CURS 6

METODE NUMERICE PENTRU ECUAȚII DIFERENȚIALE ORDINARE

Partea I (Rezumat)

6-I METODE NUMERICE PENTRU ECUAȚII DIFERENȚIALE DE ORDINUL ÎNTÂI

În această secțiune se vor prezenta metode numerice pentru ecuații şi sisteme de ecuații diferențiale de ordinul întâi – problema cu valori inițiale. O ecuație sau sistem de ordin mai mare decât unu se pot reduce la un sistem echivalent de ordinul unu, prin adăugarea de funcții 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

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

■

Notă:

Pentru sistemele de ordinul doi s-au desvoltat şi metode specifice acestor sisteme. (Aceasta, datorită faptului că ecuațiile diferențiale ale mişcării în mecanică, în particular în probleme de vibrații, sunt de ordinul doi.)

1 Problema cu valori inițiale (considerații generale)

Fie ecuația

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

cu condiția inițială

$$x(t_0) = x_0 \tag{1'}$$

Ecuația (1) cu condiția inițială (1') constituie o problemă cu valori inițiale (sau o problemă Cauchy). Dacă funcția f îndeplineşte următoarele condiții 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 funcției $|f|$ pe D, problema are o soluție unică $x(t)$ definită pe intervalul $|t - t_0| \le \alpha$, unde $\alpha = \min(a, b/M)$.

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

■

Pentru un sistem de m ecuații diferențiale cu m funcții 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

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

cu condiția inițială

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

Cu domeniul $D = I \times \Omega$, unde unde I este definit de $|t - t_0| \le a$, iar Ω de $|x_i - x_i^{(0)}| \le b_i$, $i = \overline{1, m}$, condițiile 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 funcției $|f_i|$ pe D, şi cu $M = \max_{i=1,m} M_i$. Dacă 1' şi 2' sunt îndeplinite, atunci există o soluție 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, condiția 2' este îndeplinită dacă **f** are derivate parțiale în raport cu $x_j, j = \overline{1,m}$, continue pe $I \times \Omega$.

Notă: Condiția Lipschitz pentru funcția **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-∞ ■

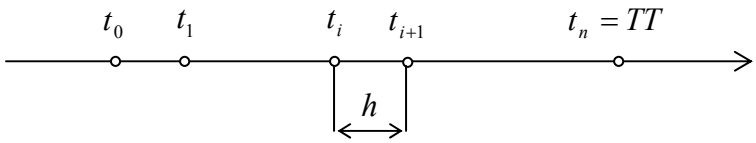
În ceea ce urmează vom considera probleme cu valori inițiale (1, 1') şau (2, 2'), pentru care vom presupune îndeplinite condițiile de existență şi unicitate ale soluției. Considerăm calculul soluției pentru un interval de integrare $[t_0, TT]$, inclus în intervalul de existență a soluției. Metodele numerice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

2 Operatori de integrare numerică (intr-un singur pas, în mai mulți paşi, expliciți, impliciți)

Găsirea soluției ecuației (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) Ecuația (1) se cere să fie satisfăcută în punctele t_i, iar între aceste puncte, variația funcției $x(t)$ se estimează.



Vom nota în ceea ce urmează:

$x(t_i)$ = soluția exactă;

x_i = soluția 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 soluția la momentul t_{i+1} în funcție de soluția 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 funcție numai de x_i, şi eventual x_{i+1}, operatorul se zice *într-un pas*, altfel se zice *în mai mulți 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 impliciți conduce la rezolvarea ecuației (3) în necunoscuta x_{i+1}, printr-o metodă pentru ecuații neliniare. O comparație între operatori într-un singur pas şi în mai mulți paşi se va face în 4.8.

Distanța 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:

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

Ecuația (1) este

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

şi prin derivare succesivă obținem:

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

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

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 diferența divizată, obținând 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

Definiție: Ordin

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

Definiții 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 funcție $x(t)$ 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 față, 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 funcția *f* să aibă derivate până la ordinul *p*, ceea ce, în general, nu este cerut pentru existența soluției. Totuşi, pentru multe din problemele practice, această condiție 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 importanță teoretică. Ea serveşte la demonstrarea teoremelor de existență, şi la exemplificarea noțiunilor de convergență şi stabilitate pe exemplul unei metode simple.

3.3 Metode Runge-Kutta

3.3.1 Construcția metodelor Runge-Kutta

Metodele Runge-Kutta (abreviat RK) utilizează desvoltarea în serie Taylor, dar înlocuiesc calculul derivatelor de ordin superior, cu calculul funcției f în puncte de forma $(t + h\alpha, x + h\phi)$, unde α şi ϕ sunt definiți de coeficienții 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 ținut 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 soluția calculată în t_i, prin $x(t_i)$ soluția exactă, şi că punem condiția $x_i = x(t_i)$ (până la termenul de ordinul *p* în *h*).

O caracteristică a metodei este numărul de evaluări al membrului doi al ecuației (1) sau sistemului (2), pe un pas. Acest număr este numit "numărul de evaluări de funcții". O metodă RK care face *q* evaluări de funcții va fi numită "cu *q*-trepte" (*q*-stage). Pentru a obține 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 coeficienți ai metodei, iar $k_m = k_m(t_i, x_i, h)$. Se obține

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

În (9, 10) funcțiile 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:

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

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

Se obişnuieşte ca coeficienții 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)

Condiții pentru coeficienții metodei:

- Coeficienții ω_m îndeplinesc condiția de *consistență*:

$$\sum_{m=1}^{q} \omega_m = 1 \qquad (14)$$

Aceasta asigură convergența metodei – v. 3.3.3.

- Coeficienții α_m, β_{mj} sunt supuşi la condițiile:

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

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

Aceste condiții simplifică deducerea coeficienților pentru metodele de ordin mai mare ca 2. Pentru justificări ale condițiilor (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 aproximațiile x_j cu soluția 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 condiția 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 obține o metodă de ordin *p*, coeficienții α_m, β_{mj} şi ω_m se determină din condiția de mai sus, cu respectarea condițiilor (14, 14').

Eroarea de trunchiere globală pe pasul $i+1$, este eroarea aproximației 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 obține 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 obțin 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:

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array}$$

■

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 puțin 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 Convergență şi consistență

Metoda RK se zice convergentă dacă, pentru $h \to 0$, soluția calculată tinde la soluția 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, condiția 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 inițiale) dacă avem

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

Cu (20) şi expresiile (11, 12) ale funcțiilor 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 condiția de consistență (20) este echivalentă cu condiția

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

Se demonstrează că, *consistența este o condiție necesară şi suficientă pentru convergență* (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 coeficienților 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 condiția de consistență (21) şi condițiile (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:

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

în care:

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

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

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

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

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 ecuații, 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 soluția

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

Funcțiile k_m sunt definite de (11) şi revin la calculul funcției *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 funcția *f* pe *aceleaşi* puncte ca (24) – adică având *aceeaşi* coeficienți α_m, β_{mj}. Fie cea de-a doua metodă, care calculează soluția

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

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 estimează 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 ecuațiilor diferențiale. 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 toleranță impusă – v. mai jos.

Metode Dormand-Prince (DOPRI):

Dormand & Prince au construit metode mai precise, de ordine $p+1(p)$, în care se minimizează coeficienții erorii în metoda de ordin mai mare. Soluția 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. Coeficienții 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 ecuațiilor diferențiale de ordinul întâi, existente în momentul de față.

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 toleranța *tol* specificată de utilizator, adică:

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

Eliminând *C* între ultimele două realții, 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 siguranță, 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: $err = \| \hat{\mathbf{x}}_1 - \mathbf{x}_1 \|$.

Observații

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

$rel \cong \frac{Ch^{p+1}}{|\, x_1 + err \,|}$, $tolrel \approx \frac{Ch_{opt}^{p+1}}{|\, x_1 + err \,|}$, 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, soluția x_2 cu doi paşi de mărime *h* şi soluția X_2 cu un pas dublu 2*h*, şi estimând eroarea prin diferența celor două soluții. Pentru o metodă de ordinul *p* se obține (Hairer et al. (1987)):

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

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

Soluția

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

este o aproximație 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 soluției. 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 toleranța *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 soluției 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 toleranța *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 globală, şi locală respectiv, pe pasul *i*+1. După definițiile din 3.3.1, avem:

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

şi definim $\bar{e}_0 = 0$, şi

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

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

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

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 ecuației (1) în jurul unei soluții a acesteia. Fie ecuația diferențială

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

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

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

Considerăm o *perturbație* $\delta x(t)$ a soluției (provenind dintr-o perturbare a condiției inițiale), unde $| \delta x(t) | \le \varepsilon$:

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

şi scăzând relația (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 $\delta x(t)$. Obținem:

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

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

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

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

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

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

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

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

În fine, notând x în loc de y, ecuația (35') devine o ecuație de tipul

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

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

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

Problema (36, 36') constituie testul pentru stabilitatea liniară a metodei – numit şi testul *Dalquist*. Soluția 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 $x = 0$.

Observație

Punctele fixe ale unei ecuații (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 soluțiile ecuației $x = g(x)$). 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 ecuației $x' = f(t,x)$, 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 ecuației sunt şi puncte fixe ale metodei RK ■

Definiție

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

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

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

Pentru un sistem de *m* ecuații diferențiale (3)

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

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

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

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 funcției **f** în raport cu **x**. Aproximăm $\mathbf{J}(t) = \mathbf{A}$ = constant. Cu aceasta, schimbând notația $\delta\mathbf{x} \mapsto \mathbf{x}$, 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*×*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 ecuațiile (38) se decuplează, sistemul reducându-se la *m* ecuații independente de forma (31). Într-adevăr, dacă vectorii proprii sunt $\{\mathbf{v}_j,\ j = \overline{1,m}\}$, definiți 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 condiții inițiale arbitrare, de exemplu

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

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

Revenind la ecuația (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 $x_1 = R(h\lambda)$.

Definiție

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

■

Regiunea de stabilitate absolută pentru o metodă, este mulțimea 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 mulțimea

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

Uneori, regiunea de stabilitate este definită împreună cu frontiera sa, prin condiția $|R| \le 1$. 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 omițând 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 funcția de stabilitate este

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

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

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

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

Din aceasta rezultă că funcția de stabilitate a unei metode explicite cu q trepte este un polinom de gradul q în h. Condiția $|R| < 1$ 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 ecuații diferențiale *rigide* – v. 5, la care metodele explicite nu mai convin.

Pentru reprezentarea regiunilor de stabilitate liniară în cazul λ = complex, pentru metodele RKp, p =1, 2, 3, 4 – v. Hairer & Wanner (1991), Cartwright & Piro (1992). Intersecțiile regiunilor cu axa reală dau intervalele de stabilitate pentru cazul λ = real. Aceste intervale se găsesc din condiția $|R| < 1$, unde R este definit de (37) cu λ = real şi negativ (conform ipotezei Re(λ) < 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 diferență constă în aceea că, pentru o problemă neliniară, metodele RK pot conține pe lângă punctele fixe ale problemei – v. Observația din 3.3.7 – şi puncte fixe adiționale. Excepție face metoda Euler care are numai punctele fixe ale problemei. Punctele fixe adiționale 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. Diferența între problemele liniare şi neliniare constă în aceea că, pentru probleme neliniare bazinul de atracție este

mărginit, în timp ce pentru o problemă liniară acesta este nemărginit. Astfel, pentru o problemă liniară există convergență pentru orice condiții inițiale, cu condiția ca $h\lambda$ să fie în interiorul regiunii de stabilitate, în timp ce pentru o problemă neliniară este necesar, în plus, ca condițiile inițiale să fie conținute în bazinul de atracție. Pentru desvoltări, trimitem la Cartwright & Piro (1992).

În practica de calcul s-a constatat că pentru un răspuns haotic, unde calculația 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 operațiile aritmetice, vectorizarea ciclurilor DO în subrutina de integrare a sistemului dat, opțiunile de "build" (ca optimizarea codului), etc. V. raportul Chisăliță 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 inițiale – v. Dormand and Prince (1978), Brankin and Gladwell (1994). Problema consideră mişcarea relativă a două puncte materiale care interacționează prin legea atracției universale, şi este descrisă, în coordonate carteziene, de sistemul de ecuații diferențiale:

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

în care $r = (x^2 + y^2)^{1/2}$. Se consideră conditiile inițiale 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 $e < 1$. Soluția 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 ecuația lui Kepler: $u - e\sin u = t$. Soluția 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 periodicității soluției. Pentru rezolvarea numerică, sistemul dat se transformă într-un sistem echivalent de 4 ecuații 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 condițiile inițiale: $x(0) = 1 - e$, $y(0) = 0$, $v(0) = 0$, $w(0) = \sqrt{(1+e)/(1-e)}$.

Calculăm soluția pe intervalul [0, 20], adică peste trei perioade, pentru valorile e = 0.1 şi e = 0.9 ale excentricității. 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 împărțit în 20, şi respectiv în 200, sub-intervale. Rezultatele mai precise se obțin pentru împărțirea 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. Toleranța *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 împărțit în 20, şi respectiv în 200 sub-intervale. Rezultatele mai precise se obțin pentru împărțirea în 20 sub-intervale – în acest caz pasul maxim posibil este 1.0 (lungimea sub-intervalului)..

Pentru soluția exactă, ecuația lui Kepler se rezolvă prin metoda punctului fix, cu toleranța *eps* =1D-13. În tabelele următoare este dată eroarea absolută maximă şi minimă a soluției calculate, la timpul cel mai apropiat de 3 perioade. În paranteze se indică funcția – 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

Toleranța *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

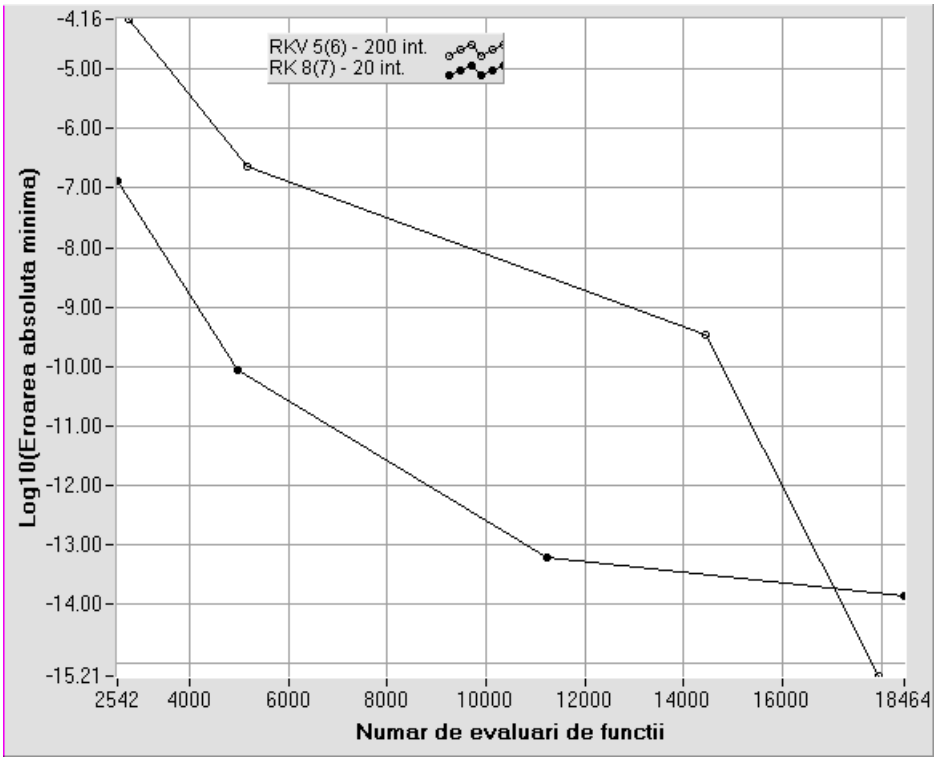
e = 0.9, Metoda RK 8(7) – 20 sub-intervale:

Erori absolute extreme la t = 18.00

Toleranța *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

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

Următorul grafic dă o comparație a eficienței 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 funcție de numărul de evaluări de funcții. Metoda este cu atât mai eficientă, cu cât realizează o precizie dată cu un număr mai mic de evaluări de funcții.



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

Observații

a.ch. – Noiembrie 2008

- Testul cel mai sever este cazul $e = 0.9$. Cu acelaşi pas (în RK4), sau aceeaşi toleranță (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 toleranțe, 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 toleranța 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 menținerea erorii globale aproximativ proporțională cu valorea *tol* (v. referințele 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 ecuații. Acest număr este referit ca "numărul de evaluări de funcții" 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 prețul măririi numărului de paşi sau a numărului total de evaluări de funcții. Se remarcă creşterea preciziei cu creştera ordinului metodei. Din comparația eficienței celor trei metode rezultă că, pentru problema considerată, metoda RK 8(7) oferă cel mai bun raport precizie/număr de evaluări de funcții – cu excepția cazului unei toleranțe apropiată de cea minimă admisă (2.22D-15), când metoda RKV 5(6) este superioară ■