ccc} A_1 & & \\ & \ddots & \\ & & A_m \end{array} \right], \quad B = \left[\begin{array}{ccc} B_1 & & \\ & \ddots & \\ & & B_m \end{array} \right], \\ \\ C = \left[\begin{array}{ccc} C_1 & \cdots & C_m \end{array} \right], \end{array} \right. \tag{3.60}$$

blocurile nediagonale ale matricelor A şi B fiind nule.

Realizarea (3.60) se numeşte *cu intrări decuplate* (sau *decuplată la intrare*).

C2. Realizarea "eficientă"

Procedăm ,,prin dualitate" faţă de cazul B2. Aceasta înseamnă că, aducând la acelaşi numitor toate elementele lui $T(s)$, scriem

$$\begin{aligned} T(s) &= \left[\begin{array}{ccc} T_1(s) & \cdots & T_m(s) \end{array} \right] = \\ &= p^{-1}(s) \left[\begin{array}{ccc} N_1^*(s) & \cdots & N_m^*(s) \end{array} \right] = p^{-1}(s)N(s)\,, \end{aligned} \tag{3.61}$$

unde $p(s) \stackrel{\text{def}}{=} \text{cmmmc}(p_1(s), \cdots, p_m(s))$, $N_i^*(s)$ au expresii similare cu (3.55) şi (3.56), deci numărătorul $N \in \mathcal{R}^{1\times m}[s]$ se scrie sub forma (3.57), unde $\nu_k \in \mathcal{R}^{1\times m}$ au componentele $\nu_k^*(i)$, $i = 1:m$, $k = 1:n$.

Matricea de transfer factorizată $T(s) = p^{-1}(s)N(s)$ se realizează direct în forma *standard observabilă* (**O**), unde acum polinomul caracteristic al matricei A_o coincide cu $p(s)$, iar B_o conţine, pe fiecare linie, coeficienţii lui N, adică $B_o \in \mathcal{R}^{n\times m}$.

Observaţia 3.2 Realizarea obţinută este observabilă. Dacă factorizarea $p^{-1}N$ este coprimă la stânga (i.e. polinoamele $p(s)$ şi fiecare $N_i^*(s)$ nu au un polinom divizor comun) atunci realizarea astfel obţinută este şi controlabilă, deci de ordin minim. $\diamond$

D. Sisteme cu mai multe intrări şi mai multe ieşiri (MIMO)

Se dă matricea de transfer cu l linii şi m coloane $T(s) = [\, T_{ij}(s) \,]$.

D1. Realizarea "naivă"

Această realizare se bazează pe scrierea

$$y_i(s) = \sum_{j=1}^{m} T_{ij}(s) u_j(s), \quad i = 1 : l \tag{3.62}$$

şi este de ordin

$$n = \sum_{i=1}^{l} \sum_{j=1}^{m} n(i,j) \tag{3.63}$$

unde $n(i,j)$ e gradul numitorului lui $T_{ij}(s)$.

D2. Realizarea "eficientă"

În acest caz, o realizare eficientă se poate construi în două moduri.

1^0. Partiţionăm $T(s)$ pe coloane

$$T(s) = \left[\begin{array}{ccc} T_1(s) & \ldots & T_m(s) \end{array} \right] \tag{3.64}$$

şi aducem fiecare coloană $T_j(s) \in \mathcal{R}^l(s)$ la acelaşi numitor ca la punctul **B**, adică scriem $T_j(s) = N_j(s) p_j^{-1}(s)$, unde $N_j(s) \in \mathcal{R}^l[s]$, $p(s) \in \mathcal{R}[s]$. Deci avem

$$\begin{aligned} T(s) &= \left[\begin{array}{ccc} N_1(s)p_1^{-1}(s) & \ldots & N_m(s)p_m^{-1}(s) \end{array} \right] = \\ &= \left[\begin{array}{ccc} N_1(s) & \ldots & N_m(s) \end{array} \right] \left[\begin{array}{ccc} p_1(s) & & \\ & \ddots & \\ & & p_m(s) \end{array} \right]^{-1} = \\ &\stackrel{\text{def}}{=} N(s) p^{-1}(s)\,, \end{aligned} \tag{3.65}$$

unde

$$\begin{aligned} p_j(s) = \;& s^{n(j)} + a_1(j) n^{(j)-1} + \ldots + a_{n(j)}(j), \quad a_k(j) \in \mathcal{R}, \\ N_j(s) = \;& \nu_1(j) s^{(j)-1} + \ldots + \nu_{n(j)}(j), \quad \nu_k(j) \in \mathcal{R}^l. \end{aligned} \tag{3.66}$$

Întrucât relaţia $y(s) = T(s)u(s)$ se scrie

$$y(s) = \sum_{j=1}^{m} N_j(s) p_j^{-1}(s) u_j(s)\,, \tag{3.67}$$

iar fiecare coloană $N_j(s)p_j^{-1}(s)$ se realizează în forma standard controlabilă $(A_c(j), B_c(j), C_c(j))$, ca în cazul B2, obţinem realizarea *standard controlabilă* (decuplată la intrare)

$$\left\{ \begin{array}{l} A_c = \left[\begin{array}{ccc} A_c(1) & & \\ & \ddots & \\ & & A_c(m) \end{array} \right], \quad B_c = \left[\begin{array}{ccc} B_c(1) & & \\ & \ddots & \\ & & B_c(m) \end{array} \right], \\ \\ C_c = \left[\begin{array}{ccc} C_c(1) & \cdots & C_c(m) \end{array} \right]. \end{array} \right. \tag{3.68}$$

Ordinul $n = n_c$ este suma gradelor $n(j)$ ale numitorilor comuni pe coloană. (Se poate arăta că aceste grade coincid cu indicii de controlabilitate ai perechii (A_c, B_c)).

2^0. Partiţionând $T(s)$ pe linii şi procedând prin dualitate faţă de punctul precedent, obţinem realizarea *standard observabilă* (decuplată la ieşire)

$$\left\{ \begin{array}{l} A_o = \left[\begin{array}{ccc} A_o(1) & & \\ & \ddots & \\ & & A_o(l) \end{array} \right], \quad B_o = \left[\begin{array}{c} B_o(1) \\ \vdots \\ B_o(l) \end{array} \right] \\ \\ C_o = \left[\begin{array}{ccc} C_o(1) & & \\ & \ddots & \\ & & C_o(l) \end{array} \right]. \end{array} \right. \tag{3.69}$$

Ordinul $n = n_o$ al realizării (3.69) este suma gradelor numitorilor comuni pe linie. (Se poate arăta că aceste grade coincid cu indicii de observabilitate ai perechii (C_o, A_o)).

Observaţia 3.3 În general, niciuna dintre cele două realizări (3.68) şi (3.69) nu este minimală căci, în general, perechea (C_c, A_c) nu rezultă observabilă iar perechea (A_o, B_o) nu rezultă controlabilă. $\diamond$

În finalul acestei secţiuni precizăm că obţinerea efectivă a realizărilor de mai sus necesită, în primul rând, un mecanism de manipulare a matricelor bloc cum este cel oferit de MATLAB (sau cel care poate fi creat

cu relativă uşurinţă de utilizator în orice limbaj de programare). Cu un astfel de mecanism la dispoziţie, scrierea algoritmilor de construcţie a realizărilor prezentate devine extrem de simplă şi, din acest motiv, face obiectul exerciţiilor.

3.4 Conversii de modele

Conversia de la un model de tip matrice de transfer la un model de stare, pe scurt $(\mathbf{T}) \rightarrow (\mathbf{S})$, constă din aplicarea uneia dintre procedurile de realizare, care au fost discutate în paragraful anterior. Conversia inversă $(\mathbf{S}) \rightarrow (\mathbf{T})$, respectiv calculul matricei de transfer asociate unui model de stare dat, se face separat pentru fiecare pereche (u_j, y_i) de intrări şi ieşiri, adică în esenţă se referă la un sistem simplu (SISO). Deoarece formele alternative de reprezentare ale unei matrice de transfer se referă tot la sisteme simple, adică la funcţii de transfer, în acest paragraf ne vom referi numai la aceste obiecte.

Calculul funcţiei de transfer (Conversia $(\mathbf{S}) \rightarrow (\mathbf{T})$)

Se dă un sistem simplu definit prin

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases}, \tag{3.70}$$

unde matricele $A \in \mathcal{R}^{n\times n},\ B \in \mathcal{R}^{n},\ C \in \mathcal{R}^{1\times n}$ sunt cunoscute.

Pentru a calcula funcţia de transfer, adică două polinoame $p(s),\ N(s)$ astfel încât

$$T(s) \stackrel{\text{def}}{=} C(sI - A)^{-1}B = \frac{N(s)}{p(s)}, \tag{3.71}$$

constatăm că funcţia de transfer a unui sistem cu reacţie unitară (vezi figura 3.4) este

$$T_0(s) = \frac{T(s)}{1 + T(s)} = \frac{N(s)}{p(s) + N(s)} = \frac{N(s)}{p_0(s)} \tag{3.72}$$

unde $p_0(s) = p(s) + N(s)$. Deci

$$N(s) = p_0(s) - p(s). \tag{3.73}$$

(Menţionăm că rădăcinile polinomului $N(s)$ sunt zerourile sistemului $S = (A, B, C)$).

Problema s-a redus la a calcula $p(s)$ şi $p_0(s)$.

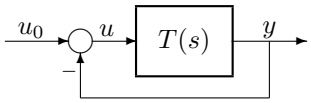

Figura 3.4: Conexiune cu reacţie unitară.

Dar $p(s)$ este polinomul caracteristic al lui A adică

$$p(s) = \prod_{i=1}^{n}(s - \lambda_i)\,, \tag{3.74}$$

unde λ_i sunt polii lui $T(s)$, adică valorile proprii ale matricei A. Analog, $p_0(s)$ este polinomul caracteristic al matricei A_0 a sistemului în circuit închis obţinut punând $u = u_0 - y$, adică $u = u_0 - Cx$. Deci $A_0 = A - BC$.

Prin urmare, pentru a calcula funcţia de transfer $T(s)$ se procedează astfel:

1. Se calculează valorile proprii λ_i ale matricei A utilizând algoritmul QR.
2. Se formează $A_0 = A - BC$.
3. Se calculează valorile proprii λ_{0i} ale matricei A_0 utilizând algoritmul QR.
4. Se calculează (coeficienţii lui) $p(s) = \prod_{i=1}^{n}(s - \lambda_i)$.
5. Se calculează $p_0(s) = \prod_{i=1}^{n}(s - \lambda_{0i})$.
6. Se calculează $N(s) = p_0(s) - p(s)$.

Dacă sistemul $S = (A, B, C)$ nu este controlabil şi observabil (ceea ce este probabil să se întâmple dacă provine dintr-un sistem multiplu) atunci p şi N, adică p şi p_0 au un cel mai mare divizor comun (cmMdc) $\neq 1$, care coincide cu produsul $\prod(s - \lambda_{fi})$, unde $\lambda_{fi} \in \sigma(A) \cap \sigma(A_0)$ sunt polii ficşi sistemului (A, B, C). Acest divizor poate fi eliminat prin inspecţie înainte de pasul 4, dar decizia poate fi afectată de erorile de rotunjire inerente.

Pentru o funcţie de transfer strict proprie definim, în afară de reprezentarea "raţională"

$$T(s) = \frac{N(s)}{p(s)}\,, \tag{3.75}$$

ca raport de două polinoame, încă două reprezentări:

$$(\mathbf{T}_1) \qquad T(s) = \delta \frac{\prod_{j=1}^{n-k}(s-\mu_j)}{\prod_{i=1}^{n}(s-\lambda_i)} \tag{3.76}$$

şi

$$(\mathbf{T}_2) \qquad T(s) = \sum_{i=1}^{n} \frac{r_i}{s-\lambda_i}\,, \tag{3.77}$$

unde λ_i sunt polii, μ_j sunt zerourile, δ este amplificarea la infinit, numărul întreg $k \geq 1$ este excesul poli - zerouri sau ,,tipul" lui $T(s)$, iar r_i sunt reziduurile lui $T(s)$ în polii λ_i (a doua reprezentare există ca atare numai dacă toţi polii lui $T(s)$ sunt simpli.)

Relativ la reprezentarea $(\mathbf{T}_1)$ discutăm în primul rând:

Conversia $(\mathbf{S}) \rightarrow (\mathbf{T}_1)$

Se dă modelul de stare $S = (A, B, C)$. Pentru a calcula $T(s)$ sub forma $(\mathbf{T}_1)$ se procedează astfel:

1. Se calculează valorile proprii λ_i ale matricei A.
2. Se calculează zerourile μ_j ale sistemului (A, B, C).
3. Se alege $\sigma_0 \in \mathcal{R}$, astfel încât $\sigma_0 \neq \lambda_i,\ \mu_j,\ \forall(i,j)$ şi se calculează
$$\delta = \frac{C(\sigma_0 I - A)^{-1}B}{\frac{\prod_{j=1}^{n-k}(\sigma_0-\mu_j)}{\prod_{i=1}^{n}(\sigma_0-\lambda_i)}}\,.$$

Observaţia 3.4 La pasul 2 se pot utiliza încă cel puţin două metode.

a) *Metoda generală* ţine seama de faptul că zerourile sistemului (A, B, C) (considerat controlabil şi observabil) coincid cu valorile proprii generalizate finite ale fascicolului

$$\lambda \begin{bmatrix} I_n & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} A & B \\ C & 0 \end{bmatrix}, \tag{3.78}$$

deci se calculează utilizând algoritmul **QZ**, vezi [3].

b) *Metoda specifică* în cazul $l = 1,\ m = 1$ considerat "simulează", în limbajul descrierii de stare $S = (A, B, C)$, procedura de inversare a sistemului propriu având funcţia de transfer $s^k T(s)$, unde (A, B, C) este

o realizare a lui $T(s)$. Fie k ordinul relativ al lui (A, B, C), adică primul întreg ≥ 1 astfel încât $\delta \stackrel{\text{def}}{=} CA^{k-1}B \neq 0$. Din ecuaţiile de stare

$$\left\{ \begin{array}{l} \dot{x} = Ax + Bu \\ y = Cx, \end{array} \right. \tag{3.79}$$

prin derivarea succesivă a ieşirii y ţinând seama de faptul că $CA^iB = 0$, $i = 0 : k-2$, obţinem

$$\begin{array}{rcl} \dot{y} & = & CAx \\ & \vdots & \\ y^{(k-1)} & = & CA^{k-1}x \\ y^{(k)} & = & CA^kx + CA^{k-1}Bu \stackrel{\text{def}}{=} CA^kx + \delta u\,, \end{array} \tag{3.80}$$

unde, prin ipoteză, $\delta \neq 0$. În consecinţă, sistemul

$$\left\{ \begin{array}{ll} \dot{x} & = Ax + Bu \\ y^{(k)} & = CA^kx + \delta u \end{array} \right. \tag{3.81}$$

este inversabil iar inversul său are ecuaţiile

$$\left\{ \begin{array}{l} \dot{x} = (A - \delta^{-1}BCA^k)x + \delta^{-1}By^{(k)} \\ u = -\delta^{-1}CA^kx + \delta^{-1}y^{(k)}. \end{array} \right. \tag{3.82}$$

Mai mult, matricea $A - \delta^{-1}BCA^k$ are exact k valori proprii nule (corespunzătoare celor k valori proprii generalizate infinite ale fascicolului (3.78)) precum şi $n-k$ valori proprii egale cu zerourile lui (A, B, C). Prin urmare ordinul relativ k coincide cu excesul poli - zerouri iar $\delta = CA^{k-1}B$ este amplificarea la infinit, adică $s^kT(s) \to \delta$, când $s \to \infty$. În definitiv, zerourile (finite) μ_j, $j = 1 : n-k$ ale sistemului (A, B, C) coincid cu cele $n-k$ valori proprii nenule ale matricei $A - \delta^{-1}BCA^k$. $\diamond$

În ceea ce priveşte celelalte conversii, dăm numai câteva sugestii. (Menţionăm că procedurile de calcul polinomial care apar în schemele de calcul de mai jos sunt discutate în secţiunea finală a acestui capitol.)

Conversia $(\mathbf{T}_1) \to (\mathbf{T})$

1. Se calculează $p(s) = \prod_{i=1}^{n}(s - \lambda_i)$.
2. Se calculează $N(s) = \prod(s - \mu_i)$.
2. $N(s) \leftarrow \delta N(s)$

Observaţia 3.5 Uneori în loc de algoritmul $(\mathbf{S}) \to (\mathbf{T})$ prezentat mai sus se utilizează $(\mathbf{S}) \to (\mathbf{T}_1)$ urmat de $(\mathbf{T}_1) \to (\mathbf{T})$. $\diamond$

Conversia $(\mathbf{T}) \to (\mathbf{T}_1)$

1. Se construieşte o realizare de stare a lui $T(s)$.
2. Se aplică procedura de conversie $(\mathbf{S}) \to (\mathbf{T}_1)$.

Conversia $(\mathbf{T}_1) \to (\mathbf{S})$

1. Se construieşte reprezentarea raţională a lui $T(s)$.
2. Se aplică algoritmul de realizare $(\mathbf{T}) \to (\mathbf{S})$.

Relativ la reprezentarea $(\mathbf{T}_2)$, menţionăm numai că ea se obţine din $(\mathbf{T})$ calculând polii λ_i cu algoritmul QR aplicat matricei companion A_c (respectiv A_o) şi apoi calculând reziduurile r_i cu formula cunoscută

$$r_i = \frac{N(\lambda_i)}{p'(\lambda_i)}\,, \tag{3.83}$$

evaluările polinoamelor realizându-se cu algoritmul (schema) lui Horner (vezi secţiunea următoare).

Alternativ, se construieşte o realizare de stare (A, B, C) a lui $T(s)$ şi se calculează valorile proprii λ_i (presupuse distincte) precum şi vectorii proprii x_i ai lui A. Considerând matricea $X = \left[\begin{array}{ccc} x_1 & \dots & x_n \end{array}\right]$ şi efectuând transformarea

$$X^{-1}AX = \left[\begin{array}{ccc} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{array}\right], \quad X^{-1}B = \left[\begin{array}{c} \beta_1 \\ \vdots \\ \beta_n \end{array}\right], \tag{3.84}$$

$$CX = \left[\begin{array}{ccc} \gamma_1 & \cdots & \gamma_n \end{array}\right],$$

avem

$$r_i = \gamma_i\beta_i, \quad i = 1 : n. \tag{3.85}$$

3.5 Algoritmi de calcul polinomial

În secţiunile precedente ale capitolului de faţă s-a evidenţiat faptul că procesarea modelelor sistemice de tip matrice de transfer se reduce, în ultimă instanţă, la efectuarea unui set de operaţii cu polinoame. De aceea reunim aici procedurile numerice de bază pentru rezolvarea problemelor de calcul polinomial.

Notăm un polinom $a(s)$ de grad n prin

$$a(s) = a_0 s^n + a_1 s^{n-1} + \cdots + a_n\,, \quad a_0 \neq 0 \tag{3.86}$$

şi îi asociem vectorul

$$a = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} \tag{3.87}$$

de lungime $n+1$. [5]

Se ştie că polinoamele de grad $\leq m$, cu operaţiile de adunare şi înmulţire cu scalari definesc un *spaţiu vectorial* de dimensiune $m+1$. În acest spaţiu, cu m fix [6] şi $n \leq m$, polinomul $a(s)$ se reprezintă prin vectorul

$$T^{m-n}\, a = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ a_0 \\ \vdots \\ a_n \end{bmatrix} \in \mathcal{R}^{m+1}\,, \tag{3.88}$$

unde T este operatorul de ,,deplasare (*shift*) cu un pas în jos urmat de adăugarea unui zero pe prima poziţie" definit prin

$$(Ta)_i = a_{i-1},\ i \geq 1,\ \ (Ta)_0 = 0. \tag{3.89}$$

Reţinând acest fapt simplu, în continuare trecem în revistă modalităţile de efectuare a operaţiilor principale cu polinoame.

1^0. **Suma** $c(s) \stackrel{\text{def}}{=} a(s)+b(s)$ a două polinoame $a(s),\ b(s)$ de grade $n,\ m$ cu $n \leq m$ [7] este de grad m, deci

$$c = T^{m-n} a + b. \tag{3.90}$$

Rezultă următorul algoritm.

[5] Numerotarea elementelor unui vector (tablou unidimensional) se face în mod diferit în diverse limbaje formale de programare, respectiv începând cu 0, ca în limbajul **C**, sau cu 1 ca în MATLAB. Trecerea la cea de a doua variantă reprezintă un exerciţiu util pentru cititor.

[6] De fiecare dată când efectuăm o operaţie alegem m minim necesar.

[7] Altfel facem suma $b(s) + a(s)$.

Algoritmul 3.1 (Se dau vectorii $a \in \mathcal{R}^{n+1}$ şi $b \in \mathcal{R}^{m+1}$, cu $m \geq n$, ai coeficienţilor polinoamelor $a(s)$ şi $b(s)$. Algoritmul calculează vectorul $c \in \mathcal{R}^{m+1}$ al coeficienţilor polinomului sumă $c(s) = a(s) + b(s)$.)

1. `Pentru` $i = 0 : m - n - 1$
   1. $c_i = b_i$
2. `Pentru` $i = m - n : m$
   1. $c_i = a_{i-(m-n)} + b_i$

2^0. **Produsul** $p(s) \stackrel{\text{def}}{=} a(s)b(s)$ a două polinoame $a(s)$, $b(s)$ de grade $n \geq m$ [8] este de grad $m + n$. Avem

$$a(s)b(s) = \left(\sum_{i=0}^{n} a_i s^{n-i}\right)\left(\sum_{j=0}^{m} b_j s^{m-j}\right) = \sum_{k=0}^{m+n} p_k s^{n+m-k}, \tag{3.91}$$

unde

$$p_k = \sum_{i+j=k} a_i b_j = \left\{ \begin{array}{l} \sum_{j=0}^{k} a_{k-j} b_j \\ \\ \sum_{i=0}^{k} a_i b_{k-i} \end{array} \right. . \tag{3.92}$$

(Pentru comoditatea scrierii am presupus că $a_i = 0$ pentru $i > n$ şi $b_j = 0$ pentru $j > m$. În general, operaţia (3.92) reprezintă *convoluţia discretă* a şirurilor $(a_i)_{i\geq 0}$, $(b_j)_{j\geq 0}$).

Din (3.92) se constată uşor ca vectorul $p \in \mathcal{R}^{m+n+1}$, al coeficienţilor polinomului produs, se poate exprima într-o formă matriceală în modul următor

$$\begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_m \\ \vdots \\ p_n \\ p_{n+1} \\ \vdots \\ p_{n+m} \end{bmatrix} = \begin{bmatrix} a_0 & 0 & \cdots & 0 \\ a_1 & a_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_m & a_{m-1} & \cdots & a_0 \\ \vdots & \vdots & & \vdots \\ a_n & a_{n-1} & \cdots & a_{n-m} \\ 0 & a_n & \cdots & a_{n-m+1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \end{bmatrix}, \tag{3.93}$$

[8] Altfel facem produsul $b(s)a(s)$.

adică, pentru a putea descrie produsul unui polinom $a(s)$ de grad n cu un polinom $b(s)$ de grad m, îi asociem lui $a(s)$ matricea cu $m+1$ coloane

$$A = \begin{bmatrix} a_0 & & \\ \vdots & \ddots & \\ a_n & & a_0 \\ & \ddots & \vdots \\ & & a_n \end{bmatrix} \in \mathcal{R}^{(m+n+1)\times(m+1)}, \tag{3.94}$$

şi scriem

$$p = Ab. \tag{3.95}$$

În mod analog, datorită comutativităţii produsului de polinoame, avem şi $p = Ba$ unde matricea cu $n+1$ coloane $B \in \mathcal{R}^{(m+n+1)\times(n+1)}$ se formează, în aceeaşi manieră, cu coeficienţii lui $b(s)$.

Obţinem următorul algoritm.

Algoritmul 3.2 (Se dau vectorii $a \in \mathcal{R}^{n+1}$ şi $b \in \mathcal{R}^{m+1}$, cu $m \geq n$, ai coeficienţilor polinoamelor $a(s)$ şi $b(s)$. Algoritmul calculează vectorul $c \in \mathcal{R}^{m+n+1}$ al coeficienţilor polinomului produs $c(s) = a(s)b(s)$.)

1. `Pentru` $k = 0 : m+n$
   1. $p_k = 0$
2. `Pentru` $j = 0 : m$
   1. `Pentru` $i = 0 : n$
      1. $p_{i+j} = p_{i+j} + a_i b_j$

Observaţia 3.6 Operaţiile 1^0 şi 2^0 definesc *inelul comutativ* $\mathcal{R}[s]$ al polinoamelor cu coeficienţi reali. $\diamond$

3^0. **Teorema împărţirii întregi** pentru două polinoame $a(s), b(s)$ de grade $n \leq m$ afirmă existenţa unică a două polinoame $q(s)$ de grad $m-n$ şi $r(s)$ de grad cel mult $n-1$ astfel încât

$$b(s) = a(s)q(s) + r(s). \tag{3.96}$$

Dacă $n > m-n$, ţinând seama de exprimarea matriceală a produsului de

polinoame, (3.96) se scrie

$$\begin{bmatrix} b_o \\ \vdots \\ b_{m-n} \\ --- \\ b_{m-n+1} \\ \vdots \\ b_n \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} a_0 & & \\ \vdots & \ddots & \\ a_{m-n} & \cdots & a_0 \\ --- & --- & --- \\ a_{m-n+1} & \cdots & a_1 \\ \vdots & \cdots & \vdots \\ a_n & \cdots & a_{2n-m} \\ & \ddots & \vdots \\ & & a_n \end{bmatrix} \begin{bmatrix} q_0 \\ \vdots \\ q_{m-n} \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ --- \\ r_0 \\ \vdots \\ r_{2n-m-1} \\ \vdots \\ r_{n-1} \end{bmatrix} \tag{3.97}$$

deci, cu partiţii evidente, avem

$$b_1 = A_1 q \tag{3.98}$$

$$b_2 = A_2 q + r \tag{3.99}$$

adică

$$r = b_2 - A_2 q. \tag{3.100}$$

Cu alte cuvinte, vectorul q al coeficienţilor polinomului cât se calculează rezolvând sistemul inferior triunghiular nesingular format din primele $m - n + 1$ ecuaţii

$$A_1 q = b_1\,, \quad (a_0 \neq 0), \tag{3.101}$$

după care r se calculează cu (3.100) ca reziduu al sistemului $A_2 q = b_2$, format cu celelalte n ecuaţii.

4^0. **Reprezentarea prin fracţie continuă finită** a unei funţii raţionale este o aplicaţie directă a algoritmului de împărţire întreagă. Fie

$$T(s) = \frac{N(s)}{p(s)} \tag{3.102}$$

strict proprie [9] şi fie k_1 primul coeficient nenul al lui N astfel încât $N(s) = k_1 p_1(s)$ unde $p_1(s)$ e monic şi de grad strict mai mic decât $p(s)$. Avem

$$T(s) = \frac{k_1 p_1(s)}{p(s)} = \frac{k_1}{\frac{p(s)}{p_1(s)}} = \frac{k_1}{q_1(s) + \frac{N_1(s)}{p_1(s)}}, \tag{3.103}$$

[9] Altfel extragem partea întreagă ν_0 şi punem $q_0(s) = \nu_0$.

unde q_1 e câtul iar N_1 e restul împărţirii lui p la p_1, deci

$$T_1(s) \stackrel{\text{def}}{=} \frac{N_1(s)}{p_1(s)}$$

e strict proprie. Putem scrie

$$T(s) = \frac{k_1}{q_1(s) + T_1(s)} = \frac{k_1 \frac{1}{q_1(s)}}{1 + T_1(s)\frac{1}{q_1(s)}}\,. \tag{3.104}$$

Dacă privim funcţia raţională $T(s)$ ca funcţie de transfer a unui sistem simplu atunci putem asocia relaţiei (3.104) schema bloc din figura 3.5. Procedura de mai sus se repetă pentru $T_1(s)$. În general,

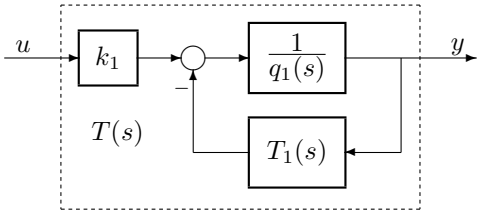

Figura 3.5: Corespondentul sistemic al primei etape de descompunere în fracţie continuă finită a funcţiei raţionale $T(s)$.

$$T_{i-1}(s) \stackrel{\text{def}}{=} \frac{k_i p_i(s)}{p_{i-1}(s)} = \frac{k_i}{q_i + \dfrac{k_{i+1}p_{i+1}(s)}{p_i(s)}}, \quad i = 1, 2, \ldots \tag{3.105}$$

unde $T_0(s) = T(s)$. Deoarece gradele lui p_i scad strict cu i şi $\operatorname{grad} p_0(s) = n < \infty$, procedura are un număr finit de paşi. (În mod analog, fracţia continuă asociată unui număr raţional e finită şi reciproc.)

Calculul reprezentării prin fracţie continuă finită a unei funcţii raţionale date constă în determinarea tripletelor $(k_i,\ q_i(s),\ p_i(s)),\ i = 1, 2, \ldots$ prin aplicarea repetată a algoritmului de împărţire întreagă. Scrierea efectivă a algoritmului este propusă ca exerciţiu cititorului.

5^0. **Calculul celui mai mare divizor comun (cmMdc)** a două polinoame $a(s)$ şi $b(s)$ se face cel mai bine cu algoritmul lui Euclid, deci consti-

tuie o altă aplicaţie directă a algoritmului de împărţire întreagă. Reamintim schema de calcul care descrie algoritmul lui Euclid, unde presupunem $n = \text{grad}\, a(s) \leq \text{grad}\, b(s) = m$ [10].

1. $r(s) = 1$
2. `Cât timp` $r(s) \not\equiv 0$
   1. Se calculează polinoamele $q(s)$ şi $r(s)$ astfel încât $b(s) = a(s)q(s)+r(s)$ utilizând algoritmul de împărţire întreagă.
   2. $b(s) \leftarrow a(s)$
   3. $a(s) \leftarrow r(s)$
3. cmMdc $= b(s)$

De notat faptul că la implementarea schemei de mai sus condiţia din instrucţiunea 2 trebuie relaxată introducându-se o toleranţă corespunzătoare.

6^0. **Calculul celui mai mic multiplu comun (cmmmc)** a două polinoame $a(s)$ şi $b(s)$ se face pe baza relaţiei

$$\text{cmmmc}(a(s), b(s)) = \frac{a(s)b(s)}{\text{cmMdc}(a(s), b(s))}\,. \tag{3.106}$$

7^0. **Calculul rădăcinilor unui polinom** dat prin coeficienţi, i.e. rezolvarea ecuaţiilor polinomiale $p(s) = 0$, este o problema căreia i s-a acordat, de foarte mult timp, o atenţie cu totul deosebită şi pentru care există numeroşi algoritmi ,,candidat", e.g. metoda bisecţiei, metode de interpolare (liniară sau pătratică), metodele Newton, Laguerre, Bairstow, etc. Subliniem, încă odată, că metoda cu cele mai bune performanţe numerice este aplicarea algoritmului **QR** matricelor companion A_c sau A_o din (3.40), respectiv (3.42) (evident, după împărţirea prin coeficientul termenului de grad maxim).

8^0. **Calculul coeficienţilor unui polinom** atunci când se cunosc rădăcinile λ_i, $i = 1 : n$ se face pe baza relaţiei

$$p(s) = \prod_{i=1}^{n}(s - \lambda_i), \tag{3.107}$$

i.e. prin "acumularea" produselor după schema:

[10] Altfel facem $a(s) \leftrightarrow b(s)$.

1. $p(s) = 1$
2. `Pentru` $i = 1 : n$
    1. $p(s) \leftarrow p(s)(s - \lambda_i)$

unde la începutul pasului curent 2.1 $p(s) \stackrel{\text{def}}{=} p_{i-1}(s)$ este un polinom monic de grad $i-1$, iar după execuţia lui un polinom monic $p(s) \stackrel{\text{def}}{=} p_i(s)$ de grad i. Deci

$$\begin{aligned} p_i(s) &\stackrel{\text{def}}{=} s^i + a_1^* s^{i-1} + \cdots + a_i^* = \\ &= (s^{i-1} + a_1 s^{i-2} + \cdots + a_{i-1})(s - \lambda_i) = \\ &= s^i + (a_1 - \lambda_i)s^{i-1} + (a_2 - \lambda_i a_1)s^{i-2} + \cdots + a_i - \lambda_i a_{i-1}. \end{aligned}$$

Obţinem următorul algoritm.

Algoritmul 3.3 (Se dau rădăcinile $\lambda_i \in \mathcal{C},\ i = 1 : n$ ale unui polinom. Algoritmul calculează coeficienţii $a_0 = 1,\ a_i,\ i = 1 : n$ ai acestuia).

1. $a_0 = 1$
2. `Pentru` $k = 1 : n$
    1. $a_k = 0$
3. `Pentru` $i = 1 : n$
    1. `Pentru` $k = i : -1 : 1$
        1. $a_k \leftarrow a_k - \lambda_i a_{k-1}$

9^0. **Evaluarea valorii unui polinom** $p(s)$ sau/şi a derivatei sale pentru o valoare $s = \sigma \in \mathcal{C}$ precizată se face cu *algoritmul lui Horner*. Reamintim acest algoritm, cu precizarea că, în contextul scalar discutat, el este optimal din punctul de vedere al eficienţei.

Algoritmul 3.4 (**Horner**) (Se dau coeficienţii $a_i,\ i = 0 : n$ ai unui polinom $p(s)$, în ordinea descrescătoare a puterilor nedeterminatei, şi numărul $\sigma \in \mathcal{C}$. Algoritmul calculează valoarea $\alpha = p(\sigma)$ a polinomului şi valoarea $\delta = p'(\sigma)$ a derivatei acestuia).

1. $\alpha = a_0$
2. $\delta = na_0$
3. `Pentru` $i = 1 : n - 1$

   1. $\alpha \leftarrow \alpha\sigma + a_i$
   2. $\delta \leftarrow \delta\sigma + (n-i)a_i$

4. $\alpha \leftarrow \alpha\sigma + a_n$

Algoritmul Horner generalizat care rezolvă problema calculului coeficienţilor polinomului $p(s+\sigma)$ face obiectul exerciţiului 3.15.

Observaţia 3.7 Aşa cum s-a mai precizat în capitolul 2, algoritmul Horner este util şi pentru evaluarea polinoamelor matriceale. Într-un astfel de context algoritmul nu mai este însă optimal din punctul de vedere al numărului de operaţii aritmetice scalare (vezi [VI] şi exerciţiul 3.16). $\diamond$

Programe MATLAB disponibile

Pentru realizarea unor conexiuni specifice sunt disponibile funcţiile **append** (care efectuează produsul direct), **parallel**, **series**, **feedback** şi **cloop** (sistem cu reacţie unitară). Conexiunile de tip general se realizează apelând **bklbuild** şi **conect**. Pentru construcţia realizării de stare în forma standard controlabilă a unei matrice de transfer cu o singură intrare este disponibilă funcţia **tf2ss** ("**t**ransfer **f**unction **t(w)o**(!) **s**tate **s**pace"). Conversia inversă se efectuează cu funcţia **ss2tf**. Numele celorlalte funcţii de conversie se formează similar, ţinându-se seama că pentru codificarea formei ($\mathbf{T}_1$) este folosită sigla **zp** ("zerouri – poli"). Calculul descompunerii în fracţii simple (forma ($\mathbf{T}_2$)) se poate face cu funcţia **residue**. Transformările nesingulare de stare (3.3) se pot calcula cu ajutorul funcţiei **ss2ss**.

Pentru implementarea algoritmilor de calcul polinomial din § 3.5, menţionăm în primul rând convenţia MATLAB de reprezentare a polinoamelor prin vectorii *linie* ai coeficienţilor în ordinea descrescătoare a puterilor nedeterminatei (primul element are indicele 1). Zerourile (rădăcinile) unui polinom se reprezintă prin vectori coloană. Adunarea polinoamelor se face prin adunarea vectorilor coeficienţilor după egalarea dimensiunilor prin completarea lor corespunzătoare cu zerouri. Pentru înmulţire se utilizează funcţia **conv** iar pentru împărţire întreagă **deconv**. Funcţia **roots** calculează rădăcinile aplicând algoritmul **QR** matricei companion iar funcţia **poly** calculează coeficienţii din rădăcini (pentru argument matriceal calculează polinomul caracteristic). În sfârşit, calculul valorii unui polinom, inclusiv pentru argument matriceal, se efectuează cu funcţia **polyval**.

Exerciţii

E 3.1 Se dau două sisteme $S_i = (A_i, B_i, C_i, D_i)$, $i = 1, 2$. Să se scrie algoritmii de construcţie a matricelor sistemelor $S = (A, B, C, D)$, rezultate prin interconectarea paralel, serie şi în circuit închis a sistemelor date.

Indicaţie. Se utilizează relaţiile (3.16), (3.21) şi (3.28).

E 3.2 Se dau trei sisteme $S_i = (A_i, B_i, C_i, D_i)$, $i = 1 : 3$. Să se scrie algoritmii de construcţie a matricelor sistemelor $S = (A, B, C, D)$, rezultate prin interconectarea sistemelor date conform schemelor din figura 3.6.

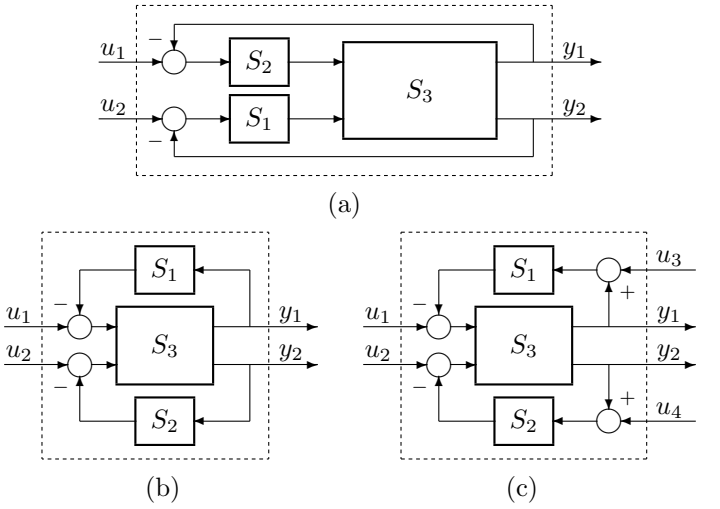

Figura 3.6: Conexiuni pentru exerciţiul 3.2.

Cum procedaţi dacă sistemele S_i, $i = 1 : 3$ sunt date prin reprezentările lor de transfer ?

E 3.3 Se dă sistemul $S = (A, B, C, D)$ ale cărui matrice au structura

$$A = \begin{bmatrix} A_1 & 0 \\ A_{21} & A_2 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ G_2 \end{bmatrix},$$

$$C = \begin{bmatrix} H_1 & C_2 \end{bmatrix}, \quad D = \begin{bmatrix} D \end{bmatrix}.$$

Să se scrie o procedură de reprezentare a lui S sub forma unei conexiuni a) paralel

b) serie
a două sisteme S_1, S_2, *convenabil* definite.

Indicaţie. a) Cu notaţia (3.13), consideraţi transformarea $\tilde{x}_2 = x_2 + Vx_1$ şi determinaţi matricea V astfel încât $\tilde{A}_{21} = 0$. În ce condiţii acest lucru este posibil?
b) Cu referire la (3.21), consideraţi factorizarea

$$\begin{bmatrix} A_{21} & G_2 \\ H_1 & D \end{bmatrix} = \begin{bmatrix} B_2 \\ D_2 \end{bmatrix} \begin{bmatrix} C_1 & D_1 \end{bmatrix}.$$

Pentru calculul factorizării utilizaţi **DVS**. Prin urmare numărul minim al terminalelor comune $l_1 = m_2$ coincide cu rangul matricei din membrul stâng.

E 3.4 Se dă o matrice de transfer $T(s)$, nu neapărat strict proprie, ale cărei coloane (linii) sunt aduse la acelaşi numitor comun. Să se scrie un program MATLAB de construcţie a unei realizări de stare $S = (A, B, C, D)$ a lui $T(s)$ utilizând structura (3.68) (respectiv (3.69)).

E 3.5 Se dă o matrice de transfer $T(s)$, nu neapărat strict proprie, sub forma $T(s) = N(s)/p(s)$, unde $p(s)$ este un polinom de grad n iar $N(s) \in \mathcal{R}^{l\times m}[s]$. Scrieţi prin inspecţie două realizări ale lui $T(s)$. Precizaţi ce relaţie există între ordinul n al acestor realizări şi ordinele n_I, n_{II} obţinute în secţiunea 3.3D.

Indicaţie. Realizările cerute se obţin generalizând convenabil formele standard (**C**) şi (**O**).

E 3.6 Se dă un sistem $S = (A, B, C, D)$ precum şi doi întregi $i \in 1 : l$, $j \in 1 : m$. Să se scrie algoritmii care calculează funcţia de transfer T_{ij} în formele (**T**), ($\mathbf{T}_1$) şi ($\mathbf{T}_2$).

Indicaţie. Vezi procedurile de conversie (**S**) $\rightarrow$ (**T**), (**S**) $\rightarrow$ ($\mathbf{T}_1$) şi (**S**) $\rightarrow$ ($\mathbf{T}_2$) din § 3.4.

E 3.7 Se dă o funcţie de transfer $T(s)$, nu neapărat strict proprie. Să se scrie şi să se discute comparativ câţiva algoritmi de conversie (**T**) $\rightarrow$ ($\mathbf{T}_1$) şi (**T**) $\rightarrow$ ($\mathbf{T}_2$).

E 3.8 Analizaţi posibilităţile de construcţie a unei realizări de stare a unei funcţii de transfer $T(s)$ date în forma ($\mathbf{T}_1$), *fără a construi*, în prealabil, reprezentarea raţională a lui $T(s)$.

E 3.9 Să se scrie, la nivel de operaţii scalare, algoritmul de împărţire întreagă a două polinoame.

E 3.10 Utiliând algoritmul lui Euclid de calcul al cmMdc a două polinoame, scrieţi o procedură de calcul a cmMdc a l polinoame.

Indicaţie. Operaţia $p_1 \circ p_2 = \text{cmMdc}(p_1, p_2)$ este asociativă.

E 3.11 Utiliând algoritmii stabiliţi în exerciţiile precedente scrieţi o procedură de calcul a cmmmc a l polinoame $p_i(s)$, $l \geq 2$.

E 3.12 Se dau l funcţii raţionale, fiecare definită ca raport de două polinoame. Să se scrie algoritmul de aducere a celor l funcţii la acelaşi numitor comun [11].

E 3.13 Să se scrie algoritmul de adunare a două sau mai multe funcţii (matrice) raţionale [12].

E 3.14 Să se scrie algoritmul de înmulţire a două sau mai multe funcţii (matrice) raţionale [13].

E 3.15 Să se scrie algoritmul (lui Horner generalizat) de calcul al coeficienţilor polinomului $p(s + \sigma)$, unde $p(s)$ este dat prin vectorul coeficienţilor iar σ este un scalar dat, utilizând un volum minim de memorie. Elaboraţi o procedură de aplicare a algoritmului "cu hârtia şi creionul".

Indicaţie. Toate calculele se pot desfăşura în locaţiile de memorie asociate vectorului coeficienţilor polinomului iniţial.

E 3.16 Se dau un polinom $p(s)$ de grad m şi o matrice $A \in \mathcal{R}^{n \times n}$. Să se scrie un algoritm de calcul al matricei $B = p(A)$. Căutaţi un exemplu care arată că în cazul matriceal schema lui Horner nu este optimală din punctul de vedere al numărului de operaţii.

E 3.17 Să se studieze problema inversării unei matrice de transfer (pătrate) $T(s)$, precizând condiţiile în care inversa este tot o matrice de transfer. Formulaţi aceeaşi problemă relativ la o realizare $S = (A, B, C, D)$ a lui $T(s)$ şi comentaţi.

E 3.18 Ce puteţi afirma relativ la problema anterioară în cazul $l \neq m$?

[11] Vezi procedurile de realizare "eficiente".
[12] Vezi conexiunea paralel.
[13] Vezi conexiunea serie.

Bibliografie

[1] **Ionescu V.** Teoria Sistemelor. Sisteme Liniare. EDP, Bucureşti 1985.

[2] **Varga A., Sima V.** *Numerically Stable Algorithm for Transfer Function Matrix Evaluation*, Int J. Control, Vol. 33, pp. 1123 – 1133, 1981.

[3] **Emami A., Naeini A., Van Dooren P.** *Computation of Zeros of Linear Multivariable Systems*, Automatica, Vol. 4, pp. 415 – 430, 1982.

[4] **Barnett S.** *Matrices, Polynomials and Linear Time - Invariant Systems*, IEEE Trans. AC-18, No. 1, pp. 1 – 10, 1973.

Capitolul 4

Calculul răspunsului în timp al sistemelor liniare

Analiza comportării sistemelor dinamice include, în mod necesar, determinarea evoluţiei temporale a ieşirilor (şi, eventual, a stării) la diverse tipuri de intrări externe. În faza de concepţie a unui sistem, astfel de determinări nu pot fi realizate decât într-un mediu de simulare în care testele se fac sau pe modele la scară sau prin rezolvarea ecuaţiilor ce descriu sistemul respectiv. Un mediu de simulare pentru sisteme cu timp continuu ce foloseşte tehnica numerică de calcul presupune, în mod obligatoriu, o discretizare a timpului şi calculul răspunsului în momentele de timp discret corespunzătoare. În acest scop se utilizează o procedură de discretizare adecvată a cărei alegere poate influenţa în mod determinant precizia rezultatelor.

4.1 Răspunsul liber al sistemelor liniare

Problema determinării răspunsului liber (la intrare nulă) al unui sistem liniar $S = (A, B, C, D)$ constă în calculul funcţiei de ieşire $y(t)$ definită prin

$$\begin{cases} \dot{x}(t) = Ax(t), & x(0) = x \\ y(t) = Cx(t), \end{cases} \tag{4.1}$$

pe un interval fixat $[0,\, t_f]$.

Soluţia ecuaţiei diferenţiale omogene (4.1) este

$$x(t) = e^{tA}x(0), \tag{4.2}$$

unde, prin definiţie, $\Phi(t) = e^{tA}$ este matricea de tranziţie a sistemului considerat. Prin urmare, ţinând seama de ecuaţia ieşirii, răspunsul liber are expresia simplă

$$y(t) = Ce^{tA}x(0).$$

Pentru a evalua numeric în mod eficient valorile $y(t)$, divizăm intervalul $[\,0,\, t_f\,]$ într-un număr N de subintervale egale, i.e. considerăm $t_f = Nh$, unde h este pasul de simulare, utilizat pentru calculul valorilor dorite $y(kh)$, $k = 0 : N-1$, ale răspunsului, iar N este numărul de paşi necesar pentru acoperirea intervalului prescris. Scriind relaţia (4.2) la două momente de timp discret $t = kh$ şi $t = (k+1)h$, obţinem evident

$$\left\{ \begin{array}{rl} x((k+1)h) & = e^{hA}x(kh) \\ y(kh) & = Cx(kh)\,, \end{array} \right. \tag{4.3}$$

adică $y(kh)$ rezultă direct în funcţie de $x(kh)$ care se determină recurent pentru $k = 0 : N-1$, ţinând seama că stare iniţială $x(0) = x$ este cunoscută. Prin urmare, totul se reduce la calculul lui $F = e^{hA}$, utilizând, de exemplu, aproximaţia Padé. De fapt, sistemul (4.3) constituie reprezentarea discretizată exactă – cu element de extrapolare de ordinul 0 – a sistemului (4.1) (vezi § 4.3).

Observaţia 4.1 Din acest punct de vedere, dacă pasul de simulare h nu este impus, atunci el se poate alege egal cu pasul de discretizare, utilizând condiţia cunoscută $h\|A\| < \frac{1}{2}$. Pe de altă parte, dacă t_f nu este nici el precizat dar se urmăreşte obţinerea regimului tranzitoriu "esenţial", atunci – în ipoteza că matricea A este stabilă cu valorile proprii satisfăcând condiţia $\mathrm{Re}\,\lambda(A) < -\alpha$, unde $\alpha > 0$ – putem evalua constanta de timp dominantă prin

$$T = \frac{1}{\alpha}$$

iar timpul de răspuns (pentru banda de 5% din jurul regimului staţionar constant) prin

$$t_f = (4 \div 5)\, T = \frac{4 \div 5}{\alpha}.$$

În definitiv, un N ales astfel încât

$$Nh \approx \frac{4 \div 5}{\alpha}\,.$$

asigură dezideratul propus. (În legătură cu aceasta, vezi şi observaţia 5.1). ◇

În concluzie, convenind să memorăm valorile $y_k \stackrel{\text{def}}{=} y(kh)$ într-o matrice Y de forma $l \times N$ astfel încât $Y(:,k) = y_{k-1},\ k = 1:N$, calculul răspunsului liber al sistemelor liniare se poate face utilizând următoarea procedură.

Algoritmul 4.1 (Se dau sistemul liber (A, C), starea inițială $x(0) = x$, intervalul de simulare $[0,\ t_f]$ precum şi parametrii h, N astfel încât $Nh = t_f$. Se calculează răspunsul liber Y în momentele $t_k = kh,\ k = 0 : N-1$ şi se returnează $x = x(t_f)$.)

1. Se calculează $F = e^{hA}$.
2. `Pentru` $k = 1 : N$
   1. $Y(:,k) = Cx$
   2. $x \leftarrow Fx$

Comentarii. Starea finală $x(t_f) = x$, furnizată de algoritmul 4.1, este necesară în calitatea de nouă stare inițială pentru o eventuală continuare a simulării pe un interval de timp $[\,t_f,\ t'_f\,]$, ulterior lui t_f. $\diamond$

4.2 Răspunsul la intrări liniar generate

Problema determinării răspunsului unui sistem liniar $S = (A, B, C, D)$ constă în calculul funcţiei de ieşire $y(t)$, definită prin

$$(S) \quad \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x \\ y(t) = Cx(t) + Du(t), \end{cases} \tag{4.4}$$

pe un interval fixat $[0,\ t_f]$, pe care funcţia de intrare $u(t)$ se consideră cunoscută.

Soluţia ecuaţiei diferenţiale (4.4) este

$$x(t) = e^{tA}x(0) + \int_0^t e^{(t-\tau)A}Bu(\tau)\,d\tau. \tag{4.5}$$

În cazul unei intrări de tip impuls $u(t) = u_0\delta(t)$, $u_0 \in \mathcal{R}^m$ (v. fig. 4.1 a)) se obţine

$$x(t) = e^{tA}x(0) + e^{tA}Bu_0, \tag{4.6}$$

respectiv

$$y(t) = Ce^{tA}x(0) + \left[\, Ce^{tA}B + D\delta(t) \,\right] u_0, \tag{4.7}$$

deci răspunsul poate fi calculat (mai puţin termenul impulsiv $Du_0\delta(t)$) utilizând algoritmul 4.1 cu datele A, C şi $x = x(0) \leftarrow x(0) + Bu_0$, unde Bu_0

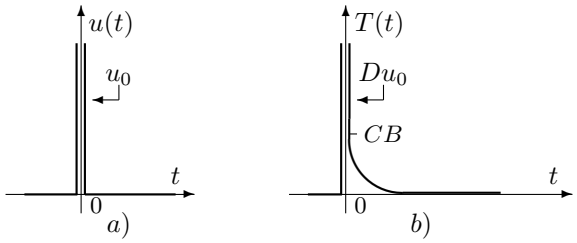

Figura 4.1: Funcţie de intrare tip impuls (a) şi funcţia pondere a unui sistem liniar (b).

este saltul traiectoriei de stare (4.6) datorat impulsului aplicat la intrare în momentul $t = 0$.

Totodată, din (4.7) rezultă că matricea pondere (v. fig. 4.1 b))

$$T(t) \stackrel{\text{def}}{=} Ce^{tA}B\,1(t) + D\delta(t), \tag{4.8}$$

poate fi calculată pe coloane determinând succesiv (prin metoda indicată mai sus) răspunsurile sistemului în stare iniţială nulă $x(0) = 0$ la impulsurile unitate $u_0 = e_i \in \mathcal{R}^m$, $i = 1 : m$. Odată $T(t)$ cunoscut, răspunsul la o intrare $u(t)$ oarecare, e.g. continuă pe porţiuni, poate fi, în principiu, obţinut utilizând relaţia

$$y(t) = Ce^{tA}x(0) + \int_0^t T(t-\tau)u(\tau)\,d\tau. \tag{4.9}$$

Deoarece nu există metode generale de reprezentare a unei funcţii de intrare $u(t)$ „arbitrare", iar evaluarea numerică a integralei din (4.9) este dificilă [1], în cele ce urmează vom considera situaţia (particulară, dar frecvent întâlnită în aplicaţii) în care $u(t)$ este *liniar generată*, adică poate fi modelată ca ieşire a unui sistem liniar liber (S_F) de forma

$$(S_F) \quad \begin{cases} \dot{x}_F(t) = A_F x_F(t), & x_F(0) = x_F \\ u(t) = C_F x_F(t), \end{cases} \tag{4.10}$$

unde A_F, C_F sunt matrice date, iar starea iniţială $x_F \in \mathcal{R}^{n_F}$ este un vector arbitrar, dar fixat. (Altfel spus, precizând x_F fixăm implicit intrarea în

[1] Câteva metode de reprezentare aproximativă a unor funcţii de intrare de forma generală, precum şi metodele corespunzătoare de calcul al răspunsului, sunt prezentate într-o altă secţiune a acestui capitol.

clasa tuturor intrărilor $u(t)$ ce pot fi obţinute ca ieşiri ale lui S_F). Sistemul S_F se numeşte *generator* sau *filtru de formare* al funcţiei de intrare u. Conectând sistemele (S) şi (S_F) în serie putem scrie

$$(S^0)\ \begin{cases} \begin{bmatrix} \dot{x}(t) \\ \dot{x}_F(t) \end{bmatrix} = \begin{bmatrix} A & BC_F \\ 0 & A_F \end{bmatrix} \begin{bmatrix} x(t) \\ x_F(t) \end{bmatrix}, \qquad \begin{bmatrix} x(0) \\ x_F(0) \end{bmatrix} = \begin{bmatrix} x \\ x_F \end{bmatrix} \\ \\ y(t) = \begin{bmatrix} C & DC_F \end{bmatrix} \begin{bmatrix} x(t) \\ x_F(t) \end{bmatrix}, \end{cases} \tag{4.11}$$

i.e. calculul răspunsului sistemului (S) la intrarea liniar generată $u(t)$ se reduce la calculul răspunsului liber al sistemului (S^0), prin metoda (exactă) expusă în paragraful anterior.

Având în vedere că aplicând transformarea Laplace ecuaţiilor (4.10) ale sistemului (S_F) obţinem

$$u(s) = C_F(sI - A_F)^{-1}x_F\,, \tag{4.12}$$

reţeta generală de construcţie a sistemului generator (S_F) constă în realizarea transformatei $u(s)$, interpretată ca funcţie de transfer a unui sistem SIMO, sub forma unui triplet (A_F, b_F, C_F) cu o singură intrare, unde vectorul $b_F = x_F$ este stare iniţială a lui (S_F). De aici rezultă că funcţia de intrare $u(t)$ este liniar generată dacă şi numai dacă transformata sa Laplace este o matrice raţională strict proprie.

În concluzie, calculul răspunsului sistemelor liniare la intrări liniar generate se poate face utilizând următoarea procedură.

Algoritmul 4.2 (Se dau sistemul (A, B, C, D), starea iniţială $x(0) = x$, intervalul de simulare $[\,0,\ t_f\,]$, parametrii $h,\ N$ astfel încât $Nh = t_f$, precum şi transformata $u(s)$ sub forma matricei de transfer a unui sistem SIMO. Se calculează răspunsul Y în momentele $t_k = kh,\ k = 0 : N-1$ şi se returnează $x = x(t_f)$.)

1. Se realizează $u(s)$ sub forma (A_F, b_F, C_F).
2. Se construiesc matricele

$$A \leftarrow A^0 = \begin{bmatrix} A & BC_F \\ 0 & A_F \end{bmatrix}, \quad x \leftarrow x^0 = \begin{bmatrix} x \\ b_F \end{bmatrix}$$

$$C \leftarrow C^0 = \begin{bmatrix} C & DC_F \end{bmatrix}.$$

3. Se aplică algoritmul 4.1 de calcul al răspunsului liber cu datele de intrare A, C şi $x(0) = x$.

4. $x = x(1:n)$.

Pentru exemplificare, vom considera câteva cazuri particulare importante.

A. Intrări polinomiale

Intrările polinomiale sunt de forma

$$u(t) = \sum_{k=1}^{p} \alpha_k t^{k-1}\,, \quad t \geq 0\,, \tag{4.13}$$

unde coeficienţii $\alpha_k \in \mathcal{R}^m$ sunt vectori daţi. Notând $u_k \stackrel{\text{def}}{=} \alpha_k(k-1)!$ avem

$$u(t) = \sum_{k=1}^{p} u_k \frac{t^{k-1}}{(k-1)!}\,, \quad t \geq 0\,, \tag{4.14}$$

iar transformata Laplace corespunzătoare este

$$u(s) = \sum_{k=1}^{p} \frac{u_k}{s^k} = \frac{u_1 s^{p-1} + \cdots + u_p}{s^p}\,. \tag{4.15}$$

Deci, o realizare de stare a lui (S_F), în forma standard controlabilă, (vezi capitolul 3) este

$$\begin{aligned} A_F &= \begin{bmatrix} 0 & \cdots & 0 & 0 \\ 1 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}, \qquad b_F = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ C_F &= \begin{bmatrix} u_1 & \cdots & u_{p-1} & u_p \end{bmatrix}. \end{aligned} \tag{4.16}$$

În particular, dacă $p = 1$ atunci $u(t) = u_1\, 1(t)$ sau

$$u(t) = u_1, \quad t \geq 0 \tag{4.17}$$

este o intrare *treaptă* de amplitudine $u_1 \in \mathcal{R}^m$ dată (v. fig. 4.2 a)). În acest caz filtrul de formare este

$$\begin{aligned} A_F &= [\,0\,], \quad b_F = [\,1\,] \\ C_F &= [\,u_1\,] \end{aligned} \tag{4.18}$$

iar matricele sistemului (S^0) sunt

$$\begin{aligned} A^0 &= \begin{bmatrix} A & Bu_1 \\ 0 & 0 \end{bmatrix}, \quad x^0 = \begin{bmatrix} x \\ 1 \end{bmatrix} \\ C^0 &= \begin{bmatrix} C & Du_1 \end{bmatrix}. \end{aligned} \tag{4.19}$$

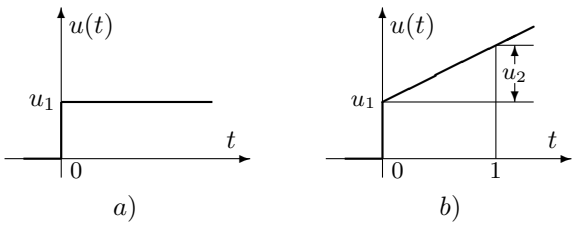

Figura 4.2: Funcţiile de intrare tip treaptă (a) şi rampă (b).

În mod analog, dacă $p = 2$ atunci

$$u(t) = u_1 + u_2 t, \quad t \geq 0 \tag{4.20}$$

este o intrare *rampă* de amplitudine iniţială $u(0) = u_1$ şi pantă $u_2 \in \mathcal{R}^m$ date (v. fig. 4.2 b)). În acest caz filtrul de formare este

$$\begin{aligned} A_F &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad b_F = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ C_F &= \begin{bmatrix} u_1 & u_2 \end{bmatrix}. \end{aligned} \tag{4.21}$$

iar matricele sistemului (S^0) sunt

$$\begin{aligned} A^0 &= \begin{bmatrix} A & Bu_1 & Bu_2 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad x^0 = \begin{bmatrix} x \\ 1 \\ 0 \end{bmatrix} \\ C^0 &= \begin{bmatrix} C & Du_1 & Du_2 \end{bmatrix}. \end{aligned} \tag{4.22}$$

Clasa tuturor intrărilor polinomiale de forma (4.14), în care vectorii $u_k,\ k = 1 : p$ sunt arbitrari, se obţine considerând în loc de (4.16) forma

standard observabilă

$$A_F = \begin{bmatrix} 0 & I_m & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & I_m \\ 0 & 0 & \cdots & 0 \end{bmatrix}, \quad b_F = \begin{bmatrix} u_1 \\ \cdots \\ u_{p-1} \\ u_p \end{bmatrix}$$
$$C_F = \begin{bmatrix} I_m & 0 & \cdots & 0 \end{bmatrix}, \tag{4.23}$$

unde vectorul $b_F = x_F$ se alege pentru a preciza intrarea în clasa considerată. (Observăm că, acum, S_F este de ordin $n_F = pm$, unde p este ordinul transformatei Laplace (4.15), iar m este numărul de intrări ale sistemului (4.4)).

În particular, dacă $u(t) = u_1\, 1(t)$ este o intrare treaptă de amplitudine arbitrară, atunci filtrul de formare este

$$\begin{array}{l} A_F = [\,0\,], \quad b_F = [u_1] \\ C_F = [I_m], \end{array} \tag{4.24}$$

iar matricele sistemului (S^o) sunt

$$A^o = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad x^o = \begin{bmatrix} x \\ u_1 \end{bmatrix}$$
$$C^o = \begin{bmatrix} C & D \end{bmatrix}. \tag{4.25}$$

B. Intrări armonice

Intrările armonice sunt de forma

$$u(t) = \gamma_1 \cos \omega t + \gamma_2 \sin \omega t\,, \quad t \geq 0\,, \tag{4.26}$$

unde pulsaţia ω este fixată iar coeficienţii γ_1, $\gamma_2 \in \mathcal{R}^m$ sunt vectori daţi. Avem

$$u(s) = \frac{\gamma_1 s + \gamma_2 \omega}{s^2 + \omega^2}\,, \tag{4.27}$$

deci o realizare a lui (S_F) este

$$A_F = \begin{bmatrix} 0 & -\omega^2 \\ 1 & 0 \end{bmatrix}, \quad b_F = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
$$C_F = \begin{bmatrix} \gamma_1 & \omega\gamma_2 \end{bmatrix}. \tag{4.28}$$

Cazurile relativ mai complicate, de tipul

$$u = \sum_{k=1}^{p} \gamma_k \cos k\omega t + \delta_k \sin k\omega t\,, \tag{4.29}$$

$$u = e^{\alpha t}(\gamma_1 \cos \omega t + \gamma_2 \sin \omega t)\,, \tag{4.30}$$

în care coeficienţii sunt eventual arbitrari, se tratează analog.

4.3 Răspunsul la intrări etajate

Domeniul de aplicaţie al metodei bazate pe utilizarea sistemului generator, expusă în paragraful anterior, poate fi sensibil lărgit, considerând că funcţia de intrare $u(t)$ este *liniar generată pe porţiuni*, e.g. are o expresie de tip (4.14) pe fiecare subinterval $[\,kh,\,(k+1)h\,)$, $k = 0 : N-1$ al unei diviziuni uniforme cu pasul h al intervalului $[\,0,\,t_f\,]$.

În cel mai simplu caz (care în acelaşi timp este şi cel mai frecvent întâlnit în aplicaţii) funcţia de intrare $u(t)$ este constantă pe porţiuni (sau *etajată*), i.e. avem

$$u(t) = u_k, \quad t \in [\,kh,\,(k+1)h\,) \tag{4.31}$$

unde amplitudinile $u_k \in \mathcal{R}^m$ sunt vectori daţi, în general variabili de la pas la pas (v. fig. 4.3).

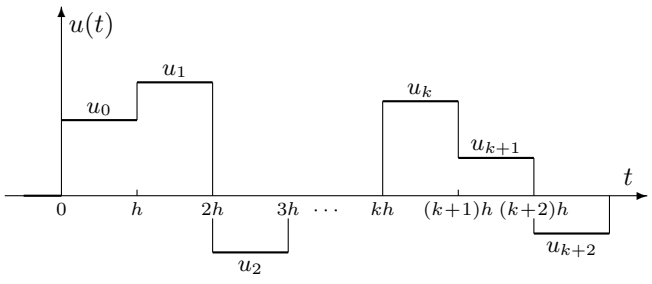

Figura 4.3: Funcţie de intrare etajată.

Utilizând relaţia (4.6) scrisă sub forma

$$x(t) = e^{(t-kh)A}x(kh) + \int_{kh}^{t} e^{(t-\tau)A}Bu(\tau)\,d\tau,$$

în care punem $t = (k+1)h$, în virtutea lui (4.31) obţinem uşor

$$\begin{cases} x((k+1)h) = & Fx(kh) + Gu_k, \\ y(kh) = & Cx(kh) + Du_k, \end{cases} \tag{4.32}$$

unde

$$F = e^{hA}, \quad G = \int_0^h e^{\eta A}\, d\eta\, B. \tag{4.33}$$

Prin definiţie, sistemul liniar discret $S_d = (F, G, C, D)$ constituie *reprezentarea discretizată* (pe scurt discretizatul) cu pasul h al sistemului liniar $S = (A, B, C, D)$. Matricele F şi G, necesare pentru construcţia lui S_d, se determină calculând exponenţiala matriceală $F^o = e^{hA^o}$, unde matricea A^o are expresia (4.25), corespunzătoare cazului în speţă. (Amintim că, pe fiecare subinterval, funcţia de intrare (4.31) este de tip treaptă). Procedând în acest mod, se obţine

$$e^{hA^o} = \begin{bmatrix} F & G \\ 0 & I_m \end{bmatrix}. \tag{4.34}$$

În concluzie, convenind să dispunem valorile date u_k într-o matrice U de forma $m \times N$, astfel încât $U(:,k) = u_{k-1}$, $k = 1:N$, calculul răspunsului sistemelor liniare la intrări etajate se poate face utilizând următoarea procedură

Algoritmul 4.3 (Se dau sistemul (A, B, C, D), starea iniţială $x(0) = x$, intervalul de simulare $[\,0,\, t_f\,]$, parametrii h, N astfel încât $Nh = t_f$ precum şi tabloul U ce conţine palierele funcţiei de intrare etajate $u(t)$. Se calculează răspunsul Y în momentele $t_k = kh$, $k = 0:N-1$ şi se returnează $x = x(t_f)$).

1. Se calculează $F = e^{hA}$ şi G în acord cu (4.34), unde A^o are expresia (4.25).
2. `Pentru` $k = 1:N$
   1. $Y(:,k) = Cx + DU(:,k)$
   2. $x \leftarrow Fx + GU(:,k)$.

Idei asemănătoare se aplică în cazul utilizării unor reprezentări ale funcţiei de intrare $u(t)$, mai complicate decât (4.31). În cazul general, corespunzător lui (4.14), putem scrie

$$u(t) = \sum_{i=1}^{p} u_k^{(i)} \frac{(t-kh)^{i-1}}{(i-1)!}\,, \quad t \in [\,kh,\, (k+1)h\,) \tag{4.35}$$

unde coeficienţii $u_k^{(i)}$ au o semnificaţie evidentă, în particular $u_k^{(1)} = u(kh)$ coincide cu u_k din (4.31). În locul lui (4.32), acum se obţine o realizare discretizată de forma

$$\begin{cases} x((k+1)h) = \ Fx(k) \ + \sum_{i=1}^{p} G_i u_k^{(i)} \\ \qquad\qquad y(kh) = \ Cx(kh) + Du_k^{(1)}, \end{cases} \tag{4.36}$$

în care matricele $F = e^{hA}$ şi

$$G_i \stackrel{def}{=} \int_0^h e^{(h-\tau)A} \frac{\tau^{i-1}}{(i-1)!}\, d\tau \ , \quad i = 1:p$$

se determină în mod corespunzător, calculând exponenţiala matriceală $F^o = e^{hA^o}$, unde A^o se construieşte cu datele din (4.23). (Stabilirea formei lui F^o, în cazul de faţă, este propusă cititorului ca exerciţiu. Evident, avem $G_1 = G$ şi în cazul $p = 1$ ecuaţiile (4.36) se reduc la (4.32)).

Pentru determinarea coeficienţilor $u_k^{(i)}$, în special dacă asigurarea ,,racordării" aproximaţiilor (4.35) pe intervale adiacente este importantă, se recomandă utilizarea metodelor de interpolare tip spline, [IX]. Subliniem că în acest caz, sistemul (4.36) nu este cauzal în sensul că, în general, coeficienţii $u_k^{(i)}$, deci starea $x((k+1)h)$ rezultată din prima ecuaţie (4.36), depind de valorile viitoare ale intrării $u(t)$ [2]. În schimb, procedurile de calcul astfel obţinute (net superioare tehnicilor de extrapolare clasice) combină precizia aproximărilor spline cu eficienţa şi siguranţa metodelor de calcul ale exponenţialei matriceale, concurând deseori cu succes procedurile alternative de tip general, bazate pe integrarea numerică a ecuaţiei diferenţiale (4.4).

Fără a intra în detalii, menţionăm numai că utilizarea metodelor de integrare numerică (tip Runge - Kutta, predictor - corector, etc., în general cu pas variabil) devine obligatorie în cazul general, în care matricea A a sistemului liniar $S = (A, B, C, D)$ este variabilă în timp. În locul exponenţialei matriceale $F = e^{hA}$, pe fiecare subinterval $[\, t_k,\, t_{k+1}\,)$, acum se utilizează matricea de tranziţie $F_k = \Phi(t_{k+1}, t_k)$, calculată pe coloane prin integrarea ecuaţiei diferenţiale omogene (4.1) cu condiţiile iniţiale $x(t_k) = e_j \in \mathcal{R}^n$, $j = 1:n$.

[2] Acest fenomen poate fi evitat numai în cazul $p = 2$, când în (4.35) se obţine $u_k^{(1)} = u_k$ şi $u_k^{(2)} = (u_{k+1} - u_k)/h$. Efectuând în (4.36) schimbarea de variabilă de stare $x((k+1)h) \leftarrow x((k+1)h) - G_2 u_{k+1}/h$, ecuaţiile (4.36) pot fi scrise sub forma uzuală (4.32). În cazul $p = 4$, corespunzător funcţiilor spline cubice, frecvent utilizate în practică, acest artificiu nu mai este eficient.

4.4 Răspunsul staţionar

Considerăm contextul din § 4.2, unde presupunem, în plus, că matricea A este stabilă, adică $\mathrm{Re}\,\lambda(A) < 0$, iar funcţia de intrare u este *persistentă*, adică $\mathrm{Re}\,\lambda(A_F) \geq 0$.

Procedura generală de separare a componentelor *tranzitorie* şi *„staţionară"* (*asimptotică* sau *permanentă*) ale răspunsului $y(t)$ constă în a scrie

$$x(t) = \xi(t) + Vx_F(t), \tag{4.37}$$

unde $\xi(t)$ este componenta *tranzitorie* a traiectoriei de stare, adică $\xi(t) \to 0$ când $t \to \infty$, iar matricea constantă $V \in \mathcal{R}^{n\times n_F}$ trebuie determinată. Derivând relaţia (4.37) şi ţinând seama de ecuaţiile sistemelor (S) şi (S_F), obţinem

$$\begin{aligned} \dot{x} &= \dot{\xi} + V\dot{x_F} = \dot{\xi} + VA_Fx_F = \\ &= Ax + Bu = \\ &= A(\xi + Vx_F) + BC_Fx_F = \\ &= A\xi + (AV + BC_F)x_F. \end{aligned} \tag{4.38}$$

Prin urmare, dacă matricea V este aleasă astfel încât

$$VA_F = AV + BC_F \tag{4.39}$$

atunci din (4.38) rezultă $\dot{\xi}(t) = A\xi(t)$, deci $\xi(t) \to 0$ când $t \to \infty$, $\forall \xi(0)$ în virtutea ipotezei de stabilitate $\mathrm{Re}\,\lambda(A) < 0$. (În fapt, dacă $x(0)$ este precizat, atunci din (4.37) avem $\xi(0) = x(0) - Vx_F(0)$).

Cu alte cuvinte, considerând un moment iniţial flotant $t_0 \in \mathcal{R}$, presupunând mai sus $t \geq t_0$ şi făcând $t_0 \to -\infty$, obţinem din nou $\xi(t) \to 0$, deci $x(t) \to Vx_F(t)$, $\forall x(t_0)$. În virtutea acestui fapt, spunem că $Vx_F(t)$ este componenta *„staţionară"* (*asimptotică* sau *permanentă*) a traiectoriei de stare $x(t)$.

Observaţia 4.2 După cum se ştie, ecuaţia matriceală Sylvester (4.39) are soluţie unică V oricare ar fi membrul drept BC_F dacă şi numai dacă $\lambda(A) \cap \lambda(A_F) = \emptyset$, (vezi capitolul 1) adică nici o valoare proprie a lui A nu coincide cu nici o valoare proprie a lui A_F. În cazul de faţă această condiţie este evident satisfăcută în virtutea ipotezelor de stabilitate $\mathrm{Re}\,\lambda(A) < 0$ şi persistenţă $\mathrm{Re}\,\lambda(A_F) \geq 0$. $\diamond$

În continuare obţinem

$$y(t) = C(\xi(t) + Vx_F(t)) + DC_Fx_F(t)\,, \tag{4.40}$$

deci avem

$$y(t) = \eta(t) + Wx_F(t)\,, \tag{4.41}$$

unde

$$\eta(t) = C\xi(t) \tag{4.42}$$

este (prin definiţie) componenta *tranzitorie* a răspunsului, iar matricea W este definită prin

$$W = CV + DC_F\,. \tag{4.43}$$

Având în vedere că, în esenţă, $x_F(t) = e^{tA_F}x_F(0)$ este cunoscut, componenta *staţionară* $Wx_F(t)$ a răspunsului este complet determinată prin (4.43) în funcţie de soluţia V a ecuaţiei matriceale algebrice liniare (4.39).

În concluzie, calculul răspunsului staţionar al sistemelor liniare la intrări persistente liniar - generate se poate face utilizând următoarea procedură.

> **Algoritmul 4.4** (Se dau sistemul (A, B, C, D) precum şi transformata $u(s)$ sub forma matricei de transfer a unui sistem SIMO. Se calculează răspunsul staţionar $Wx_F(t)$ pe un interval $[\,0,\, t_f\,]$ convenabil precizat).
>
> 1. Se realizează $u(s)$ sub forma (A_F, b_F, C_F).
> 2. Se calculează matricea V rezolvând ecuaţia matriceală Sylvester (4.39).
> 3. Se calculează matricea W în acord cu (4.43).
> 4. Se determină răspunsul staţionar aplicând algoritmul 4.1 cu datele de intrare A_F, W şi $x_F = b_F$.

Pentru exemplificare, vom considera câteva cazuri particulare importante.

Răspunsul staţionar la intrări polinomiale

Intrările polinomiale au un filtru de formare S_F definit prin matricele A_F, C_F date în (4.16). Prin urmare, partiţionând V pe coloane sub forma $V = [\,v_1\ \ldots\ v_p\,]$ unde $v_i \in \mathcal{R}^n$, ecuaţia (4.39) se scrie

$$\begin{bmatrix} v_1 & v_2 & \ldots & v_p \end{bmatrix} \begin{bmatrix} 0 & \ldots & 0 & 0 \\ 1 & \ldots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \ldots & 1 & 0 \end{bmatrix} =$$

$$= A \begin{bmatrix} v_1 & v_2 & \ldots & v_p \end{bmatrix} + B \begin{bmatrix} u_1 & \ldots & u_{p-1} & u_p \end{bmatrix}. \tag{4.44}$$

Obţinem

$$\left\{\begin{array}{ll} v_2 & = Av_1 + Bu_1 \\ \vdots & \\ v_p & = Av_{p-1} + Bu_{p-1} \\ 0 & = Av_p + Bu_p \end{array}\right. \tag{4.45}$$

deci vectorii $v_p \dots v_1$ rezultă succesiv rezolvând sistemele liniare

$$\begin{array}{l} Av_p = -Bu_p \\ Av_k = -Bu_k + v_{k+1}, \quad k = p-1:1\,. \end{array} \tag{4.46}$$

(Evident, matricea A trebuie triangularizată o singură dată.)

În sfârşit, din (4.43), unde $W = [\, w_1 \ \dots \ w_p \,]$ rezultă uşor

$$w_k = Cv_k + Du_k\,, \quad k = 1:p\,. \tag{4.47}$$

Observaţia 4.3 Unicitatea soluţiei V este în acest caz garantată dacă A e inversabilă (adică $0 \notin \lambda(A)$), ceea ce evident rezultă din ipoteza A stabilă, adoptată mai sus. Altfel spus, S poate avea soluţia polinomială $Vx_F(t)$, dar ea nu este soluţie asimptotică (pentru $t \to \infty$), decât dacă A este stabilă. $\diamond$

În particular, răspunsul staţionar la o intrare treaptă de amplitudine $u_1 \in \mathcal{R}^m$ corespunde la $p = 1$ şi $x_F(t) = 1$, deci are expresia

$$W = [\, -CA^{-1}B + D \,]u_1 \stackrel{\text{def}}{=} T(0)u_1,$$

unde

$$T(0) = \left[\, C(sI - A)^{-1}B + D \,\right]\Big|_{s=0}.$$

Răspunsul staţionar la intrări armonice

Intrările armonice au un filtru de formare definit prin

$$\begin{array}{l} A_F = \left[\begin{array}{cc} 0 & -\omega^2 \\ 1 & 0 \end{array}\right], \\ C_F = \left[\begin{array}{cc} \gamma_1 & \omega\gamma_2 \end{array}\right]. \end{array} \tag{4.48}$$

Partiţionând V pe coloane sub forma $V = [v_1 \ \ \omega v_2]$ ecuaţia (4.39) se scrie

$$[v_1 \ \ \omega v_2] \left[\begin{array}{cc} 0 & -\omega^2 \\ 1 & 0 \end{array}\right] = A[v_1 \ \ \omega v_2] + B[\gamma_1 \ \ \omega\gamma_2] \tag{4.49}$$

Obţinem

$$\left\{ \begin{array}{rl} \omega v_2 & = Av_1 + B\gamma_1 \\ -\omega v_1 & = Av_2 + B\gamma_2 \end{array} \right. \tag{4.50}$$

deci vectorii v_i, $i = 1:2$ rezultă rezolvând sistemul liniar

$$\left[\begin{array}{cc} -A & \omega I \\ -\omega I & -A \end{array} \right] \left[\begin{array}{c} v_1 \\ v_2 \end{array} \right] = \left[\begin{array}{c} B\gamma_1 \\ B\gamma_2 \end{array} \right] . \tag{4.51}$$

În sfârşit, din (4.43), unde $W = [w_1 \;\; \omega w_2]$, rezultă uşor

$$w_i = Cv_i + D\gamma_i, \quad i = 1:2 \tag{4.52}$$

Observaţia 4.4 Unicitatea soluţiei V este în acest caz garantată dacă $i\omega \notin \lambda(A)$, adică în absenţa rezonanţei pe frecvenţă ω a intrării, ceea ce rezultă evident din ipoteza A stabilă, adoptată mai sus. $\diamond$

4.5 Calculul caracteristicilor de frecvenţă

Deseori, problema de calcul abordată în finalul paragrafului precedent intervine sub o formă modificată, adică se cere calculul răspunsului armonic pentru diverse valori ale frecvenţei ω renunţând la ipoteza A stabilă în favoarea condiţiei mai slabe $i\omega \notin \lambda(A)$ de absenţă a rezonanţelor pe frecvenţele ω prescrise. Evident, aceasta înseamnă calculul caracteristicilor de frecvenţă ale sistemului considerat, ceea ce, tradiţional, se face recurgând la "complexificarea" acestuia.

Concret, aceasta înseamnă că în locul funcţiei de intrare reale (4.26), i.e. $u(t) = \gamma_1 \cos \omega t + \gamma_2 \sin \omega t$ se consideră intrarea complexă

$$u_c(t) = \gamma_c e^{i\omega t} , \tag{4.53}$$

unde $\gamma_c \in \mathcal{C}^m$ este un vector dat cu componente complexe. Avem

$$u_c(s) = \frac{\gamma_c}{s - i\omega} \tag{4.54}$$

deci o realizare (complexă) a lui (S_F) este

$$\begin{array}{l} A_F = i\omega, \quad b_F = 1, \\ C_F = \gamma_c. \end{array} \tag{4.55}$$

Dacă scriem acum (4.39) sub forma

$$i\omega V = AV + B\gamma_c \tag{4.56}$$

obţinem imediat

$$V = (i\omega I - A)^{-1}B\gamma_c, \tag{4.57}$$

iar din (4.43) rezultă

$$W = CV + D\gamma_c = [C(i\omega I - A)^{-1}B + D]\gamma_c \stackrel{\text{def}}{=} T(i\omega)\gamma_c \tag{4.58}$$

unde

$$T(i\omega) = C(i\omega I - A)^{-1}B + D\,, \quad \omega \in \mathcal{R} \tag{4.59}$$

reprezintă caracteristica (complexă) de frecvenţă a sistemului (S).

Problema calculului lui $T(i\omega)$ pentru diverse valori $\omega_k \in \mathcal{R}\,,\ k = 1 : N$ ale lui ω presupune deci

1. `Pentru` $k = 1 : N$
    1. Rezolvarea sistemelor matriceale liniare $(i\omega_k I - A)V_k = B$.
    2. Calculul matricei $T(i\omega_k) = CV_k + D$.

Deosebirea importantă faţă de cazul considerat anterior al intrărilor polinomiale constă aici, (în afara necesităţii evidente de a utiliza aritmetica complexă), în faptul că, aparent, la pasul 1.1 matricele $(i\omega_k I - A)$ trebuiesc triangularizate pentru fiecare k în parte, ceea ce în cazul în care A este de forma generală presupune, în total, $N\frac{n^3}{3}$ operaţii (în afara celor Nmn^2 necesare pentru determinarea coloanelor lui V). Pentru a evita aceasta, amintim că rezolvarea unui sistem $Ax = b$ cu A superior (sau inferior) Hessenberg cere numai n^2 operaţii. Prin urmare, în cazul de faţă se recomandă determinarea unei transformări prealabile M astfel încât

$$\tilde{A} = MAM^{-1}$$

să fie, de exemplu, superior Hessenberg. În acest scop, se utilizează eliminarea gaussiană cu pivotare parţială în aritmetică complexă şi rezultă $M = M_{n-1}P_{n-1}\ldots M_2P_2$, unde M_k sunt matrice inferior triunghiulare elementare, iar P_k sunt matrice de permutare adecvat alese. (Alternativ, dar mai costisitor, se poate utiliza procedura ortogonală în care $M = U = U_{n-1}\cdots U_2$ este o secvenţă de reflectori). Aplicând transformarea M matricelor B şi C, adică $\tilde{B} = MB,\ \tilde{C} = CM^{-1}$ şi definind $\tilde{V} = MV$, problema calculului lui $T(i\omega_k)$ se reduce la

1. `Pentru` $k = 1 : N$

    1. Rezolvarea sistemelor matriceale liniare superior Hessenberg $(i\omega_k I - \tilde{A})\tilde{V}_k = \tilde{B}$.
    2. Calculul matricei $T(i\omega_k) = \tilde{C}\tilde{V}_k + D$.

Aplicarea acestei scheme de calcul necesită, la pasul 1.1, în total, numai Nmn^2 operaţii, mai precis, $Nm\frac{n^2}{2}$ operaţii pentru triangularizare plus $Nm\frac{n^2}{2}$ pentru rezolvarea efectivă.

Procedura de calcul poate fi rezumată astfel.

Algoritmul 4.5 (Se dau A, B, C, D şi $\omega_k \in \mathcal{R}$, $k = 1 : N$, de regulă în scară logaritmică. Se calculează $T(i\omega_k)$, $k = 1 : N$).

1. Se determină M astfel încât $A \leftarrow \tilde{A} = MAM^{-1}$ este superior Hessenberg.
2. $B \leftarrow MB$
3. $C \leftarrow CM^{-1}$
4. `Pentru` $k = 1 : N$
    1. Se formează matricea $A_0 = i\omega_k I - A$.
    2. Se face triangularizarea $A_0 \leftarrow R = NA_0$.
    3. $B_0 = NB$
    4. Se rezolvă $A_0\tilde{V} = B_0$ în raport cu $\tilde{V}$.
    5. $T(i\omega_k) = C\tilde{V} + D$

Comentarii. Matricele A_0 şi B_0 sunt tablouri de lucru. La pasul 4.2. se efectuează eliminarea gaussiană cu pivotare parţială în aritmetică complexă astfel încât R rezultă superior triunghiulară. La pasul 4.3 transformarea N se aplică lui B iar la pasul 4.4 se procedează ca în cap. 1, § 2. $\diamond$

Observaţia 4.5 Aducerea prealabilă a lui A la forma Schur (pentru a evita paşii 4.2, 4.3) este mai puţin eficientă. $\diamond$

4.6 Răspunsul sistemelor liniare discrete

Datorită specificului recurent al ecuaţiilor de stare în cazul discret, matricea de tranziţie este $\Phi(k) = A^k$, $k \in \mathcal{N}$. În particular, matricea de tranziţie

într-un pas $\Phi(1) = A$ este dată, deci procedurile de calcul prezentate mai sus se simplifică. În algoritmii 4.1 şi 4.3 se elimină paşii 1, înlocuind peste tot perechea discretizată (F, G) cu perechea (A, B) a sistemului discret considerat. În algoritmul 4.2, în locul lui $u(s)$ se consideră transformata $\mathcal{Z}$ corespunzătoare $u(z)$. Consideraţiile din § 4.4 rămân valabile înlocuind condiţiile de stabilitate şi persistenţă prin versiunile lor discrete $|\lambda(A)| < 1$ şi, respectiv, $|\lambda(A_F)| \geq 1$. (Vezi § 5.1). În sfârşit, în versiunea discretă a algoritmului 4.5, numerele $i\omega_k$ situate pe axa imaginară se înlocuiesc cu numere $e^{i\theta_k}$, $k = 1 : N$, convenabil plasate pe cercul unitate T. Detaliile de redactare sunt propuse cititorului ca exerciţii.

Programe MATLAB disponibile

Pentru calculul şi reprezentarea grafică a răspunsului liber al unui sistem liniar este disponibilă funcţia **initial**, care foloseşte algoritmul 4.1. Răspunsul în stare iniţială nulă la o intrare (scalară) de tip impuls sau treaptă unitate se poate calcula utilizând funcţiile **impulse** şi **step**, care implementează algoritmii corespunzători 4.1 şi 4.2 considerând $x(0) = b$ şi respectiv $x(0) = 0$. Funcţia **lsim** implementează algoritmul 4.3 utilizând o aproximare liniară pe porţiuni a funcţiei de intrare $u(t)$, vezi formula (4.35) în cazul $p = 2$. În cazul discret, în aceleaşi scopuri, se utilizează funcţiile **dinitial**, **dimpulse**, **dstep** şi **dlsim**.

Pentru construcţia reprezentării discretizate cu pasul h a unei perechi (A, B) este disponibilă funcţia **c2d**, bazată pe formula de calcul exactă (4.33). Diverse metode de discretizare aproximativă pot fi utilizate apelând funcţia **c2dm**.

Pentru calculul şi reprezentarea grafică a caracteristicilor de frecvenţă ale unui sistem liniar sunt disponibile funcţiile **nyquist**, **bode** şi **nichols** care apelează procedura **freqresp**, bazată pe algoritmul 4.5. În cazul discret, în acelaşi scop se utilizează funcţiile **dnyquist**, **dbode** şi **dnichols**. Pentru funcţii de transfer sunt disponibile versiunile simple **freqs** şi respectiv **freqz**.

Exerciţii

E 4.1 Prin analogie cu algoritmii din text, scrieţi algoritmii de calcul al răspunsului în timp al unui sistem liniar, discret $S = (A, B, C, D)$ definit

prin ecuaţiile de stare

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), & x(0) = x \\ y(k) \ = Cx(k) + Du(k) \end{cases} \tag{4.60}$$

E 4.2 Reluaţi exerciţiul 4.1 în ipoteza că matricele sistemului discret S au formele particulare obţinute prin aplicarea procedurilor de realizare expuse în capitolul 1, § 3. Consideraţi succesiv cazurile SISO, MISO, SIMO şi în fiecare caz scrieţi algoritmii propuşi cât mai amănunţit posibil, evaluând memoria necesară precum şi numărul de operaţii. Formulaţi concluziile rezultate.

E 4.3 Scrieţi algoritmul de calcul al răspunsului unui sistem liniar discret S descris printr-o relaţie intrare - ieşire de forma

$$y(k) = \sum_{i=1}^{n} a_i y(k-i) + \sum_{i=0}^{d} b_i u(k-i), \quad k = 0 : N-1 \tag{4.61}$$

precizând cu grijă datele iniţiale necesare.

E 4.4 Reluaţi exerciţiul anterior aplicând transformata $\mathcal{Z}$ ecuaţiei (4.61) pentru a obţine $T(z)$ şi utilizaţi în acest context procedurile stabilite la exerciţiul 4.2. Formulaţi concluziile rezultate.

E 4.5 Cum se calculează răspunsul în timp al unui sistem liniar discret având funcţia de transfer dată sub forma

$$T(z) = k\,\frac{\prod(z-\mu_i)}{\prod(z-\lambda_i)}, \quad \text{respectiv} \quad T(z) = \sum \frac{r_i}{z-\lambda_i} \ ? \tag{4.62}$$

E 4.6 Reluaţi exerciţiul 4.1 pentru sistemul liniar discret definit prin

$$\begin{cases} x(k+1) = Ax(k) + \sum_{i=0}^{d} B_i u(k-i), & x(0) = x \\ \\ y(k) \ = x(k) + \sum_{i=0}^{d} D_i u(k-i) \end{cases} \tag{4.63}$$

în care $d \geq 1$ reprezintă întârzierea (pură) maximă pe canalul de intrare, iar A, C şi B_i, D_i, $i = 1 : d$ sunt matrice date.

E 4.7 Calculaţi matricele de transfer $T(z)$ pentru sistemele din exerciţiile 4.1 şi 4.6, indicând o procedură eficientă de calcul al răspunsului staţionar la intrare treaptă.

E 4.8 Elaboraţi proceduri eficiente de calcul pentru caracteristicile de frecvenţă $T(e^{i\theta})$, $\theta \in [\,0, 2\pi\,)$, unde $T(z)$ sunt matricele de transfer calculate în exerciţiul precedent.

E 4.9 Consideraţi sistemul liniar descris prin

$$\begin{cases} \dot{x}(t) = Ax(t) + \sum_{i=0}^{d} B_i u(t-ih), \quad x(0) = x_0 \\ \\ y(t) = Cx(t) + \sum_{i=0}^{d} D_i u(t-ih) \end{cases} \tag{4.64}$$

în care $d \geq 1$, iar funcţia de intrare $u(t)$ este etajată cu pasul h. Scrieţi algoritmul de calcul al răspunsului $y(kh)$, $k = 0 : N-1$.

E 4.10 Formulaţi şi rezolvaţi versiunile continue ale exerciţiilor 4.2 – 4.5, 4.7 şi 4.8.

E 4.11 Scrieţi algoritmul de calcul al răspunsului sistemului liniar (4.4) la o intrare impuls $u(t) = u_0\delta(t)$, unde vectorul $u_0 \in \mathcal{R}^m$ este dat. Cum procedaţi în cazul $u(t) = \sum_{k=0}^{N-1} u_k\delta(t-kh)$, $t_f = Nh$?

E 4.12 Formulaţi şi rezolvaţi versiunea discretă a exerciţiului 4.11.

E 4.13 Scrieţi algoritmul de calcul al răspunsului sistemului liniar (4.4) la un semnal de intrare periodic de formă dreptunghiulară, presupunând că perioada acestuia este ph, $p \geq 1$.

E 4.14 Elaboraţi algoritmul de calcul al răspunsului în timp al sistemului liniar (4.4) pentru un semnal de intrare periodic având formă de dinţi de fierăstrău cu perioada ph, $p \geq 1$ şi panta u_2 i.e. $u(t) = u_2(t - kph)$, $t \in [\,kph,\, (k+1)ph\,)$, $k \in \mathcal{N}$.

E 4.15 Analizaţi posibilităţile de calcul al regimului staţionar rezultat în condiţiile exerciţiilor 4.13 şi 4.14.

E 4.16 Propuneţi un analog al formulei (4.3) pentru sistemul liniar cu întârziere descris de ecuaţia diferenţială $\dot{x}(t) = Ax(t) + Bx(t-\tau)$ cu condiţia iniţială $x(t) = u(t)$, considerată cunoscută pentru $t \in [-\tau,\, 0\,]$, unde $\tau > 0$ reprezintă întârzierea (internă), iar A şi B sunt matrice date. Indicaţi o procedură de calcul aproximativ al răspunsului liber $y(kh) = Cx(kh)$, $k = 0 : N-1$.

Indicaţie. Consideraţi mai întâi ecuaţia corespunzătoare cazului $n = 1$ şi alegeţi h astfel încât $\tau = dh$, unde, de exemplu, $d = 10$. Cum se pot calcula valorile $x(kh)$, $h = 1 : d$?

E 4.17 Scrieţi algoritmul de calcul al răspunsului $y(kh)$, $k = 1 : N$, al sistemului (4.1) dacă în locul stării iniţiale $x(0) = x_0$ se dă starea finală $x(t_f) = \xi$.

E 4.18 Scrieţi algoritmul de calcul al răspunsului $y(kh)$, $k = 0 : N$, al sistemului (4.1) dacă între $x(0)$ şi $x(t_f)$ se dă o relaţie de forma $Lx(0) + Mx(t_f) = d$, unde L, M sunt două matrice de ordin n iar $d \in \mathcal{R}^n$ este un vector dat. (Cazul $L = I_n$, $M = 0$ corespunde algoritmului 4.1, iar cazul $L = 0$, $M = I_n$ corespunde exerciţiului 4.17).

Indicaţie. Spre deosebire de algoritmul 4.1 şi exerciţiul 4.17, care în esenţă efectuează integrarea sistemului $\dot{x} = Ax$ în timp direct şi, respectiv, în timp invers, problema bilocală considerată acum nu are întotdeauna soluţie unică. Considerând $t = t_f$ în expresia (4.2) a răspunsului liber, obţinem $x(t_f) = e^{t_f A}x(0)$, deci trebuie să avem $(L + Me^{t_f A})x(0) = d$. Acest sistem de n ecuaţii liniare cu n necunoscute $x(0) \in \mathcal{R}^n$ are soluţie unică dacă şi numai dacă matricea $\bar{L} = L + Me^{t_f A}$ rezultă inversabilă. În acest caz, se calculează $x(0) = \bar{L}^{-1}d$ şi se aplică algoritmul 4.1.

E 4.19 Formulaţi şi rezolvaţi exerciţiile 4.17 şi 4.18 în cazul $u(\cdot) \neq 0$.

E 4.20 Formulaţi şi rezolvaţi versiunile discrete ale exerciţiilor 4.17 – 4.19, precizând ipotezele necesare pentru buna formulare a problemelor de calcul astfel obţinute.

E 4.21 Scrieţi un algoritm pentru rezolvarea *ecuaţiei diferenţiale matriceale Sylvester*

$$\dot{X}(t) = AX(t) + X(t)B, \tag{4.65}$$

cu condiţia iniţială $X(0) = X$, unde A, B, X sunt matrice date de dimensiuni corespunzătoare.

Indicaţie. Analogul formulei (4.2) în cazul de faţă este

$$X(t) = e^{tA}X(0)e^{tB}.$$

În cazul $B = A^T$ se obţine *ecuaţia diferenţială matriceală Liapunov.*

E 4.22 Reluaţi exerciţiul precedent pentru ecuaţia

$$\dot{X}(t) = AX(t) + X(t)B + C \tag{4.66}$$

cu $X(0) = X$, unde, în plus, C este o matrice dată, eventual variabilă în timp.

Indicaţie. Analogul formulei (4.5) este

$$X(t) = e^{tA}X(0)e^{tB} + \int_0^t e^{(t-\tau)A}Ce^{(t-\tau)B}\,d\tau.$$

Bibliografie

[1] **Lupaş L., Bogdan M.** *Forced Response Computation of Linear Time - Invariant Multivariable Systems*, Rev. Roum. Sci. Tehn. – Electrotehn. et Energ., Vol. 19, No. 3, pp. 475 – 490, 1974.

[2] **Lupaş L.** *A Numerical Method of Evaluating the Transient Response of Linear Time - Invariant Multivariable Systems*, Rev. Roum. Sci. Tehn. – Electrotehn. et Energ., Vol. 19, No. 4, pp. 639 - 648, 1974.

[3] **Enright W.** *On the Efficient and Reliable Numerial Solution of Large Linear Systems of ODE's*, IEEE Trans. AC - 24, No. 6, pp. 905 – 908, 1979.

[4] **Laub A.J.** *Efficient Multivariable Frequency Response Computations*, IEEE Trans. AC - 26, No. 2, pp. 407 - 408, 1981.

Capitolul 5

Proceduri de analiză sistemică

În acest capitol vom discuta procedurile de analiză numerică a proprietăţilor sistemice fundamentale (stabilitate, controlabilitate şi observabilitate, etc.) precum şi procedurile conexe de minimizare dimensională şi echilibrare a modelelor sistemice liniare.

5.1 Stabilitatea sistemelor liniare

Stabilitatea unui sistem liniar $S = (A, B, C)$ este o proprietate ce depinde numai de matricea A a sistemului şi, în ultimă instanţă, numai de matricea de tranziţie $\Phi(t)$ asociată lui A. Mai precis, sistemul S este *stabil* dacă $\Phi(t) \to 0$ când $t \to \infty$. Deoarece matricea de tranziţie are expresii diferite în cazurile continuu şi discret, adică $\Phi(t) = e^{tA},\ t \in \mathcal{R}$, respectiv $\Phi(t) = A^t,\ t \in \mathcal{N}$, proprietatea de stabilitate are şi ea exprimări diferite în cele două cazuri.

Definiţia 5.1 *Matricea A a unui sistem continuu este* stabilă *dacă toate valorile proprii ale lui A au partea reală strict negativă, deci sunt plasate în* semiplanul stâng deschis $\mathcal{C}^-$ *al planului complex $\mathcal{C}$.*

Pe scurt, criteriul (condiţia necesară şi suficientă) de stabilitate în cazul continuu este $\lambda(A) \subset \mathcal{C}^-$ sau, mai sugestiv, $\mathrm{Re}\lambda(A) < 0$.

Definiţia 5.2 *Matricea A a unui sistem discret este* stabilă *dacă toate valorile proprii ale lui A au modul strict subunitar, deci sunt plasate în* discul unitate deschis $\mathcal{D}_1(0)$ *din planul complex $\mathcal{C}$.*

Pe scurt, criteriul de stabilitate în cazul discret este $|\lambda(A)| < 1$.

În general, o valoare proprie λ a lui A satisfăcând condiţia de stabilitate $\mathrm{Re}\lambda < 0$, respectiv $|\lambda| < 1$, se numeşte valoare proprie stabilă. Prin urmare, matricea A este stabilă dacă şi numai dacă toate valorile sale proprii sunt stabile.

Deoarece în ambele cazuri, continuu şi discret, proprietatea de stabilitate vizează o anumită plasare a valorilor proprii, iar acestea se calculează eficient utilizând algoritmul **QR**, obţinem următorul test de stabilitate pentru sisteme liniare.

Algoritmul 5.1 (Se testează stabilitatea matricei A).

1. Se calculează valorile proprii $\lambda(A)$ utilizând algoritmul QR (fără acumularea transformărilor).
2. Se determină $\alpha = \max \mathrm{Re}\lambda(A)$ în cazul continuu, respectiv $\rho = \max |\lambda(A)|$ în cazul discret. Matricea A este *stabilă* dacă şi numai dacă $\alpha < 0$, respectiv $\rho < 1$.

Observaţia 5.1 Proprietatea de stabilitate este o proprietate binară, i.e. poate fi numai adevărată sau falsă. În aplicaţii interesează deseori evaluări cantitative ale unor "margini" sau "rezerve" de stabilitate. Asemenea evaluări sunt furnizate de parametri α şi ρ calculaţi mai sus. De exemplu, dacă un sistem continuu este stabil, adică $\alpha < 0$, atunci $|\alpha|$ reprezintă inversul constantei de timp dominante, deci relaţia

$$t_r \approx \frac{4 \div 5}{|\alpha|}$$

oferă o evaluare orientativă a timpului de răspuns (definit pentru banda de 5% din jurul regimului staţionar constant) al sistemului considerat. Pe de altă parte, dacă matricea discretă A rezultă prin discretizarea cu pasul h a unei matrice A_c, aparţinând unui sistem continuu atunci avem $\lambda(A) = e^{h\lambda A_c}$, în particular $\rho = e^{h\alpha_c}$, deci numărul N_r de paşi necesar pentru stabilizarea răspunsului discret poate fi evaluat cu formula

$$N_r \approx \frac{4 \div 5}{|ln\rho|}.$$

$\diamond$

În general, numărul $\rho = \rho(A)$ se numeşte *rază spectrală* a matricei A.

5.2 Descompunerea spectrală

În anumite probleme de analiză şi sinteză structurală în care matricea A nu este stabilă, interesează evidenţierea subspaţiilor invariante ale matricei A generate de vectorii proprii asociaţi valorilor proprii stabile şi respectiv nestabile.

Pentru precizare, fie sistemul liniar $S = (A, B, C)$ definit prin ecuaţiile de stare

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{5.1}$$

unde $x \in \mathcal{R}^n$ este un vector cu n componente.

Considerăm transformarea de coordonate $\tilde{x} = Ux$, în care $U \in \mathcal{R}^{n\times n}$ este matricea ortogonală ce aduce matricea A la *forma Schur reală* $S \stackrel{\text{def}}{=} \tilde{A} = UAU^T$. Punem în evidenţă partiţiile conforme

$$U = \begin{bmatrix} U_a \\ U_b \end{bmatrix} \begin{matrix} \}n_a \\ \}n_b \end{matrix}, \qquad S \stackrel{\text{def}}{=} \tilde{A} = \begin{bmatrix} \overbrace{A_a}^{n_a} & \overbrace{A_{ab}}^{n_b} \\ 0 & A_b \end{bmatrix} \begin{matrix} \}n_a \\ \}n_b \end{matrix} \tag{5.2}$$

unde dimensiunile blocurilor satisfac relaţia $n_a + n_b = n$. Se constată uşor că un vector $\tilde{x}$ de forma

$$\tilde{x} = \begin{bmatrix} x_a \\ 0 \end{bmatrix} \begin{matrix} \}n_a \\ \}n_b \end{matrix}, \tag{5.3}$$

unde x_a este arbitrar, se transformă prin $\tilde{A}$ într−un vector de aceeaşi formă, mai precis avem

$$\tilde{A}\tilde{x} = \begin{bmatrix} A_a x_a \\ 0 \end{bmatrix} \begin{matrix} \}n_a \\ \}n_b \end{matrix},$$

Acest fapt arată că

(i) subspaţiul $\mathcal{X}_a \subset \mathcal{R}^n$ al vectorilor de forma (5.3) este invariant în raport cu $\tilde{A}$ şi restricţia lui $\tilde{A}$ la acest subspaţiu coincide cu blocul stânga - sus A_a al lui $\tilde{A}$.

În coordonatele iniţiale $x = U^T\tilde{x}$ obţinem

$$x = \begin{bmatrix} U_a^T & U_b^T \end{bmatrix} \begin{bmatrix} x_a \\ 0 \end{bmatrix} = U_a^T x_a,$$

deci

(j) coloanele matricei U_a^T formează o bază ortogonală a subspaţiului $\mathcal{X}_a$.

Proprietăţiile (i) şi (j) de mai sus sunt esenţiale pentru orice procedură de calcul a subspaţiilor invariante $\mathcal{X}_a$, definite prin condiţii spectrale impuse restricţiei A_a corespunzătoare.

De exemplu, dacă se cere ca evoluţia sistemului S pe subspaţiul $\mathcal{X}_a$ să fie stabilă atunci matricea A_a trebuie să fie stabilă, deci $Re\lambda(A_a) < 0$. *Cel mai mare* subspaţiu A-invariant $\mathcal{X}_a$ cu această proprietate se notează cu $\mathcal{X}^-(A)$ şi corespunde situaţiei în care matricea A_a din (5.2) conţine *toate* valorile proprii ale lui A cu parte reală negativă.

Această situaţie se realizează uşor dacă presupunem că matricea S din (5.2) reprezintă *forma Schur reală ordonată* a lui A, în care ordonarea valorilor proprii pe diagonala lui S se face *în sensul crescător al părţilor reale*, deci valorile proprii λ_i ale lui A apar pe diagonala lui S în ordinea

$$Re\lambda_1 \leq Re\lambda_2 \leq \ldots \leq Re\lambda_n. \tag{5.4}$$

În acest caz submatricea A_a satisface condiţia de stabilitate şi are dimensiunea maximă dacă n_a se consideră egal cu numărul n^- al valorilor proprii stabile ale lui A.

Procedura de determinare a subspaţiului A-invariant $\mathcal{X}^-(A)$ asociat valorilor proprii stabile ale lui A poate fi rezumată astfel.

Algoritmul 5.2 (Se determină o bază ortogonală a subspaţiului A-invariant stabil $\mathcal{X}^-(A)$).

1. Se calculează forma Schur reală $S = UAU^T$ a lui A, utilizând algoritmul QR cu acumularea transformărilor U.
2. Se ordonează S, în acord cu criteriul (5.4), utilizând transformări de asemănare ortogonală $S \leftarrow PSP^T$ şi efectuând acumularea acestora conform schemei $U \leftarrow PU$.
3. Se determină numărul $n_a \stackrel{\text{def}}{=} n_-$ al valorilor proprii stabile inspectând diagonala lui S. O bază ortogonală a subspaţiului $\mathcal{X}^-(A)$ este formată din primele n_- coloane ale matricei U^T.

Comentarii. Algoritmul 5.2 utilizează exclusiv transformări ortogonale şi, ca atare, este numeric stabil. În cazul discret se obţin concluzii similare utilizând *forma Schur reală ordonată* a lui A, în care ordonarea valorilor proprii pe diagonala lui S se face *în sensul crescător al modulelor*, deci astfel încât

$$|\lambda_1| \leq |\lambda_2| \leq \ldots \leq |\lambda_n|. \tag{5.5}$$

$\diamond$

Revenim acum la sistemul liniar $S = (A, B, C)$ adus la forma (5.2) cu transformarea ortogonală $\tilde{x} = Ux$. Considerăm transformarea suplimentară $z = V_0^{-1}\tilde{x}$ sau, echivalent, $\tilde{x} = V_0 z$, în care matricea $V_0 \in \mathcal{R}^{n\times n}$ are structura

$$V_0 = \begin{array}{c} \overbrace{}^{n_a} \quad \overbrace{}^{n_b} \\ \left[\begin{array}{cc} I & Y \\ 0 & I \end{array}\right] \begin{array}{l} \}n_a \\ \}n_b \end{array} \end{array} \tag{5.6}$$

Se constată imediat că avem

$$\tilde{A}V_0 = V_0 A_0, \quad \text{unde} \quad A_0 \stackrel{\text{def}}{=} \left[\begin{array}{cc} A_a & 0 \\ 0 & A_b \end{array}\right], \tag{5.7}$$

dacă şi numai dacă matricea Y satisface ecuaţia matriceală Sylvester

$$A_a Y - Y A_b = -A_{ab}. \tag{5.8}$$

Pe de altă parte, se ştie că ecuaţia (5.8) are o soluţie unică Y dacă şi numai dacă matricele A_a şi A_b nu au valori proprii comune, pe scurt

$$\lambda(A_a) \cap \lambda(A_b) = \emptyset. \tag{5.9}$$

Deoarece condiţia (5.9) este evident satisfăcută în situaţia creată aplicând algoritmul 5.2, în care A_a şi A_b colectează toate valorile proprii stabile şi, respectiv, nestabile ale lui A, concludem că prin schimbarea

$$x \stackrel{\text{def}}{=} Vz = U^T\tilde{x} = U^T V_0 z,$$

sistemul $S = (A, B, C)$ poate fi întotdeauna adus la forma

$$\left\{\begin{array}{l} \dot{z} = A_0 z + B_0 u \\ y = C_0 z \end{array}\right. \tag{5.10}$$

unde

$$\begin{array}{l} A_0 \stackrel{\text{def}}{=} V^{-1}AV = \left[\begin{array}{cc} A_a & 0 \\ 0 & A_b \end{array}\right], \quad B_0 \stackrel{\text{def}}{=} V^{-1}B = \left[\begin{array}{c} B_a \\ B_b \end{array}\right], \\ C_0 \stackrel{\text{def}}{=} \quad CV \quad = \left[\begin{array}{cc} C_a & C_b \end{array}\right], \end{array} \tag{5.11}$$

în care matricea A_a colectează toate valorile proprii ale matricei A cu $Re\lambda(A) < 0$, deci, este *stabilă*, iar matricea A_b colectează toate valorile proprii ale matricei A cu $Re\lambda(A) \geq 0$, deci este *total nestabilă*, pe scurt

$$\lambda(A_a) \subset \mathcal{C}^-, \quad \lambda(A_b) \subset \bar{\mathcal{C}}^+, \tag{5.12}$$

unde $\bar{\mathcal{C}}^+$ este semiplanul drept închis al planului complex $\mathcal{C}$.

Definiţia 5.3 *Forma bloc-diagonală (5.10), (5.11) se numeşte* descompunerea spectrală *a sistemului* $S = (A, B, C)$ *şi corespunde reprezentării lui* S *prin conexiunea paralel a două sisteme,* $S_a = (A_a, B_a, C_a)$ *şi* $S_b = (A_b, B_b, C_b)$.

În acord cu (5.12), S_a este stabil de ordin $n_a = n^-$, unde n^- este numărul valorilor proprii stabile ale lui A iar S_b este total nestabil de ordin $n_b = \bar{n}^+$, unde $\bar{n}^+$ este numărul valorilor proprii nestabile (cu parte reală strict pozitivă sau nulă) ale lui A. În consecinţă, S_a se numeşte *partea stabilă* iar S_b se numeşte *partea total nestabilă* a sistemului S.

În particular, dacă matricea A este *dihotomică*, adică nu are valori proprii pe axa imaginară (cu $Re\lambda = 0$), atunci matricea A_b din (5.11) este *antistabilă*, adică $\lambda(-A_b) \subset \mathcal{C}^-$, iar S_b se numeşte *partea antistabilă* a sistemului dihotomic S.

Conform celor de mai sus, matricea V din (5.11) este

$$V = U^T V_0 = \left[\begin{array}{cc} U_a^T & U_b^T \end{array} \right] \left[\begin{array}{cc} I & Y \\ 0 & I \end{array} \right],$$

deci avem

$$V \stackrel{\text{def}}{=} \left[\begin{array}{cc} V_a & V_b \end{array} \right] = \left[\begin{array}{cc} U_a^T & U_b^T + U_a^T Y \end{array} \right]. \tag{5.13}$$

Punând

$$z = \left[\begin{array}{c} z_a \\ z_b \end{array} \right]$$

obţinem

$$x = Vz = V_a z_a + V_b z_b,$$

unde coloanele matricei $V_a = U_a^T$ constituie o bază ortogonală a *subspaţiului stabil* $\mathcal{X}^-(A) = \mathcal{X}_a$, corespunzător vectorilor z cu $z_b = 0$ (vezi (5.3)).

În mod analog, coloanele matricei V_b constituie o bază (în general neortogonală, deoarece $Y \neq 0$) a *subspaţiului total nestabil* $\bar{\mathcal{X}}^+(A) = \mathcal{X}_b$, corespunzător vectorilor z cu $z_a = 0$. (Spre deosebire de (5.2), în (5.11) avem $A_{ab} = 0$, astfel încât acum şi $\mathcal{X}_b$ este $\tilde{A}$-invariant cu restricţia A_b).

Procedura de determinare a descompunerii spectrale (5.10), (5.11) poate fi rezumată astfel.

Algoritmul 5.3 (Se construieşte descompunerea spectrală a sistemului $S = (A, B, C)$ şi se determină o bază ortogonală a subspaţiului A-invariant stabil $\mathcal{X}^-(\mathcal{A})$).

1. Se aplică algoritmul 5.2.

2. Se face

$$B \leftarrow UB = \begin{bmatrix} B_a \\ B_b \end{bmatrix}, \quad C \leftarrow CU^T = \begin{bmatrix} C_a & C_b \end{bmatrix}.$$

3. Se rezolvă ecuaţia matriceală Sylvester (5.8), ţinând seama că matricele A_a, A_b sunt deja în forma Schur reală (vezi (5.2)).
4. Se face

$$B_a \leftarrow B_a - YB_b, \quad C_b \leftarrow C_aY + C_b.$$

Comentarii. Spre deosebire de algoritmul 5.2, algoritmul 5.3 utilizează transformări de asemănare neortogonală (vezi forma (5.13) a lui V_0), care în general ridică probleme de condiţionare. În practică, numărul de condiţie $\mathrm{cond}(V_0)$ se poate evalua eficient utilizând estimatorul de condiţie pentru matrice triunghiulare propus în LINPACK [III]. În ceea ce priveşte însăşi condiţionarea problemei rezolvării ecuaţiei matriceale Sylvester (5.8), aceasta poate fi apreciată (destul de nesigur) evaluând apropierea dintre spectrele matricelor A_a şi A_b, e.g. prin

$$\delta \stackrel{\mathrm{def}}{=} \min \mathrm{Re}\lambda(A_b) - \max \mathrm{Re}\lambda(A_a).$$

(Relativ la această chestiune, vezi cap. 1). În cazul discret se obţin concluzii similare pe baza aceleiaşi ecuaţii (5.8). Dacă este necesar, aceasta poate fi scrisă sub forma specifică

$$A_aYA_b^{-1} - Y = -A_{ab}A_b^{-1}, \tag{5.14}$$

căreia i se pot aplica rezultatele corespunzătoare din capitolul 1. $\diamond$

Opţional, algoritmul 5.3 poate fi completat pentru a furniza baza $V_b = U_b^T + U_a^TY$ a subspaţiului $\bar{\mathcal{X}}^+(A)$ sau/şi matricea de transformare (neortogonală) $T \in \mathcal{R}^{n\times n}$ a tripletului $S = (A, B, C)$ la forma (5.11), (5.12). Prin definiţie avem

$$z = Tx, \tag{5.15}$$

unde, conform notaţiilor de mai sus

$$T = V^{-1} = V_0^{-1}U = \begin{bmatrix} I & -Y \\ 0 & I \end{bmatrix} \begin{bmatrix} U_a \\ U_b \end{bmatrix} = \begin{bmatrix} U_a - YU_b & U_b \end{bmatrix}. \tag{5.16}$$

Prin urmare, matricea T poate fi calculată pe loc în tabloul U furnizat de algoritmul 5.2, efectuând actualizarea

$$U_a \leftarrow U_a - YU_b.$$

Alternativ, pentru a evita distrugerea bazei ortogonale U_a a subspaţiului $\mathcal{X}^-(A)$, se poate recurge la calculul matricei $V = T^{-1}$ din (5.13), corespunzătoare schimbării de coordonate $x = Vz$.

5.3 Controlabilitatea şi observabilitatea sistemelor liniare

Controlabilitatea unui sistem liniar $S = (A, B, C)$ este o proprietate ce depinde numai de perechea (A, B). Pe scurt, S este *controlabil* dacă orice tranziţie de stare dorită poate fi realizată prin alegerea adecvată a funcţiei de intrare.

Pentru precizare, considerăm sistemul liniar discret

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{5.17}$$

în care, ca de obicei, $A \in \mathcal{R}^{n\times n}$, $B \in \mathcal{R}^{n\times m}$ şi $C \in \mathcal{R}^{l\times n}$ sunt matrice constante.

Presupunând $x(0) = 0$, din (5.17) rezultă succesiv

$$\begin{aligned} x(1) &= Bu(0), \\ x(2) &= ABu(0) + Bu(1), \end{aligned}$$

şi în general

$$x(k) = A^{k-1}Bu(0) + \ldots + ABu(k-2) + Bu(k-1). \tag{5.18}$$

Introducând notaţiile

$$R_k = \left[\begin{array}{cccc} B & AB & \ldots & A^{k-1}B \end{array}\right], \quad u_k = \left[\begin{array}{c} u(k-1) \\ u(k-2) \\ \vdots \\ u(0) \end{array}\right], \tag{5.19}$$

în care vectorul u_k conţine k blocuri, fiecare cu m componente, iar R_k este o matrice bloc de forma $n \times mk$, relaţiile (5.18) se scriu pe scurt

$$R_k u_k = x(k). \tag{5.20}$$

Considerând că în (5.20) pasul $k \geq 1$ este *fixat* iar $x(k) \in \mathcal{R}^n$ este un vector arbitrar care reprezintă starea dorită la pasul k, vom spune că perechea

discretă (A, B) este *controlabilă în k paşi* dacă sistemul de ecuaţii liniare (5.20) are o soluţie u_k, adică *există* un şir de intrări $u(0), u(1), \ldots, u(k-1)$ care realizează tranziţia de stare $0 \to x(k)$. După cum se ştie, sistemul (5.20) este *compatibil oricare ar fi membrul drept* dacă şi numai dacă matricea R_k este *epică*, adică

$$rang R_k = n. \tag{5.21}$$

Mai mult, în acest caz *soluţia normală* u_k^* a sistemului (5.20) este unic determinată şi are expresia

$$u_k^* = R_k^T (R_k R_k^T)^{-1} x(k), \tag{5.22}$$

deci avem

$$\|u_k^*\|^2 = \min_{R_k u = x(k)} \|u_k\|^2, \tag{5.23}$$

unde norma (euclidiană) a unui vector u_k de forma (5.19) este evident

$$\|u_k\|^2 = \sum_{t=0}^{k-1} \|u(t)\|^2. \tag{5.24}$$

Interpretând expresia (5.24) ca "efort" de comandă necesar pentru realizarea tranziţiei $0 \to x(k)$, soluţia normală (5.22) se numeşte sugestiv comandă de normă sau efort minim.

În rezumat, perechea (A, B) este controlabilă în k paşi dacă şi numai dacă este satisfăcută condiţia de rang (5.21), şi în acest caz comanda de normă minimă are expresia (5.22).

În general, considerând mai sus un număr de paşi k arbitrar, obţinem noţiunea de controlabilitate. Prin urmare, perechea discretă (A, B) este *controlabilă* dacă există $k \geq 1$ astfel încât oricare ar fi $x(k) \in \mathcal{R}^n$ există o soluţie u_k a sistemului (5.20) sau, echivalent, este îndeplinită condiţia de rang (5.21).

Mai mult, în acest caz avem întotdeauna $k \leq n$, deoarece, în virtutea teoremei Cayley-Hamilton, eventualele blocuri $A^{k-1}B$ cu $k > n$ din (5.19) sunt în mod necesar liniar dependente de precedentele, deci nu contribuie la satisfacerea condiţiei de rang (5.21). Pe scurt, un sistem liniar discret controlabil este întotdeauna controlabil în cel mult n paşi.

Notând pentru simplitate $R_n = R$, deci

$$R = \left[\begin{array}{cccc} B & AB & \ldots & A^{n-1}B \end{array} \right], \tag{5.25}$$

precum şi $u_n = \mathbf{u}$, relaţiile (5.20) pentru $k = n$ se scriu

$$R\mathbf{u} = x(n) \tag{5.26}$$

iar condiţia de rang (5.21), echivalentă cu controlabilitatea perechii (A, B), devine

$$\text{rang}R = n. \tag{5.27}$$

În cazul sistemelor cu timp continuu, consideraţii similare conduc în ultimă instanţă tot la condiţia (5.27). De aceea, în continuare vom adopta următoarea definiţie unificatoare, pur algebrică.

Definiţia 5.4 *Perechea* (A, B) *este* controlabilă *dacă şi numai dacă este satisfăcută condiţia de rang (5.27).*

Matricele R *şi* R_k *definite prin relaţiile (5.25) şi (5.19) se numesc* matrice de controlabilitate *şi respectiv* matrice de controlabilitate în k paşi *ale perechii* (A, B).

În cazul în care perechea (A, B) *este controlabilă, cel mai mic întreg* $k \geq 1$ *pentru care condiţia de rang (5.21) este satisfăcută se numeşte* indice de controlabilitate *al perechii* (A, B) *şi se notează cu* ν.

Observăm că, în cazul discret, ν reprezintă cel mai mic număr k de paşi (deci timpul minim) în care este posibilă tranziţia de stare $0 \to x(k)$, oricare ar fi starea dorită $x(k) \in \mathcal{R}^n$. De asemenea, dacă perechea (A, B) are o singură intrare, deci $m = 1$, atunci R este pătrată şi nesingulară iar $\nu = n$. Din contră, dacă $m > 1$ atunci în general avem $\nu < n$.

Observabilitatea unui sistem liniar $S = (A, B, C)$ este proprietatea duală controlabilităţii şi, în consecinţă, depinde numai de perechea (C, A). Pe scurt, S este *observabil* dacă starea iniţială este unic determinată cunoscând funcţiile de intrare şi ieşire, obţinute e.g. prin măsurători efectuate la terminalele lui S.

Pentru precizare, considerăm din nou sistemul liniar discret descris prin ecuaţiile de stare (5.17). Presupunând că $u(t) = 0, t \geq 0$, din (5.17) rezultă $x(t) = A^t x(0)$, astfel încât şirul de ieşiri corespunzător stării iniţiale $x(0)$ este

$$y(t) = CA^t x(0), \quad t = 0 : k-1. \tag{5.28}$$

Introducând notaţiile

$$Q_k = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{bmatrix}, \quad y_k = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(k-1) \end{bmatrix}, \tag{5.29}$$

în care vectorul y_k conţine k blocuri, fiecare cu l componente, iar Q_k este o matrice bloc de forma $kl \times n$, relaţiile (5.28) se scriu pe scurt

$$Q_k x(0) = y_k. \tag{5.30}$$

Considerând că în (5.30) pasul $k \geq 1$ este *fixat*, vom spune că perechea discretă (C, A) este *observabilă în k paşi* dacă sistemul de ecuaţii liniare (5.30) are cel mult o soluţie x(0), adică starea iniţială $x(0)$ este *unic* determinată de şirul de ieşiri $y(0)$, $y(1)$, ..., $y(k-1)$. După cum se ştie, sistemul (5.30) este *unic determinat* dacă şi numai dacă matricea Q_k este *monică*, adică

$$\operatorname{rang} Q_k = n. \tag{5.31}$$

Mai mult, în acest caz *pseudosoluţia* $x^*(0)$ în sensul CMMP a sistemului (5.30) este unic determinată şi are expresia

$$x^*(0) = (Q_k^T Q_k)^{-1} Q_k^T y_k, \tag{5.32}$$

deci avem

$$\|y_k - Q_k x^*(0)\|^2 = \min_{x(0) \in \mathcal{R}^n} \|y_k - Q_k x(0)\|^2, \tag{5.33}$$

unde evident

$$\|y_k - Q_k x(0)\|^2 = \sum_{t=0}^{k-1} \|y_k - Cx(k)\|^2. \tag{5.34}$$

Interpretând reziduurile $r(k) = y(k) - Cx(k)$ ca erori sau "zgomote" de măsură care afectează ieşirile măsurate $y(k)$, pseudosoluţia $x^*(0)$ se numeşte sugestiv estimare optimală (în sensul CMMP) a stării iniţiale $x(0)$.

În rezumat, perechea (C, A) este observabilă în k paşi dacă şi numai dacă este îndeplinită condiţia de rang (5.31) şi în acest caz estimarea optimală a stării iniţiale $x(0)$ are expresia (5.32).

În general, considerând mai sus un număr de paşi k arbitrar, obţinem noţiunea de observabilitate. Prin urmare, , perechea discretă (C, A) este *observabilă* dacă există $k \geq 1$ astfel încât este îndeplinită condiţia de rang (5.31). Mai mult, în acest caz, la fel ca în cazul controlabilităţii, putem considera $k \leq n$.

Notând pentru simplitate $Q_n = Q$, deci

$$Q = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}, \tag{5.35}$$

precum şi $y_n = \mathbf{y}$, relaţiile (5.30) pentru $k = n$ se scriu

$$Qx(0) = \mathbf{y} \tag{5.36}$$

iar condiţia de rang (5.31), echivalentă cu observabilitatea perechii (C, A), devine

$$\operatorname{rang} Q = n. \tag{5.37}$$

În cazul sistemelor cu timp continuu, consideraţii similare conduc în ultimă instanţă tot la condiţia (5.37). În consecinţă, mai departe vom adopta următoarea definiţie unificatoare.

Definiţia 5.5 *Perechea* (C, A) *este* observabilă *dacă şi numai dacă este satisfăcută condiţia de rang (5.37).*

Matriceile Q *şi* Q_k *definite prin relaţiile (5.35) şi (5.29) se numesc* matrice de observabilitate *şi respectiv* matrice de observabilitate în k paşi *ale perechii* (C, A).

În cazul în care perechea (C, A) *este observabilă, cel mai mic întreg* $k \geq 1$ *pentru care condiţia de rang (5.31) este satisfăcută se numeţte* indice de observabilitate *al perechii* (C, A).

Examinând în paralel relaţiile (5.18)-(5.27) şi (5.28)-(5.37), concludem că noţiunile de controlabilitate şi observabilitate sunt *duale*, în sensul că ele se corespund prin corespondenţa

$$A \leftarrow A^T, \quad C \leftarrow B^T,$$

în particular avem

$$Q(C, A) = R^T(A^T, C^T).$$

În consecinţă, mai departe vom prezenta procedurile de calcul asociate controlabilităţii, rămânând ca dualele lor să fie formulate "prin dualitate" şi discutate în detaliu de către cititorul interesat.

5.4 Teste elementare de controlabilitate

În general procedurile de testare a controlabilităţii unei perechii date (A, B) sunt de două tipuri.

Primul tip de proceduri, numite ad-hoc *elementare*, testează controlabilitatea perechii (A, B) utilizând direct definiţia 5.3 sau proprietăţi echivalente cu aceasta. Aceste proceduri sunt în general neeficiente şi pot fi aplicate cu succes numai pentru perechi de dimensiuni mici.

Al doilea tip de proceduri utilizează transformări de asemănare pentru a aduce perechea iniţială (A, B) la o formă "simplă", pe care proprietatea de controlabilitate să poată fi testată direct ("prin inspecţie"). Aceste proceduri, numite în continuare *de transformare*, sunt cele mai utilizate în aplicaţii, cu atât mai mult cu cât ele facilitează şi rezolvarea altor probleme de calcul conexe vizând, de exemplu, descompunrea controlabilă, testarea stabilizabilităţii, alocarea polilor etc. Cea mai eficientă procedură de transformare, care constă în aducerea perechii (A, B) la forma (superior) Hessenberg prin transformări de asemănare ortogonale, va fi prezentată în paragraful următor.

Observaţia 5.2 Proprietatea de controlabilitate este o proprietate binară, exprimată prin condiţia de rang (5.27). De aceea, pe de o parte nu pot exista proceduri numerice absolut sigure de testare a controlabilităţii iar, pe de altă parte, pentru reducerea riscurilor implicate de deciziile de rang, se recomandă utilizarea transformărilor ortogonale şi, în special, a descompunerii valorilor singulare (**DVS**). $\diamond$

Din clasa procedurilor elementare face parte în primul rând procedura "directă", bazată pe construcţia matricei de controlabilitate R şi testarea condiţiei de rang (5.27).

Construcţia matricei R se face recurent, ţinând seama că prin definiţie avem $R = R_n$, unde matricele şirului $R_k,\ k \geq 1$ au expresia (5.19) iar $R_1 = B$. Pentru a obţine relaţia de recurenţă care leagă doi termeni succesivi R_k şi R_{k+1}, observăm că avem

$$R_{k+1} \stackrel{\text{def}}{=} \left[\begin{array}{c|ccccc} B & AB & A^2B & \cdots & A^kB \end{array}\right] = \left[\begin{array}{c|c} B & AR_k \end{array}\right], \quad (5.38)$$

respectiv

$$R_{k+1} \stackrel{\text{def}}{=} \left[\begin{array}{ccccc|c} B & AB & \cdots & A^{k-1}B & A^kB \end{array}\right] = \left[\begin{array}{c|c} R_k & B_{k+1} \end{array}\right], \quad (5.39)$$

unde submatricea $B_{k+1} = A^kB$, formată din ultimele m coloane ale lui R_{k+1}, rezultă în funcţie de submatricea corespunzătoare B_k a lui R_k prin relaţia evidentă

$$B_{k+1} = AB_k, \quad k \geq 1 \quad (5.40)$$

iar $B_1 = B$.

Se constată imediat că procedura de construcţie a matricei R bazată pe relaţia (5.38) este relativ neeconomică, întrucât necesită efectuarea produselor AR_k, unde matricea R_k are mk coloane, $k \geq 1$, în timp ce procedura bazată pe relaţia (5.39) necesită efectuarea produselor AB_k,unde la fiecare

pas $k \geq 1$ matricea B_k are numai m coloane. De asemenea, este clar că la fiecare pas matricea B_k este disponibilă în R_k, mai precis avem

$$B_k = R_k(:, (k-1)m+1 : km) \tag{5.41}$$

astfel încât nu este necesară nici alocarea unui tablou suplimentar de lucru pentru memorarea matrice curente B_k, nici calculul pe loc în B, conform schemei $B \leftarrow AB$, al termenilor şirului B_k, ceea ce ar implica distrugerea matricei iniţiale B.

În rezumat, procedura de testare a controlabilităţii perechii (A, B) bazată pe utilizarea relaţiilor (5.39) – (5.41) poate fi formulată astfel.

Algoritmul 5.4 (Se testează controlabilitatea perechii (A, B) construind matricea de controlabilitate R).

1. $R(:, 1 : m) \leftarrow B_1 = B$
2. `Pentru` $k = 2 : n$
   1. $R(:, (k-1)m+1 : km) \leftarrow B_k = AB_{k-1}$
3. Se calculează $r = \text{rang}R$ utilizând algoritmul DVS.
4. `Dacă` $r = n$ `atunci`
   1. `Tipăreşte` "Perechea (A, B) este controlabilă."

În general, în ciuda simplităţii sale, algoritmul 5.4 *nu este recomandabil* întrucât matricea R este de mari dimensiuni iar calculul coloanelor sale la pasul 2.1 implică riscuri la efectuarea testului de rang de la pasul 3.

De exemplu, dacă $m = 1$, deci matricea $B = b$ se reduce la o singură coloană, atunci, conform relaţiei (5.40), coloanele $b, Ab, \ldots, A^k b, \ldots$ ale matricei (patrate) R coincid cu vectorii obţinuţi aplicând *metoda puterii* vectorului iniţial $b_1 = b$. Prin urmare, dacă $n \gg 1$ iar matricea A are o valoare proprie dominantă, deci

$$|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|,$$

atunci ultimele coloane ale lui R sunt practic aliniate cu vectorul propriu x_1 al lui A asociat cu λ_1. Aceasta înseamnă că, chiar dacă perechea (A, b) este controlabilă, matricea de controlabilitate R este numeric aproape singulară, adică

$$\text{cond}(R) = \frac{\sigma_1}{\sigma_n} \gg 1,$$

unde $\text{cond}(R)$ este numărul de condiţionare la inversare al matricei R (în raport cu norma spectrală) iar σ_i, $i = 1 : n$ sunt valorile singulare ale lui R, ordonate astfel încât $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$. Fenomenul poartă numele de

necontrolabilitate numerică a perechii (A, b) şi în general nu poate fi evitat, indiferent de testul de rang utilizat la pasul 3 al algoritmului de mai sus.

Din acest punct de vedere, în cazul $m > 1$ se recomandă efectuarea testului de rang în bucla 2, conform schemei următoare, care, în plus, furnizează şi indicele de controlabilitate ν al perechii (A, B).

Algoritmul 5.5 (Se testează controlabilitatea perechii (A, B) construind matricea de controlabilitate R. Se presupune $m > 1$.)

1. $R(:, 1:m) \leftarrow B_1 = B$
2. `Pentru` $k = 2 : n$
   1. Se calculează $r = \text{rang}R$ utilizând algoritmul DVS.
   2. `Dacă` $r = n$ `atunci`
      1. $\nu = k$
      2. `Tipăreşte` "Perechea (A, B) este controlabilă."
      3. `stop`
   3. $R(:, (k-1)m + 1 : km) \leftarrow B_k = AB_{k-1}$

Subliniem însă că schema de mai sus este în orice caz neeficientă din punctul de vedere al efortului de calcul datorită apelării repetate a algoritmului **DVS** într-o formă brută, care nu exploatează rezultatele calculelor anterioare. Cel puţin aceeaşi siguranţă a testului este asigurată de algoritmul 5.7 prezentat mai departe, cu un efort de calcul considerabil mai redus.

În afară de procedura "directă", bazată pe testarea condiţiei de rang (5.27), din grupul metodelor elementare fac parte o serie de proceduri derivate pe baza unor criterii echivalente cu aceasta. Cele mai importante sunt rezumate mai jos pentru a simplifica referirile ulterioare.

Propoziţia 5.1 *Controlabilitatea perechii (A, B) este echivalentă cu oricare dintre condiţiile*

1. Nu există nici un vector $h \neq 0$ astfel încât $A^T h = \lambda h, \lambda \in \mathcal{C}$ şi $B^T h = 0$.

2. $\text{rang} \begin{bmatrix} A - \lambda I & B \end{bmatrix} = n, \quad \forall \lambda \in \lambda(A).$

3. Oricare ar fi vectorul $g \in \mathcal{R}^m$ cu proprietatea $Bg \neq 0$, există o matrice $F \in \mathcal{R}^{m \times n}$ astfel încât perechea cu o singură intrare $(A - BF, Bg)$ este controlabilă.

În general, testele de controlabilitate derivate prin aplicarea directă a criteriilor de mai sus nu sunt recomandabile din punct de vedere numeric. Criteriul 1 necesită calculul valorilor şi vectorilor proprii $x_i \in \mathcal{C}^n$ ai matricei A (ceea ce implică aplicarea algoritmului **QR** cu acumularea transformărilor de asemănare) precum şi testarea condiţiilor $B^T x_i \neq 0,\ i = 1:n$. Criteriul 2 necesită numai calculul valorilor proprii, în schimb presupune aplicarea repetată a algoritmului **DVS** pentru testarea condiţiilor de rang. În sfârşit, criteriul 3 este interesant deoarece reduce cazul general $m > 1$ la cazul aparent mai simplu $m = 1$. Deoarece controlabilitatea este o proprietate generică, matricea F precum şi vectorul g se pot alege aleator. Această idee poate fi deseori exploatată cu succes în legătură cu algoritmul 5.6, prezentat mai departe, dar în legătură cu aceasta trebuie să facem următoare observaţie de principiu.

Observaţia 5.3 De obicei în aplicaţii testele de controlabilitate nu se aplică *per se*, ci fac parte din anumite proceduri de sinteză mai generale, în care constituie condiţii de calcul sau/şi de validare a rezultatelor. (Un exemplu în acest sens îl constituie procedurile de alocare a polilor, prezentate în capitolul 6). De aceea este important ca secvenţele de calcul să se înlănţuie în mod natural, transformările datelor necesare la un moment dat fiind utilizate ca atare în fazele ulterioare ale procedurii considerate. $\diamond$

Din perspectiva observaţiei 5.3, criteriul 3 devine mai puţin semnificativ, întrucât presupune distrugerea perechii date (A, B) şi înlocuirea ei cu o pereche $(A-BF, Bg)$ transformată aleator precum şi renunţarea nemotivată la libertăţile suplimentare de sinteză existente în cazul $m > 1$ al perechilor cu mai multe intrări faţă de cazul $m = 1$.

În concluzie, testele "elementare" de controlabilitate sunt în general nesatisfăcătoare din punctul de vedere al eficienţei şi siguranţei numerice. Aplicarea lor practică este limitată la sisteme de mici dimensiuni, pentru care, eventual, pot constitui mijloace de calcul "cu hârtia şi creionul".

Teste elementare de observabilitate

În general, dacă o procedură **proc** calculează un rezultat $R = \mathbf{proc}(A, B)$ privind controlabilitatea perechii (A, B) atunci rezultatul dual Q, privind observabilitatea perechii (C, A), se obţine cu secvenţa

1. $R_d = \mathbf{proc}(A^T, C^T)$
2. $Q = R_d^T$

De exemplu, dacă $\mathbf{proc}(A, B)$ este algoritmul 5.4, care produce matricea de controlabilitate R a perechii (A, B), atunci secvenţa de mai sus produce

matricea de observabilitate Q a perechii (C, A).

5.5 Forma Hessenberg controlabilă

Considerăm sistemul liniar $S = (A, B, C)$ şi fie $\tilde{S} = (\tilde{A}, \tilde{B}, \tilde{C})$ un sistem *asemenea* cu S, având matricele

$$\tilde{A} = TAT^{-1}, \quad \tilde{B} = TB, \quad \tilde{C} = CT^{-1}, \tag{5.42}$$

unde $T \in \mathcal{R}^{n\times n}$ este o matrice nesingulară arbitrară. Avem

$$\tilde{A}\tilde{B} = TAB, \quad \tilde{A}^2\tilde{B} = \tilde{A}(\tilde{A}\tilde{B}) = TA^2B, \ \ \ldots$$

astfel încât între matricele de controlabilitate şi observabilitate ataşate sistemelor S şi $\tilde{S}$ există relaţiile

$$\tilde{R} = TR, \quad \tilde{Q} = QT^{-1}. \tag{5.43}$$

Deoarece multiplicarea cu o matrice nesingulară nu modifică rangul, din relaţiile (5.43) rezultă că matricele R şi $\tilde{R}$ au acelaşi rang, deci perechile (A, B) şi $(\tilde{A}, \tilde{B})$ sunt simultan controlabile sau necontrolabile. În mod similar (sau prin dualitate), perechile (C, A) şi $(\tilde{C}, \tilde{A})$ sunt simultan observabile sau neobservabile. Pe scurt, proprietăţile de controlabilitate şi observabilitate, precum şi rangurile matricelor corespunzătoare de controlabilitate şi observabilitate, sunt invarianţi în raport cu transformările de asemănare (5.42).

Dacă în (5.42) presupunem că matricea $T = U$ este ortogonală, deci sistemele S şi $\tilde{S}$ sunt *ortogonal asemenea*

$$\tilde{A} = UAU^T, \quad \tilde{B} = UB, \quad \tilde{C} = CU^T, \tag{5.44}$$

atunci relaţiile (5.43) devin

$$\tilde{R} = UR, \quad \tilde{Q} = QU^T. \tag{5.45}$$

Prin urmare, nu numai proprietăţile (binare!) de controlabilitate şi observabilitate ci şi valorile singulare $\sigma_i(R)$ şi $\sigma_i(Q)$, $i \in 1 : n$ ale matricelor de controlabilitate şi observabilitate sunt invarianţi ai sistemului (A, B, C) în raport cu transformările de asemănare ortogonală (5.44).

Observaţia 5.4 Invarianţii ortogonali $\sigma_i(R)$ şi $\sigma_i(Q)$, $i \in 1 : n$ au o deosebită importanţă în analiza sistemelor liniare, întrucât reprezintă o

măsură cantitativă a "gradelor" de controlabilitate şi observabilitate ale sistemului $S = (A, B, C)$. În special, cea mai mică valoare singulară $\sigma_n(R) \geq 0$ a lui R reprezintă distanţa (măsurată în norma spectrală) dintre perechea dată (A, B) şi mulţimea perechilor necontrolabile de aceeaşi formă. $\diamond$

În cele ce urmează vom descrie procedurile de aducere a perechii (A, B) la o formă simplă utilizând transformările de asemănare ortogonală (5.44), cu scopul identificării invarianţilor (ortogonali ai) acestei perechi şi, în particular, al testării controlabilităţii acesteia.

În consecinţă, vom considera o pereche (A, B) cu m intrări, nu neapărat controlabilă, şi pentru claritate vom discuta succesiv cazurile $m = 1$ şi $m > 1$. Peste tot vom presupune $B \neq 0$ şi vom nota

$$r \stackrel{\text{def}}{=} \text{rang} R, \quad \bar{r} \stackrel{\text{def}}{=} n - r, \tag{5.46}$$

unde evident r satisface inegalitatea $1 \leq r \leq n$.

Cazul $m = 1$

În cazul unei perechi (A, b) de ordin n cu o singură intrare, forma "simplă", rezultată prin aplicarea transformării (5.44), se numeşte *forma superior Hessenberg* şi este definită prin

$$\tilde{A} = UAU^T = \begin{bmatrix} \overbrace{A_R}^{r} & \overbrace{A_{R\bar{R}}}^{\bar{r}} \\ 0 & A_{\bar{R}} \end{bmatrix} \begin{matrix} \}r \\ \}\bar{r} \end{matrix}, \qquad \tilde{b} = Ub = \begin{bmatrix} b_R \\ 0 \end{bmatrix} \begin{matrix} \}r \\ \}\bar{r} \end{matrix} \tag{5.47}$$

în care perechea (A_R, b_R) de ordin r se află în forma superior Hessenberg ireductibilă

$$A_R = \begin{bmatrix} x & x & \dots & x & x \\ h_2 & x & \dots & x & x \\ & \ddots & \ddots & \vdots & \vdots \\ & & \ddots & x & \vdots \\ & & & h_r & x \end{bmatrix}, \quad b_R = \begin{bmatrix} h_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{5.48}$$

unde

$$h_i \neq 0, \quad i = 1 : r \tag{5.49}$$

iar x denotă elemente a căror valoare numerică nu are importanţă în context.

Dacă perechea inițială (A, b) este *controlabilă* atunci în (5.47) avem $r = n$, astfel încât blocurile $A_{\bar{R}}$ și $A_{R\bar{R}}$ dispar iar perechea $(\tilde{A}, \tilde{b}) = (A_R, b_R)$ are structura ireductibilă (5.48) cu $r = n$, adică

$$\tilde{A} = UAU^T = \begin{bmatrix} x & x & \dots & x & x \\ h_2 & x & \dots & x & x \\ & \ddots & \ddots & \vdots & \vdots \\ & & \ddots & x & \vdots \\ & & & h_n & x \end{bmatrix}, \quad \tilde{b} = Ub = \begin{bmatrix} h_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (5.50)$$

unde

$$h_i \neq 0, \quad i = 1 : n. \quad (5.51)$$

De aceea, structura ireductibilă din (5.50) se numește *forma superior Hessenberg controlabilă* a perechii (A, b).

Pe de altă parte, dacă perechea $(\tilde{A}, \tilde{b})$ are forma (5.50), nu neapărat ireductibilă, iar $k \geq 2$ este primul întreg pentru care $h_k = 0$, atunci această pereche admite evident o descompunere (deflație) de tip (5.47), în care $r = k - 1$. În acest sens, mai departe vom spune că structura (5.50), în general reductibilă, constituie *forma superior Hessenberg* (completă) a perechii (A, b).

Afirmația următoare relativă la (5.50) se demonstrează ușor prin calcul direct.

Propoziția 5.2 *Matricea de controlabilitate $\tilde{R}$ a unei perechi $(\tilde{A}, \tilde{b})$ de forma (5.50) este superior triunghiulară, mai precis avem*

$$\tilde{R} \stackrel{\text{def}}{=} \begin{bmatrix} \tilde{b} & \tilde{A}\tilde{b} & \dots & \tilde{A}^{n-1}\tilde{b} \end{bmatrix} = \begin{bmatrix} \tilde{r}_{11} & x & \dots & x \\ & \tilde{r}_{22} & \dots & x \\ & & \ddots & \vdots \\ & & & \tilde{r}_{nn} \end{bmatrix}, \quad (5.52)$$

în care elementele diagonale ale lui $\tilde{R}$ sunt

$$\tilde{r}_{kk} = \prod_{i=1}^{k} h_i, \quad k = 1 : n. \quad (5.53)$$

Prin urmare, perechea inițială (A, b) este controlabilă, deci matricea $R = U\tilde{R}$ este nesingulară, dacă și numai dacă sunt satisfăcute condițiile (5.51).

Pe o cale încă mai simplă, aceeaşi concluzie rezultă utilizând criteriul de controlabilitate furnizat de punctul 2 al propoziţiei 5.1. Într-adevăr, avem

$$\left[\begin{array}{cc} \tilde{b} & \tilde{A}-\lambda I \end{array}\right] = \left[\begin{array}{ccccc} h_1 & x & \dots & x & x \\ & h_2 & \dots & x & x \\ & & \ddots & \vdots & \vdots \\ & & & h_n & x \end{array}\right], \tag{5.54}$$

deci (5.51) implică evident rang $\left[\begin{array}{cc} \tilde{b} & \tilde{A}-\lambda I \end{array}\right] = n, \ \forall \lambda \in \mathcal{C}$. Demonstraţia implicaţiei inverse este propusă cititorului ca exerciţiu (de atenţie!).

La fel de simplu se demonstrează afirmaţia următoare relativă la (5.47).

Propoziţia 5.3 *Matricea de controlabilitate $\tilde{R}$ a unei perechi $(\tilde{A}, \tilde{b})$ de forma (5.47) are structură superior trapezoidală*

$$\tilde{R} = \left[\begin{array}{cc} R_R & x \\ 0 & 0 \end{array}\right], \tag{5.55}$$

în care $R_R \in \mathcal{R}^{r\times r}$ este matricea de controlabilitate a perechii (A_R, b_R).

Prin urmare, ordinul r al perechii (A_R, b_R) coincide cu rangul matricei de controlabilitate $R = U\tilde{R}$ al perechii (A, b) dacă şi numai dacă sunt satisfăcute condiţiile (5.49), deci perechea (A_R, b_R) din (5.47) este controlabilă.

În rezumat, sunt posibile două cazuri:

1. Perechea (A, b) este controlabilă, deci $r \stackrel{\text{def}}{=} \text{rang}R = n$.
În acest caz perechea (A, b) este ortogonal asemenea cu o pereche $(\tilde{A}, \tilde{b})$ în forma superior Hessenberg controlabilă (5.50), (5.51).

2. Perechea (A, b) nu este controlabilă, deci $r \stackrel{\text{def}}{=} \text{rang}R < n$.
În acest caz perechea (A, b) este ortogonal asemenea cu o pereche $(\tilde{A}, \tilde{b})$ în forma bloc superior triunghiulară (5.47), în care perechea (A_R, b_R) este controlabilă de ordin r, deci are structura (5.48), (5.49). (În particular, (5.47) poate coincide cu forma superior Hessenberg completă (5.50), în care r este cel mai mic întreg ≥ 1 astfel încât $h_{r+1} = 0$).

Aducerea unei perechi (A, b) cu o singură intrare la forma superior Hessenberg completă (5.50) se face printr-o procedură directă, a cărei schemă de calcul este următoarea.

1. Se determină un reflector $U_1 \in \mathcal{R}^{n\times n}$ astfel încât ultimele $n-1$ componente ale vectorului transformat $b \leftarrow U_1 b$ să fie nule.

2. Se aplică U_1 lui A, deci se calculează $A \leftarrow U_1 A U_1$.

3. Se aduce matricea A la forma superior Hessenberg $A \leftarrow \tilde{A} = \tilde{U}A\tilde{U^T}$, utilizând algoritmul **HQ**, [VI,IX].

Pasul 1 este elementar. Reflectorul U_1 este definit prin

$$U_1 = I_n - \frac{u_1 u_1^T}{\beta_1}$$

în care componentele $u_{i1},\ i = 1 : n$ ale vectorului $u_1 \in \mathcal{R}^n$ precum şi scalarul β_1 se determină cu formulele cunoscute

Procedura $U = \mathbf{H0}(b)$

1. $\sigma = sgn(b_1)(\sum_{i=1}^n b_i^2)^{1/2}$
2. $u_{11} = b_1 + \sigma; \quad u_{i1} = b_i, \quad i = 2 : n$
3. $\beta_1 = \sigma u_{11}$
4. $b_1 \leftarrow h_1 = -\sigma; \quad b_i = 0, \quad i = 2 : n$

iar ultima instrucţiune efectuează transformarea $b \leftarrow U_1 b$.

Pasul 2 se desfăşoară în două faze. În prima fază se calculează $A \leftarrow U_1 A$, aplicând U_1 matricei A partiţionate pe coloane. Obţinem

Procedura $A = \mathbf{H1}(U, A)$

1. `Pentru` $j = 1 : n$
   1. $\tau = (\sum_{i=1}^n u_{i1} a_{ij})/\beta_1$
   2. $a_{ij} \leftarrow a_{ij} - \tau u_{i1}, \ \ i = 1 : n.$

În a doua fază se calculează $A \leftarrow AU_1$ aplicând U_1 matricei A partiţionate pe linii, unde U_1 e simetric deci $AU_1 = (U_1 A^T)^T$. Obţinem

Procedura $A = \mathbf{H2}(A, U)$

1. `Pentru` $i = 1 : n$
   1. $\tau = (\sum_{i=1}^n u_{i1} a_{ji})/\beta_1$
   2. $a_{ij} \leftarrow a_{ij} - \tau u_{j1}, \ \ j = 1 : n.$

Pasul 3 constă din $n-2$ etape. La fiecare etapă se efectuează calcule similare cu cele descrise la paşii 1 şi 2 de mai sus, operând efectiv asupra unor perechi "restante" $(\bar{A}, \bar{b})$ de ordin din ce în ce mai mic. Pentru a simplifica referirile ulterioare şi, în special, redactarea algoritmului general, corespunzător cazului $m > 1$, mai jos vom trece în revistă principalele momente ale calculului.

Presupunem că înainte de etapa k, $k = 2 : n-1$, perechea (A, b) (parţial transformată în etapele anterioare) are structura

$$A = \left[\begin{array}{cccc|ccc} x & \dots & x & x & x & \dots & x \\ h_2 & \dots & x & x & x & \dots & x \\ & \ddots & \vdots & \vdots & \vdots & & \vdots \\ & & h_{k-1} & x & x & \dots & x \\ \hline & & & a_{k,k-1} & a_{kk} & \dots & a_{kn} \\ & 0 & & \vdots & \vdots & & \vdots \\ & & & a_{n,k-1} & a_{nk} & \dots & a_{nn} \end{array}\right] = \left[\begin{array}{c|c|c} \tilde{A}_k & & X_k \\ \hline 0 & \bar{b}_k & \bar{A}_k \end{array}\right], \tag{5.56}$$

$$b = \left[\begin{array}{c} h_1 \\ 0 \\ \vdots \\ 0 \\ \hline 0 \\ \vdots \\ 0 \end{array}\right] = \left[\begin{array}{c} \tilde{b}_k \\ \hline 0 \end{array}\right]$$

în care perechea $(\tilde{A}_k, \tilde{b}_k)$ este deja în formă superior Hessenberg completă de ordin $k-1$.

La etapa k se efectuează următoarele calcule:

3.1. Se determină un reflector $U_k \in \mathcal{R}^{n\times n}$ astfel încât ultimele $n-k$ elemente din coloana $k-1$ a matricei A transformate să fie nule.

3.2. Se aplică U_k lui A, deci se calculează $A \leftarrow U_k A U_k$.

La pasul 3.1 reflectorul U_k este definit prin

$$U_k = I_n - \frac{u_k u_k^T}{\beta_k},$$

în care vectorul $u_k \in \mathcal{R}^n$ are primele $k-1$ componente nule. În virtutea acestui fapt, U_k are structura bloc

$$U_k = \left[\begin{array}{cc} I_{k-1} & 0 \\ 0 & \bar{U}_k \end{array}\right], \tag{5.57}$$

iar datorită acesteia premultiplicarea cu U_k nu modifică primele $k-1$ linii iar postmultiplicarea cu U_k nu modifică primele $k-1$ coloane ale tablourilor

transformate. Prin urmare, la pasul 3.2 obţinem

$$U_kA = \left[\begin{array}{c|c|c} \tilde{A}_k & & X_k \\ \hline 0 & \bar{U}_k\bar{b}_k & \bar{U}_k\bar{A}_k \end{array}\right] \stackrel{\text{def}}{=} \left[\begin{array}{c|c|c} \tilde{A}_k & & \\ \hline 0 & \bar{U}_k\bar{b}_k & Y_k \end{array}\right] \tag{5.58}$$

precum şi

$$(U_kA)U_k = \left[\begin{array}{c|c|c} \tilde{A}_k & & \\ \hline 0 & \bar{U}_k\bar{b}_k & Y_k\bar{U}_k \end{array}\right] \stackrel{\text{def}}{=} \left[\begin{array}{c|c|c} \tilde{A}_{k+1} & & X_{k+1} \\ \hline 0 & \bar{b}_{k+1} & \bar{A}_{k+1} \end{array}\right] \tag{5.59}$$

$$U_kb = b = \left[\begin{array}{c} \tilde{b}_{k+1} \\ \hline \\ 0 \end{array}\right]$$

unde, datorită alegerii lui U_k de la pasul 3.1, perechea $(\tilde{A}_{k+1}, \tilde{b}_{k+1})$ rezultă în forma superior Hessenberg completă de ordin k.

Astfel procedura a progresat, adică dimensiunea părţii transformate (marcate cu tildă) a crescut, iar cea a părţii restante (barate) a scăzut cu o unitate.

După a $(n-1)$-a etapă vom obţine (A, b) în forma dorită.

Pentru organizarea procedurală a calculelor, este esenţial să observăm că, în acord cu (5.57), reflectorul U_k este complet determinat de blocul său $\bar{U}_k$, care este el însuşi un reflector de ordin $n-k+1$. Ţinând seama de această observaţie, precum şi de modul de acţiune al lui $\bar{U}_k$ în (5.58), rezultă că pasul 3.1 se reduce la aplicarea procedurii **H0** vectorului

$$(\mathbf{H0}) \qquad \bar{b}_k = A(k:n, k-1)$$

(În acest fel se determină componentele nenule u_{ik}, $i = k:n$ ale vectorului u_k, scalarul β_k precum şi vectorul transformat $\bar{b}_k \leftarrow \bar{U}_k\bar{b}_k$ din (5.59)).

Similar, pasul 3.2 se reduce la efectuarea transformărilor $\bar{A}_k \leftarrow \bar{U}_k\bar{A}_k$ şi $Y_k \leftarrow Y_k\bar{U}_k$ din (5.58) şi (5.59), ceea ce necesită aplicarea procedurilor **H1** şi **H2** submatricelor lui A definite prin

$$(\mathbf{H1}) \qquad \bar{A}_k = A(k:n, k:n)$$

şi respectiv

$$(\mathbf{H2}) \qquad Y_k = A(1:n, k:n).$$

În sfârşit, ultima relaţie (5.59) arată că reflectorul U_k lasă b invariant, deci fiecare etapă k, $k = 2:n-1$ conservă automat asemănarea ortogonală a perechilor transformate (A, b).

În rezumat, procedura de aducere a perechii (A, b) cu o singură intrare la forma superior Hessenberg completă poate fi formulată astfel.

1. $U_1 = \mathbf{H0}(b)$.
2. $A = \mathbf{H1}(U_1, A); \quad A = \mathbf{H2}(A, U_1)$.
3. `Pentru` $k = 2 : n-1$
   1. $U_k = \mathbf{H0}(A(k:n, k-1))$.
   2. $A(k:n, k:n) = \mathbf{H1}(U_k, A(k:n, k:n))$;
      $A(:, k:n) = \mathbf{H2}(A(:, k:n), U_k)$

Construcţia matricei ortogonale $U = U_{n-1} \cdots U_2 U_1$ din (5.50), i.e. acumularea transformărilor, se face după schema

1. $U = U_1$
2. Pentru $k = 2 : n-1$
   1. $U(k:n, 1:n) = \mathbf{H1}(U_k, U(k:n, 1:n))$.

În practică, se preferă deseori obţinerea formei superior Hessenberg (5.47),(5.48),(5.49), ceea ce evident presupune testarea condiţiei $\|\bar{b}_k\| \neq 0$ înainte de parcurgerea etapei k, $k = 2 : n-1$. Se obţine următoarea procedură de calcul.

Algoritmul 5.6 (Construieşte forma superior Hessenberg (5.47), (5.48), (5.49) a unei perechi (A, b) de ordin n cu o singură intrare şi acumulează transformările. Se presupune $b \neq 0$).

1. $k = 1$
2. $U_1 = \mathbf{H0}(b)$
3. $A = \mathbf{H1}(U_1, A), \quad A = \mathbf{H2}(A, U_1)$
4. $U = U_1$
5. CONTINUĂ='da'
6. `Cât timp` CONTINUĂ='da'
   1. $r = k$
   2. `Dacă` $k = n$
      `atunci`
      1. CONTINUĂ ='nu',
      2. `tipăreşte` "Perechea (A, b) este controlabilă"

`altfel`

1. $g = A(k+1:n, k)$
2. `Dacă` $\|g\| = 0$
   `atunci`
   1. CONTINUĂ ='nu'
   `altfel`
   1. $k \leftarrow k+1$ % Etapa $k, k \geq 2$
   2. $U_k = \mathbf{H0}(g)$
   3. $A(k:n, k:n) = \mathbf{H1}(U_k, A(k:n, k:n))$, $A(:, k:n) = \mathbf{H2}(A(:, k:n), U_k)$
   4. $U(k:n, 1:n) = \mathbf{H1}(U_k, U(k:n, 1:n))$

Comentarii. CONTINUĂ este o variabilă logică ce determină avansarea contorului de etapă k. În final, variabila r furnizează rangul matricei de controlabilitate R a perechii date (A, b). În cazul $r = n$, deci dacă perechea (A, b) este controlabilă (vezi pasul 6.2) numărul de operaţii necesar este de ordinul $5\frac{n^3}{3}$. $\diamond$

Cazul $m > 1$

În cazul unei perechi (A, B) de ordin n cu m intrări, $m \geq 1$, procedura de transformare are la bază următorul rezultat general.

Propoziţia 5.4 (Lema de deflaţie controlabilă). *Oricare ar fi perechea (A, B) cu $rang B = r_1 > 0$, există o matrice ortogonală $U_1 \in \mathcal{R}^{n\times n}$ astfel încât*

$$A \leftarrow \tilde{A} = U_1 A U_1^T = \begin{bmatrix} A_1 & X \\ G & F \end{bmatrix}, \quad B \leftarrow \tilde{B} = U_1 B = \begin{bmatrix} H_1 \\ 0 \end{bmatrix} \tag{5.60}$$

unde matricea $H_1 \in \mathcal{R}^{r_1\times m}$ este epică, i.e. $rang H_1 = r_1$. Mai mult, perechea (A, B) este controlabilă dacă şi numai dacă perechea redusă (F, G), de ordin $n - r_1 < n$, este controlabilă.

Demonstraţie. Matricea U_1 se determină aplicând lui B procedura de descompunere a valorilor singulare (**DVS**) sau procedura de triangularizare ortogonală cu pivotarea coloanelor. În primul caz avem

$$U_1 B V = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix}, \tag{5.61}$$

unde U_1 şi $V \in \mathcal{R}^{m\times m}$ sunt ortogonale iar Σ_1 este diagonală de ordin $r_1 > 0$ cu elementele diagonale pozitive. Partiţionând $V = [V_1\ V_2]$, unde V_1 are r_1 coloane, obţinem

$$\tilde{B} = U_1 B = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = \begin{bmatrix} \Sigma_1 V_1^T \\ 0 \end{bmatrix},$$

unde $H_1 = \Sigma_1 V_1^T$ este evident epică de rang r_1. În al doilea caz obţinem

$$U_1 B P = \begin{bmatrix} R_1 & R_2 \\ 0 & 0 \end{bmatrix}, \tag{5.62}$$

unde U_1 este ortogonală iar $P \in \mathcal{R}^{m\times m}$ este o matrice de permutare a coloanelor lui B, aleasă pe parcursul procedurii de triangularizare ortogonală cu reflectori astfel încât matricea superior triunghiulară R_1 de ordin r_1 să rezulte nesingulară. În acest caz matricea $\tilde{B} = U_1 B$ rezultă din nou în forma (5.60) cu $H_1 = [\,R_1\ \ R_2\,]P^T$ [1].

După determinarea lui U_1 printr-una din procedurile (5.61) sau (5.62) se aplică U_1 lui A, i.e. se calculează $\tilde{A} = U_1 A U_1^T$, şi în acord cu (5.60) se evidenţiază perechea redusă (F, G) de ordin $n - r_1$ cu r_1 intrări.

A doua afirmaţie din lemă se demonstrează ca în cazul $m = 1$, utilizând punctul 2 al propoziţiei 5.1. $\diamond$

În ipoteza $G \neq 0$, perechii (F, G) cu $rang\ G = r_2 > 0$ i se poate aplica din nou o transformare de tip (5.60). Se obţine

$$F \leftarrow \tilde{F} = \bar{U}_2 F \bar{U}_2^T = \begin{bmatrix} A_2 & X \\ G_{nou} & F_{nou} \end{bmatrix}, \quad G \leftarrow \tilde{G} = \bar{U}_2 G = \begin{bmatrix} H_2 \\ 0 \end{bmatrix}$$

sau echivalent

$$\begin{aligned} A \leftarrow \tilde{A} = U_2 A U_2^T &= \left[\begin{array}{c|cc} A_1 & X & X \\ \hline H_2 & A_2 & X \\ 0 & G_{nou} & F_{nou} \end{array}\right], \\ B \leftarrow \tilde{B} = U_2 B &= \left[\begin{array}{c} H_1 \\ \hline 0 \\ 0 \end{array}\right], \end{aligned} \tag{5.63}$$

[1] În cazul $m = 1$, considerat anterior, avem $r_1 = 1$, U_1 este reflectorul care anulează ultimele $n - 1$ elemente ale matricei $B = b \in \mathcal{R}^n$ iar $P = 1$. Această constatare evidenţiază clar complicaţiile de calcul ce apar în cazul $m > 1$ faţă de cazul simplu $m = 1$.

unde matricele $\bar{U}_2$ precum şi

$$U_2 = \begin{bmatrix} I_{r_1} & 0 \\ 0 & \bar{U}_2 \end{bmatrix} \tag{5.64}$$

sunt ortogonale iar matricea H_2 este epică de rang r_2.

Forma finală a perechii (A, B), obţinută prin aplicarea repetată a procedurii de deflaţie descrisă în propoziţia 5.4, se numeşte *forma bloc-superior Hessenberg* şi este definită prin

$$A \leftarrow \tilde{A} = UAU^T = \begin{bmatrix} A_R & A_{R\bar{R}} \\ 0 & A_{\bar{R}} \end{bmatrix}, \quad B \leftarrow \tilde{B} = UB = \begin{bmatrix} B_R \\ 0 \end{bmatrix}, \tag{5.65}$$

în care perechea (A_R, B_R), de ordin r, se găseşte în forma *bloc-superior Hessenberg controlabilă*

$$A_R = \begin{bmatrix} A_1 & X & \cdots & X & X \\ H_2 & A_2 & \cdots & X & X \\ & \ddots & \ddots & \vdots & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & H_k & A_k \end{bmatrix}, \quad B_R = \begin{bmatrix} H_1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \tag{5.66}$$

unde toate blocurile $H_i \in \mathcal{R}^{r_i \times r_{i-1}}$ sunt epice, i.e.

$$\text{rang} H_i = r_i > 0, \quad i = 1 : k, \quad r_0 \stackrel{\text{def}}{=} m. \tag{5.67}$$

Evident avem

$$r = \sum_{i=1}^{k} r_i \leq n \tag{5.68}$$

şi în virtutea condiţiilor (5.67) perechea (A_R, B_R) din (5.65) este controlabilă. Mai mult, *perechea iniţială* (A, B) *este controlabilă* dacă şi numai dacă $r = n$, i.e. $(\tilde{A}, \tilde{B}) = (A_R, B_R)$, şi în acest caz numărul k de blocuri din (5.66) coincide cu *indicele de controlabilitate* ν al lui (A, B).

Matricea ortogonală $U \in \mathcal{R}^{n \times n}$ din (5.65) rezultă prin acumularea transformărilor parţiale, i.e.

$$U = U_k \cdots U_2 U_1, \tag{5.69}$$

în care U_1 se determină ca în demonstraţia lemei 5.4, U_2 are forma (5.64) etc. Subliniem că în virtutea structurii lui U_2, premultiplicarea cu U_2 nu modifică primele r_1 linii, iar postmultiplicarea cu U_2^T nu modifică primele

r_1 coloane ale matricei asupra căreia acţionează transformarea corespunzătoare. De asemenea, proprietăţi asemănătoare au toate matricele $U_i,\ i = 2:k$ din (5.69).

În rezumat, aducerea pe loc a perechii iniţiale (A, B) la forma bloc-superior Hessenberg (5.65), (5.66) se face printr-o succesiune de transformări ortogonale de asemănare

$$A \leftarrow U_i A U_i^T, \quad B \leftarrow U_i B, \quad i = 1:k. \tag{5.70}$$

La etapa $i = 1$ se operează asupra perechii iniţiale (A, B) ca în demonstraţia propoziţiei 5.4, iar la etapele $i \geq 2$ se operează analog asupra perechii reduse corespunzătoare (F, G), conţinute în tabloul A. În particular, transformările $U_i,\ i \geq 2$ nu modifică forma lui B obţinută după prima etapă, deci pentru $i \geq 2$ relaţiile (5.70) se reduc la

$$A \leftarrow U_i A, \quad A \leftarrow A U_i^T. \tag{5.71}$$

Pentru a descrie precis algoritmul astfel obţinut, notăm cu

$$[\, r_1, U_1, H_1 \,] = \mathbf{red}(B) \tag{5.72}$$

procedura de reducere a lui B decrisă în demonstraţia propoziţiei 5.4, vezi relaţiile (5.60) şi (5.61) sau (5.62). (Amintim că la etapele $i \geq 2$ procedura **red** se aplică unui bloc G, localizat în tabloul A). De asemenea, notăm transformările (5.71) respectiv prin

$$A = \mathbf{mul1}(U_i, A), \quad A = \mathbf{mul2}(A, U_i),$$

convenind să indicăm explicit de fiecare dată elementele lui A care sunt efectiv modificate.

Acumularea transformărilor din (5.69) se face conform schemei

$$U \leftarrow U_i U, \quad i = 2:k \tag{5.73}$$

cu iniţializarea $U = U_1$.

Algoritmul 5.7 (Dată perechea (A, B) de ordin n cu m intrări şi $B \neq 0$ algoritmul construieşte forma bloc-superior Hessenberg (5.65), (5.66), (5.67) şi acumulează transformările. Perechea rezultată suprascrie pe cea iniţială).

1. $k = 1$
2. $[r_1, U_1, H_1] = \mathbf{red}(B)$

3. $A = \mathbf{mul1}(U_1, A), \quad A = \mathbf{mul2}(A, U_1)$
4. $U = U_1$
5. $r_v = 0, \quad r = r_1$
6. CONTINUĂ = 'da'
7. `Cât timp` CONTINUĂ = 'da'
   1. $\nu = k$
   2. `Dacă` $r = n$
      `atunci`
      1. CONTINUĂ = 'nu'
      2. `Tipăreşte` "Perechea (A, B) este controlabilă"
      `altfel`
      1. $G = A(r+1 : n, r_v + 1 : r)$
      2. `Dacă` $\|G\| = 0$
         `atunci`
         1. CONTINUĂ = 'nu'
         `altfel`
         1. $k \leftarrow k + 1 \quad$ % (etapa k, $k \geq 2$)
         2. $[r_k, U_k, H_k] = \mathbf{red}(G)$
         3. $A(r+1 : n, r+1 : n) =$
            $\mathbf{mul1}(U_k, A(r+1 : n, r+1 : n),$
            $A(1 : n, r+1 : n) = \mathbf{mul2}(A(1 : n, r+1 : n), U_k)$
         4. $U(r+1 : n, 1 : n) = \mathbf{mul1}(U_k, U(r+1 : n, 1 : n))$
         5. $r_v = r, \quad r \leftarrow r + r_k$

Comentarii. Având în vedere că numărul k al etapelor de reducere nu este apriori cunoscut, utilizăm şi aici variabila binară CONTINUĂ cu semnificaţia cunoscută. În final, variabila r furnizează rangul matricei de controlabilitate R a perechii date (A, B). În cazul $r = n$, i.e. dacă perechea (A, B) este controlabilă, algoritmul se opreşte la instrucţiunea 7.2. iar ν este indicele de controlabilitate. Din contră, dacă algoritmul se opreşte datorită faptului că blocul curent G este nul, atunci perechea (A, B) nu este controlabilă, iar perechea transformată $(\tilde{A}, \tilde{B})$, furnizată de algoritmul 5.7, are forma (5.65), în care $r < n$. $\diamond$

Forma Hessenberg observabilă

Pentru analiza proprietăţilor de observabilitate a unei perechi (C, A) de ordin n cu l ieşiri, $l \geq 1$, se procedează prin dualitate. Presupunem $C \neq 0$

şi fie $q_1 = rang\, C > 0$. Lema de deflaţie observabilă (vezi propoziţia 5.4) afirmă existenţa unei matrice ortogonale $U_1 \in \mathcal{R}^{n\times n}$, astfel încât

$$\begin{aligned} A \leftarrow \tilde{A} = U_1 A U_1^T = & \begin{bmatrix} A_1 & H \\ X & F \end{bmatrix}, \\ C \leftarrow \tilde{C} = \quad CU_1^T = & \begin{bmatrix} G_1 & 0 \end{bmatrix}, \end{aligned} \tag{5.74}$$

unde matricea $G_1 \in \mathcal{R}^{l\times q_1}$ este *monică*, i.e. rang$G_1 = q_1$. În plus, perechea (C, A) este observabilă dacă şi numai dacă perechea redusă (H, F) este observabilă.

Prin aplicarea repetată a acestei proceduri de deflaţie, sau echivalent, prin aplicarea schemei de dualizare relativ la algoritmul 5.7, se obţine *forma bloc-inferior Hessenberg* a perechii (C, A), definită prin

$$\begin{aligned} A \leftarrow \tilde{A} = UAU^T = & \begin{bmatrix} A_Q & 0 \\ A_{\bar{Q}Q} & A_{\bar{Q}} \end{bmatrix}, \\ C \leftarrow \tilde{C} = \quad CU^T = & \begin{bmatrix} C_Q & 0 \end{bmatrix}, \end{aligned} \tag{5.75}$$

în care perechea (C_Q, A_Q) de ordin q se află în forma *bloc-inferior Hessenberg observabilă*

$$A_Q = \begin{bmatrix} A_1 & G_2 & & & \\ X & A_2 & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ X & X & \cdots & \ddots & G_k \\ X & X & \cdots & \cdots & A_k \end{bmatrix}, \quad C_Q = \begin{bmatrix} G_1 & 0 & \cdots & 0 \end{bmatrix}, \tag{5.76}$$

unde toate blocurile $G_i \in \mathcal{R}^{q_{i-1}\times q_i}$ sunt monice, i.e.

$$\text{rang}G_i = q_i > 0, \quad i = 1:k, \quad q_0 \stackrel{\text{def}}{=} l. \tag{5.77}$$

Evident avem

$$q = \sum_{i=1}^{k} q_i \leq n \tag{5.78}$$

şi în virtutea condiţiilor (5.77) perechea (C_Q, A_Q) din (5.75) este observabilă. În plus, *perechea iniţială* (C, A) este *observabilă* dacă şi numai dacă $q = n$, i.e. $(\tilde{C}, \tilde{A}) = (C_Q, A_Q)$, şi în acest caz numărul k de blocuri din (5.76) coincide cu *indicele de observabilitate* al lui (C, A).

În sfârşit, la fel ca în cazul controlabilităţii, matricea ortogonală $U \in \mathcal{R}^{n\times n}$ din (5.75) rezultă din acumularea transformărilor parţiale, i.e.

$$U = U_k \cdots U_2 U_1,$$

în care U_1 se determină ca în (5.74).

5.6 Descompunerea controlabilă

Fie (A, B) o pereche, în general necontrolabilă, de ordin n cu m intrări, unde $m \geq 1$. Perechea bloc-superior Hessenberg $(\tilde{A}, \tilde{B}) = (UAU^T, UB)$, obţinută aplicând algoritmul 5.7 expus în paragraful anterior, are forma (5.65) cu $r \leq n$ şi constituie prin definiţie, *descompunerea controlabilă* a perechii date (A, B). (În cazul $m = 1$ se utilizează (5.47)). Perechea controlabilă (A_R, B_R) de ordin r cu m intrări reprezintă *partea controlabilă*, iar matricea $A_{\bar{R}}$ de ordin $\bar{r} = n - r$ reprezintă *partea total necontrolabilă* a lui (A, B).

Semnificaţia sistemică a descompunerii controlabile este evidenţiată de graful de semnal asociat, vezi figura 5.1(a), din care se vede că partea necontrolabilă nu este influenţată de intrarea u nici direct, nici indirect (i.e. prin intermediul părţii controlabile), iar matricea $A_{R\bar{R}}$ caracterizează acţiunea părţii total necontrolabile asupra părţii controlabile. (De aceea, în aplicaţii, partea necontrolabilă modelează perturbaţiile exterioare ce acţionează asupra sistemului dat, considerat controlabil). Prin definiţie, valorile proprii ale matricei $A_{\bar{R}}$ constituie valorile proprii necontrolabile sau *polii fixi ai perechii* (A, B).

Principalele elemente de analiză a proprietăţilor de controlabilitate asociate unei perechi oarecare (A, B) sunt furnizate direct de algoritmul 5.7 şi pot fi rezumate astfel.

1. *Testarea controlabilităţii* se face pe baza egalităţii $r = n$. Altfel spus, perechea (A, B) este *controlabilă* dacă şi numai dacă partea sa total necontrolabilă este vidă.

2. *Spaţiul controlabil* $\mathcal{R} \subset R^n$ al perechii (A, B) este definit prin $\mathcal{R} = \mathrm{Im}R$, unde R este matricea de controlabilitate corespunzătoare. (Amintim că prin $\mathrm{Im}R$, unde matricea R are n linii, notăm subspaţiul liniar al lui $\mathcal{R}^n$ generat de coloanele lui R, vezi (5.26)). Avem evident $r = \mathrm{rang}R = \dim\mathcal{R}$, unde r este ordinul perechii (A_R, B_R) din (5.65). Mai mult, partiţionând conform matricea de transformare U, i.e.

$$U = \begin{bmatrix} U_R \\ U_{\bar{R}} \end{bmatrix} \begin{matrix} \}r \\ \}\bar{r} = n - r \end{matrix} , \tag{5.79}$$

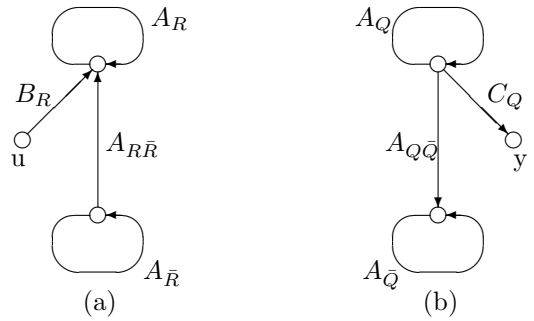

Figura 5.1: Semnificaţia sistemică a *descompunerii controlabile* (a) şi a *descompunerii observabile* (b).

şi, utilizând aceleaşi raţionamente ca în § 2, constatăm că – în coordonatele iniţiale – coloanele matricei U_R^T formează o *bază ortogonală* a subspaţiului controlabil $\mathcal{R}$.

În coordonatele lui $\mathcal{R}^n$ corespunzătoare reprezentării (5.65) subspaţiul controlabil $\mathcal{R}$ este format din totalitatea vectorilor de forma

$$\tilde{x} = \begin{bmatrix} x_R \\ 0 \end{bmatrix} \begin{matrix} \}r \\ \}\bar{r} \end{matrix} \quad , \tag{5.80}$$

iar structura perechii transformate $(\tilde{A}, \tilde{B})$ din (5.65) arată că

(i) subspaţiul $\mathcal{R}$ este A-invariant şi

(ii) conţine $\mathrm{Im}B$.

Mai mult, în virtutea faptului că perechea (A_R, B_R) este controlabilă (deci nu poate fi descompusă la rândul ei ca în (5.65)), subspaţiul $\mathcal{R}$ este *cel mai mic* subspaţiu al lui $\mathcal{R}^n$ cu proprietăţile de mai sus.

În legătură cu aceasta, subliniem că noţiunea de subspaţiu necontrolabil al perechii (A, B) nu are semnificaţie geometrică invariantă deoarece orice subspaţiu $\bar{\mathcal{R}}$ (vezi figura 5.2) cu proprietatea $R^n = \mathcal{R} \bigoplus \bar{\mathcal{R}}$ poate fi numit astfel. În aplicaţii se consideră de obicei complementul ortogonal $\bar{\mathcal{R}} = \mathcal{R}^{\perp}$, pentru care matricea $U_{\bar{R}}^T$ defineşte evident o bază ortogonală. (Simplitatea acestor construcţii se datorează faptului că matricea de transformare (5.80) este ortogonală, i.e. coloanele lui U^T constituie o bază ortogonală a întregului spaţiu al stărilor $\mathcal{R}^n$).

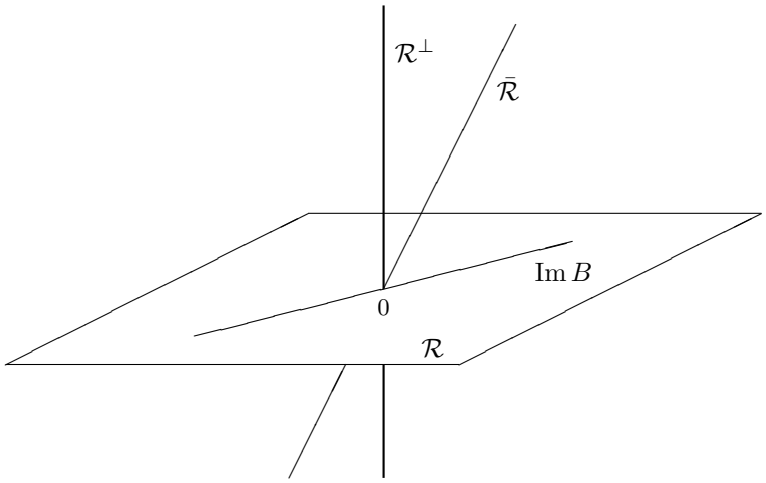

Figura 5.2: Reprezentarea geometrică a spaţiului controlabil.

Descompunerea observabilă

Fie (C, A) o pereche în general neobservabilă de ordin n cu l ieşiri, unde $l \geq 1$. Perechea bloc-inferior Hessenberg $(\tilde{C}, \tilde{A}) = (CU^T, UAU^T)$, obţinută aplicând schema de dualizare relativ la algoritmul 5.7, expus în paragraful anterior, are forma (5.75) cu $q \leq n$, şi constituie, prin definiţie, *descompunerea observabilă* a perechii date (C, A).

Semnificaţia sistemică a descompunerii observabile este evidenţiată de graful de semnal asociat, vezi figura 5.1(b). Prin definiţie, valorile proprii ale matricei $A_{\bar{Q}}$ constituie valorile proprii neobservabile sau *polii ficşi ai perechii* (C, A).

Principalele elemente de analiză a proprietăţilor de observabilitate asociate unei perechi oarecare (C, A) sunt propuse ca exerciţii.

5.7 Stabilizabilitatea şi detectabilitatea sistemelor liniare

În problemele de sinteză vizând stabilizarea sistemelor liniare prin reacţie după stare sau prin compensare dinamică, interesează o proprietate mai

slabă decât controlabilitatea, numită *stabilizabilitate* şi definită cu referire la descompunerea controlabilă (5.65).

Definiţia 5.6 *Perechea* (A, B) *este* stabilizabilă *dacă partea sa total necontrolabilă* $A_{\bar{R}}$ *este stabilă sau, echivalent, polii ficşi ai perechii* (A, B) *sunt stabili.*

În consecinţă, principalele elemente de analiză a proprietăţilor de stabilizabilitate asociate unei perechi (A, B) rezultă combinând procedurile de analiză a proprietăţilor fundamentale de stabilitate şi controlabilitate, stabilite în §§ 1, 2 şi §§ 5, 6.

1. *Testarea stabilizabilităţii* se face pe baza definiţiei 5.6, i.e. construind descompunerea controlabilă (5.65) a perechii (A, B) şi apoi aplicând algoritmul 5.1 matricei $A_{\bar{R}}$. Alternativ, se poate construi descompunerea spectrală a perechii (A, B) ca în §2, după care se aplică algoritmul 5.7 perechii total nestabile (A_b, B_b).

2. *Spaţiul stabilizabil* $\mathcal{S} \subset R^n$ al perechii (A, B) este definit prin $\mathcal{S} = \mathcal{R} + \mathcal{X}^-(A)$, astfel încât evidenţierea sa se face construind descompunerea controlabilă (5.65) a perechii (A, B) şi, apoi, aplicând algoritmul 5.2 matricei $A_{\bar{R}}$.

În urma acestor două transformări se obţine, evident, *descompunerea stabilizabilă* a perechii date (A, B), care evidenţiază partea stabilizabilă (controlabilă sau necontrolabilă dar stabilă), respectiv partea total nestabilizabilă (necontrolabilă şi nestabilă) a perechii considerate. (În aplicaţii, partea nestabilizabilă modelează perturbaţiile exterioare persistente ce acţionează asupra sistemului dat, considerat stabilizabil).

În problemele de proiectare a estimatoarelor de stare stabile interesează o proprietate mai slabă decât observabilitatea, numită *detectabilitate* şi definită prin dualitate faţă de stabilizabilitate cu referire la descompunerea observabilă (5.75).

Definiţia 5.7 *Perechea* (C, A) *este* detectabilă *dacă partea total neobservabilă* $A_{\bar{Q}}$ *este stabilă sau, echivalent, polii ficşi ai perechii* (C, A) *sunt stabili.*

Principalele elemente de analiză a proprietăţilor de detectabilitate asociate unei perechi (C, A) sunt propuse ca exerciţii.

5.8 Realizări minimale

Principalul rezultat al teoriei realizării sistemelor liniare afirmă că orice sistem $S = (A, B, C)$ este echivalent intrare-ieşire, i.e. are aceeaşi matrice de transfer, cu un sistem de ordin minim $S_m = (A_m, B_m, C_m)$ care este simultan controlabil şi observabil. Mai mult, sistemul S_m este determinat până la o transformare de asemănare şi, în esenţă, coincide cu partea simultan controlabilă şi observabilă a sistemului dat S.

Definiţia 5.8 *Un sistem $S_m = (A_m, B_m, C_m)$ controlabil şi observabil, echivalent intrare-ieşire cu S se numeşte* realizare minimală *a lui S.*

Mai general, orice sistem $S_m = (A_m, B_m, C_m)$ controlabil şi observabil se numeşte minimal.

În consecinţă, o realizare minimală (i.e. de ordin minim) a unei matrice de transfer date $T(s)$ poate fi construită aplicând următoarea procedură.

Algoritmul 5.8 (Construieşte o realizare minimală a matricei de transfer $T(s)$).

1. Se construieşte o realizare observabilă $S = (A, B, C)$ a lui $T(s)$, utilizând formele standard observabile, expuse în capitolul 3.
2. Se construieşte descompunerea controlabilă
$$\tilde{A} = UAU^T = \begin{bmatrix} A_R & A_{R\bar{R}} \\ 0 & A_{\bar{R}} \end{bmatrix}, \qquad \tilde{B} = UB = \begin{bmatrix} B_R \\ 0 \end{bmatrix},$$
utilizând algoritmul 5.7 şi se aplică transformarea U matricei C, i.e.
$$\tilde{C} = CU^T = [\, C_R \;\; C_{\bar{R}} \,].$$
3. Se reţine partea controlabilă $S_m = (A_R, B_R, C_R)$ a tripletului transformat, unde
$$A_R = \tilde{A}(1:r, 1:r), \quad B_R = \tilde{B}(1:r, :)$$
este partea controlabilă a perechii (A, B) iar
$$C_R = \tilde{C}(:, 1:r).$$

Procedurile de identificare sistemică, bazate pe teoria realizării sistemelor liniare, sunt prezentate în ANEXA A.

Procedurile de reducţie dimensională aproximativă a unor sisteme controlabile şi observabile (deci minimale) au la bază conceptul fundamental de realizare echilibrată şi vor fi expuse în §10.

5.9 Gramieni de controlabilitate şi observabilitate

Pentru a introduce cât mai simplu noţiunea de gramian în contextul sistemic al discuţiei noastre, considerăm din nou sistemul liniar discret $S = (A, B, C)$ definit prin relaţiile (5.17).

Utilizând matricea de controlabilitate în k paşi R_k a perechii (A, B) şi ţinând seama de relaţiile (5.19) şi (5.22) definim *gramianul de controlabilitate în k paşi* prin

$$Y_k \stackrel{\text{def}}{=} R_k R_k^T = \sum_{i=0}^{k-1} A^i B B^T (A^T)^i. \tag{5.81}$$

Evident, Y_k este o matrice $n \times n$ simetrică pozitiv semidefinită, pe scurt $Y_k = Y_k^T \geq 0,\ k \geq 1$. Ţinând seama de relaţia (5.38) putem scrie

$$Y_{k+1} = R_{k+1} R_{k+1}^T = \left[\begin{array}{cc} B & AR_k \end{array}\right] \left[\begin{array}{c} B^T \\ R_k^T A^T \end{array}\right],$$

prin urmare şirul Y_k satisface ecuaţia matriceală recurentă Liapunov discretă (DEML)

$$Y_{k+1} = BB^T + AY_kA^T, \tag{5.82}$$

cu condiţia iniţială $Y_1 = BB^T$ (sau $Y_0 = 0$).

Prin dualitate, utilizând matricea de observabilitate în k paşi Q_k a perechii (C, A), definim *gramianul de observabilitate în k paşi* prin

$$X_k \stackrel{\text{def}}{=} Q_k^T Q_k = \sum_{i=0}^{k-1} (A^T)^i C^T C A^i, \tag{5.83}$$

unde $X_k = X_k^T \geq 0,\ k \geq 1$ satisface DEML

$$X_{k+1} = C^T C + A^T X_k A \tag{5.84}$$

cu condiţia iniţială $X_1 = C^T C$ (sau $X_0 = 0$).

Dacă matricea discretă A este *stabilă*, i.e. $|\lambda(A)| < 1$, atunci se poate arăta uşor că pentru $k \to \infty$ şirurile (5.81) şi (5.83) converg către limitele $Y = Y^T \geq 0$ şi respectiv $X = X^T \geq 0$ care satisfac evident ecuaţiile matriceale algebrice Liapunov discrete (DEMAL)

$$Y = BB^T + AYA^T \tag{5.85}$$

şi respectiv

$$X = C^T C + A^T X A. \tag{5.86}$$

În cazul sistemelor cu timp continuu *stabile*, i.e. $\mathrm{Re}\lambda(A) < 0$, consideraţii similare conduc la ecuaţiile matriceale algebrice Liapunov (EMAL)

$$0 = BB^T + AY + YA^T, \tag{5.87}$$

respectiv

$$0 = C^T C + A^T X + XA. \tag{5.88}$$

Având în vedere că, în condiţiile de *stabilitate* formulate mai sus, ecuaţiile (5.85) – (5.88) au soluţii unice, în continuare vom adopta următoarea definiţie unificatoare.

Definiţia 5.9 *Matricele simetrice $Y, X \in \mathcal{R}^{n\times n}$, unic definite ca soluţii ale EMAL (5.87), (5.88) (respectiv DEMAL (5.85), (5.86) în cazul discret), se numesc* gramian de controlabilitate *al perechii (A, B), respectiv* gramian de observabilitate *al perechii (C, A).*

Rezultatul următor se demonstrează uşor utilizând propoziţia 5.1 şi are o importanţă deosebită pentru tot restul capitolului.

Propoziţia 5.5 *Sistemul liniar stabil $S = (A, B, C)$ este* controlabil*, respectiv* observabil*, dacă şi numai dacă gramianul corespunzător Y, respectiv X, este o matrice* pozitiv definită*.*

Observaţia 5.5 Utilizând rezultatele obţinute în capitolul 1 precum şi în §2 al acestui capitol se poate vedea uşor că ecuaţiile (5.85) – (5.88) au soluţii unice chiar dacă, în locul condiţiei de stabilitate, este îndeplinită numai condiţia (mai slabă) de *dihotomie*, i.e $Re\lambda(A) \neq 0$ (respectiv $|\lambda(A)| \neq 1$ în cazul discret). Mai mult, în acest caz $S = (A, B, C)$ este controlabil, respectiv observabil, dacă şi numai dacă gramianul corespunzător este o matrice simetrică inversabilă a cărei inerţie este dictată de numerele valorilor proprii stabile şi nestabile ale lui A. $\diamond$

În cele ce urmează nu vom exploata implicaţiile importante ale observaţiei 5.5, ci vom presupune mereu că sistemul considerat $S = (A, B, C)$ este stabil.

Adoptând această ipoteză, profităm de ocazie pentru a efectua o scurtă trecere în revistă a unor proceduri iterative de rezolvare a DEMAL. (Procedurile "de transformare", bazate pe utilizarea formei Schur, au fost discutate în capitolul 1).

Pentru precizare, în continuare ne referim la ecuaţia (5.86).

Schema de iterare "directă", care utilizează ecuaţia recurentă corespunzătoare (5.84), este neeficientă deoarece, de regulă, convergenţa $X_k \to X$ este lentă (cu atât mai lentă cu cât raza spectrală $\rho(A) < 1$ a matricei discrete A este mai apropiată de 1, vezi observaţia 5.1). Pe de altă parte, matricele X_k sunt pozitiv semidefinite, ceea ce permite utilizarea avantajoasă pe parcursul calculelor a unor factorizări de tip Cholesky. Având în vedere aceste observaţii de principiu, în practică se recomandă două tipuri de scheme iterative.

Algoritmii "de dublare" urmăresc accelerarea convergenţei. Ideea constă în a efectua iterarea cu pas mereu dublat, i.e. conform schemei

$$X_1 \to X_2 \to X_4 \to X_8 \to \ldots,$$

fără a mai calcula iteraţiile intermediare. Notăm

$$X^{(s)} = X_k, \quad k = 2^s, \;\; s \geq 0. \tag{5.89}$$

Prin calcul direct, din (5.83) sau (5.84) rezultă

$$\begin{aligned} X^{(0)} &= X_1 = C^T C, \\ X^{(1)} &= X_2 = A^T X^{(0)} A + X^{(0)}, \\ X^{(2)} &= X_4 = (A^2)^T X^{(1)} A^2 + X^{(1)} \end{aligned}$$

şi în general

$$X^{(s+1)} = (A^k)^T X^{(s)} A^k + X^{(s)}. \tag{5.90}$$

Prin urmare, schema de iterare accelerată, bazată pe aplicarea formulei de dublare (5.90), poate fi rezumată astfel.

Algoritmul 5.9 (Rezolvă DEMAL (5.86) în cazul A stabilă utilizând un algoritm de dublare).

1. CONTINUĂ = 'da'
2. $X = C^T C$
3. `Cât timp` CONTINUĂ = 'da'
   1. $X(+1) = AXA^T + X$
   2. `Dacă` $\|X(+1) - X\| < tol\|X\|$
      `atunci`
      1. CONTINUĂ = 'nu'

      `altfel`

1. $X = X(+1)$
2. $A \leftarrow A^2$.

La fiecare iteraţie sunt necesare $2n^3$ înmulţiri (se exploatează simetria lui X), deci schema se justifică dacă $\rho(A)$ nu este prea aproape de 1 i.e. convergenţa este suficient de rapidă.

Algoritmii "de rădăcină pătrată" urmăresc exploatarea caracterului pozitiv (semi)definit al matricelor X_k din (5.83) prin determinarea factorilor Cholesky

$$X_k = L_k^T L_k \geq 0, \tag{5.91}$$

unde, evident, se poate lua $L_1 = C$. Dacă pentru $k \geq 1$ matricea L_k este cunoscută atunci L_{k+1} se determină din (5.84), observând că trebuie să avem

$$X_{k+1} = L_{k+1}^T L_{k+1} = C^T C + A^T L_k^T L_k A = \begin{bmatrix} C^T & (L_k A)^T \end{bmatrix} \begin{bmatrix} C \\ L_k A \end{bmatrix}.$$

Este clar că putem lua

$$\begin{bmatrix} L_{k+1} \\ 0 \end{bmatrix} = U_k \begin{bmatrix} C \\ L_k A \end{bmatrix}, \tag{5.92}$$

unde U_k este o matrice *ortogonală* (e.g. o secvenţă de reflectori) care aduce matricea bloc din dreapta la forma *superior triunghiulară*. (În consecinţă, toate matricele şirului $L_k,\ k \geq 2$ rezultă superior triunghiulare).

Prin urmare, schema de iterare "compactă", bazată pe aplicarea relaţiei (5.92) poate fi rezumată astfel.

Algoritmul 5.10 (Rezolvă DEMAL (5.86) în cazul A stabilă utilizând un algoritm de rădăcină pătrată, bazat pe triangularizarea ortogonală).

1. CONTINUĂ = 'da'
2. $L = C$
3. `Cât timp` CONTINUĂ = 'da'
   1. $\begin{bmatrix} L(+1) \\ 0 \end{bmatrix} = U \begin{bmatrix} C \\ AL \end{bmatrix}$,
   2. `Dacă` $\|L(+1) - L\| < tol\|L\|$
      `atunci`
      1. CONTINUĂ = 'nu'
      `altfel`

1. $L = L(+1)$.

Datorită convergenţei în general lente, schema de rădăcină pătrată, a cărei idee a fost expusă mai sus relativ la DEMAL (5.86), se utilizează avantajos mai ales pentru rezolvarea DEML (5.84) şi în special a unor ecuaţii matriceale recurente de tip Riccati, la care păstrarea caracterului pozitiv (semi-)definit al soluţiei prin iterare directă este problematică. În legătură cu această menţiune, observăm că schemele de rădăcină pătrată rămân valabile ca atare chiar dacă sistemul discret $S = (A, B, C)$ este variabil în timp, i.e. matricele $A = A_k$, $B = B_k$, $C = C_k$ variază de la pas la pas.

5.10 Echilibrarea sistemelor liniare

Considerăm sistemul liniar stabil, controlabil şi observabil $S = (A, B, C)$ şi fie $Y = Y^T > 0$ şi $X = X^T > 0$ gramienii de controlabilitate şi observabilitate, definiţi ca soluţii unice ale EMAL (5.87) şi respectiv (5.88).

Utilizând transformarea de asemănare

$$\tilde{A} = TAT^{-1}, \;\; \tilde{B} = TB, \;\; \tilde{C} = CT^{-1}, \tag{5.93}$$

obţinem uşor

$$0 = \tilde{B}\tilde{B}^T + \tilde{A}\tilde{Y} + \tilde{Y}\tilde{A}^T,$$
$$0 = \tilde{C}^T\tilde{C} + \tilde{A}^T\tilde{X} + \tilde{X}\tilde{A},$$

unde gramienii transformaţi au expresiile

$$\tilde{Y} = TYT^T, \quad \tilde{X} = T^{-T}XT^{-1}. \tag{5.94}$$

În particular avem

$$\tilde{Y}\tilde{X} = T(YX)T^{-1}, \tag{5.95}$$

deci valorile proprii $\lambda(YX)$ ale produsului YX sunt *invarianţi* ai sistemului $S = (A, B, C)$ în raport cu transformarea (5.93). (Se poate arăta că aceste valori proprii sunt pozitive şi coincid cu pătratul valorilor singulare Hankel ale lui S, [X]). Mai mult, aceste valori proprii pot fi interpretate – într-un sens ce va fi precizat mai departe – ca expresii cantitative ale "gradelor" de transfer intrare-ieşire ataşate (celor n componente ale stării $x \in \mathcal{R}^n$ a) sistemului $S = (A, B, C)$.

Pe de altă parte, utilizând transformarea de asemănare (5.93), în care matricea $T = U$ este ortogonală, relaţiile (5.94) devin

$$\tilde{Y} = UYU^T, \quad \tilde{X} = UXU^T. \tag{5.96}$$

Prin urmare, în acest caz se conservă *separat* valorile proprii (evident pozitive) ale matricelor simetrice pozitiv definite Y şi X. Interpretând aceste valori proprii ca expresii cantitative ale "gradelor" de controlabilitate şi respectiv observabilitate ataşate lui S, putem spune sugestiv că aceste grade sunt *invarianţi ortogonali* ai sistemului S. De aceea, în problemele de analiză *separată* a controlabilitaţii şi observabilităţii, abordate în §§3–8, am utilizat cu succes transformări ortogonale.

Din contră, în acest paragraf, dedicat analizei proprietăţilor de transfer ale sistemului S, rezultate (într-un sens precizat) prin "compunerea" proprietăţilor de controlabilitate şi observabilitate, vom utiliza transformări (5.93) de formă generală, în particular, cu scopul "echilibrării" (egalării) gradelor corespunzătoare de controlabilitate şi observabilitate.

Enunţul precis al problemei echilibrării este următorul.

Se dă sistemul $S = (A, B, C)$, presupus stabil, controlabil şi observabil. Să se determine o matrice de transformare $T \in \mathcal{R}^{n\times n}$ inversabilă astfel încât gramienii transformaţi (5.94) să fie diagonali şi egali, i.e.

$$\tilde{X} = \tilde{Y} = \Sigma, \tag{5.97}$$

unde Σ este o matrice diagonală cu elemente pozitive. (Conform relaţiei (5.95) aceste elemente reprezintă valorile singulare Hankel ale lui S).

Definiţia 5.10 *Dacă matricea inversabilă $T \in \mathcal{R}^{n\times n}$ este o soluţie a problemei echilibrării formulate mai sus relativ la sistemul S, atunci sistemul transformat $\tilde{S} = (\tilde{A}, \tilde{B}, \tilde{C})$, definit prin relaţiile (5.93), se numeşte* reprezentare echilibrată *a lui S.*

Mai general, orice sistem $\tilde{S} = (\tilde{A}, \tilde{B}, \tilde{C})$ cu proprietatea (5.97) se numeşte echilibrat.

Soluţia problemei echilibrării în formularea de mai sus este simplă din punct de vedere principial şi, în esenţă, se reduce la diagonalizarea simultană a două matrice simetrice, ambele pozitiv definite, prin transformări de congruenţă de forma (5.94).

Vom determina matricea căutată T sub forma

$$T = T_3 T_2 T_1 \,, \tag{5.98}$$

unde transformările $T_k,\ k = 1 : 3$ se determină succesiv.

În prima etapă considerăm factorizarea Cholesky a matricei $Y > 0$, i.e.

$$Y = MM^T, \tag{5.99}$$

unde M este *inferior triunghiulară inversabilă* şi luăm $T_1 = M^{-1}$. Aplicând (5.94) obţinem

$$\begin{aligned} Y_1 &\stackrel{\text{def}}{=} T_1 Y T_1^T = M^{-1}(MM^T)M^{-T} = I, \\ X_1 &\stackrel{\text{def}}{=} M^T X M. \end{aligned} \tag{5.100}$$

În a doua etapă considerăm descompunerea spectrală a matricei $X_1 > 0$, obţinută aplicând algoritmul QR simetric cu acumularea transformărilor, i.e.

$$X_1 = U^T \Sigma^2 U, \tag{5.101}$$

unde coloanele matricei ortogonale U^T sunt vectorii proprii ai matricei (simetrice) X_1 iar notaţia $\Sigma^2 \stackrel{\text{def}}{=} \text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ se justifică ţinând seama că valorile proprii $\lambda i \stackrel{\text{def}}{=} {\sigma_i}^2$ ale matricei X_1 sunt strict pozitive. Convenim să alegem $\sigma_i > 0,\ i = 1 : n$ şi luăm $T_2 = U$. Aplicând încă o dată (5.94) obţinem

$$\begin{aligned} Y_2 &\stackrel{\text{def}}{=} T_2 Y_1 T_2^T = UU^T = I, \\ X_2 &\stackrel{\text{def}}{=} U(U^T \Sigma^2 U)U^T = \Sigma^2. \end{aligned} \tag{5.102}$$

Acum ambele matrice X_2 şi Y_2 sunt diagonale. Pentru a le face egale, luăm $T_3 = \Sigma^{1/2}$ şi aplicând ultima dată (5.94) obţinem într-adevăr

$$\begin{aligned} \tilde{Y} &= T_3 Y_3 T_3^T = \Sigma^{1/2}\Sigma^{1/2} = \Sigma, \\ \tilde{X} &= \Sigma^{-1/2}(\Sigma^2)\Sigma^{-1/2} = \Sigma. \end{aligned} \tag{5.103}$$

În definitiv transformarea căutată (5.98) este

$$T = \Sigma^{1/2} U M^{-1}, \tag{5.104}$$

iar elementele pozitive ale matricei diagonale Σ coincid cu invarianţii $\lambda^{1/2}(YX)$.

Este clar că procedura de mai sus admite o versiune duală, care începe cu factorizarea Cholesky $X = L^T L$ a matricei $X > 0$, unde L este *superior triunghiulară inversabilă*, etc.

Pentru a evidenţia "simetria" de principiu a procedurii de echilibrare, considerăm ambele factorizări Cholesky

$$Y = MM^T, \quad X = L^T L \tag{5.105}$$

şi formăm produsul LM [2]. Aplicând algoritmul de descompunere a valorilor singulare (**DVS**) obţinem

$$LM = V^T \Sigma U, \tag{5.106}$$

unde U, V sunt matrice ortogonale, $\Sigma \stackrel{\text{def}}{=} (\sigma_1, \ldots, \sigma_n)$ este diagonală, iar

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0 \tag{5.107}$$

sunt valorile singulare corespunzătoare. Transformarea căutată T poate fi exprimată prin formulele echivalente

$$T = \Sigma^{1/2} U M^{-1}, \quad \text{respectiv} \quad T = \Sigma^{-1/2} V L \tag{5.108}$$

şi conduce la realizarea egalităţii dorite (5.97), unde în plus Σ satisface condiţia de ordonare (5.107).

Prin urmare, procedura de echilibrare "simetrică", bazată pe relaţiile (5.105) şi (5.106), poate fi rezumată astfel.

Algoritmul 5.11 (Se echilibrează sistemul $S = (A, B, C)$, presupus stabil, controlabil şi observabil).

1. Se aduce matricea A la forma Schur, i.e. $A \leftarrow U_0 A U_0^T$, utilizând algoritmul **QR** cu acumularea transformărilor.
2. Se face $B \leftarrow U_0 B$ şi $C \leftarrow C U_0^T$.
3. Se rezolvă EMAL (5.87) şi (5.88), ţinând seama că matricele A şi A^T sunt triunghiulare.
4. Se calculează factorii Cholesky L şi M conform relaţiei (5.105).
5. Se efectuează produsul LM şi se calculează DVS (5.106).
6. Se calculează matricea de transformare T conform (5.108).
7. Se face $A \leftarrow TAT^{-1}$, $B \leftarrow TB$ şi $C \leftarrow CT^{-1}$.

Comentarii. Paşii 1 şi 2 au drept scop reducerea efortului de calcul implicat de rezolvarea ambelor EMAL (5.87) şi (5.88) prin metodele expuse în capitolul 1, întrucât matricea A este adusă la forma Schur o singură dată. Utilizând o procedură specială de tip rădăcină pătrată, propusă de Hammarling [17], este posibilă contragerea paşilor 3 şi 4, i.e. calculul direct al factorilor Cholesky L şi M fără a determina explicit soluţiile X şi Y

[2] Având în vedere definiţiile (5.81) şi (5.83), matricele M şi L din (5.105) constituie versiuni "prescurtate" ale matricelor de controlabilitate şi observabilitate, deci matricea $H = LM$ are o semnificaţie similară în raport cu matricea Hankel $H_{\alpha\beta} = Q_\alpha R_\beta$.

ale EMAL. În general, procedura de echilibrare rezumată prin algoritmul 5.11, care în esenţă reprezintă analogul sistemic al procedurii **DVS**, este susceptibilă de ameliorări vizând eliminarea a cât mai multe cantităţi intermediare în favoarea operării directe asupra datelor primare $S = (A, B, C)$ ale sistemului considerat. Căile prin care acest obiectiv de principiu poate fi cel mai bine realizat urmează încă a fi descoperite. $\diamond$

Dintre aplicaţiile procedurii de echilibrare, expunem mai jos numai tehnicile de reducţie dimensională a unui sistem $S = (A, B, C, D)$ în scopul eliminării stărilor neesenţiale din punctul de vedere al transferului intrare-ieşire.

Aprecierea caracterului esenţial, respectiv neesenţial, al variabilelor de stare $x_i \in \mathcal{R},\ i = 1 : n$ se face pe forma echilibrată $\tilde{S} = (\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})$ satisfăcând condiţia (5.97),în care elementele matricei diagonale Σ sunt ordonate în acord cu (5.107). Subliniem că matricele $\tilde{A}, \tilde{B}, \tilde{C}$ ale lui $\tilde{S}$ rezultă aplicând algoritmul 5.11 tripletului (A, B, C) iar $\tilde{D} = D$.

Fie $\varepsilon > 0$ un număr dat suficient de mic, înţeles ca prag de semnificaţie, şi fie t cel mai mare număr întreg astfel încât

$$0 \le t \le n \quad \text{cu} \quad \sigma_i \ge \varepsilon,\ \ i = 1 : t. \tag{5.109}$$

Sistemul de ordin redus S_t va fi de ordin t şi, în esenţă, se obţine eliminând stările $x_j,\ j = t+1 : n$ corespunzătoare valorilor singulare nesemnificative $\sigma_j < \varepsilon,\ j = t+1 : n$.

Pentru a descrie succint cele două modalităţi principale de calcul ale matricelor sistemului redus S_t, considerăm partiţia matricelor sistemului echilibrat $\tilde{S}$ definită prin

$$\begin{aligned} \tilde{A} &= \left[\begin{array}{cc} A_1 & A_{12} \\ A_{21} & A_2 \end{array}\right], \quad \tilde{B} = \left[\begin{array}{c} B_1 \\ B_2 \end{array}\right], \\ \tilde{C} &= \left[\begin{array}{cc} C_1 & C_2 \end{array}\right], \quad \tilde{D} = D\,, \end{aligned} \tag{5.110}$$

unde A_1 şi B_1 au t linii iar A_1 şi C_1 au t coloane.

I. Prima modalitate de reducţie corespunde *trunchierii* directe a lui $\tilde{S}$, conform relaţiei

$$S_t = (A_1, B_1, C_1, D_1), \tag{5.111}$$

prin urmare, matricele sistemului S_t, dispuse sugestiv sub forma

$$S_t = \left[\begin{array}{cc} A_1 & B_1 \\ C_1 & D_1 \end{array}\right], \tag{5.112}$$

rezultă prin inspecţie din (5.110).

II. A doua modalitate de reducţie, preferenţial utilizată în practică, corespunde unei *trunchieri compensate* în scopul conservării transferului în regim staţionar constant al sistemelor S şi $\tilde{S}$, i.e. al matricei de amplificare

$$K \stackrel{\text{def}}{=} T(0) = D + (sI - A)^{-1}B\Big|_{s=0}. \tag{5.113}$$

Utilizând idei specifice metodei perturbaţiilor singulare, scriem ecuaţiile lui $\tilde{S}$ sub forma

$$\begin{array}{rcl} \dot{x}_1 &=& A_1 x_1 + A_{12} x_2 + B_1 u \\ \dot{x}_2 &=& A_{21} x_2 + A_2 x_2 + B_2 u \\ y &=& C_1\, x_1 + C_2\, x_2 + D\, u \end{array} \tag{5.114}$$

şi presupunând $x_2 = \text{const}$, i.e. $\dot{x}_2 = 0$, din a doua ecuaţie găsim

$$x_2 = -A_2^{-1}(A_{21}x_1 + B_2 u)\,.$$

Introducând această expresie în celelalte două ecuaţii, obţinem

$$\dot{x}_1 = (A_1 - A_{12}{A_2}^{-1}A_{21})x_1 + (B_1 - A_{12}{A_2}^{-1}B_2)u$$

$$y = (C_1 - C_2{A_2}^{-1}A_{21}x_1 + (\,D - C_2{A_2}^{-1}B_2)u$$

deci trunchierea compensată S_t a lui S este

$$S_t = (A_1 - A_{12}A_2^{-1}A_{21}, B_1 - A_{12}A_2^{-1}B_2, C_1 - C_2A_2^{-1}A_{21}, D - C_2A_2^{-1}B_2). \tag{5.115}$$

Prin urmare, matricele lui S_t se calculează acum utilizând (5.112) şi efectuând actualizarea

$$S_t \leftarrow S_t - \left[\begin{array}{c} A_{12} \\ C_2 \end{array}\right] A_2^{-1} \left[\begin{array}{cc} A_{21} & B_2 \end{array}\right]. \tag{5.116}$$

În cazul sistemelor discrete (presupuse de asemenea stabile, controlabile şi observabile) raţionamentele şi rezultatele de mai sus rămân valabile în cea mai mare parte. În particular, algoritmul 5.11 se păstrează ca atare, înlocuind peste tot EMAL cu DEMAL corespunzătoare. De asemenea, pentru a construi trunchierea compensată discretă, care conservă matricea de amplificare

$$K \stackrel{\text{def}}{=} T(1) = D + (zI - A)^{-1}B\Big|_{z=1}, \tag{5.117}$$

scriem ecuaţiile lui $\tilde{S}$ sub forma

$$\begin{array}{rcl} x_1(+1) &=& A_1 x_1 + A_{12} x_2 + B_1 u \\ x_2(+1) &=& A_{21} x_1 + A_2 x_2 + B_2 u \\ y &=& C_1\, x_1 + C_2\, x_2 + D\, u \end{array} \tag{5.118}$$

şi presupunând x_2 =const, i.e. $x_2(+1) = x_2$, din a doua ecuaţie găsim

$$x_2 = (I - A_2)^{-1}(A_{21}x_1 + B_1u)$$

Prin urmare, trunchierea compensată a unui sistem discret este definită tot de (5.115) sau (5.116), în care matricea $(sI - A_2)|_{s=0} = -A_2$ se înlocuieşte cu $(zI - A_2)|_{z=1} = I - A_2$. Redactarea algoritmilor este propusă cititorului sub formă de exerciţii.

Programe MATLAB disponibile

Testarea stabilităţii se face conform algoritmului 5.1, i.e. calculând valorile proprii ale matricei A a sistemului cu funcţia **eig**. Pentru obţinerea formei Schur ordonate, necesară în algoritmul 5.2, se utilizează funcţia **schord**. Construcţia matricei de controlabilitate R a perechii (A, B) se face cu funcţia **ctrb**, care implementează algoritmul 5.4, mai precis versiunea acestuia bazată pe formula (5.38). Pentru construcţia formei bloc-superior Hessenberg a unei perechi (A, B) este disponibilă funcţia **ctrbf**, care implementează algoritmul 5.7 cu blocurile permutate în ordine inversă. Construcţia unei realizări minimale (vezi algoritmul 5.8) se poate face utilizând funcţia **minreal**. Pentru calculul gramienilor de controlabilitate şi observabilitate sunt disponibile funcţiile **gram** şi **dgram**, care apelează, evident, **lyap** şi **dlyap**. Echilibrarea unei realizări stabile şi minimale se efectuează cu **balreal**, care implementează o versiune neoptimizată a algoritmului 5.11. (În particular, valorile singulare nu rezultă ordonate ca în (5.107)). Trunchierea compensată a realizării echilibrate furnizate de **balreal** se face interactiv, pe bază opţiunii utilizatorului ce apelează funcţia corespunzătoare **modred**.

Exerciţii

E 5.1 Conform teoremei lui Liapunov, matricea A a unui sistem continuu (discret) este stabilă dacă şi numai dacă oricare ar fi matricea $Q = Q^T > 0$ ecuaţia matriceală algebrică Liapunov (discretă)

$$A^TX + XA = -Q \quad (\ A^TXA - X = -Q\ )$$

are o soluţie $X = X^T > 0$. Explicaţi de ce enunţul de mai sus (de mare valoare teoretică) nu conduce la un test de stabilitate numeric eficient în raport cu algoritmul 5.1.

E 5.2 Un polinom $p(s) = a_0 s^n + a_1 s^{n-1} + \ldots + a_n$ este stabil (sau "hurwitzian") dacă toate rădăcinile sale au partea reală strict negativă. Scrieţi algoritmul de testare a stabilităţii polinomului $p(s)$ utilizând criteriul Routh-Hurwitz.

Indicaţie. Se utilizează algoritmul de eliminare gaussiană *fără pivotare* relativ la matricea Hurwitz asociată lui $p(s)$, ceea ce corespunde exact formulării propuse de Routh pentru criteriul în discuţie.

Observaţie. Întrucât în absenţa unei strategii de pivotare adecvate, algoritmul de eliminare gaussiană este *numeric nestabil* (apariţia pivoţilor nuli corespunde aşa-numitelor cazuri "critice" ale algoritmului Routh), în practică testarea stabilităţii polinomului $p(s)$ se face construind matricea companion şi aplicând algoritmul 5.1.

E 5.3 Formulaţi şi rezolvaţi versiunea discretă a exerciţiului 5.2.

E 5.4 Matricea A a unui sistem continuu se numeşte *admisibilă* dacă $\mathrm{Re}\lambda(A) < -\eta$, unde $\eta > 0$ este un număr real dat. Formulaţi testul corespunzător de admisibilitate. (Algoritmul 5.1 corespunde evident cazului $\eta = 0$).

E 5.5 Formulaţi şi rezolvaţi versiunea discretă a exerciţiului 5.4.

E 5.6 Utilizând algoritmul **QZ** în locul algoritmului **QR**, scrieţi algoritmul de testare a stabilităţii sistemului generalizat (de tip descriptor), descris prin

$$\begin{aligned} E\dot{x} &= Ax + Bu, \\ y &= Cx, \end{aligned}$$

unde matricea $E \in \mathcal{R}^{n\times n}$ este nesingulară dar poate fi rău condiţionată la inversare. (Prin urmare, trebuie evitată formarea matricelor $A_0 = E^{-1}A$ etc).

Indicaţie. Valorile proprii ale matricei A_0 coincid cu valorile proprii generalizate ale fascicolului $\lambda E - A$.

E 5.7 Scrieţi algoritmul de calcul a unei baze ortogonale a subspaţiului A-invariant total nestabil $\bar{\mathcal{X}}^+(A)$.

E 5.8 Utilizând rezultatele furnizate de algoritmul 3 (vezi observaţia 5.4), indicaţi o procedură de calcul a unei baze ortogonale a lui $\bar{\mathcal{X}}^+(A)$.

E 5.9 Scrieţi versiunile discrete ale algoritmilor 5.2 şi 5.3 (vezi observaţia 5.2).

E 5.10 Presupunând dată o funcţie de transfer $T(s)$ într-una din formele

$$T(s) = \delta \frac{\prod(s-\mu_j)}{\prod(s-\lambda_i)}, \quad \text{respectiv} \quad T(s) = \sum \frac{r_i}{s-\lambda_i},$$

scrieţi o realizare de stare $S = (A, b, c^T)$ în forma (5.11) şi indicaţi baze (ortogonale) pentru subspaţiile $\mathcal{X}^-(A)$ şi $\bar{\mathcal{X}}^+(A)$. Cum trebuie procedat dacă $T(s) = N(s)/p(s)$ este dată ca raport a două polinoame ?

E 5.11 Formulaţi şi rezolvaţi versiunea discretă a exerciţiului 5.10.

E 5.12 Presupunând că matricea A este simplă, scrieţi algoritmul de descompunere spectrală bazat pe calculul valorilor şi vectorilor proprii ai matricei A şi comparaţi-l cu algoritmul 5.3. Care este preferabil şi de ce ?

E 5.13 Utilizând algoritmul **QZ** în locul algoritmului **QR**, analizaţi posibilităţile de extindere a algoritmilor 5.2 şi 5.3 la cazul sistemelor generalizate (de tip descriptor, vezi exerciţiul 5.6).

E 5.14 Formulaţi şi rezolvaţi exerciţiul 5.12 în cazul sistemelor generalizate.

E 5.15 Discutaţi controlabilitatea şi observabilitatea unui sistem $S = (A, b, c^T)$ scris în forma standard controlabilă. Idem, pentru forma standard observabilă.

E 5.16 Se dă o funcţie de transfer discretă $T(z) = N(z)/p(z)$. Să se scrie o realizare de stare $S = (A, b, c^T)$ astfel încât problema (5.26) a determinării şirului de comenzi de efort minim să fie cât mai simplă. Scrieţi algoritmul de calcul corespunzător.

E 5.17 Se dă o funcţie de tranfer discretă $T(z) = N(z)/p(z)$. Să se scrie o realizare de stare $S = (A, b, c^T)$ astfel încât problema (5.36) a determinării unei estimări optimale a stării iniţiale să fie cât mai simplă. Scrieţi algoritmul de calcul corespunzător.

E 5.18 Analizaţi posibilităţile de extindere a soluţiilor exerciţiilor 5.15 şi 5.16 în cazul unei matrice de transfer $T(z) \in \mathcal{R}^{l\times m}(z)$. Consideraţi separat cazurile $m = 1,\ l = 1$ şi $m, l > 1$.

E 5.19 Scrieţi o realizare de stare $S = (A, b, c^T)$ a funcţiei de transfer $T(z) = N(z)/p(z)$ astfel încât $R(A, b) = I$. Idem, astfel încât $Q(c^T, A) = I$. În ce condiţii putem avea simultan $R(A, b) = Q(c^T, A) = I$?

E 5.20 Formulaţi analogul algoritmilor 5.4 şi 5.5 în cazul observabilităţii (fără a recurge la schema de dualizare din § 4).

E 5.21 Formulaţi şi discutaţi realizarea numerică a testelor elementare de observabilitate, bazate pe duala propoziţiei 5.1.

E 5.22 Se dau perechea observabilă (C, A) şi un vector $\mathbf{y} \in \mathcal{R}^{nl}$. Scrieţi algoritmul de calcul al pseudosoluţiei $x^* = x^*(0)$ a sistemului de ecuaţii (5.36), utilizând transformări ortogonale.

E 5.23 Se dau perechea controlabilă (A, B) şi un vector $x = x(n) \in \mathcal{R}^n$. Scrieţi algoritmul de calcul al soluţiei normale $\mathbf{u}^*$ a sistemului de ecuaţii (5.26), utilizând transformări ortogonale.

E 5.24 Obţineţi condiţii suficiente simple de controlabilitate şi observabilitate a conexiunilor serie, paralel şi în circuit închis. Ce puteţi afirma despre stabilitatea acestor conexiuni ?

E 5.25 Extindeţi propoziţia 5.1 precum şi duala sa în cazul sistemelor generalizate (vezi exerciţiul 5.6). Discutaţi realizarea numerică a testelor corespunzătoare.

E 5.26 Se dă un triplet $S = (A, B, C)$. Indicaţi (cel puţin două) proceduri de calcul a unei baze ortogonale pentru *subspaţiul controlabil* $\mathcal{R} \stackrel{\text{def}}{=} \text{Im}R$ al perechii (A, B). Idem, pentru *subspaţiul neobservabil* $\mathcal{N} \stackrel{\text{def}}{=} \text{Ker}Q$.

E 5.27 Formulaţi problemele de calcul (5.26) şi (5.36) în cazul unui sistem S nu neapărat controlabil şi observabil.

E 5.28 Scrieţi programul MATLAB care implementează algoritmul 5.6 şi testaţi-l considerând perechi (A, b) adecvat alese.

Indicaţie. Scrieţi o funcţie MATLAB care generează o pereche $(\tilde{A}, \tilde{b})$ de ordin n dat, nu neapărat controlabilă, utilizând forma (5.47), în care perechea controlabilă (A_R, b_R) de ordin $r \leq n$ este în FSC, iar $A_{R\bar{R}}$ şi $A_{\bar{R}}$ sunt generate aleator. Perechea dorită (A, b) se obţine aplicând lui $(\tilde{A}, \tilde{b})$ o transformare de asemănare oarecare, i.e. $(A, b) = (S\tilde{A}S^{-1}, S\tilde{b})$, unde matricea nesingulară S poate fi, de asemenea, aleasă aleator.

E 5.29 Utilizând programul MATLAB ce implementează algoritmul 5.4, bazat pe construcţia matricei de controlabilitate, ilustraţi fenomenul de necontrolabilitate numerică a perechilor (A, b) cu o singură intrare, urmărind variaţia numărului de condiţionare $\text{cond}(R)$ în raport cu creşterea

ordinului perechii. Comentaţi comparativ rezultatele numerice obţinute utilizând procedura din exerciţiul 5.28.

E 5.30 Scrieţi funcţiile MATLAB ce implementează procedura de deflaţie controlabilă descrisă în propoziţia 5.4 utilizând **DVS**, procedura de triangularizare ortogonală cu pivotarea coloanelor şi, respectiv, procedura de eliminare gaussiană cu pivotare completă.

E 5.31 Scrieţi programul MATLAB care implementează algoritmul 5.7 şi testaţi-l considerând perechi (A, B) adecvat alese.

Indicaţie. Generarea perechilor controlabile (A_R, B_R) cu $m \geq 1$ are la bază utilizarea FSC cu intrări decuplate descrise în capitolul 3, § 3. Indicele de controlabilitate ν al perechii astfel obţinute coincide cu cel mai mare indice de controlabilitate al perechilor diagonale din FSC.

E 5.32 Consideraţi un sistem $S = (A, B, C)$, nu neapărat controlabil sau observabil, având perechea (A, B) de forma (5.65), unde (A_R, B_R) este controlabilă şi fie $C_R = C(:, 1:r)$. Arătaţi că S şi $S_R \stackrel{\text{def}}{=} (A_R, B_R, C_R)$ au aceeaşi matrice de transfer, mai precis avem

$$T(s) \stackrel{\text{def}}{=} C(sI - A)^{-1}B = C_R(sI - A_R)^{-1}B_R. \tag{5.119}$$

Egalitatea (5.119) arată că polii ficşi $\lambda(A_{\bar{R}})$ corespunzători părţii necontrolabile a perechii (A, B) sunt *poli simplificabili* ai lui $T(s)$. Prin dualitate, aceeaşi proprietate aparţine polilor ficşi $\lambda(A_{\bar{Q}})$ corespunzători părţii neobservabile a perechii (C, A).

E 5.33 Arătaţi că în cazul SISO, sistemul $S = (A, b, c^T)$ este o realizare minimală a funcţiei de transfer ireductibile $T(s) = N(s)/p(s)$ dacă şi numai dacă ordinul matricei A coincide cu gradul numitorului $p(s)$. Ce se întâmplă în cazul sistemelor cu mai multe intrări sau/şi cu mai multe ieşiri ?

E 5.34 O matrice de transfer $T(s)$ se numeşte stabilă dacă polii tuturor elementelor lui $T(s)$ sunt stabili, i.e. au partea reală strict negativă, respectiv au modulul strict subunitar. Arătaţi că $T(s)$ este stabilă dacă şi numai dacă orice realizare minimală $S_m = (A_m, B_m, C_m)$ a lui $T(s)$ este stabilă. Scrieţi algoritmul de testare a stabilităţii unei matrice de transfer $T(s)$ date.

E 5.35 Enumeraţi câteva proprietăţi semnificative ale realizărilor stabilizabile sau/şi detectabile.

E 5.36 Indicaţi câteva metode de calcul a unei baze a *subspaţiului stabilizabil* $\mathcal{S} = \mathcal{R}+\mathcal{X}^-(A)$. Idem, pentru *subspaţiul nedetectabil* $\mathcal{T} = \mathcal{N}\cap\bar{\mathcal{X}}^+(A)$.

Indicaţie. În cazul în care coloanele matricelor $S_k \in \mathcal{R}^{n\times n_k}$ constituie sisteme de generatori pentru subspaţiile $\mathcal{X}_k \subset \mathcal{R}^n$, adică $\mathcal{X}_k = \mathrm{Im}\, S_k,\ k = 1:2$, atunci coloanele matricei bloc

$$S = [\, S_1\ S_2\,]$$

constituie un sistem de generatori ai subspaţiului $\mathcal{X}_1 + \mathcal{X}_2$. Prin dualitate, dacă liniile matricelor $T_k \in \mathcal{R}^{n_k\times n}$ constituie un sistem de anulatori pentru subspaţiile $\mathcal{X}_k \subset \mathcal{R}^n$, adică $\mathcal{X}_k = \mathrm{Ker} T_k,\ k = 1:2$, atunci liniile matricei

$$T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

constituie un sistem de anulatori ai subspaţiului $\mathcal{X} = \mathcal{X}_1 \cap \mathcal{X}_2$. În ambele cazuri, pentru a obţine o bază a lui $\mathcal{X}$, se utilizează **DVS**.

E 5.37 Analizaţi posibilitatea combinării algoritmilor de dublare şi de rădăcină pătrată din § 9, utilizând schema "candidat"

$$\begin{bmatrix} L \\ 0 \end{bmatrix} \leftarrow U \begin{bmatrix} L \\ LA \end{bmatrix}, \quad A \leftarrow A^2.$$

E 5.38 Formulaţi versiunile algoritmilor 5.9 şi 5.10 corespunzătoare rezolvării DEMAL (5.85) fără a recurge la schema de dualizare expusă în finalul § 4.

E 5.39 Arătaţi că dacă X, Y sunt soluţiile ecuaţiilor (5.87), (5.88) (respectiv (5.85), (5.86) în cazul discret) atunci avem $tr(B^TXB) = tr(C^TYC)$, unde $tr(A)$ este *urma* matricei pătrate A.

Numărul $\delta \geq 0$ introdus prin egalitatea de mai sus coincide cu norma H_2 a funcţiei de transfer $T(s)$, definită prin

$$\|T\|_2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} trT^H(i\omega)T(i\omega)d\omega\,.$$

(În cazul discret integrala se efectuează pe cercul unitate).

Bibliografie

[1] **Ionescu V., Popeea C.** CONDUCEREA STRUCTURALĂ A SISTEMELOR LINIARE, ET Bucureşti, 1986.

[2] **Ionescu V.** INTRODUCERE ÎN TEORIA STRUCTURALĂ A SISTEMELOR LINIARE, EA Bucureşti, 1975.

[3] **Aplevich J.D.** *Direct Computation of Canonical Forms for Linear Systems by Elementary Matrix Operations*, IEEE Trans. AC - 19, No. 2, pp. 124 – 126, 1974.

[4] **Aplevich J.D.** *Tableau Methods for Analysis and Design of Linear Systems*, Automatica, Vol. 15, pp. 419 – 429, 1979.

[5] **Varga A.** *Numerically Reliable Algorithm to Test Controllability*, Electron. Lett. 15, pp. 454 – 455, 1981.

[6] **Varga A.** *Numerically Stable Algorithm for Standard Controllability Form*, Electron. Lett. 17, pp. 74 – 75, 1981.

[7] **Klema V., Laub A.J.** *The Singular Value Decomposition: Its Computation and Some Applications*, IEEE Trans. AC - 25, No. 2, pp. 164 – 176, 1980.

[8] **Moore B.C.** *Principal Component Analysis in Linear Systems Controlability, Observability and Model Reduction*, IEEE Trans. AC - 26, No. 2, pp. 17 – 32, 1981.

[9] **Laub A.J., Heath M.T., Paige C.C., Ward R.C.** *Computation of System Balancing Transformations*, IEEE Trans. AC - 32, No. 2, 1987.

[10] **Uhlig F.** *Simultaneous Block - Diagonalization of Two Real Symmetric Matrices.* Linear Algebra and Its Applications, Vol. 7, pp. 281 – 289, 1973.

[11] **Heath M.T., Laub A.J., Paige C.C., Ward R.C.** *Computing SVD of a Product of Two Matrices*, SIAM J. Sci. Stat. Comp., Vol. 7, pp. 1147 – 1159, 1986.

[12] **Hammarling S.J.** *Numerical Solution of the Stable, Non-negative Definite Lyapunov Equations*, IMA J. numer. Anal., Vol. 2, pp. 303 – 323, 1982.