CURS 6 - Rezumat

METODE NUMERICE PENTRU ECUAŢII DIFERENŢIALE ORDINARE

6-I METODE NUMERICE PENTRU ECUAŢII DIFERENŢIALE DE ORDINUL ÎNTÂI

În această secţiune se vor prezenta metode numerice pentru ecuaţii şi sisteme de ecuaţii diferenţiale de ordinul întâi – problema cu valori iniţiale. O ecuaţie sau sistem de ordin mai mare decât unu se pot reduce la un sistem echivalent de ordinul unu, prin adăugarea de funcţii necunoscute.

Exemplu: Fie sistemul de ordinul doi,

$$x'' = f(t, x, y, x', y')$$
$$y'' = g(t, x, y, x', y')$$

Punând $u = x', v = y'$, sistemul devine

$$x' = u$$
$$y' = v$$
$$u' = f(t, x, y, u, v)$$
$$v' = g(t, x, y, u, v)$$

■

Pentru sistemele de ordinul doi, se vor da metode speciale în Secţiunea 7-II.

1 Problema cu valori iniţiale (consideraţii generale)

Fie ecuaţia

$$\frac{dx}{dt} = f(t, x) \qquad (1)$$

cu condiţia iniţială

$$x(t_0) = x_0 \qquad (1')$$

Ecuaţia (1) cu condiţia iniţială (1') constituie o problemă cu valori iniţiale (sau o problemă Cauchy). Dacă funcţia f îndeplineşte următoarele condiţii pe domeniul $D = I \times \Omega$, unde I este definit de $|t - t_0| \le a$, iar Ω de $|x - x_0| \le b$:

1) f este definită şi continuă pe D;

2) f este lipschitziană în raport cu x, adică: există o constantă pozitivă A, astfel că pentru orice $t \in I$ şi orice $x, x^* \in \Omega$ avem

$$| f(t, x^*) - f(t, x) | \le A | x^* - x |,$$

atunci: notând cu M marginea superioară a funcţiei $|f|$ pe D, problema are o soluţie unică $x(t)$ definită pe intervalul $| t - t_0 | \le \alpha$, unde $\alpha = \min(a, b/M)$.

În particular, condiţia 2 este îndeplinită dacă f are derivată parţială în raport cu x, mărginită în D (sau, mai mult, continuă pe D).

■

Pentru un sistem de m ecuaţii diferenţiale cu m funcţii necunoscute, fie $\mathbf{x} = [x_1 \dots x_m]^T$, $\mathbf{f} = [f_1(t, \mathbf{x}) \dots f_m(t, \mathbf{x})]^T$, $\mathbf{x}^{(0)} = [x_1^{(0)} \dots x_m^{(0)}]^T$, şi sistemul

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(t, x_1, \dots, x_m) \qquad (2)$$

cu condiţia iniţială

$$\mathbf{x}(t_0) = \mathbf{x}^{(0)} \qquad (2')$$

Cu domeniul $D = I \times \Omega$, unde unde I este definit de $| t - t_0 | \le a$, iar Ω de $| x_i - x_i^{(0)} | \le b_i$, $i = \overline{1, m}$, condiţiile 1 şi 2 devin:

1') $\mathbf{f}$ este definită şi continuă pe domeniul D;

2') $\mathbf{f}$ este lipschitziană în raport cu argumentele $x_1, \dots, x_m$, adică: există constantele pozitive A_j, $j = \overline{1, m}$, astfel că pentru orice $t \in I$ şi orice $\mathbf{x}, \mathbf{x}^* \in \Omega$ avem:

$$| f_i(t, \mathbf{x}^*) - f_i(t, \mathbf{x}) | \le \sum_{j=1}^{m} A_j | x_j^* - x_j |, \quad i = \overline{1, m}.$$

Notăm cu M_i marginea superioară a funcţiei $| f_i |$ pe D, şi cu $M = \max_{i=1, m} M_i$. Dacă 1' şi 2' sunt îndeplinite, atunci există o soluţie unică $\mathbf{x}(t)$ definită pe intervalul

$|t-t_0| \le \alpha$, unde $\alpha = \min(a, b_1/M, \ldots, b_m/M)$. În particular, condiţia 2' este îndeplinită dacă **f** are derivate parţiale în raport cu x_j, $j = \overline{1,m}$, continue pe $I \times \Omega$.

Notă: Condiţia Lipschitz pentru funcţia **f** se poate considera şi sub forma:

$$\| \mathbf{f}(t, \mathbf{x}^*) - \mathbf{f}(t, \mathbf{x}) \| \le A \| \mathbf{x}^* - \mathbf{x} \|,$$

iar marginea M este dată de $\| \mathbf{f}(t, \mathbf{x}) \| \le M$, pentru $(t, \mathbf{x}) \in I \times \Omega$. Norma considerată este norma-∞ ■

În ceea ce urmează vom considera probleme cu valori iniţiale (1, 1') şau (2, 2'), pentru care vom presupune îndeplinite condiţiile de existenţă şi unicitate ale soluţiei. Considerăm calculul soluţiei pentru un interval de integrare $[t_0, TT]$, inclus în intervalul de existenţă a soluţiei. Metodele numerice vor fi prezentate pentru o singură ecuaţie diferenţială (1), şi vor fi generalizate la sisteme (2).

2 Operatori de integrare numerică (intr-un singur pas, în mai mulţi paşi, expliciţi, impliciţi)

Găsirea soluţiei ecuaţiei (1) printr-o metodă numerică se va numi integrare numerică sau integrare *pas cu pas*. Metoda constă în următoarele:

a) Intervalul de integrare $[t_0, TT]$ se divizează prin punctele t_i, $i = \overline{0,n}$, unde $t_n = TT$.

b) Ecuaţia (1) se cere să fie satisfăcută în punctele t_i, iar între aceste puncte, variaţia funcţiei $x(t)$ se estimează.

Vom nota în ceea ce urmează:

$x(t_i)$ = soluţia exactă;

x_i = soluţia calculată în t_i;

$x(t_i) = x_i + e_i$,

unde e_i este eroarea de trunchiere globală a metodei, pe pasul i.

Un *operator de integrare numerică* este reprezentat de o formulă care dă soluţia la momentul t_{i+1} în funcţie de soluţia calculată la *k* momente anterioare

$t_i, t_{i-1}, \ldots, t_{i-k+1}$, şi anume:

$$x_{i+1} = g(x_{i+1}, x_i, \ldots, x_{i-k+1}) \qquad (3)$$

- Dacă în membrul doi din (3), *g* este funcţie numai de x_i, şi eventual x_{i+1}, operatorul se zice *într-un pas*, altfel se zice *în mai mulţi paşi* (şi anume, în *k* paşi). Adică: $x_{i+1} = g(x_{i+1}, x_i)$; sau $x_{i+1} = g(x_i)$.
- Dacă în membrul doi din (3) apare şi x_{i+1}, operatorul se zice *implicit*, în caz contrar se zice *explicit*.

Integrarea prin operatori impliciţi conduce la rezolvarea ecuaţiei (3) în necunoscuta x_{i+1}, printr-o metodă pentru ecuaţii neliniare. O comparaţie între operatori într-un singur pas şi în mai mulţi paşi se va face în 4.8.

Distanţa dintre două puncte succesive de diviziune a intervalului de integrare se zice *pas* de integrare:

$$h_{i+1} = t_{i+1} - t_i .$$

Cazul comun este acela în care pasul este constant: $h_{i+1} = h$. Avem

$t_{i+1} = t_i + h$ (h = constant).

Există însă, algoritmi care utilizează paşi variabili.

3 Operatori într-un singur pas (Taylor, Euler, Runge-Kutta)

3.1 Serii Taylor, eroare de trunchiere, ordin al metodei

Se desvoltă *x*(*t*) în serie Taylor în jurul lui *t* până la termenul de ordinul *p*. De exemplu pentru *p* = 3, avem:

$$x(t+h) = x(t) + hx'(t) + \frac{h^2}{2!}x''(t) + \frac{h^3}{3!}x^{(3)}(t) + \ldots \qquad (4)$$

Ecuaţia (1) este

$x' = f(t, x)$

şi prin derivare succesivă obţinem:

$$x'' = f_t' + f_x' x'; \; x' = f$$

$$x^{(3)} = f_{tt}'' + f_{xt}'' x' + f_x' x''; \; x'' = \ldots$$

Eroarea în desvoltarea (4) este dată de restul seriei Taylor

$$T_4 = \frac{1}{4!} h^4 x^{(4)}(\xi); \; \xi \in (t, t+h)$$

Eroarea T_4 se numeşte *eroarea de trunchiere locală*. Derivata $x^{(4)}$ în ξ se poate aproxima prin derivata în *t*, şi aceasta din urmă prin diferenţa divizată, obţinând estimarea $T_4 \approx \frac{1}{4!} h^3 [x^{(3)}(t+h) - x^{(3)}(t)]$.

În general, considerând desvoltarea până la ordinul $p \geq 1$, eroarea de trunchiere locală este

$$T_p = \frac{1}{(p+1)!} h^{p+1} x^{(p+1)}(\xi); \quad \xi \in (t, t+h)$$

sau

$$T_p = O(h^{p+1})$$

Eroarea de trunchiere *globală* e_p este eroarea produsă de eroarea locală în calculul lui $x(t_n)$, adică eroarea după *n* paşi – unde $n = (t_n - t_0)/h$ – şi ea va fi de ordinul nT_p, adică de ordinul h^p. Avem următoarea

Definiţie: Ordin

(1) Dacă eroarea de trunchiere globală este de ordinul h^p, metoda (sau operatorul) se zice de *ordinul p* ■

Definiţii echivalente ale ordinului sunt următoarele:

(2) Metoda este de ordinul *p* dacă formula metodei coincide cu seria Taylor trunchiată până la termenul de ordinul *p* inclusiv ■

(3) Metoda este de ordinul *p* dacă formula metodei este exactă pentru un

polinom de gradul p (şi nu mai este exactă, pentru un polinom de gradul $p+1$).

Formula (3) a metodei se zice "exactă" pentru o funcţie $x(t)$ dacă, din ipoteza că în membrul doi avem $x_j = x(t_j),\ j = \overline{i, i-k+1}$, rezultă ca avem şi $x_{i+1} = x(t_{i+1})$ ■

În cazul de faţă, formula metodei este chiar seria Taylor (4) trunchiată, scrisă pentru $t = t_i,\ t + h = t_{i+1}$, şi anume:

$$x_{i+1} = x_i + hf_i + \frac{h^2}{2!}x_i'' + \frac{h^3}{3!}x_i^{(3)} + \ldots + \frac{h^p}{p!}x_i^{(p)}, \quad i \geq 0 \tag{5}$$

în care $f_i = f(t_i, x_i)$, iar $x_i'', x_i^{(3)}, \ldots$ reprezintă derivatele calculate în t_i.

Avantaje şi dezavantaje ale metodei seriei Taylor

c) Avantajele sunt simplitatea metodei şi precizia mare care poate fi atinsă. Precizia creşte cu ordinul *p*, dar calculul cere evaluarea a mai multor derivate.

d) Dezavantajul principal constă în calculul derivatelor de ordin superior. Mai mult, trebuie ca funcţia *f* să aibă derivate până la ordinul *p*, ceea ce, în general, nu este cerut pentru existenţa soluţiei. Totuşi, pentru multe din problemele practice, această condiţie este realizată.

3.2 Metoda Euler

Metoda Euler corespunde cazului în care *p* = 1. Formula metodei este, cf. (4),

$$x_{i+1} = x_i + hf(t_i, x_i) \tag{6}$$

Metoda are avantajul că nu cere decât calculul lui *f*. Ordinul ei este *p* = 1, şi pentru a atinge o precizie convenabilă, pasul *h* trebuie luat foarte mic. Metoda are mai degrabă o importanţă teoretică. Ea serveşte la demonstrarea teoremelor de existenţă, şi la exemplificarea noţiunilor de convergenţă şi stabilitate pe exemplul unei metode simple.

3.3 Metode Runge-Kutta

3.3.1 Construcţia metodelor Runge-Kutta

Metodele Runge-Kutta (abreviat RK) utilizează desvoltarea în serie Taylor, dar înlocuiesc calculul derivatelor de ordin superior, cu calculul funcţiei f în puncte de forma $(t+h\alpha, x+h\phi)$, unde α şi ϕ sunt definiţi de coeficienţii metodei.

Reluând desvoltarea Taylor cu rest, avem:

$$x(t_{i+1}) = x(t_i) + hf_i + \frac{h^2}{2!} f_i' + \ldots + \frac{h^p}{p!} f_i^{(p-1)} + O(h^{p+1}) \tag{7}$$

în care s-a ţinut cont de $x'(t) = f(t, x(t))$, iar $f_i = f\,|_{t=t_i}$ şi $f_i^{(n)} = (df^{(n)}/dt^n)\Big|_{t=t_i}$.

Reamintim că notăm prin x_i soluţia calculată în t_i, prin $x(t_i)$ soluţia exactă, şi că punem condiţia $x_i = x(t_i)$ (până la termenul de ordinul p în h).

O caracteristică a metodei este numărul de evaluări al membrului doi al ecuaţiei (1) sau sistemului (2), pe un pas. Acest număr este numit "numărul de evaluări de funcţii". O metodă RK care face q evaluări de funcţii va fi numită "cu q-trepte" (q-stage). Pentru a obţine o metodă cu q-trepte, punem:

$$x_{i+1} = x_i + h\phi(t_i, x_i, h) \tag{8}$$

în care

$$\phi(t_i, x_i, h) = \sum_{m=1}^{q} \omega_m k_m \tag{9}$$

unde ω_m sunt coeficienţi ai metodei, iar $k_m = k_m(t_i, x_i, h)$. Se obţine

$$x_{i+1} = x_i + h\sum_{m=1}^{q} \omega_m k_m \tag{10}$$

În (9, 10) funcţiile k_m se definesc astfel:

a) Pentru o metodă explicită:

$$k_m = f(t_i + h\alpha_m,\ x_i + h\sum_{j=1}^{m-1} \beta_{mj} k_j) \tag{11}$$

şi $\alpha_1 = 0$, astfel că avem:

$k_1 = f(t_i, x_i)$,

$k_2 = f(t_i + h\alpha_2, x_i + h\beta_{21}k_1)$,

etc.

b) Pentru o metodă implicită:

$$k_m = f(t_i + h\alpha_m, x_i + h\sum_{j=1}^{q}\beta_{mj}k_j) \tag{12}$$

Coeficienţii α_m se mai zic *noduri,* iar ω_m se mai zic *ponderi*.

Se obişnuieşte ca coeficienţii α_m, β_{mj} şi ω_m, să se dea în *tabloul Butcher*:

$\boldsymbol{\alpha}$	$\mathbf{B}$
	$\boldsymbol{\omega}$

în care: $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \ldots\ \alpha_q]^T$, $\mathbf{B} = [\beta_{mj}]$, şi $\boldsymbol{\omega} = [\omega_1\ \omega_2\ \ldots\ \omega_q]$.

Pentru o metodă explicită ($\alpha_1 = 0$, şi $\beta_{mj} = 0$ pentru $j > m-1$) tabloul Butcher este:

0					
α_2	β_{21}				
α_3	β_{31}	β_{32}			
$\vdots$	$\vdots$	$\vdots$	$\vdots$		
α_q	β_{q1}	β_{q2}	$\ldots$	$\beta_{q,q-1}$	
	ω_1	ω_2	$\ldots$	ω_{q-1}	ω_q

(13)

Condiţii pentru coeficienţii metodei:

- Coeficienţii ω_m îndeplinesc condiţia de *consistenţă*:

$$\sum_{m=1}^{q}\omega_m = 1 \tag{14}$$

Aceasta asigură convergenţa metodei – v. 3.3.3.

- Coeficienţii α_m, β_{mj} sunt supuşi la condiţiile:

$$\sum_{j=1}^{m-1}\beta_{mj} = \alpha_m, \qquad m = \overline{2,q} \tag{14'}$$

adică: $\beta_{21} = \alpha_2$, $\beta_{31} + \beta_{32} = \alpha_3$, …, $\beta_{q1} + \beta_{q2} + \cdots + \beta_{q,q-1} = \alpha_q$.

Aceste condiţii simplifică deducerea coeficienţilor pentru metodele de ordin mai mare ca 2. Pentru justificări ale condiţiilor (14'), v. Ralston & Rabinowitz (1978), şi Isaacson & Keller (1966).

Ordin:

Eroarea de trunchiere *locală* T_{i+1}, pe pasul $i+1$, se defineşte ca eroarea formulei (8) a metodei, când înlocuim aproximaţiile x_j cu soluţia exactă $x(t_j)$. Adică, definim T_{i+1} prin:

$$x(t_{i+1}) = x(t_i) + h\phi(t_i, x(t_i), h) + T_{i+1}, \qquad (15)$$

Dacă

$$T_{i+1} = O(h^{p+1}) \qquad (15')$$

metoda se zice de ordinul *p*. (Mai precis, *p* este cel mai mare întreg pentru care avem (15')). Aceasta revine la condiţia ca ca formula (8) să coincidă cu seria Taylor a lui $x(t_{i+1}) = x(t_i + h)$, trunchiată până la termenii de ordinul *p* în *h* inclusiv. Pentru a obţine o metodă de ordin *p*, coeficienţii α_m, β_{mj} şi ω_m se determină din condiţia de mai sus, cu respectarea condiţiilor (14, 14').

Eroarea de trunchiere globală pe pasul $i+1$, este eroarea aproximaţiei x_{i+1}, adică

$e_{i+1} = x(t_{i+1}) - x_{i+1}$.

În 3.3.6 se va arăta că $T_{i+1} = O(h^{p+1}) \Rightarrow e_{i+1} = O(h^p)$. Astfel, o metodă RK de ordinul *p* are o eroare globală de ordinul h^p.

În ceea ce urmează vom analiza numai metodele RK explicite. Pentru metodele implicite trimitem la Hairer & Wanner (1991).

Exemplu:

Metoda RK explicită, cu 2-trepte şi de ordinul 2 ($q = 2$ şi $p = 2$). Se obţine o familie cu un parametru, de metode explicite RK cu 2-trepte, de ordinul doi, definite de formulele:

$$x_{i+1} = x_i + h[(1-\omega_2)k_1 + \omega_2 k_2]$$

$$k_1 = f(t_i, x_i)$$

$$k_2 = f(t_i + \frac{h}{2\omega_2}, x_i + \frac{h}{2\omega_2} k_1)$$

Metode cunoscute se obţin cu $\omega_2 = \frac{1}{2}, \frac{3}{4}, 1$. De exemplu, pentru $\omega_2 = 1$ metoda se zice metoda Runge de ordinul 2, iar tabloul Butcher (13) este:

0	
$\frac{1}{2}$	$\frac{1}{2}$
	0 1

■

3.3.2 Ordin şi număr de trepte (evaluări de funcţii / pas)

Se arată că, în general, pentru ca o metodă explicită să aibă ordinul *p*, ea trebuie să aibă $q \geq p$ trepte, şi anume: pentru *p* = 1, 2 ,3 4, avem $q_{\min} = p$; pentru *p* > 4, $q_{\min} > p$. Mai precis, avem următoarele rezultate datorate lui Butcher (Hairer, Nørsett, & Wanner (1987)):

c) Pentru $p \geq 5$ nu există metode explicite de ordin *p*, cu $q = p$ trepte.

d) Pentru $p \geq 7$ nu există metode explicite de ordin *p*, cu $q = p+1$ trepte.

e) Pentru $p \geq 8$ nu există metode explicite explicite de ordin *p*, cu $q = p+2$ trepte.

Aceste rezultate sunt numite "barierele Butcher". Pentru *p* = 9, 10 se cunosc numai margini pentru $q_{\min}$, iar pentru $p > 10$ nu se cunosc evaluări pentru $q_{\min}$. Rezultatele anterioare se pot sintetiza în tabloul următor (Cartwright & Piro (1992)):

p	1	2	3	4	5	6	7	8	9	10
$q_{\min}(p)$	1	2	3	4	6	7	9	11	12 … 17	13 … 17

Ordinul maxim pentru care avem *q* = *p* este $p = 4$. Din acest motiv, metoda RK de ordinul 4 este cea mai frecvent utilizată. (Pentru $p > 4$ trebuie adăugate cel puţin

două trepte, ceea ce măreşte timpul de calcul şi introduce erori de rotunjire suplimentare.). În ceea ce priveşte metodele RK *implicite*, pentru orice număr de trepte *q*, există metode de ordinul $p = 2q$. V. Hairer, Nørsett, & Wanner (1987).

3.3.3 Convergenţă şi consistenţă

Metoda RK se zice convergentă dacă, pentru $h \to 0$, soluţia calculată tinde la soluţia exactă (pe fiecare t_i). Considerând intervalul de integrare $[t_0, t_i]$, şi notând $c = t_i - t_0$, numărul de paşi de integrare va fi $i = c/h$, sau $ih = c$. Astfel, condiţia se exprimă prin limita:

$$\lim_{\substack{h\to 0 \\ ih=c}} x_i = x(t_i)$$

Metoda RK, definită de (8) se zice *consistentă* (cu problema cu valori iniţiale) dacă avem

$$\phi(t_i, x(t_i), 0) = f(t_i, x(t_i)) \qquad (20)$$

Cu (20) şi expresiile (11, 12) ale funcţiilor k_m, avem

$$\phi(t_i, x(t_i), 0) = \sum_{m=1}^{q} \omega_m (k_m)\,|_{h=0} = f(t_i, x(t_i)) \sum_{m=1}^{q} \omega_m$$

şi condiţia de consistenţă (20) este echivalentă cu condiţia

$$\sum_{m=1}^{q} \omega_m = 1 \qquad (21)$$

Se demonstrează că, *consistenţa este o condiţie necesară şi suficientă pentru convergenţă* (Cartwright & Piro (1992)).

3.3.4 Metode RK de ordinul 4

O metodă RK explicită, de ordinul 4 (abreviat RK4), este definită de (10) cu $q = 4$:

$$x_{i+1} = x_i + h(\omega_1 k_1 + \omega_2 k_2 + \omega_3 k_3 + \omega_4 k_4)$$

în care, conform (11), avem:

$$k_1 = f(t_i, x_i),$$

$$k_2 = f(t_i + h\alpha_2, x_i + h\beta_{21} k_1),$$

$$k_3 = f(t_i + h\alpha_3, x_i + h(\beta_{31} k_1 + \beta_{32} k_2))$$

$$k_4 = f(t_i + h\alpha_4,\ x_i + h(\beta_{41}k_1 + \beta_{42}k_2 + \beta_{43}k_3))$$

Deducerea coeficienţilor metodei conduce la o familie cu doi parametri – v. Hairer, Nørsett, & Wanner (1987), Ralston & Rabinowitz (1978). Cele mai uzuale metode RK4 sunt definite de următoarele tablouri Butcher:

"Metoda" RK4

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{1}{6}$

"Regula 3/8"

0				
$\frac{1}{3}$	$\frac{1}{3}$			
$\frac{2}{3}$	$-\frac{1}{3}$	$\frac{1}{2}$		
1	1	-1	1	
	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$

Se verifică condiţia de consistenţă (21) şi condiţiile (14'). Prima metodă este cea mai uzuală, fiind denumită "Metoda" RK de ordinul 4. A doua este ceva mai precisă decât prima (Hairer et al. (1987)).

Explicit, "Metoda" RK4 este dată de formulele:

$$x_{i+1} = x_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{22}$$

în care:

$$\begin{aligned} k_1 &= f(t_i,\ x_i) \\ k_2 &= f(t_i + \tfrac{1}{2}h,\ x_i + \tfrac{1}{2}hk_1) \\ k_3 &= f(t_i + \tfrac{1}{2}h,\ x_i + \tfrac{1}{2}hk_2) \\ k_4 &= f(t_i + h,\ x_i + hk_3) \end{aligned} \tag{23}$$

Pentru un sistem de ecuaţii diferenţiale (de ordinul întâi), formulele metodei RK4 sunt similare cu (19, 20), variabilele scalare *x*, *f*, *k*, înlocuindu-se cu vectorii **x**, **f**, **k**:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \tag{22a}$$

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(t_i,\ \mathbf{x}_i) \\ \mathbf{k}_2 &= \mathbf{f}(t_i + \tfrac{1}{2}h,\ \mathbf{x}_i + \tfrac{1}{2}h\mathbf{k}_1) \\ \mathbf{k}_3 &= \mathbf{f}(t_i + \tfrac{1}{2}h,\ \mathbf{x}_i + \tfrac{1}{2}h\mathbf{k}_2) \\ \mathbf{k}_4 &= \mathbf{f}(t_i + h,\ \mathbf{x}_i + h\mathbf{k}_3) \end{aligned} \tag{23a}$$

În programarea formulelor (22a, 23a) vectorii se reprezintă prin tablouri: `x(0:n), f(1:m), k(1:m)`. Reamintim că *m* desemnează numărul de ecuaţii, iar *n* numărul paşilor de integrare.

Metode RK de ordin mai înalt

Cel mai înalt ordin pentru care s-au construit metode RK explicite este 10: Curtis (18 trepte, 1975) şi Hairer (17 trepte, 1978).

3.3.5 Metode RK îmbricate

Fie o metodă RK de ordin *p,* cu *q* trepte, care calculează soluţia

$$x_{i+1} = x_i + h\sum_{m=1}^{q} \omega_m k_m \qquad (24)$$

Funcţiile k_m sunt definite de (11) şi revin la calculul funcţiei *f* în puncte de forma

$$(t_i + h\alpha_m,\ x_i + h\sum_{j=1}^{m-1} \beta_{mj} k_j).$$

Ideea metodei îmbricate este de a o combina metoda (24) cu o metodă RK de ordin p' (uzual $p' = p+1$ sau $p' = p-1$), cu acelaşi număr de trepte *q*, şi care să calculeze funcţia *f* pe *aceleaşi* puncte ca (24) – adică având *aceeaşi* coeficienţi α_m, β_{mj}. Fie cea de-a doua metodă, care calculează soluţia

$$\hat{x}_{i+1} = x_i + h\sum_{m=1}^{q} \hat{\omega}_m k_m . \qquad (25)$$

În (24) şi (25), *q* este numărul de trepte din metoda de ordin mai mare. Pentru fixarea ideilor să presupunem că $p' > p$: atunci $q = q_{\min}(p')$, iar metoda de ordin *p* va avea numărul de trepte $q > q_{\min}(p)$. Astfel, metoda de ordin mai mic are trepte – sau grade de libertate – suplimentare. Coeficienţii metodei imbricate se determină astfel ca ei să minimizeze coeficienţii care definesc eroarea în una din cele două metodele. Soluţia $\hat{x}_{i+1}$ se utilizează pentru estimarea erorii de trunchiere prin (citeşte $\cong$ ‘egal prin estimare’):

$$T_p \cong \hat{x}_{i+1} - x_{i+1}, \qquad (26)$$

O astfel de metodă se va nota RK $p(p')$ – exemplu RK 4(5).

Explicit, eroarea de trunchiere locală se estimeză prin

$$T_p \cong h\sum_{m=1}^{q}(\hat{\omega}_m - \omega_m)k_m \qquad (26')$$

Metode Runge-Kutta-Fehlberg:

Fehlberg a construit astfel de metode de ordine $p(p+1)$, care să minimizeze coeficienţii erorii în metoda de ordin mai mic. Ele sunt numite metode Runge-Kutta-Fehlberg (RKF). Cele mai cunoscute sunt metodele RKF 4(5) şi 7(8). Cea mai utilizată dintre acestea este metoda de ordinul 4 cu 6 trepte (p = 4, p' = 5, $q(p') = 6$), definită de următorul tablou Butcher – în ultima linie sunt daţi coeficienţii $\hat{\omega}_m$:

Metoda RKF 4(5)

0						
$\frac{1}{4}$	$\frac{1}{4}$					
$\frac{3}{8}$	$\frac{3}{32}$	$\frac{9}{32}$				
$\frac{12}{13}$	$\frac{1932}{2197}$	$\frac{7296}{2197}$				
1	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		
$\frac{1}{2}$	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	
ω	$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	0
$\hat{\omega}$	$\frac{16}{135}$	0	$\frac{6656}{12825}$	$\frac{28561}{56430}$	$-\frac{9}{50}$	$\frac{2}{55}$

Metoda RKF 4(5) se găseşte implementată în multe pachete de programe pentru integrarea numerică a ecuaţiilor diferenţiale. Implementarea conduce la o metodă RKF cu *pas variabil*: estimarea (26) se utilizează pentru a controla eroarea metodei (24) şi a modifica pasul dacă eroarea depăşeşte o toleranţă impusă – v. mai jos.

Metode Dormand-Prince (DOPRI):

Dormand & Prince au construit metode mai precise, de ordine $p+1(p)$, în care se minimizează coeficienţii erorii în metoda de ordin mai mare. Soluţia calculată este dată de metoda cu ordinul $p+1$, iar metoda de ordinul p se utilizează numai pentru controlul pasului. Acestea sunt metodele DOPRI 5(4) – ordin 5, cu 7 trepte, şi DOPRI 8(7) – ordin 8, cu 12 trepte. Coeficienţii metodelor, ca şi coduri Fortran, sunt date în tratatul Hairer, Nørsett, & Wanner (1987). Codurile Fortran găsesc şi

la adresa: `http://www.unige.ch/math/folks/hairer/software.html`. Metode DOPRI sunt prezentate în Dormand (1996). Coduri Fortran sunt date la adresa: `ftp://ftp.tees.ac.uk/pub/j.r.dormand/`.

Metodele DOPRI sunt cele mai precise metode explicite pentru integrarea numerică a ecuaţiilor diferenţiale de ordinul întâi, existente în momentul de faţă.

Alegerea pasului

Considerăm o metodă imbricată cu $p < p'$, şi un pas curent, notând pentru simplificare $x_0 = x_i$ şi $x_1 = x_{i+1}$. Eroarea de trunchiere locală estimată conform (26) este:

$$T_p \cong \hat{x}_1 - x_1$$

Avem:

$$T_p \cong \hat{x}_1 - x(t_0 + h) + x(t_0 + h) - x_1 \cong O(h^{p'+1}) + O(h^{p+1})$$

sau, cu $p < p'$, avem eroarea absolută:

$$err = |T_p| \cong Ch^{p+1}$$

Pasul optim este cel pentru care eroarea este aproximativ egală cu toleranţa *tol* specificată de utilizator, adică:

$$tol \approx Ch_{opt}^{p+1},$$

Eliminând *C* între ultimele două realţii, rezultă:

$$\frac{tol}{err} \approx \left(\frac{h_{opt}}{h} \right)^{p+1}$$

din care,

$$h_{opt} \approx \left(\frac{tol}{err} \right)^{\frac{1}{p+1}} h$$

Pentru siguranţă, în program se pune:

$$h_{opt} = 0.9 \left(\frac{tol}{err} \right)^{\frac{1}{p+1}} h$$

În formula anterioară, $err = |\hat{x}_1 - x_1| = |T_p|$ unde T_p este estimarea (26') a erorii.

Pentru un sistem, modulul se înlocuieşte cu norma: $err = \|\hat{\mathbf{x}}_1 - \mathbf{x}_1\|$.

Observaţii

1) Dacă se cere specificare toleranţei *tolrel* la eroarea relativă în modul *rel*, atunci avem $rel \cong \frac{err}{|x_1 + err|}$ şi rezultă

$$rel \cong \frac{Ch^{p+1}}{|x_1 + err|},\ tolrel \approx \frac{Ch_{opt}^{p+1}}{|x_1 + err|},\ \text{de unde}$$

$$\frac{tolrel}{rel} \approx \left(\frac{h_{opt}}{h}\right)^{p+1},$$

$$h_{opt} = \left(\frac{tolrel}{rel}\right)^{\frac{1}{p+1}} h$$

În formula anterioară, *rel* este dat de expresia de mai sus în care *err* este estimarea (26') în modul. Pentru un sistem, avem $rel = \max_j \frac{err_j}{|x_{1j} + err_j|}$.

2) Eroarea *err* se mai estimează şi prin aşa numita extrapolare Richardson, calculând în paralel, soluţia x_2 cu doi paşi de mărime *h* şi soluţia X_2 cu un pas dublu 2*h*, şi estimând eroarea prin diferenţa celor două soluţii. Pentru o metodă de ordinul *p* se obţine (Hairer et al. (1987)):

$$x(t_0 + 2h) - x_2 = \frac{x_2 - X_2}{2^p - 1} + O(h^{p+2}),$$

$$T_p = \frac{x_2 - X_2}{2^p - 1}.$$

Soluţia

$$\hat{x}_2 = x_2 + T_p$$

este o aproximaţie a lui $x(t_0 + 2h)$, cu o eroare de ordinul *p*+1. Pentru controlul erorii avem:

$$err \cong \frac{|x_2 - X_2|}{2^p - 1},\ rel \cong \frac{err}{|\hat{x}_2|}.$$

Pentru sistem, în *err* modulul se înlocuieşte cu norma, iar $rel = \max_j \frac{err_j}{|\hat{x}_{2j}|}$, unde

err_j este estimarea *err* pentru coordonata *j* a soluţiei. Estimările *err, rel* se pot folosi în formulele anterioare pentru h_{opt}.

3) Codurile care implementează metode cu pas variabil utilizează fie *tol*, fie *tolrel*, fie ambele, împreună cu alte mecanisme de control al pasului care previn creşterea sau scăderea excesivă a pasului. De exemplu, în unul din cele mai noi coduri – v. RKSUITE în Brankin and Gladwell (1994), utilizatorul specifică toleranţa *TOL* a erorii relative, iar testul de eroare cere ca pe fiecare pas *i*:

$$|eroare(j)| \le TOL * \max(mag(j),\ prag(j))$$

unde $mag(j)$ este o mărime medie a coordonatei *j* a soluţiei x_i pe pasul considerat, iar *prag* este un tablou specificat de utilizator. Astfel, dacă *prag*(*j*) > *mag*(*j*) rezultă un test de eroare absolută cu toleranţa *tol* = *TOL*∗*prag*(*j*), iar pentru *prag*(*j*) < *mag*(*j*) rezultă un test de eroare relativă cu *tolrel* = *TOL*.

■

3.3.6 Estimarea erorii de trunchiere globale

Notăm acum cu $\bar{e}_{i+1}$ şi $\bar{T}_{i+1}$, *modulul* erorii de trunchiere locală, şi respectiv globală, pe pasul *i*+1. După definiţiile din 3.3.1, avem:

$$\bar{e}_{i+1} = |x(t_{i+1}) - x_{i+1}|, \qquad (27)$$

şi definim $\bar{e}_0 = 0$, şi

$$\bar{T}_{i+1} = |x(t_{i+1}) - x(t_i) - h\phi(t_i, x(t_i), h)| \qquad (28)$$

Se arată că: eroarea de trunchiere globală este $\bar{e}_i = O(h^p)$.

Eroarea de trunchiere locală (în modul) se poate scrie sub forma

$$\bar{T}_{i+1} = \psi(x(t_i))h^{p+1} + O(h^{p+2}) \qquad (30)$$

Primul termen se zice eroarea de trunchiere locală *principală*.

Pentru un sistem, în expresiile anterioare modulul se înlocuieşte cu norma.

3.3.7 Stabilitatea metodelor RK (stabilitatea absolută liniară)

Ne vom limita la stabilitatea *liniară* a metodei. Aceasta se studiază prin liniarizarea ecuaţiei (1) în jurul unei soluţii a acesteia. Fie ecuaţia diferenţială

$$x'(t) = f(t, x) \tag{31}$$

şi $x = \varphi(t)$ o soluţie netedă a acesteia, adică

$$\varphi'(t) = f(t, \varphi(t)). \tag{32}$$

Considerăm o *perturbaţie* $\delta x(t)$ a soluţiei (provenind dintr-o perturbare a condiţiei iniţiale), unde $|\delta x(t)| \le \varepsilon$:

$$\delta x(t) = x(t) - \varphi(t), \quad x(t) = \varphi(t) + \delta x(t)$$

şi scăzând relaţia (32) din (31) rezultă

$$\frac{d}{dt}\delta x(t) = f(t, \varphi(t) + \delta x(t)) - f(t, \varphi(t)) \tag{33}$$

Desvoltăm membrul doi în (33) în jurul lui $\varphi(t)$ pînă la termenul de ordinul întâi în $\delta x(t)$. Obţinem:

$$\frac{d}{dt}\delta x(t) = \frac{\partial f}{\partial x}(t, \varphi(t))\delta x(t) + \ldots = J(t)\delta x(t) + \ldots \tag{34}$$

în care $J(t) = (\partial f / \partial x)|_{(t, \varphi(t))}$. Ecuaţia (34) *liniarizată* se obţine neglijând termenii nescrişi, şi anume:

$$\frac{d}{dt}\delta x(t) = J(t)\delta x(t)$$

În fine, în primă aproximaţie, considerăm $J(t) = J$ = constant (şi anume $J = J(t^*)$, unde $t^* \in (t, t+h)$) şi avem

$$\frac{d}{dt}\delta x(t) = J\delta x(t) \tag{35}$$

Ecuaţia (35) poate fi scalată, astfel că perturbaţia să fie de mărime *arbitrară*. Punem

$y(t) = C\delta x(t)$, unde C este o constantă, şi ecuaţia (35) devine

$$y' = Jy. \qquad (35')$$

În fine, notând *x* în loc de *y* , ecuaţia (35') devine o ecuaţie de tipul

$$x' = \lambda x \qquad (36)$$

în care, în general, λ va fi considerat complex (v. mai jos, cazul unui sistem). Ecuaţiei (36) îi ataşăm o condiţie iniţială arbitrară:

$$x(t_0) = x^{(0)} \qquad (36')$$

Problema (36, 36') constituie testul pentru stabilitatea liniară a metodei – numit şi testul *Dalquist*. Soluţia exactă a problemei este:

$$x(t) = x^{(0)} e^{\lambda(t-t_0)} \qquad (37)$$

Dacă $\mathrm{Re}(\lambda) < 0$, atunci avem $t \to \infty \Rightarrow x(t) \to 0$. Se zice că problema are un punct fix stabil, în *x* = 0.

Observaţie

Punctele fixe ale unei ecuaţii (31) sunt valorile *x* pentru care avem $f(t, x) = 0$, $\forall t \geq t_0$. Punctele fixe ale unei metode numerice explicite $x_{i+1} = g(x_i)$, sunt date de $x_{i+1} = x_i$ (adică de soluţiile ecuaţiei $x = g(x)$). Punctele fixe ale unei metode RK definită de (8) sunt date de

$$\phi(t_i, x_i, h) = \sum_{m=1}^{q} \omega_m k_m = 0$$

în care, pentru o metodă explicită:

$$k_m = f(t_i + h\alpha_m,\ x_i + h\sum_{j=1}^{m-1} \beta_{mj} k_j)$$

Dacă X este un punct fix al ecuaţiei $x' = f(t, x)$, atunci avem $f(t, X) = 0, \forall t \geq t_0$, şi rezultă: $k_1 = f(t, X) = 0$, $k_2 = f(t, X) = 0$, … , sau $k_m = 0, m = \overline{1, q}$. Pentru o metodă implicită avem acelaşi rezultat. Urmează că punctele fixe ale ecuaţiei sunt şi puncte fixe ale metodei RK ■

Definiţie

O metodă numerică este stabilă liniar dacă, aplicată ecuaţiei (36) avem:

$i \to \infty \Rightarrow x_i \to 0$,

adică metoda *păstrează* stabilitatea punctului fix $x = 0$ ■

Pentru un sistem de m ecuaţii diferenţiale (3)

$$\mathbf{x}' = \mathbf{f}(t, \mathbf{x})$$

avem, analog cu cazul unei singure ecuaţii: fie soluţia $\boldsymbol{\varphi}(t)$

$$\boldsymbol{\varphi}'(t) = \mathbf{f}(t, \boldsymbol{\varphi}(t)),$$

punem

$$\delta\mathbf{x}(t) = \mathbf{x}(t) - \boldsymbol{\varphi}(t)$$

şi rezultă

$$\frac{d}{dt}\delta\mathbf{x}(t) = \mathbf{f}(t, \mathbf{x}) - \mathbf{f}(t, \boldsymbol{\varphi}) = \mathbf{J}(t)\delta\mathbf{x}(t) + \ldots$$

în care $\mathbf{J}(t) = [\partial f_i / \partial x_k]|_{(t, \boldsymbol{\varphi}(t))}$ este jacobianul funcţiei **f** în raport cu **x**. Aproximăm **J**(t) = **A** = constant. Cu aceasta, schimbând notaţia $\delta\mathbf{x} \mapsto \mathbf{x}$, modelul liniar este

$$\begin{aligned} &\mathbf{x}' = \mathbf{A}\mathbf{x} \\ &\mathbf{x}(t_0) = \mathbf{x}^{(0)} \end{aligned} \qquad (38)$$

în care **A** este o matrice constantă $m \times m$. Presupunem, pentru simplificare, că **A** că are valori proprii λ_j, $j = \overline{1, m}$ distincte (şi, în general, complexe). Presupunem că valorile proprii au partea reală negativă: atunci avem un punct fix stabil în **x** = **0**.

Întrucât valorile proprii sunt distincte, există o bază ortogonală formată din vectorii proprii în care matricea **A** se diagonalizează, iar ecuaţiile (38) se decuplează, sistemul reducându-se la m ecuaţii independente de forma (31). Într-adevăr, dacă vectorii proprii sunt $\{\mathbf{v}_j,\ j = \overline{1, m}\}$, definiţi de $\mathbf{A}\mathbf{v}_j = \lambda_j \mathbf{v}_j$, punem $\mathbf{x} = \sum_j y_j \mathbf{v}_j$ şi avem $\mathbf{x}' = \sum_j (y_j)' \mathbf{v}_j$, $\mathbf{A}\mathbf{x} = \sum_j \lambda_j y_j \mathbf{v}_j$. Înlocuind în (35) rezultă

$$y_j' = \lambda_j y_j, \quad j = \overline{1, m}$$

la care adăugăm condiţii iniţiale arbitrare, de exemplu

$$y_j(t_0) = y_j^{(0)}, \quad j = \overline{1, m}$$

Astfel, în ipotezele făcute, analiza stabilităţii pentru sistemul (38) se poate face pe o singură ecuaţie de forma (36) ■

Revenind la ecuaţia (36) să-i aplicăm metoda explicită RK de ordinul 2, considerată în 3.3.1:

$x_{i+1} = x_i + h[(1-\omega_2)k_1 + \omega_2 k_2]$.

Cu $f(t,x) = \lambda x$, rezultă $k_1 = \lambda x_i$, $k_2 = \lambda(x_i + \frac{h}{2\omega_2}\lambda x_i)$, şi

$$x_{i+1} = x_i + h[(1-\omega_2)\lambda x_i + \omega_2\lambda(x_i + \frac{h}{2\omega_2}\lambda x_i)] = x_i + h(\lambda x_i + \frac{h}{2}\lambda^2 x_i)$$

Avem:

$x_{i+1} = R(h\lambda)\cdot x_i$,

unde

$$R(h\lambda) = 1 + h\lambda + \frac{(h\lambda)^2}{2}$$

Să observăm că, cu $x_0 = 1$, avem $x_1 = R(h\lambda)$.

Definiţie

$R(h\lambda)$ se numeşte *funcţia de stabilitate* a metodei. Ea poate fi considerată ca soluţia numerică după un pas, a problemei liniare de test (36, 36'), cu $x^{(0)} = 1$ ■

Regiunea de stabilitate absolută pentru o metodă, este mulţimea valorilor h şi λ (h = real şi nenegativ, λ = complex), pentru care avem $x_i \to 0$ pentru $i \to \infty$, adică punctul fix $x = 0$ (originea) este stabil. Pentru aceasta este necesar şi suficient ca să avem $|R| < 1$. Punând $z = h\lambda$, regiunea de stabilitate este mulţimea

$S = \{z \in \mathbf{C}; |R(z)| < 1\}$.

Uneori, regiunea de stabilitate este definită împreună cu frontiera sa, prin condiţia $|R| \le 1$. Pentru metoda explicită RK de ordinul 2, regiunea de stabilitate va fi dată de:

$|1 + z + z^2/2| < 1$

Pentru un sistem, λ va fi valoarea proprie de modul maxim a matricii jacobian **A**. Să considerăm acum, cazul general al unei metode explicite cu *p* trepte, de ordinul *p* (adică, $p \le 4$). Considerăm desvoltarea lui *x*(*t*) în serie Taylor, până la ordinul *p*. Cu $x' = \lambda x$, rezultă $x'' = \lambda x' = \lambda^2 x$, şi în general, $x^{(r)} = \lambda^r x, \ r \le p$, astfel că avem: $x(t_{i+1}) = x(t_i) + h\lambda x(t_i) + \frac{h^2}{2!}\lambda^2 x(t_i) + \ldots + \frac{h^p}{p!}\lambda^p x(t_i) + O(h^{p+1})$

În fine, cu $x_i = x(t_i)$, şi omiţând restul $O(h^{p+1})$, avem:

$$x_{i+1} = (1 + h\lambda + \frac{h^2}{2!}\lambda^2 + \ldots + \frac{h^p}{p!}\lambda^p)x_i$$

care arată că funcţia de stabilitate este

$$R = 1 + h\lambda + \frac{(h\lambda)^2}{2!} + \ldots + \frac{(h\lambda)^p}{p!}; \quad p \le 4. \tag{39}$$

Pentru o metodă explicită de ordin *p*, cu $q > p$ trepte, funcţia de stabilitate va fi

$$R = 1 + h\lambda + \frac{(h\lambda)^2}{2!} + \ldots + \frac{(h\lambda)^p}{p!} + \sum_{j=p+1}^{q} \gamma_j (h\lambda)^j, \tag{40}$$

unde γ_j sunt definiţi de coeficienţii metodei. De exemplu, pentru metoda DOPRI 5(4) – cu 6 trepte (treapta 7 se utilizează numai pentru estimarea erorii) – termenul adiţional în (40) este $(h\lambda)^6/600$. (Hairer & Wanner (1991).

Din aceasta rezultă că funcţia de stabilitate a unei metode explicite cu *q* trepte este un polinom de gradul *q* în *h*. Condiţia $|R| < 1$ conduce la o regiune de stabilitate mărginită. (Dacă aceasta ar fi nemărginită nu putem avea $|R| < 1$, întrucât $|h\lambda| \to \infty \Rightarrow |R| \to \infty$). Metodele RK implicite pot avea regiuni de stabilitate nemărginite. Aceste metode se aplică pentru ecuaţii diferenţiale *rigide* – v. 5, la care metodele explicite nu mai convin.

Pentru reprezentarea regiunilor de stabilitate liniară în cazul λ = complex, pentru metodele RK*p*, *p* =1, 2, 3, 4 – v. Hairer & Wanner (1991), Cartwright & Piro (1992). Intersecţiile regiunilor cu axa reală dau intervalele de stabilitate pentru cazul λ = real. Aceste intervale se găsesc din condiţia $|R| < 1$, unde *R* este definit de (37) cu λ = real şi negativ (conform ipotezei $\mathrm{Re}(\lambda) < 0$). Rezultă:

p = 1, 2: -2 < $h\lambda$ < 0; p = 3: -2.512745 < $h\lambda$ < 0; p = 4: -2.785296 < $h\lambda$ < 0.

3.3.8 Stabilitatea absolută neliniară

Stabilitatea neliniară este o problemă mult mai complexă. Ea are conexiune cu dinamica haotică. În cazul unei probleme neliniare, regiunea de stabilitate a unei metode RK poate fi diferită de regiunea ei de stabilitate liniară. Cea mai importantă diferenţă constă în aceea că, pentru o problemă neliniară, metodele RK pot conţine pe lângă punctele fixe ale problemei – v. Observaţia din 3.3.7 – şi puncte fixe adiţionale. Excepţie face metoda Euler care are numai punctele fixe ale problemei. Punctele fixe adiţionale sunt numite puncte fixe *fantomă*. Recent (1991) s-a arătat că, în unele cazuri, puncte fixe fantomă pot exista la orice lungime a pasului (diferită de zero), adică la paşi pentru care $h\lambda$ este în regiunea de stabilitate liniară absolută. Dacă un asemenea punct fix este stabil la paşi oricât de mici, atunci o traiectorie (calculată) poate converge la un punct fix care nu există în dinamica problemei originale. Diferenţa între problemele liniare şi neliniare constă în aceea că, pentru probleme neliniare bazinul de atracţie este mărginit, în timp ce pentru o problemă liniară acesta este nemărginit. Astfel, pentru o problemă liniară există convergenţă pentru orice condiţii iniţiale, cu condiţia ca $h\lambda$ să fie în interiorul regiunii de stabilitate, în timp ce pentru o problemă neliniară este necesar, în plus, ca condiţiile iniţiale să fie conţinute în bazinul de atracţie. Pentru desvoltări, trimitem la Cartwright & Piro (1992).

În practica de calcul s-a constatat că pentru un răspuns haotic, unde calculaţia se face pe un mare număr de paşi (sute de mii sau milioane), codul Runge-Kutta de ordinul 4 – formulele (22a, 23a) – este foarte *sensibil* la mici schimbări ca: utilizarea de variabile locale, asocierea în operaţiile aritmetice, vectorizarea ciclurilor DO în subrutina de integrare a sistemului dat, opţiunile de "build" (ca optimizarea codului), etc. V. raportul Chisăliţă A. & al. (1998).

3.3.9 Exemplu de test – Problema celor două corpuri

Următoarea problemă, constituită de problema celor două corpuri în cazul mişcării eliptice, este luată ca test pentru metodele de integrare numerică a problemei cu valori iniţiale – v. Dormand and Prince (1978), Brankin and Gladwell (1994). Problema consideră mişcarea relativă a două puncte materiale care interacţionează

prin legea atracţiei universale, şi este descrisă, în coordonate carteziene, de sistemul de ecuaţii diferenţiale:

$\ddot{x} = -x/r^3,\ \ddot{y} = -y/r^3,$

în care $r = (x^2 + y^2)^{1/2}$. Se consideră conditiile iniţiale pentru cazul mişcării eliptice $x(0) = 1-e,\ \dot{x}(0) = 0,\ y(0) = 0,\ \dot{y}(0) = \sqrt{(1+e)/(1-e)}$,

în care $e < 1$. Soluţia analitică este dată de:

$$x = \cos u - e, \quad y = \sqrt{1-e^2}\sin u, \quad \dot{x} = \frac{-\sin u}{1-e\cos u}, \quad \dot{y} = \frac{\sqrt{1-e^2}\cos u}{1-e\cos u}$$

în care u se determină din ecuaţia lui Kepler: $u - e\sin u = t$. Soluţia este periodică cu perioada minimă $T = 2\pi$, iar orbita este o elipsă cu excentricitatea e şi semi-axa mare egală cu 1. Problema reprezintă un test sever, datorită periodicităţii soluţiei. Pentru rezolvarea numerică, sistemul dat se transformă într-un sistem echivalent de 4 ecuaţii de ordinul întâi:

$$\dot{x} = v, \quad \dot{y} = w, \quad \dot{v} = -x/r^3, \quad \dot{w} = -y/r^3; \quad r = (x^2 + y^2)^{1/2}$$

cu condiţiile iniţiale: $x(0) = 1-e,\ y(0) = 0,\ v(0) = 0,\ w(0) = \sqrt{(1+e)/(1-e)}$.
Calculăm soluţia pe intervalul [0, 20], adică peste trei perioade, pentru valorile $e = 0.1$ şi $e = 0.9$ ale excentricităţii. Calculul este făcut în dublă precizie, cu metodele:

(a) RK4 (pas constant) – v. codul în ANA_EcDif.

(b) Runge-Kutta-Verner 5(6), cu subrutina DIVPRK din IMSL (pas variabil) – v. "IMSL Libraries Reference" (1998) – cu argumentele: *tol* = 1D-7 … 2.23 D-16, *param*(10) = 1 (se utilizează norma-∞ a erorii). Subrutina se bazează pe codul scris de Hull, Enright şi Jackson (1976), care utilizează formulele lui Verner de ordinul 5 şi 6 – v. DVERK, in site-ul: `http://www.cs.toronto.edu/NA/index.html`. Rutina poate utiliza paşi în plaja 2.22D-15 … 2.0 (valori implicite). Intervalul de integrare s-a împărţit în 20, şi respectiv în 200, sub-intervale. Rezultatele mai precise se obţin pentru împărţirea în 200 sub-intervale – în acest caz pasul maxim posibil este 0.1 (egal cu lungimea sub-intervalului).

(c) RK 8(7), cu subrutina DIVMRK din IMSL (pas variabil). S-a utilizat apelul subrutinei DI2MRK, cu specificarea argumentelor. Toleranţa *tol* s-a luat în plaja 1D-7 … 2.23 D-15. Subrutina implementează codul din RKSUITE – metodele RK de ordine 3(2), 5(4), şi metoda Dormand şi Prince de ordin 8(7) – v. Brankin and Gladwell (1994). Ordinul metodelor este 3, 5, şi 8, respectiv. Intervalul de integrare s-a împărţit în 20, şi respectiv în 200 sub-intervale. Rezultatele mai precise se obţin pentru împărţirea în 20 sub-intervale – în acest caz pasul maxim posibil este 1.0 (lungimea sub-intervalului)..

Pentru soluţia exactă, ecuaţia lui Kepler se rezolvă prin metoda punctului fix, cu toleranţa *eps* =1D-13. În tabelele următoare este dată eroarea absolută maximă şi minimă a soluţiei calculate, la timpul cel mai apropiat de 3 perioade. În paranteze se indică funcţia – dintre x, y, $\dot{x}$, $\dot{y}$ – pentru care are loc extremul erorii absolute.

■

e = 0.1: Erori absolute extreme la t = 18.84 (RK4); 18.60 (RKV); 18.0 (RK 8(7)).

Extrem eroare absolută	Metoda		
	RK4 h = 0.01	RKV 5(6) *tol* = 1D-10	RK 8(7) *tol* = 1D-10
Maximă Minimă	7.71 D-9 ($\dot{x}$) 4.38 D-11 (x)	2.68 D-9 (y) 4.49 D-10 (x)	1.57 D-10 ($\dot{y}$) 9.63 D-11 (y)
Număr paşi Nr. apeluri FCN	2000 8000	439 3512	138 2090

e = 0.9, Metoda RK4: Erori absolute extreme la t = 18.84 (h = 0.01; 0.005) şi t = 18.849 (h = 0.001; 0.0005)

Pasul h	Eroarea absolută Maximă ($\dot{x}$)	Minimă (x)	Număr de paşi
0.01	3.52 D0[†]	3.54 D-1	2000
0.005	7.12 D-1	4.26 D-3	4000
0.001	6.02 D-4	3.33 D-7	20000
0.0005	3.28 D-5	1.82 D-8	40000

[†] Eroarea maximă are loc în $\dot{y}$

e = 0.9, Metoda RKV 5(6) – 200 sub-intervale:

Erori absolute extreme la t = 18.60

Toleranţa *tol*	Eroarea absolută Maximă ($\dot{x}$)	Minimă (y)	Număr de paşi	Număr apeluri FCN
1 D-7	3.98 D-4	6.97 D-6	315	2779
1 D-10	1.28 D-6	2.23 D-7	622	5165
1 D-13	1.92 D-9	3.35 D-10	1800	14435
2.23 D-15	2.88 D-14	6.11 D-16	2244	17959

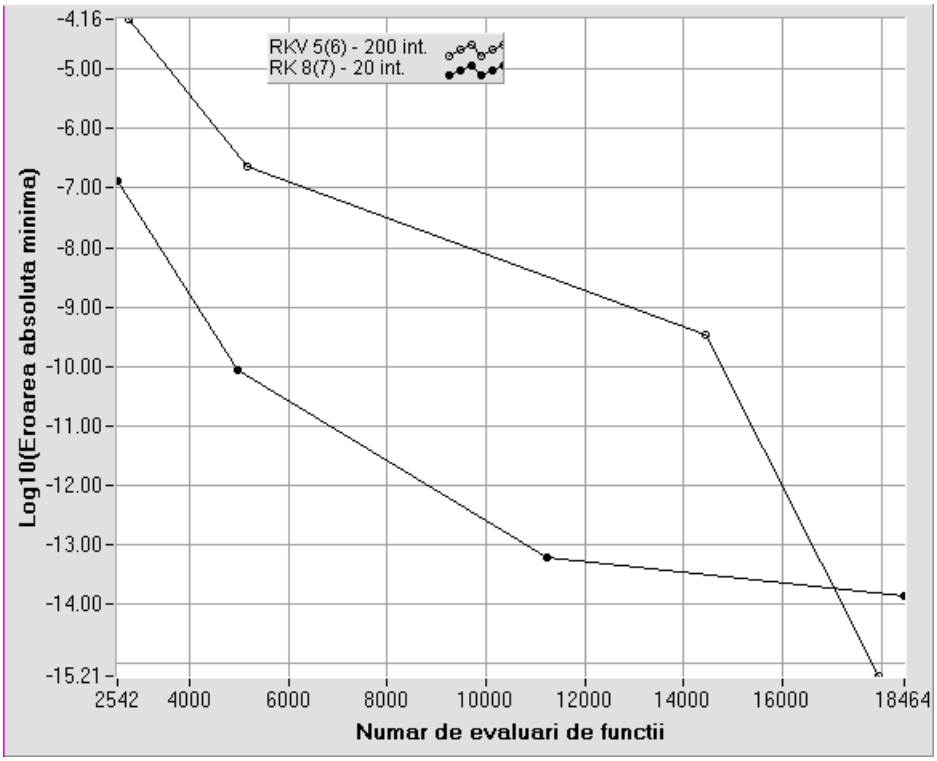
e = 0.9, Metoda RK 8(7) – 20 sub-intervale:

Erori absolute extreme la t = 18.00

Toleranţa *tol*	Eroarea absolută Maximă (x)	Minimă (y)	Număr de paşi	Număr apeluri FCN
1 D-7	2.16 D-6	1.31 D-7	152	2542
1 D-10	1.29 D-9	8.55 D-11	356	4984
1 D-13	9.00 D-13	5.96 D-14	847	11223
2.23 D-15[†]	2.21 D-13	1.38 D-14	1380	18464

[†] Toleranţa minimă admisă = 2.22 D-15.

Următorul grafic dă o comparaţie a eficienţei metodelor de mai sus, pentru cazul e = 0.9. În ordonată este reprezentat numărul $r = log_{10}$(*Eroarea absolută minimă*). El indică cea mai bună precizie atinsă de metodă (eroarea minimă este de ordinul 10^r) şi este reprezentat în funcţie de numărul de evaluări de funcţii. Metoda este cu atât mai eficientă, cu cât realizează o precizie dată cu un număr mai mic de evaluări de funcţii.



Problema celor două corpuri, $e = 0.9$: Eficienţa metodelor RKV 5(6), RK 8(7)

Observaţii

- Testul cel mai sever este cazul $e = 0.9$. Cu acelaşi pas (în RK4), sau aceeaşi toleranţă (în RKV 5(6), RK 8(7)), metodele dau rezultate cu o precizie inferioară cazului $e = 0.1$. Din acest motiv, s-au efectuat integrări cu paşi, respectiv toleranţe, mai mici. Să remarcăm că pasul $h = 0.01$ reprezintă aproximativ $T/628$, unde T este perioada mişcării. În cazul $e = 0.1$, pentru metodele RKV şi RK 8(7), s-a ales toleranţa 1D-10 pentru a avea erori comparabile cu cele din metoda RK4.
- În subrutina DIPVRK (metoda RKV 5(6)), argumentul *tol* serveşte pentru controlul normei erorii locale, în scopul de a se încerca menţinerea erorii globale aproximativ proporţională cu valorea *tol* (v. referinţele citate mai sus).
- În subrutina DI2MRK (metoda RK 8(7)), argumentul *tol* serveşte la controlul erorii relative, şi la alegerea ordinului metodei astfel: 1D-4 < *tol* ≤ 1D-2, 1D-6 < *tol* ≤ 1D-4, şi *tol* < 1D-6, produc alegerea metodei de ordinele 3(2), 5(4) şi 8(7), respectiv.
- Coloana “Număr apeluri FCN” dă numărul de apeluri ale subrutinei FCN care calculează membrii doi ai sistemului de ecuaţii. Acest număr este referit ca

"numărul de evaluări de funcţii" al metodei. Metoda RK4 face 4 apeluri ale subrutinei FCN pe un pas (În codul din Anexa, 4.1, FCN este `DERIVS`.).

- Se remarcă creşterea preciziei odată cu micşorarea pasului (RK4), sau a argumentului *tol* (RKV 5(6), RK 8(7)), dar cu preţul măririi numărului de paşi sau a numărului total de evaluări de funcţii. Se remarcă creşterea preciziei cu creştera ordinului metodei. Din comparaţia eficienţei celor trei metode rezultă că, pentru problema considerată, metoda RK 8(7) oferă cel mai bun raport precizie/număr de evaluări de funcţii – cu excepţia cazului unei toleranţe apropiată de cea minimă admisă (2.22D-15), când metoda RKV 5(6) este superioară ■

4 Operatori în mai mulţi paşi

4.1 Definiţii. Operatori liniari

Considerăm din nou, problema cu valori iniţiale (1,2)

$$x' = f(t,x); \qquad x(t_0) = x^{(0)} \tag{41}$$

pentru care se cere soluţia pe intervalul $[t_0, TT]$, şi nodurile t_j, $j = \overline{0,N}$ (unde $t_N = TT$). Notăm, ca înainte, cu $x(t_j)$ soluţia exactă şi cu x_j soluţia calculată în t_j, $j \geq 1$, iar $x_0 = x^{(0)}$. Un operator în *k*-paşi este o formulă de forma

$$x_{i+1} = g(x_{i+1}, x_i, x_{i-1}, \ldots, x_{i-k+1}), \tag{42}$$

care calculează soluţia x_{i+1} în funcţie de *k* valori x_j, $j = i, i-1, \ldots, i-k+1$ calculate anterior. La primul pas al metodei, aceste valori trebuie determinate printr-o procedură specială de start. Dacă x_{i+1} apare în membrul doi operatorul este *implicit*, altfel este *explicit*. În ceea ce urmează vom considera numai cazul operatorilor multi-pas *liniari* şi cu *pas constant h*, adică:

- $t_j = t_0 + jh, \quad j \geq 1;$
- Funcţia *g* este liniară în x_j şi în $f_j = f(t_j, x_j)$, $j = i+1, i, \ldots, i-k+1$.

Pentru convenienţă, ecuaţia (42) se scrie sub forma

$$\begin{aligned} x_{i+1} &= a'_{k-1}x_i + a'_{k-2}x_{i-1} + \ldots + a'_0 x_{i-k+1} \\ &\quad + h(b_k f(t_{i+1}, x_{i+1}) + b_{k-1} f(t_i, x_i) + \ldots b_0 f(t_{i-k+1}, x_{i-k+1})) \end{aligned} \tag{43}$$

În (43) vom presupune că cel puţin unul dintre a'_0, b_0 este diferit de zero (operatorul are *k* paşi); în rest, oricare alt coeficient poate fi zero. Dacă $b_k \neq 0$ operatorul este implicit, iar dacă $b_k = 0$ el este explicit. Condensat, (43) se scrie:

$$x_{i+1} = \sum_{l=0}^{k-1} a'_l x_{i-k+1+l} + h\sum_{l=0}^{k} b_l f_{i-k+1+l}, \quad i \geq k-1 \tag{43a}$$

Pentru simplificarea notaţiei indexate, să notăm valorile curente cum urmează: valoarea care se calculează cu x_k ($k = i+1, i = k-1$), şi cele k valori anterioare cu $x_{k-1}, x_{k-2}, \ldots, x_0$. (Aceasta nu restrânge generalitatea, coeficienţii a_l', b_l nedepinzând de i). Astfel, relaţia (44) se scrie:

$$x_k = \sum_{l=0}^{k-1} a_l x_l + h\sum_{l=0}^{k} b_l f_l \qquad (44)$$

Forma generală a unei metode multi-pas (44) este

$$\sum_{l=0}^{k} a_l x_l = h\sum_{l=0}^{k} b_l f_l \qquad (45)$$

în care s-a pus $a_l = -a_l', l = \overline{0, k-1}$.

În (45) se pun condiţiile:

$a_k = 1$ (mai general, $a_k \neq 0$, pentru explicitare în raport cu x_k)

$|a_0| + |b_0| \neq 0$ (metoda are k paşi).

Explicit:

$$a_0 x_0 + a_1 x_1 + \ldots + a_{k-1} x_{k-1} + a_k x_k = h(b_0 f_0 + b_1 f_1 + \ldots + b_{k-1} f_{k-1} + b_k x_k)$$

■

Exemple:

1) Metoda mijlocului este definită de formula

$$x_{i+1} = x_{i-1} + 2hf(t_i, x_i), \quad i \geq 1$$

şi este un operator explicit în 2 paşi.

2) Metoda trapezului este definită de

$$x_{i+1} = x_i + \frac{h}{2}(f(t_i, x_i) + f(t_{i+1}, x_{i+1})), \quad i \geq 0$$

şi este un operator implicit într-un singur pas ■

4.2 Ordin

Formula (44) sau (45) se zice "exactă" pentru o funcţie $x(t)$ dacă, din ipoteza că în membrul întâi avem $x_l = x(t_l),\quad l = \overline{i, i-k+1}$, rezultă ca avem şi $x_{i+1} = x(t_{i+1})$ – în limita erorilor de rotunjire.

Definiţie

Dacă formula (43) sau (44) este exactă pentru polinoamele de grad *p*, zicem că operatorul are ordinul *p* (sau, formula are ordinul de precizie *p*) ■

Lucrăm pe forma (45)

$$\sum_{l=0}^{k} a_l x_l = h\sum_{l=0}^{k} b_l f_l$$

Să presupunem că cerem ca (45) să aibă ordinul *p*. În acest caz $f(t, x(t)) = x'(t)$ este un polinom de grad $p-1$. Condiţia pusă revine la condiţia că formula să fie exactă pentru polinoamele $x(t) = t^q,\ q = \overline{0, p}$ (acestea alcătuiesc o bază pentru polinoamele de grad ≤ *p*). Avem $x'(t) = qt^{q-1}$. Putem presupune $t_0 = 0$ (coeficienţii nu pot depinde de t_0), avem $t_l = lh$, şi rezultă:

$x_l = x(t_l) = l^q h^q,\quad q \geq 0;$

$f_l = x'(t_l) = ql^{q-1}h^{q-1},\quad q \geq 1$, şi $f_l = 0$, pentru $q = 0$.

Ţinând cont de faptul că termenul al doilea în (44) conţine $hf_l = qlh^q$, urmează că h^q se va simplifica. Putem pune atunci, *h* = 1, şi avem:

$x_l = l^q,\quad q \geq 0;$

$f_l = ql^{q-1},\quad q \geq 1;\qquad f_l = 0$, pentru $q = 0$.

Înlocuind în (45), rezultă, pentru *q* = 0, 1 şi *q* = 2, … , *p*:

1) $q = 0$ ($x_l = 1, f_l = 0$):

$$\sum_{l=0}^{k} a_l = 0 \qquad (46a)$$

2) $q = 1$ ($x_l = l, f_l = 1$):

Rezultă, ţinând cont de (45a):

$$\sum_{l=0}^{k} l\alpha_l - \sum_{l=0}^{k} b_l = 0 \qquad (46b)$$

3) $q = 2, \ldots, p;\ p+1$ $\quad (x_l = l^q,\ f_l = lj^{q-1})$:

$$\sum_{l=0}^{k} l^q a_l - q\sum_{l=0}^{k} l^{q-1} b_l = 0, \quad q = 2, \ldots, p \qquad (46c)$$

$$\sum_{l=0}^{k} l^{p+1} a_l - (p+1)\sum_{l=0}^{k} l^p b_l \neq 0 \qquad (q = p+1) \qquad (46d)$$

Rezumând, avem condiţiile:

$$d_0 = \sum_{l=0}^{k} a_l = 0 \qquad (q = 0)$$

$$d_1 = \sum_{l=0}^{k} l a_l - \sum_{l=0}^{k} b_l = 0 \qquad (q = 1)$$

$$d_q = \sum_{l=0}^{k} l^q a_l - q\sum_{l=0}^{k} l^{q-1} b_l = 0, \qquad q = 2, \ldots, p \qquad (46)$$

$$d_{p+1} = \sum_{l=0}^{k} l^{p+1} a_l - (p+1)\sum_{l=0}^{k} l^p b_l \neq 0 \quad (q = p+1)$$

În (46) avem $l^0 = 1, l \geq 0$.

În particular, formulele explicite pentru q = 0, 1, 2, 3 sunt cele de mai jos, în care sumele se opresc la termenii de indice k, inclusiv:

$$d_0 = a_0 + a_1 + a_2 + a_3 + a_4 + \ldots$$

$$d_1 = a_1 + 2a_2 + 3a_3 + 4a_4 + \ldots - (b_0 + b_1 + b_2 + b_3 + b_4 + \ldots)$$

$$d_2 = a_1 + 4a_2 + 9a_3 + 16a_4 + \ldots - 2(b_1 + 2b_2 + 3b_3 + 4b_4 + \ldots)$$

$$d_3 = a_1 + 8a_2 + 27a_3 + 64a_4 + \ldots - 3(b_1 + 4b_2 + 9b_3 + 16b_4 + \ldots)$$

Observaţie

Cu coeficienţii din (43), $a_l = -a'_l$, $l = \overline{1, k-1}$, şi $a_k = a'_k = 1$ condiţiile (46) sunt:

$$d_0 = 1 - \sum_{l=0}^{k-1} a'_l = 0 \qquad (q = 0)$$

$$d_1 = k - \sum_{l=0}^{k-1} l a'_l - \sum_{l=0}^{k} b_l = 0 \qquad (q = 1)$$

$$d_q = k^q - \sum_{l=0}^{k} l^q a'_l - q \sum_{l=0}^{k} l^{q-1} b_l = 0, \qquad q = 2, \ldots, p$$

$$d_{p+1} = k^{p+1} - \sum_{l=1}^{k-1} l^{p+1} a'_l - (p+1) \sum_{l=0}^{k} l^p b_l \neq 0 \quad (q = p+1)$$

■

Cu cele de mai sus avem următoarea propoziţie:

Propozitie

Ordinul unei metode (45) este numărul natural p pentru care avem

$d_0 = 0, d_1 = 0, \ldots, d_p = 0$ şi $d_{p+1} \neq 0$.

Coeficienţii d_q sunt definiţi de (46) ■

Polinoamele generatoare ale metodei

Polinoamele obţinute prin înlocuirile $x_l, f_l \to r^l$ în cei doi membri din (45) se zic polinoamele generatoare ale metodei, şi anume:

$$\rho(r) = \sum_{l=0}^{k} \alpha_l r^l$$
$$\sigma(r) = \sum_{l=0}^{k} \beta_l r^l \qquad (47)$$

Observaţi că:

$d_0 = \rho(1)$; $\qquad d_1 = \rho'(1) - \sigma(1)$ ■

Consistenţă

O metodă pentru care coeficienţii verifică primele două relaţii (46) – adică, condiţiile pentru q = 0, 1:

$d_0 = 1, \quad d_1 = 0$,

se zic *consistentă*. Aceasta echivalează cu condiţia ca metoda să fie exactă pentru polinoame de gradul unu. Condiţia de consistenţă se poate exprima şi sub forma următoare, utilizând polinoamele generatoare:

$$\rho(1) = 0, \quad \rho'(1) = \sigma(1). \tag{48}$$

Exemple-2

1) Reluăm metoda mijlocului (Exemple-1), în care $k = 2$:

$$x_{i+1} - x_{i-1} = 2hf_i, \quad i \geq 1$$

Avem: $a_0 = -1, a_1 = 0, a_2 = 1; b_0 = 2$. Rezultă:

$$d_0 = a_0 + a_1 + a_2 = 0;\ d_1 = a_1 + 2a_2 - b_0 = 2 - 2 = 0$$

$$d_2 = a_1 + 4a_2 - 2(0 \cdot b_0) = 4 \neq 0$$

Astfel, ordinul metodei este $p = 1$.

2) Metoda trapezului ($k = 1$):

$$x_{i+1} - x_i = h(\tfrac{1}{2} f_{i+1} + \tfrac{1}{2} f_i), \quad i \geq 0$$

Avem: $a_0 = -1, a_1 = 1; b_0 = \tfrac{1}{2}, b_1 = \tfrac{1}{2}$, şi

$$d_0 = 0;\ d_1 = 1 - (\tfrac{1}{2} + \tfrac{1}{2}) = 0$$

$$d_2 = a_1 - 2(b_1) = 1 - 2(\tfrac{1}{2}) = 0$$

$$d_3 = a_1 - 3(b_1) = 1 - 3\tfrac{1}{2} \neq 0$$

Rezultă $p = 2$ ■

4.3 Construcţia operatorilor în mai mulţi paşi

Coeficienţii în (44), (45) se determină astfel ca formula să fie exactă pentru polinoamele de un grad dat. Dacă, din condiţiiile puse, rămân coeficienţi liberi (parametri), aceştia se vor determina astfel ca să avem îndeplinite una sau mai multe din condiţiile:

- Eroarea de trunchiere să fie cât mai mică;
- Propagarea erorilor să fie cât mai mică;
- Formula să fie cât mai simplă, de exemplu unii coeficienţi să fie zero.

În afară de aceste condiţii, vom cere ca metoda să fie stabilă, v. mai jos.

Determinarea coeficienţilor în (45) se face prin una din următoarele metode:

- metoda coeficienţilor nedeterminaţi
- prin integrare numerică
- prin derivare numerică

Acestea se expun în continuare.

4.3.1 Metoda coeficienţilor nedeterminaţi

Relaţiile (46) reprezintă un sistem de $p+1$ ecuaţii în cel mult $2k+1$ coeficienţi a_l, b_l (conform $a_k = 1$). Dacă numărul coeficienţilor este egal cu $p+1$ atunci, sistemul poate fi rezolvat în a_l, b_l. Dacă acest număr este mai mare decât $p+1$, unii coeficienţi rămân ca parametri.

Exemple-3

1) Să determinăm metodele 1-pas, de ordin doi ($k = 1$, şi $p = 2$):

$$x_{i+1} = a_0' x_i + h(b_1 f_{i+1} + b_0 f_i).$$

În forma (45), metoda se scrie:

$$x_{i+1} - a_0' x_i = h(b_1 f_{i+1} + b_0 f_i)$$

Cu $a_1 = 1$, ecuaţiile (46) sunt: $-a_0' + 1 = 0$, $1 - (b_0 + b_1) = 0$, $1 - 2b_1 = 0$. Din acestea rezultă $a_0' = 1$, $b_1 = b_0 = 1/2$, astfel că metoda căutată este metoda trapezului:

$$x_{i+1} = x_i + \frac{h}{2}(f_{i+1} + f_i).$$

2) Metode 2-pas, explicite, de ordin 1 ($k = 2$, $p = 1$; $b_1 = 0$)

$$x_{i+1} = a_1' x_i + a_0' x_{i-1} + h b_0 f_i.$$

Sau, în forma (45): $x_{i+1} - a_1' x_i - a_0' x_{i-1} = h b_0 f_i$. Condiţiile (46) sunt $1 - a_0' - a_1' = 0_1$, $1 - b_0 = 0$, astfel că metoda este

$$x_{i+1} = (1 - a_0') x_i + a_0' x_{i-1} + h f_i,$$

în care $a_0' \neq 0$ rămâne un parametru ■

4.3.2 Metode bazate pe integrare numerică (metodele Adams şi Milne)

Integrând ecuaţia (41) pe intervalul $[t_i, t_{i+1}]$, avem:

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} f(t, x(t))\, dt$$

Cu *k* aproximaţii cunoscute $x_i, x_{i-1}, \ldots, x_{i-k+1}$, pe nodurile $t_i, t_{i-1}, \ldots, t_{i-k+1}$, găsim aproximaţiile funcţiei *f* pe aceste noduri:

$$f_j = f(t_j, x_j), \quad j = \overline{i, i-k+1}.$$

Metode Adams explicite

În membrul doi, înlocuim funcţia necunoscută $f(t, x(t))$ cu polinomul de interpolare Newton pe nodurile $t_i, t_{i-1}, \ldots, t_{i-k+1}$. (*k* noduri, polinom de grad $k-1$).
Se obţine metodele Adams explicite, referite şi ca metode *Adams-Bashforth*:

$$x_{i+1} = x_i + h\sum_{l=0}^{k-1} c_l f_{i-l} = x_i + h(c_0 f_i + c_1 f_{i-1} + c_2 f_{i-2} + c_3 f_{i-3} + \ldots) \qquad (49)$$

Coeficienţii c_l din (49) sunt daţi în tabelul următor, pentru *k* = 1, 2, 3, 4.

Coeficienţii pentru c_l pentru metodele Adams explicite – ecuaţia (49)

k	f_i	f_{i-1}	f_{i-2}	f_{i-3}	f_{i-4}
1	1				
2	3/2	-1/2			
3	23/12	-16/12	5/12		
4	55/24	-59/24	37/24	-9/24	
5	1901/720	-2774/720	2616/720	-1274/720	251/720

De exemplu, metoda pentru $k = 4$ este:

$x_{i+1} = x_i + \frac{h}{24}(55 f_i - 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3})$. Formulele (49) sunt de tipul (44),

cu $a_0 = 1, a_l = 0, l \geq 1$, şi $b_{-1} = 0$, $b_l = c_l$.

Ordin:

Metodele Adams explicite au ordinul $p = k$.

Metode Adams implicite

Analog, cu aproximaţiile $f_j = f(t_j, x_j)$ pe nodurile $t_{i+1}, t_i, \ldots, t_{i-k+1}$, utilizăm polinomul de interpolare pentru f.($k+1$ noduri, polinom de grad k).

Se obţin metodele Adams implicite, referite şi ca metode *Adams-Moulton*:

$$x_{i+1} = x_i + h\sum_{l=0}^{k} c_l f_{i+1-l} = x_i + h(c_0 f_{i+1} + c_1 f_i + c_2 f_{i-1} + c_3 f_{i-2} + \ldots) \qquad (50)$$

Pentru $k = 0, 1, 2, 3, 4$, coeficienţii din (50) se dau în tabelul următor:

Coeficienţii c_l pentru metodele Adams implicite – ecuaţia (50)

k	f_{i+1}	f_i	f_{i-1}	f_{i-2}	f_{i-3}
0	1				
1	1/2	1/2			
2	5/12	8/12	-1/12		
3	9/24	19/24	-5/24	1/24	
4	251/720	646/720	-264/720	106/720	-19/720

Cazul special $k = 0$ produce metoda Euler implicită: $x_{i+1} = x_i + hf_{i+1}$. Formulele obţinute sunt de tipul (44) – cu $a_0 = 1, a_l = 0, l \geq 1$, şi $b_l = c_{l+1}$. Metodele implicite au o precizie mai mare decât cele explicite. Determinarea lui x_{i+1} din formulele de mai sus, se face cu o metodă pentru ecuaţii neliniare (de exemplu, pentru h = mic, metoda punctului fix).

Ordin:

Ordinul metodei este $p = k + 1$.

Metode Milne-Simpson (implicite)

Ecuaţia (41) se integrează pe intervalul $[t_{i-1}, t_{i+1}]$, obţinând

$$x(t_{i+1}) = x(t_{i-1}) + \int_{t_{i-1}}^{t_{i+1}} f(t, x(t))\, dt$$

Sub integrală, înlocuim *f* cu polinomul p_k utilizat în metodele Adams implicite (polinomul de interpolare pe nodurile $t_{i+1}, \ldots, t_{i-k+1}$).

Se obţin metodele:

$$x_{i+1} = x_{i-1} + h \sum_{l=0}^{k} c_l f_{i+1-l} \qquad (51)$$

Coeficienţii c_l în (51), pentru $k = 0, 1, 2, 3, 4$, sunt daţi mai jos.

Coeficienţii c_l pentru metodele Milne-Simpson – ecuaţia (51)

K	f_{i+1}	f_i	f_{i-1}	f_{i-2}	f_{i-3}
0	2				
1	0	2			
2, 3	1/3	4/3	1/3		
4	29/90	124/90	24/90	4/90	-1/90

Metoda pentru $k = 2$ este numită metoda Milne, şi este dată de

$$x_{i+1} = x_{i-1} + \frac{h}{3}(f_{i+1} + 4 f_i + f_{i-1}).$$

Ordin:

Metodele Milne-Simpson au ordinul $p = k + 1$ ■

4.3.3 Metode bazate pe derivare numerică (BDF)

În metodele anterioare s-a integrat ecuaţia (41) şi s-a utilizat polinomul de interpolare pentru funcţia $f(t, x(t))$. Considerăm acum ecuaţia (41), şi polinomul de interpolare pentru funcţia $x(t)$, pe nodurile $t_{i+1}, t_i, \ldots, t_{i-k+1}$ şi valorile $x_{i+1}, \ldots, x_{i-k+1}$ (polinom de grad k):

Avem:

$$\sum_{j=1}^{k} \frac{1}{j} \nabla^j x_{i+1} = h f_{i+1} \qquad (54)$$

Formulele (54) se numesc "formule de derivare înapoi" (*backward differentiation formulae*) şi metodele cu aceste formule se zic *metode BDF*. Ele se utilizează în integrarea numerică a ecuaţiilor diferenţiale rigide – v. 5.

Formulele (54) explicitate în termenii x_{i+1-l} sunt:

$$\sum_{l=0}^{k} c_l x_{i+1-l} = h f_{i+1} \qquad (54')$$

Pentru $k = \overline{1,6}$, coeficienţii c_l sunt daţi mai jos. Pentru $k > 6$, metodele BDF sunt instabile (Hairer et al. (1987)).

Coeficienţii c_l pentru metodele BDF – ecuaţia (54')

k	x_{i+1}	x_i	x_{i-1}	x_{i-2}	x_{i-3}	x_{i-4}	x_{i-5}
1	1	-1					
2	3/2	-2	1/2				
3	11/6	-3	3/2	-1/3			
4	25/12	-4	3	-4/3	1/4		
5	137/60	-5	5	-10/3	5/4	-1/5	
6	147/60	-6	15/2	-20/3	15/4	-6/5	1/6

Ordin: Metodele BDF au ordinul $p = k$ ■

4.4 Stabilitatea metodelor în mai mulţi paşi

Stabilitatea metodei analizează comportarea soluţiei pentru $n \to \infty$ şi $h \to 0$, cu condiţia *nh* = constant ($nh = TT - t_0$ = lungimea intervalului de integrare). Pentru $h \to 0$, din (43') se obţine:

$$x_{i+1} - a_0 x_i - a_1 x_{i-1} - \ldots - a_{k-1} x_{i-k+1} = 0 \qquad (55)$$

"x_{i+1}" din această ecuaţie, poate fi interpretat ca soluţia dată de metodă, pentru ecuaţia diferenţială $x' = 0$.

Pentru a rezolva ecuaţia liniară (şi omogenă) cu diferenţe (55), căutăm soluţii de forma $x_j = r^j$. Înlocuind, obţinem

$$r^{i+1} - a_0 r^i - a_1 r^{i-1} - \ldots a_{k-1} r^{i-k+1} = 0,$$

şi împărţind cu r^{i-k+1} rezultă

$$\rho(r) = r^k - a_0 r^{k-1} - a_1 r^{k-2} - \ldots a_{k-1} = 0. \qquad (56)$$

Remarcăm că $\rho(r)$ este primul polinom generator definit în (47). Fie $r_\nu, \nu = 1, k$ rădăcinile polinomului $\rho(r)$. Dacă rădăcinile sunt simple, un sistem de soluţii fundamentale este $\{r_1^{\,j}, \ldots, r_k^{\,j}\}$, şi soluţia generală este o combinaţie liniară de acestea:

$$x_j = \sum_{\nu=1}^{k} c_{j\nu} r_\nu^j, \quad j = i+1, \ldots, i-k+1.$$

Dacă există rădăcini multiple r_ν, cu ordinul de multiplicitate $m_\nu \geq 1$, un sistem de soluţii fundamentale corespunzând rădăcinii r_ν sunt $\{r_\nu, jr_\nu, \ldots, j^{m_\nu - 1} r_\nu\}$. Soluţia generală are forma

$$x_j = \sum_{\nu=1}^{k} p_{j\nu}(j) r_\nu^j,$$

în care $p_{j\nu}(j)$ sunt polinoame de grad $m_\nu - 1$ în *j*. În ambele cazuri, pentru ca soluţia să rămână mărginită pentru $n \to \infty$, trebuie ca rădăcinile ecuaţiei (56) să se situeze în discul unitate ($|r| \leq 1$), iar rădăcinile de modul 1 să fie simple.

Remarcăm că, pentru metode consistente polinomul $\rho(r)$ are întotdeauna o rădăcină $r = 1$, conform (48).

Definiţie

Metoda multi-pas (43) se zice *stabilă*, dacă polinomul generator $\rho(r)$ satisface *condiţia de rădăcini*, şi anume:

a) Rădăcinile sunt situate în discul unitate: $|r_v| \le 1$;

b) Rădăcinile de modul 1 sunt simple: dacă $|r_v| = 1$, atunci $\rho'(r_v) \neq 0$ ■

Exemple

1) Stabilitatea metodelor Adams (§ 4.2.2):

Polinomul generator pentru metodele Adams (explicite şi implicite) este $\rho(r) = r^k - r^{k-1}$. Radăcinile sunt $r = 1$ – simplă, şi $r = 0$ – multiplă de ordinul (k-1). Polinomul satisface condiţia de rădăcini şi, în consecinţă, metodele Adams sunt stabile.

2) Stabilitatea metodelor Milne-Simpson (§ 4.2.2):

Polinomul generator $\rho(r) = r^k - r^{k-2}$ satisface condiţia de rădăcini, şi deci, metodele sunt stabile. Existenţa rădăcinii $r = -1$ duce însă la fenomenul de "instabilitate slabă" – v. Hairer & Wanner (1991).

3) Stabilitatea metodelor BDF (§ 4.2.3):

Se arată că metodele BDF sunt stabile pentru $k \le 6$, şi instabile pentru $k \ge 7$.

■

Ordinul maxim al unei metode stabile

Ordinul maxim pentru care metoda este stabilă este dat de următorul rezultat datorat lui Dalquist, şi numit "prima barieră Dalquist".

Propoziţie

Ordinul p al unei metode liniare în k-paşi stabilă, satisface:

a) $p \le k+2$ … pentru k = par

b) $p \le k+1$ … pentru k = impar

c) $p \le k$ … pentru $b_{-1} \le 0$ (în particular, pentru o metodă explicită)

■

4.5 Convergenţa metodelor în mai mulţi paşi

Considerăm din nou, problema cu valori iniţiale (41), în care presupunem că $f(t,x)$ satisface condiţiile din secţiunea 6-I, 1, şi în consecinţă problema are soluţie unică. Considerăm integrarea numerică pe intervalul $[t_0, TT]$, inclus în intervalul de existenţă a soluţiei. Reamintim că notăm prin $x(t_j)$ şi x_j, respectiv soluţia exactă şi calculată pe punctul $t_j = t_0 + jh$. Pentru ceea ce urmeaza vom presupune că vrem să calculăm soluţia pe punctul *fixat* $t \in [t_0, TT]$, cu paşi *h* din ce în ce mai mici. Punem atunci: $t - t_0 = nh = constant$, şi avem $n = (t - t_0)/h$, astfel că $h \to 0 \Leftrightarrow n \to \infty$. Notăm cu $x_t(h)$, soluţia calculată pe punctul fixat *t*, cu pasul *h*. Convergenţa va cere ca, sub anumite ipoteze, să avem limita:

$$\lim_{\substack{h\to 0 \\ t=\text{fixat}}} x_t(h) = x(t)$$

Practic, pentru a calcula $x_t(h)$, utilizăm metoda (43) în care punem $t = t_{i+1}$, $n = i+1$, şi $(i+1)h = t_{i+1} - t_0 =$ constant. În acest caz vom scrie $x_{i+1}(h)$ în loc de $x_{t_{i+1}}(h)$.

Definiţie

1) Metoda liniară multi-pas (43) se zice convergentă dacă, pentru orice problemă cu valori iniţiale (41), următoarea condiţie este îndeplinită:

Dacă valorile de start $x_j(h)$ (pe punctele t_j, $0 \le j \le k-1$), satisfac

$x(t_j) - x_j(h) \to 0$ … pentru $h \to 0$, $j = \overline{0, k-1}$

atunci avem şi

$\forall t \in [t_0, TT]$, t = fixat, $x(t) - x_t(h) \to 0$ … pentru $h \to 0$.

2) Metoda se zice convergentă de ordinul *p* dacă, pentru orice problemă (41) cu *f* suficient derivabilă, există constantele pozitive h_0, C_0, C, astfel ca să avem:

Dacă valorile de start satisfac

$\| x(t_j) - x_j(h) \| \le C_0 h^p$ … pentru $h \le h_0$, $j = \overline{0, k-1}$

atunci

$\forall t \in [t_0, TT]$, t = fixat, $\| x(t) - x_t(h) \| \le Ch^p$ … pentru $h \le h_0$.

■

Rezultatul principal din următoarele teoreme este că proprietăţile de consistenţă + stabilitate ale unei metode, sunt condiţii necesare şi suficiente pentru convergenţa acesteia: *Convergenţă* ⇔ *Consistenţă* + *Stabilitate*.

Teorema 1

Dacă metoda (43) este convergentă, atunci ea este stabilă şi consistentă ■

Teorema 2

1) Dacă metoda (43) este stabilă şi de ordinul $p = 1$ (adică, consistentă), atunci ea este convergentă.
2) Dacă metoda (43) este stabilă şi de ordinul p, atunci ea este convergentă de ordinul p ■

4.6 Stabilitate relativă şi stabilitate slabă

Considerăm ecuaţia de test

$x' = \lambda x,\ x(0) = 1$

care are soluţia $x(t) = e^{\lambda t}$.

Definiţie

Fie o metodă (43) consistentă, şi r_0 rădăcina principală a polinomului caracteristic (57). Metoda se zice:

- *Relativ stabilă* pe intervalul $[\alpha, \beta] \ni 0$, dacă pentru orice $h\lambda$ în acest interval, rădăcinile polinomului caracteristic satisfac condiţiile:

$| r_\nu(h\lambda) | \le | r_0(h\lambda) |, \quad \nu = \overline{1, k-1}$ (61a)

şi, dacă $| r_\nu | = | r_0 |$, atunci r_ν este rădăcină simplă. (61b)

- *Absolut stabilă* pe intervalul $[\alpha, \beta]$ dacă, pentru orice $h\lambda$ în acest interval:

$$|r_\nu(h\lambda)|<1, \quad \nu=\overline{0,k-1} \tag{62}$$

- Metoda satisface condiţia *tare* de rădăcini, dacă:

$$|r_\nu(0)|<1, \quad \nu=\overline{1,k-1} \tag{63}$$

O metodă stabilă dar care nu este relativ stabilă, se zice *slab stabilă*.

■

Observaţii

- Stabilitatea absolută poate avea loc numai în cazul $\lambda<0$ (mai general, λ are partea reală negativă) – v. relaţia (60). În acest caz, stabilitatea absolută echivalează cu condiţia ca $t_j\to\infty\Rightarrow x_j\to 0$.

- Condiţia tare de rădăcini implică stabilitatea relativă: cu $r_0(0)=1$, condiţia (63) se scrie $|r_\nu(0)|<r_0(0)$, iar din continuitatea rădăcinilor ca funcţii de $h\lambda$, rezultă că aceasta are loc pe o vecinătate a lui 0. Reciproca nu este, în general adevărată.

- Întrucât definiţiile anterioare se aplică şi la sisteme, λ se consideră, în general, complex. Mulţimea valorilor $h\lambda$ pentru care au loc (60), respectiv (61), se zic *regiunea de stabilitate* relativă, respectiv absolută, ale metodei. Determinarea regiunii de stabilitate absolută este mai simplă decât cea de stabilitate relativă, şi ea se poate face pe criterii algebrice (Hurwitz-Routh, Schur – v. Ralston & Rabinowitz (1978)). Regiunile de stabilitate absolută (în planul complex), pentru metodele Adams-Bashforth şi Adams-Moulton sunt reproduse în Atkinson (1978). Din aceste diagrame rezultă că, cu cât ordinul este mai mare, regiunea de stabilitate este mai mică – v. şi Exemple-3. Totuşi, chiar pentru ordine mari, $|h\lambda|$ rămâne relativ mare, şi nu introduce restricţii majore asupra lui h, cu excepţia cazului în care $|\lambda|$ este "mare" ■

Exemple

1. Metoda mijlocului este dată de $x_{i+1}=x_{i-1}+2hf(t_i,x_i), \quad i\geq 1$, şi cu $f(t,x)=\lambda x$ devine $x_{i+1}=x_{i-1}+2h\lambda x_i$. Polinomul caracteristic este: $r^2-2(h\lambda)r-1=0$. Punem $\lambda=-K$, unde $K>0$, şi avem

$r_{0,1}(hK) = -hK \pm \sqrt{(hK)^2 + 1}$. Metoda este stabilă (| r | < 1) pentru $hK < 1$, dar avem $|r_1| > |r_0|$, deci metoda nu este relativ stabilă.

2. Metodele Adams: Verificăm condiţia (62). Pentru $\lambda = 0$, polinomul caracteristic este $\rho(r) = r^k - r^{k-1}$, cu rădăcinile $r_0 = 1$ şi $r_\nu = 0, \nu = \overline{1, k-1}$. Condiţia tare (63) este satisfăcută şi deci, metoda este relativ stabilă.

3. Să determinăm intervalul (presupunem λ real, şi negativ) de stabilitate absolută pentru metodele Adams-Moulton $k = 2$, $k = 3$ (de ordinele 3, 4) – v. Tabelul din 4.2.2. f_j se înlocuieşte cu λx_j. Pentru metoda $k = 2$ avem: $x_{i+1} = x_i + h\lambda(5x_{i+1} + 8x_i - x_{i-1})/12$. Punem $h\lambda = z$ ($z < 0$) şi $x_{i-2} = 1, x_{i-1} = r, \ldots$, rezultă $p(z) = (12 - 5z)r^2 + (12 + 8z)r + z$. Condiţiile pentru | r | < 1 sunt: $\Delta > 0$; $p(-1) > 0$; $p(1) > 0$; $-1 < (6+4z)/(12-5z) < 1$, care conduc la $-6 < z < 0$.

   Pentru metoda $k = 3$ avem $x_{i+1} = x_i + h\lambda(9x_{i+1} + 19x_i - 5x_{i-1} + x_{i-2})/24$. Cu notaţiile anterioare avem $p(z) = (24 - 9z)r^3 - (24 + 19z)r^2 + 5zr - z$. Condiţii *necesare* pentru | r | < 1 sunt (avem $z < 0$): $p(-1) < 0$; $p(1) > 0$, care conduc la $-3 < z < 0$. Se arată că acesta este rezultatul final. (Exerciţiu: verificaţi că $p''(r) > 0$, $\forall z < 0$.) ■

4.6 Eroarea de trunchiere

Considerăm metoda în k-paşi (44), în care punem acum $x'(t) = f(t, x(t))$

$$x_{i+1} = \sum_{l=0}^{k-1} a_l x_{i-l} + h \sum_{l=-1}^{k-1} b_l x'_{i-l}, \quad i \geq k-1 \tag{64}$$

În (64), x_j şi x'_j sunt aproximaţii pentru $x(t_j)$ şi $x'(t_j)$. Înlocuind x_j, x'_j cu valorile exacte, formula va avea o eroare pe care o notăm T_{i+1} şi care reprezintă eroarea de trunchiere locală pe pasul i+1:

$$x(t_{i+1}) = \sum_{l=0}^{k-1} a_l x(t_{i-l}) + h \sum_{l=-1}^{k-1} b_l x'(t_{i-l}) + T_{i+1} \tag{65}$$

Presupunem că avem $T_{i+1} = 0$, pentru cazul când x este un polinom de grad $\leq p$ (ordinul metodei este p).

Se arată că eroarea T_{i+1} are forma

$$T_{i+1} = d_p h^{p+1} x^{(p+1)}(\xi), \quad \xi \in (t_{i-k+1}, t_{i+1}) \tag{69}$$

Expresia coeficientului d_p se dă mai jos.

$$(p+1)!\, d_p = k^{p+1} - \sum_{l=0}^{k-2} a_l (k-1-l)^{p+1} - (p+1) \sum_{l=-1}^{k-2} b_l (k-1-l)^p \tag{74}$$

Explicit:

$$\begin{aligned}(p+1)!\, d_p = {} & k^{p+1} - a_0 (k-1)^{p+1} - a_1 (k-2)^{p+1} - \ldots - a_{k-2} \cdot 1 \\ & - (p+1)[b_{-1} k^p + b_0 (k-1)^p + b_1 (k-2)^p + \ldots + b_{k-2} \cdot 1]\end{aligned} \tag{74'}$$

■

Exemple

1. Metode Adams-Bashforth ($p = k$):

Nr. paşi k	Ordin p	Coeficient d_p
1	1	1/2
2	2	5/12
3	3	3/8
4	4	251/720

2. Metode Adams-Moulton ($p = k+1$):

Nr. paşi k	Ordin p	Coeficient d_p
1	2	-1/12
2	3	-1/24
3	4	-19/720
4	5	-3/160

4.7 Metode predictor-corector

4.7.1 Predictori şi corectori

Eroarea unei metode de ordin *p* este dată de (69). Pentru o ecuaţie dată, la acelaşi pas şi acelaşi ordin, mărimea erorii este dată de $|d_p|$, unde d_p este dat de (74). În general, $|d_p|$ este mai mic pentru o metodă implicită decât pentru una explicită. Un exemplu îl constituie metodele Adams – compară valorile din exemplele 1 şi 2 de mai sus. Astfel, este preferabil a se determina soluţia cu o metodă implicită. Aceasta are forma generală $x_{i+1} = g(x_{i+1}, \ldots, x_{i-k+1})$ şi determină soluţia pe un pas cu o metodă iterativă pentru rezolvarea ecuaţiei în x_{i+1}. Astfel, se cere o estimare a aproximaţiei iniţiale (la fiecare pas). Cea mai bună cale este de a calcula această aproximaţie cu o metodă explicită – care va fi numită *predictor*. Apoi se va "corecta" valoarea prin iteraţie în metoda implicită – aceasta va fi numită *corector*. Metoda obţinută prin cuplarea unui predictor şi a unui corector, se va numi o metodă *predictor-corector*. Criterii pentru alegerea predictorului şi corectorului se vor discuta în 4.7.4.

4.7.2 Convergenţa iteraţiei de punct fix

Explicitând în membrul doi termenul în x_{i+1}, metoda (64) se scrie:

$$x_{i+1} = \sum_{l=0}^{k-1} (a_l x_{i-l} + h b_l x'_{i-l}) + h b_{-1} f(t_{i+1}, x_{i+1}) \equiv g(x_{i+1}) \qquad (75)$$

Să rezolvăm (75) prin metoda punctului fix. Fie la pasul $i+1$ (fixat) estimarea $x_{i+1}^{(0)}$ a lui x_{i+1}, cu aceasta calculăm $f(t_{i+1}, x_{i+1}^{(0)})$ şi înlocuim în (74) obţinând $x_{i+1}^{(1)}$, etc. În general, la pasul *j* al iteraţiei avem:

$$x_{i+1}^{(j+1)} = \sum_{l=0}^{k-1} (a_l x_{i-l} + h b_l x'_{i-l}) + h b_{-1} f(t_{i+1}, x_{i+1}^{(j)}), \quad j \geq 0 \qquad (76)$$

şi remarcăm că de la pasul *j* la pasul *j*+1, suma din membrul doi nu se modifică. Condiţia de convergenţă în (76) este $|g'(x)| \leq \lambda < 1$, pe o vecinătate a lui $x_{i+1}^{(0)}$. Avem

$$g'(x) = hb_{-1}\frac{\partial f(t,x)}{\partial x},$$

Presupunând că derivata $\partial f / \partial x$ este mărginită într-o vecinătate I a lui $(t_{i+1}, x_{i+1}^{(0)})$ care conţine punctele $(t_{i+1}, x_{i+1}^{(j)})$:

$$\left|\frac{\partial f(t,x}{\partial x}\right| \le \lambda, \quad (t,x) \in I \tag{77}$$

rezultă că trebuie să avem:

$$hb_{-1}\lambda < 1 \tag{78}$$

Mai mult, rata convergenţei este $hb_{-1}\lambda$, astfel că pentru o iteraţie rapidă vom cere să avem $hb_{-1}\lambda << 1$. În (78) s-a presupus $b_{-1} > 0$, ceea ce are loc pentru toate metodele considerate anterior. Pentru convergenţa pe orice pas (i+1) corespunzând lui $t_{i+1} \in [t_0, TT]$, în (77) se va lua $I = [t_0, TT]$.

Observaţie

Pentru ecuaţii diferenţiale *rigide* (v. 5), unde λ este "mare", nu putem avea (77) decât pentru un pas h excesiv de mic. În acest caz se va utiliza, în loc de iteraţia de punct fix, metoda Newton

4.7.3 Estimarea de tip Milne a erorii. Modificarea pasului

Diferenţa între valoarea corectată $x_{i+1}^{(c)}$ şi valoarea prezisă $x_{i+1}^{(0)}$, constituie o estimare a erorii pe pasul i+1, numită *estimare de tip Milne*:

$$\varepsilon_{i+1} = x_{i+1}^{(c)} - x_{i+1}^{(0)}$$

Dacă în raport cu o toleranţă *tol* impusă, avem:

- $|\varepsilon_{i+1}| \le tol$: $x_{i+1}^{(c)}$ este acceptată (ca aproximaţie pentru x_{i+1}) şi calculul continuă. Dacă $|\varepsilon_{i+1}| << tol$, pasul h poate fi mărit – pentru calculul valorii următoare x_{i+2}.
- $|\varepsilon_{i+1}| > tol$: $x_{i+1}^{(c)}$ nu este acceptată şi se recalculează cu un pas h mai mic;

Utilizarea estimării de mai sus este însă supusă condiţiei ca predictorul şi corectorul să aibă *acelaşi ordin*.

4.7.4 Eroarea de trunchiere a metodei predictor-corector

Fie predictorul şi corectorul definiţi de (44), explicită şi respectiv implicită:

$$x_{i+1}^{(0)} = \sum_{l=0}^{k-1} \alpha_l x_{i-l} + h \sum_{l=0}^{k-1} \beta_l f_{i-l} \qquad \text{- predictor} \tag{79a}$$

$$x_{i+1} = \sum_{l=0}^{k-1} a_l x_{i-l} + h \sum_{l=0}^{k-1} b_l f_{i-l} + h b_{-1} f(t_{i+1}, x_{i+1}^{(0)}) \qquad \text{- corector} \tag{79b}$$

şi presupunem că facem *o singură iteraţie* în corector adică, aplicăm (79b) cu valoarea $x_{i+1}^{(0)}$ furnizată de (79a), obţinând $x_{i+1} = x_{i+1}^{(1)}$. Eroarea de trunchiere a metodei (79a, b) se obţine cum urmează.

Eroarea de trunchiere a metodei predictor-corector este de ordinul erorii de trunchiere a corectorului. Pentru aceasta trebuie să avem:

$$p^P \geq p^C - 1$$

unde p^P şi p^C sunt ordinele predictorului, respectiv corectorului.

4.7.5 Alegerea predictorului şi corectorului. Exemple

Alegerea predictorului este legată de o cât mai bună estimare a lui $x_{i+1}^{(0)}$. Între predictori de acelaşi ordin, criteriul este coeficientul d_p al erorii (cât mai mic).

Alegerea predictorului este mai puţin critică decât cea a corectorului. Alegerea corectorului este dictată în principal de proprietăţile lui de stabilitate. Între corectori de acelaşi ordin, criteriile de alegere sunt: coeficientul erorii (cât mai mic) şi regiunea de stabilitate absolută (cât mai mare). Dacă predictorul este suficient de exact – v. 4.7.4 , ordinul metodei predictor-corector este ordinul corectorului (dacă ar fi utilizat singur).

Vom da câteva exemple de metode predictor-corector, dintre cele mai utilizate. În predictor, notăm:

$$f_{i+1}^{(j)} = f(t_{i+1}, x_{i+1}^{(j)})$$

Primul exemplu este cel al unei metode de cel mai mic ordin (corector 1-pas, $p = 2$).

0. Metodă de ordin 2:

Predictor (Euler, ordin 1):

$$x_{i+1}^{(0)} = x_i + hf_i$$

Corector (Metoda trapezului, ordin 2):

$$x_{i+1}^{(j+1)} = x_i + \frac{h}{2}(f_i + f_{i+1}^{(j)})$$

1. Metodele Milne şi Hamming

Predictor (ordin 4):

$$x_{i+1}^{(0)} = x_{i-3} + \frac{4h}{3}(f_i - f_{i-1} + 2f_{i-2})$$

Corector (Milne, ordin 4):

$$x_{i+1}^{(j+1)} = x_{i-1} + \frac{h}{3}(f_{i+1}^{(j)} + 4f_i + f_{i-1}), \qquad j \geq 0$$

Coeficienţii erorii sunt: 14/45 (predictor) şi -1/90 (corector).

Corectorul nu este însă relativ stabil, pentru valori $\lambda > 0$ (ecuaţia de test este $x' = \lambda x$ – v. § 4.4).

O modificare a corectorului conduce la metoda Hamming care este stabilă pentru $h\lambda \leq 0.69$. Corectorul devine:

$$\tilde{x}_{i+1}^{(0)} = x_{i+1}^{(0)} + \tfrac{112}{121}(x_i - x_i^{(0)}) \qquad \text{- modificare}$$

$$x_{i+1}^{(j+1)} = \frac{1}{8}(9x_i - x_{i-2}) + \frac{3h}{8}(\tilde{f}_{i+1}^{(j)} + 2f_i - f_{i-1}) \qquad \text{- corector}$$

unde $\tilde{f}$ este calculat în valoarea modificată $\tilde{x}$.

2. Metode Adams – ordin 4:

Predictor (Adams-Bashforth, ordin 4):

$$x_{i+1}^{(0)} = x_i + \frac{h}{25}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})$$

Corector (Adams-Moulton, ordin 4):

$$x_{i+1}^{(j+1)} = x_i + \frac{h}{24}(9f_{i+1}^{(j)} + 19f_i - 5f_{i-1} + f_{i-2})$$

3. Metode Adams – ordin 5:

Predictor (Adams-Bashforth, ordin 5):

$$x_{i+1}^{(0)} = x_i + \frac{h}{720}(1901 f_i - 2774 f_{i-1} + 2616 f_{i-2} - 1274 f_{i-3} + 251 f_{i-4})$$

Corector (Adams-Moulton, ordin 5):

$$x_{i+1}^{(j+1)} = x_i + \frac{h}{720}(251 f_{i+1}^{(j)} + 646 f_i - 264 f_{i-1} + 106 f_{i-2} - 19 f_{i-3})$$

4.7.6 Implementarea metodelor predictor-corector

Implementarea clasică urmează, în principal, două scheme:

a) PECE: Predicţie – Evaluare – Corectare (1 iteraţie) – Evaluare

Aceasta înseamnă, pe un pas $i+1$:

1. $x_{i+1}^{(0)}$ este estimat de predictor; (P)

2. Se calculează $f(t_{i+1}, x_{i+1}^{(0)})$; (E)

3. Se face *o singură* iteratie în corector, rezultă $x_{i+1}^{(1)}$; (C)

4. Se calculează $f_{i+1} = f(t_{i+1}, x_{i+1})$ (dacă pasul este acceptat). (E)

Schema conduce la 2 evaluări de funcţii pe un pas – sub-paşii 2, 4. Dacă predictorul este de ordin ≤ (ordinul corectorului – 1), eroarea de trunchiere a schemei este de ordinul erorii corectorului.

b) P(EC)^M:

1) $x_{i+1}^{(0)}$ este estimat de predictor; (P)

2.0) Testare $| x_{i+1}^{(j+1)} - x_{i+1}^{(j)} | < tol$. Dacă este satisfăcută, se trece la pasul *i*+2. Dacă nu, se face:

2.1) Evaluare: $f_{i+1}^{(j)} = f(t_{i+1}, x_{i+1}^{(j)})$; (E)

2.2) Iterare în corector: rezultă $x_{i+1}^{(j+1)}$, şi se revine la pasul 2.0. (C)

M este numărul de iteraţii pe pasul $i+1$, şi poate varia de la un pas la altul. Schema (b) conduce la *M* evaluări de funcţii pe un pas, dar numărul total de

evaluări de funcţii (pentru întregul interval de integrare) poate fi mai mic decât în schema (a), şi în acest caz, schema (b) este mai eficientă.

Observaţie

O altă schemă (c), constă în a impune un număr fixat de iteraţii *M* în schema (b). În acest caz, la pasul 2.0 se testează dacă *numărul de iteraţii* = *M*. În cazul satisfacerii testului 2.0, pasul este urmat de evaluarea 2.1. În acest caz, schema se simbolizează prin $\mathbf{P(EC)}^M\mathbf{E}$. Cazul $M = 1$ revine la schema (a) ■

Stabilitate:

În schema (a) – PECE, utilizarea unei singure iteraţii modifică caracteristicile de stabilitate ale metodei în raport cu cele ale schemei (b), şi stabilitatea este influenţată de predictorul utilizat. Concret, schema (a) micşorează regiunea de stabilitate pe care o are corectorul utilizat singur. În schema (b) – $\mathrm{P(EC)}^M$, iterarea până la convergenţă nu modifică stabilitatea metodei şi aceasta nu este influenţată de predictorul utilizat. V. o discuţie mai amplă în Ralston & Rabinowitz (1978), şi reprezentări ale regiunii de stabilitate pentru schema (a) în Hairer et al. (1991).

Observaţie

Codurile care implementează aceste scheme utilizează procedeul *pas variabil – ordin variabil* (abreviat *VSVO*), adică în funcţie de un test pe un pas $i+1$ acceptat, la pasul următor se poate varia atât ordinul metodei cât şi mărimea pasului ■

4.7.7 Determinarea valorilor de start

La primul pas ($i = 0$), metoda în *k*-paşi cere *k* valori de start $x_0, x_{-1}, \ldots, x_{=k+1}$.

$x_0 = x^{(0)}$ este dat de condiţiile iniţiale, dar pentru celelalte valori trebuie utilizată o procedură de determinare. Aceasta poate fi:

a) Seria Taylor

b) Metode Runge-Kutta

c) Metode multi-pas de ordin mai mic

(a) Se utilizează desvoltarea lui $x(t)$ în serie Taylor, în jurul lui t_0:

$$x(t_0 + jh) = x_0 + jhx_0' + \frac{(jh)^2}{2!}x_0'' + \ldots \qquad j = -1, \ldots, -k+1$$

Derivata x_0' se calculează, cu condiţiile iniţiale, din ecuaţia dată $x' = f(t, x)$, iar derivatele $x_0^{(n)} = x^{(n)}(t_0)$ – prin derivarea ecuaţiei. Desvoltatea în serie se face până la termenul de ordinul *p* inclusiv, unde *p* este ordinul metodei.

(b) Metodele Runge-Kutta sunt auto-start şi pot astfel furniza valorile de start. Se va utiliza o metodă al cărei ordin este egal cu ordinul metodei multi-pas.

(c) Procedurile a, b, au fost utilizate înainte de apariţia implementării metodelor *VSVO* (*pas variabil - ordin variabil*). În acestea din urmă, se începe integrarea cu o metodă 1-pas (v. Exemplul 0 în 4.7.5) care calculează x_1, şi se continuă cu metode în 2, 3, …, (*k*-1) paşi. Cu cele *k* valori calculate, se continuă cu metoda în *k*-paşi. Astfel, aceste metode devin metode auto-start.

4.8 Comparaţia metodelor predictor-corector (PC) cu metodele Runge Kutta (RK)

Comparaţia se face luând în considerare următoarele criterii:

a) Necesitatea valorilor de start;

b) Precizie;

c) Număr de evaluări de funcţii / pas;

d) Numărul de evaluări suplimentare de funcţii, necesare pentru controlul erorii locale de trunchiere.

(a) Metodele R-K sunt auto-start, pe când metodele PC cer o procedură pentru valorile de start. Totuşi, în implementarea predictor-corector, ordin variabil-pas variabil, metodele PC devin auto-start.

(b) La ordine egale, precizia metodelor RK este uşor superioară. Compară rezultatele din 4.9 (v. mai jos) cu cele din 3.3.9.

(c) Pentru metodele RK, numărul de evaluări/pas este ≥ ordinul metodei. Pentru metodele PC, în implementarea PECE acest număr este 2, dar în implementarea P(EC)^M, respectiv P(EC)^{M}E, acesta depinde de numărul *M* de iteraţii (fiind egal cu *M*, respectiv *M*+1).

(d) Controlul erorii locale de trunchiere: cere evaluări suplimentare în metodele RK, în timp ce la metodele PC nu cere evaluări suplimentare.

Codurile actuale se orientează mai mult spre metodele RK, sau metode de tip similar pentru ecuaţii diferenţiale de ordinul doi (metodele Nyström) – cu excepţia ecuaţiilor diferenţiale rigide, pentru care se utilizează metode BDF şi metode RK implicite.

4.9 Exemple numerice

Vom relua problema celor două corpuri din 3.3.9, pentru $e = 0.9$, calculând soluţia pe [0, 20] în dublă precizie, cu următoarele metode predictor-corector:

- Adams de ordinele 4 şi 5 (4.7 – Exemplele 2 şi 3), cu pas constant şi cu 1 iteraţie în corector – utilizând codul din Anexa, 4.2.
- Adams, ordin variabil-pas variabil, ordinul ≤ 12, lucrând cu subrutina DIVPAG din IMSL. S-au considerat două cazuri: ordinul ≤ 5, şi ≤ 12. Întervalul de integrare s-a împărţit în 20, respectiv 200, sub-intervale. Rezultate mai precise se obţin pentru 20 sub-intervale (pasul maxim este astfel 1.).

Rezultatele se dau în tabelele următoare.

$e = 0.9$, Metoda Adams ordin 4, pas constant: Erori absolute extreme la $t = 18.84$

Pasul h	Eroarea absolută Maximă ($\dot{x}$)	Minimă (x)	Nr. apeluri DERIVS
0.01	3.65 D0[†]	3.30 D-1[†]	4010
0.001	2.70 D-2	1.14 D-5	40010
0.0005	2.09 D-3	1.14 D-6	80010

[†] Eroarea maximă are loc în $\dot{y}$ şi cea minimă în y.

$e = 0.9$, Metoda Adams ordin 5, pas constant: Erori absolute extreme la $t = 18.84$

Pasul h	Eroarea absolută Maximă ($\dot{x}$)	Minimă (x)	Nr. apeluri DERIVS
0.01	4.29 D0[†]	2.97 D-1[†]	4013
0.001	6.64 D-4	3.69 D-7	40013
0.0005	3.33 D-5	1.86 D-8	80013

[†] Eroarea maximă are loc în $\dot{y}$ şi cea minimă în $\dot{x}$.

Observaţie

Exemplele din tabelele anterioare s-au rulat şi iterând în corector până la convergenţă, cu toleranţa *tol* = 1D-10 (pentru iteraţia de punct fix). Ordinul erorii a fost aproximativ acelaşi, iar numărul de iteraţii a crescut nesemnificativ: de exemplu, pentru Adams ordinul 5, $h = 0.001$, valorile din tabelul de mai sus sunt: 1.71 D-4, 9.85 D-8, 40942 ■

$e = 0.9$, Metoda Adams, ordin maxim 5, ordin variabil-pas variabil (DIVPAG): Erori absolute extreme la $t = 18.0$

Toleranţa *TOL*	Eroarea absolută Maximă (x)	Minimă (y)	Număr de paşi	Număr apeluri FCN
1 D-10	1.11 D-6	1.05 D-8	2437	2577
1 D-15	6.84 D-11	4.44 D-13	16276	16424

$e = 0.9$, Metoda Adams, ordin maxim 12, ordin variabil-pas variabil (DIVPAG): Erori absolute extreme la $t = 18.0$

Toleranţa *TOL*	Eroarea absolută Maximă (x)	Minimă (y)	Număr de paşi	Număr apeluri FCN
1 D-10	2.07 D-07	1.44 D-08	1611	1760
1 D-15	6.28 D-12	3.25 D-13	4228	4417

Observaţii

- Pasul maxim care a putut fi utilizat de DIVPAG a fost 1, adică lungimea sub-intervalului (nu s-au impus nici pasul maxim, nici pasul minim, care pot fi specificaţi în parametrii de intrare `hmax`, `hmin`). Aceasta explică numărul redus de paşi cu care lucrează rutina chiar pentru *TOL* foarte mic.
- *TOL* este un parametru care serveşte la controlul normei erorii locale, astfel ca eroarea globală să fie proporţională cu *TOL*. Norma aleasă în exemplele de mai sus a fost norma-∞. *TOL* şi tipul de normă sunt parametri de intrare ai rutinei DIVPAG (în tabloul `param`). A nu se confunda parametrul *TOL* cu toleranţa *tol* pentru iteraţia de punct fix în 4.7.6.

- Se observă superioritatea implementării în forma ordin variabil-pas variabil, cu controlul erorii, faţă de implementarea cu pas fix.
- Comparând rezultatele cu cele obţinute prin metoda Runge-Kutta 8(7) – v. 3.3.9, se constată o precizie mai mare a acesteia din urmă, cu preţul unui număr mai mare de evaluări de funcţii: de exemplu, pentru toleranţa 2.22D-15, ordinul erorii este în plaja D-13 … D-14, la un număr de 1380 paşi şi cca. 18000 evaluări ■

5 Ecuaţii diferenţiale "rigide"

Vom face în continuare o scurtă introducere în problematica ecuaţiilor diferenţiale rigide. Pentru o tratare extensivă trimitem la tratatul Hairer et al. (1991), dedicat integrării ecuaţiilor diferenţiale rigide.

În aplicarea unei metode numerice, dimensiunea pasului se alege dintr-o condiţie de precizie a soluţiei, impunând o toleranţă pentru eroarea de trunchiere locală. De obicei, pasul rezultat este în regiunea de stabilitate a metodei. Dacă însă dimensiunea pasului este dictată mai degrabă de condiţia de stabilitate decât de condiţia de precizie, zicem că avem o problemă *rigidă*.

Exemplu – 1

Să considerăm următoarea problemă:

$$x'' + 101x' + 100x = 0; \quad x(0) = 1,\ x'(0) = 0$$

Punând ecuaţia sub forma unui sistem de ordinul întâi, avem

$$x' = u, \quad u' = -100x - 101u; \qquad x(0) = 1,\ u(0) = 0,$$

sau matriceal, $\mathbf{x}' = \mathbf{A}\mathbf{x}$, unde $\mathbf{x} = [x \quad u]^T$, iar $\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -100 & -101 \end{pmatrix}$.

Valorile proprii ale lui $\mathbf{A}$ sunt $\lambda_1 = -1,\ \lambda_2 = -100$. Soluţia exactă este de forma

$$x(t) = C_1 e^{-t} + C_2 e^{-100t} \tag{84}$$

şi cu condiţiile iniţiale date rezultă

$$x(t) = \tfrac{100}{99} e^{-t} - \tfrac{1}{99} e^{-100t}. \tag{85}$$

Termenul în e^{-100t} tinde rapid spre 0 (se amortizează): de exemplu, pentru $t = 0.1$ avem $e^{-10} = 4.54 * 10^{-5}$, astfel că pentru $t > 0.1$ avem:

$$x(t) \approx \frac{100}{99} e^{-t} \qquad (86)$$

Soluţia dată de al doilea termen din (85) zice *tranzitorie*. Soluţia (86) reprezintă soluţia *staţionară*. Pentru *t* suficient de mare (de exemplu, $t > 0.1$), soluţia tranzitorie nu mai contribuie la soluţia (85), dar determină stabilitatea metodei. Chiar şi în cazul în care termenul al doilea dispare din soluţia (84) – condiţiile iniţiale sunt astfel că rezultă $C_2 = 0$ – valoarea proprie λ_2 determină intervalul de stabilitate al metodei. Într-adevăr, să presupunem că integrăm sistemul cu metoda RK4. Intervalul de stabilitate absolută a metodei este (v. 3.3.9): $-2.785296 < h\lambda < 0$, sau $0 < h(-\lambda) < 2.785296$, ceea ce conduce (pentru λ_2) la $h < 0.02785$. Întrucât ordinul erorii globale a metodei RK4 este h^4, dacă am vrea să calculăm soluţia cu o eroare de ordinul 10^{-4}, ar fi suficient un pas de ordinul $h = 0.1$ – care este însă în afara intervalului de stabilitate. Într-adevăr, calculând cu $h = 0.025$ (pas constant) se obţine $x(10) = 4.5858514950$ E-5 care are o eroare de –6.21E-13, în timp ce cu $h = 0.28$ şi 0.3, rezultă respectiv $x(9.996) = -2.74\ldots$E+1, şi $x(9.990) = -1.1459\ldots$E+44 – care probează instabilitatea ■

Fie un sistem liniar $\mathbf{x}' = \mathbf{Ax}$, de *m* ecuaţii, şi λ_i valorile proprii ale matricii **A**. Faptul că sistemul este rigid se defineşte prin următoarele condiţii:

$$\mathrm{Re}(\lambda_i) < 0, \quad i = \overline{1,m}$$

$$\max_{i=1,m} |\mathrm{Re}(\lambda_i)| >> \min_{i=1,m} |\mathrm{Re}(\lambda_i)|$$

Cu alte cuvinte, sistemul este rigid dacă are un punct fix stabil, şi **A** are valori proprii de mărimi foarte diferite. Definiţia de mai sus are mai multe inconveniente, şi anume:

- Nu mai convine în cazul în care matricea **A** are o valoare proprie egală cu zero;
- Nu se poate aplica pentru un sistem neliniar, sau pentru o singură ecuaţie.

Se dă atunci următoarea definiţie mai puţin precisă, dar aplicabilă atât sistemelor liniare cât şi celor neliniare, cât şi pentru o singură ecuaţie (Lambert, v. Cartwright and Piro (1992)):

"Dacă o metodă numerică este forţată să utilizeze, într-un anumit interval, un *pas excesiv de mic* pentru o problemă a cărei soluţie exactă este netedă în acel interval, atunci problema se zice *rigidă* în acel interval"

(O funcţie este netedă în [*a, b*], dacă are derivată continuă în [*a, b*].)

O problemă poate fi rigidă pe unele sub-intervale ale soluţiei şi non-rigidă pe altele. Definiţia de mai sus permite codurilor pentru integrarea numerică, să "recunoască" rigiditatea problemei pe intervalul pe care se calculează soluţia (sau pe sub-intervale ale acestuia), prin faptul că rutina este forţată să micşoreze excesiv pasul, pentru a satisface toleranţa impusă erorii de trunchiere. Utilizarea unui pas foarte mic poate crea probleme datorate acumulării erorilor de rotunjire sau creşterii timpului de calcul.

Pentru o problemă rigidă controlată de un parametru, s-ar cere ca metoda de integrare să fie stabilă pentru orice dimensiune a pasului, la orice valoare a parametrului pentru care problema este stabilă. De exemplu, problema de test pentru stabilitatea absolută, $x' = \lambda x$, este stabilă pentru $\mathrm{Re}(\lambda) < 0$, iar metoda ar trebui să fie stabilă pentru orice h, oricare ar fi λ cu $\mathrm{Re}(\lambda) < 0$. Regiunea de stabilitate absolută este atunci tot semi-planul (complex) stâng. Aceasta conduce la definiţia *A*-stabilităţii:

"O metodă se zice *A*-stabilă dacă regiunea ei de stabilitate liniară absolută conţine întreg semi-planul stâng".

Metodele RK explicite nu au *A*-stabilitate, deoarece regiunile lor de stabilitate absolută sunt finite – v. 3.3.7. În schimb, unele metode RK implicite sunt *A*-stabile şi se pot aplica la ecuaţii rigide. Inconvenientul este că ele cer rezolvarea unui sistem neliniar ceea ce măreşte numărul de evaluări de funcţii. *A*-stabilitatea este o cerinţă severă pentru o metodă. Pentru metodele multi-pas, ordinul maxim al unei metode *A*-stabile este $p = 2$ (a doua "barieră Dalquist"). De exemplu, metodele Adams-Moulton sunt *A*-stabile numai pentru $k = 1$ (metoda trapezului implicită), iar metodele BDF numai pentru $k = 1$ (Euler implicită) şi $k = 2$. V. Hairer et al. (1991).

În mod curent, problemele rigide se rezolvă cu metode BDF (4.2.3), numite şi metode *Gear*. Acestea sunt tot implicite. Metodele BDF sunt implementate în subrutina DIVPAG, care permite, prin codul metodei *param*(10), alegerea

metodelor Adams (v. 3.10), sau BDF până la ordinul 5. Ordinul este limitat din considerente de stabilitate. Codul *param*(13) permite alegerea rezolvitorului sistemului neliniar (iteraţia de punct fix, metoda Newton, şi metode Newton modificate), recomandându-se alegerea metodei Newton sau Newton-modificată (Secţiunea 3-IV, 2).

Exemplu – 2

Considerăm ecuaţia van der Pol

$$x'' - \lambda(1 - x^2)x' + x = A\cos(\omega t)$$

în cazul vibraţiei libere $A = 0$, pentru valori "mici" şi "mari" ale parametrului λ – de exemplu, $\lambda = 1$ şi $\lambda = 100$. Luăm condiţiile iniţiale: $x(0) = 1,\ x'(0) = 0$ şi integrăm ecuaţia pe intervalul [0, 100], în dublă precizie, cu următoarele metode:

- Metoda RK4, pas constant (codul din ANA_EcDif): Cazul $\lambda = 1$ se integrează cu un pas $h = 0.1$ (1000 paşi). În cazul $\lambda = 100$, pasul maxim pentru stabiliate este $h = 0.0083$ (12048 paşi); cu $h = 0.084$, la $t = 0.5628$, soluţia (x, x') calculată este (3.035E+88, - 6.173E+181), şi apoi (*NaN*, *NaN*), care probează instabilitatea metodei pentru acest pas. Pasul mult mai mic necesar în cazul $\lambda = 100$, arată că ecuaţia este rigidă. Pentru o precizie comparabilă cu cea a metodelor RKV şi BDF, în cazul $\lambda = 1$ a fost necesar un pas de 2.5E-3, iar în cazul $\lambda = 100$, un pas de 2.5E-4.
- Metoda RK-Verner 5(6), pas variabil între 1 şi 2.22E-15, cu toleranţa $tol = 1D-10$ (v. 3.10): Cazul $\lambda = 1$ este integrat cu 21375 evaluări (2594 paşi, pasul de încercare (trial step) la $t = 100$, $h = 5.9E-2$). Cazul $\lambda = 100$ este integrat cu 63522 evaluări (6941 paşi, pasul de încercare la $t = 100$ este $h = 1.59E-2$).
- Metoda BDF, pas variabil – ordin variabil (ordin ≤ 5), pasul între 1.0 şi 1E-5, toleranţă 1D-10, rezolvitor Newton cu jacobianul calculat numeric (subrutina DIVPAG): Cazul $\lambda = 1$ cere 14109 evaluări (12427 paşi, pasul de încercare la $t = 100$, $h = 1.24E-2$). Cazul $\lambda = 100$ cere 4015 evaluări (3256 paşi, pasul de încercare la $t = 100$, $h = 9.86E-1$).

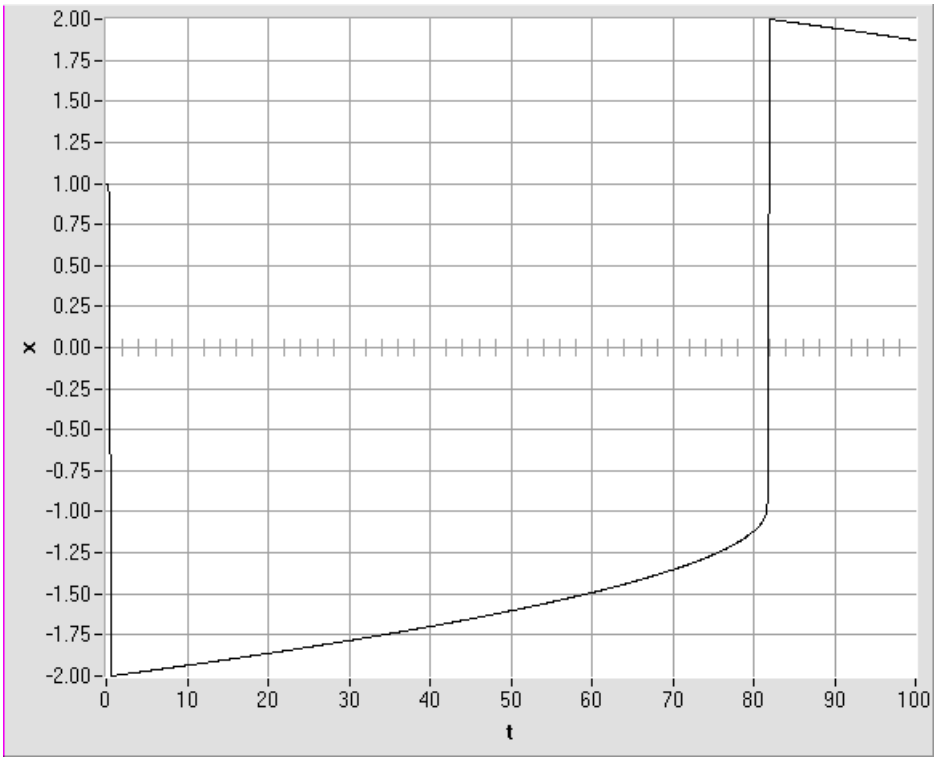
Rezultate comparative pentru ultimele două metode se dau în tabelul următor. Metodele s-au rulat cu aceeaşi toleranţă 1D-10, şi produc la $t = 100$, soluţii x care au 8 cifre semnificative identice. (Soluţiile x' au tot 8 cifre semnificative identice).

Ecuaţia van der Pol – Număr de evaluări de funcţii şi pas de încercare la $t = 100$

Metoda	$\lambda = 1$	$\lambda = 100$
RKV 5(6), *tol* = 1D-10	21375; 5.91 E-2	63522; 1.59 E-2
BDF, *tol* = 1D-10	14109; 1.24 E-2	4021; 9.86 E-1

Comparaţia eficienţei după numărul de evaluări de funcţii arată net superioritatea metodei BDF pentru cazul ecuaţiei rigide, dar şi pentru cazul ecuaţiei non-rigide.

În graficul următor se reprezintă soluţia $x(t)$ a problemei, pentru $\lambda = 100$.



Ecuaţia van der Pol, cazul $\lambda = 100$.

■

BIBLIOGRAFIE (selectivă)

Manuale de analiză numerică:

1. Atkinson K.E., "An Introduction to Numerical Analysis", John Wiley & Sons, N.Y., 1978. 2nd edition, 1989.
2. Curtis F.G., "Applied Numerical Analysis", Addison-Wesley Publishing Company, Inc., 1978.
3. Isaacson E., and Keller H.B., "Analysis of Numerical Methods", John Wiley & Sons, N.Y., 1966.
4. Kincaid D., and Cheney W., "Numerical analysis", 2nd edition, Brooks/Cole Publ. Co., 1996.
5. Ralston A., and Rabinowitz Ph., " A First Course in Numerical Analysis", McGraw-Hill, Inc., 1983.

Ecuaţii diferenţiale:

6. Cartwright J.H.E & Piro O., "The Dynamics of Runge-Kutta Methods", Int. J. Bifurcation and Chaos, 2, 427-449, 1992, http://lec.ugr.es/~julyan/publications.html
7. Chisăliţă A., Lung N, Chisăliţă G.-A., "Criterii numerice şi procedee analitice pentru identificarea răspunsului haotic", Contract 34/1998, Tema 25/155, Universitatea Tehnică din Cluj-Napoca, 1998.
8. Dormand J. R., "Numerical Methods for Differential Equations", CRC Press LLC, (1996).
9. Hairer E., Nørsett S.P., and Wanner G., "Solving Ordinary Differential Equations I (Nonstiff Problems)", Springer-Verlag, 1987.
10. Hairer E., and Wanner G., "Solving Ordinary Differential Equations II (Stiff and Differential-Algebraic Problems)", Springer-Verlag, 1991.
11. Lambert J.D., "Numerical Methods for Ordinary Differential Systems. The Initial Value Problem.", J. Wiley & Sons, 1991.

Biblioteci şi coduri:

12. Chisăliţă A, "ANA_EcDif", 2006, ftp.utcluj.ro/pub/users/chisalita/

13. "IMSL Mathematical and Statistical Libraries", Compaq Visual Fortran 6.6, IMSL Help, 1999.
14. Brankin R.W. and Gladwell I., "RKSUITE_90 Release 1.0 June 1994", http://www.netlib.org/ode/rksuite/.
15. Brankin R.W., Gladwell I., and Shampine L.F. , "RKSUITE Release 1.0 November 1991", http://www.netlib.org/ode/rksuite/.