

Komputerowa analiza zagadnień różniczkowych

7. Wybrane metody Rungego-Kutty. Zmiana kroku całkowania.

P. F. Góra

<http://th-www.if.uj.edu.pl/zfs/gora/>

2013

Metody DIRK

Jeśli spodziewamy się problemów ze stabilnością, w szczególności jeśli rozwiązujemy problemy sztywne, powinniśmy stosować metody niejawne. Są one jednak **bardzo kosztowne obliczeniowo**: jeżeli przy użyciu metody s -etapowej rozwiązujemy problem m -wymiarowy, w każdym kroku musimy rozwiązywać $s \times m$ wymiarowy układ równań algebraicznych, w ogólności nieliniowych.

Aby zmniejszyć ten koszt nie tracąc stabilności metod niejawnych, stosuje się metody DIRK (ang. *Diagonally Implicit Runge-Kutta*), zwane też metodami **półniejawnymi** (ang. *semi-implicit*). W metodach tych macierz B ma niezerowe elementy diagonalne i poddiagonalne — elementy ponaddiagonalne.

nalne są zerowe:

$$\begin{array}{c|cccccc}
 \alpha_1 & \beta_{11} & 0 & 0 & \cdots & 0 \\
 \alpha_2 & \beta_{21} & \beta_{22} & 0 & \cdots & 0 \\
 \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
 \alpha_s & \beta_{s1} & \beta_{s2} & \beta_{s3} & \cdots & \beta_{ss} \\
 \hline
 & w_1 & w_2 & w_2 & \cdots & w_s
 \end{array} \tag{1}$$

Dzięki temu, stosując s -etapową metodę do problemu m -wymiarowego, zamiast rozwiązywać **jeden** układ równań algebraicznych o wymiarze $s \times m$, należy rozwiązać s układów o wymiarze m — po jednym układzie na każdy etap. Jest to numerycznie prostsze.

Jeżeli dodatkowo $\beta_{11} = \beta_{22} = \cdots = \beta_{ss} = \gamma$, taką metodę nazywa się metodą SDIRK (*Singly Diagonally Implicit Runge-Kutta*).

Przykład

$$\begin{array}{c|cc} \gamma & \gamma & 0 \\ 1 - \gamma & 1 - 2\gamma & \gamma \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}, \quad \gamma = \frac{3 + \sqrt{3}}{6} \quad (2)$$

Sprawdzamy rząd metody. Nietrywialnymi wyrażeniami, które trzeba sprawdzić, są:

$$\frac{1}{2} \cdot \gamma + \frac{1}{2} \cdot (1 - \gamma) = \frac{1}{2} \text{ — metoda jest rzędu drugiego.}$$

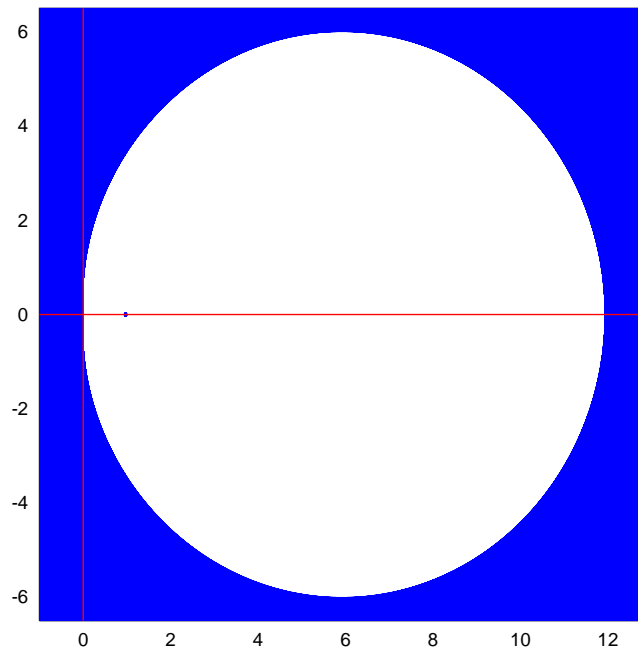
$$\frac{1}{2} \cdot \gamma^2 + \frac{1}{2} \cdot (1 - \gamma)^2 = \gamma^2 - \gamma + \frac{1}{2} = \frac{1}{3}, \quad \frac{1}{2} \cdot \gamma \cdot \gamma + \frac{1}{2} \cdot ((1 - 2\gamma) \cdot \gamma + \gamma \cdot (1 - \gamma)) = -\gamma^2 + \gamma = \frac{1}{6}$$

— metoda jest rzędu trzeciego.

$$\frac{1}{2} \cdot \gamma^3 + \frac{1}{2} \cdot (1 - \gamma)^3 = \frac{3\gamma^2 - 3\gamma + 1}{6} = \frac{1}{4}, \quad \frac{1}{2} \cdot \gamma \cdot \gamma \cdot \gamma + \frac{1}{2} \cdot (1 - \gamma) \cdot ((1 - 2\gamma) \cdot \gamma + \gamma \cdot (1 - \gamma)) = \frac{4\gamma^3 - 5\gamma^2 + 2\gamma}{2} = \frac{6 + \sqrt{3}}{36} \neq \frac{1}{8} \text{ — metoda nie jest rzędu czwartego.}$$

Obszar stabilności tej metody dany jest przez $\{z \in \mathbb{C} : |G(z)| < 1\}$, gdzie

$$G(z) = -6 \frac{(1+\sqrt{3})z^2 + 2\sqrt{3}z - 6}{[(3+\sqrt{3})z - 6]^2}.$$



Obszar niezacieniony jest obszarem nistabilności.

Przykład

Ciągle publikuje się i bada nowe metody RK. W pracy G. Yu. Kulikov, S. K. Shindin, Appl. Num. Math. **59**, 707 (2009), dyskutowane są, między innymi, własności metody

0	0	0	0	0
c_1^2	$\frac{6(c_1^2+\theta)-5}{12}$	$\frac{1-\theta}{2}$	$\frac{1-\theta}{2}$	$\frac{6(c_1^2+\theta)-5}{12}$
$1 - c_1^2$	$\frac{7-6(c_1^2+\theta)}{12}$	$\frac{\theta}{2}$	$\frac{\theta}{2}$	$\frac{5-6(c_1^2+\theta)}{12}$
1	0	$\frac{1}{2}$	$\frac{1}{2}$	0
<hr/>				
	0	$\frac{1}{2}$	$\frac{1}{2}$	0

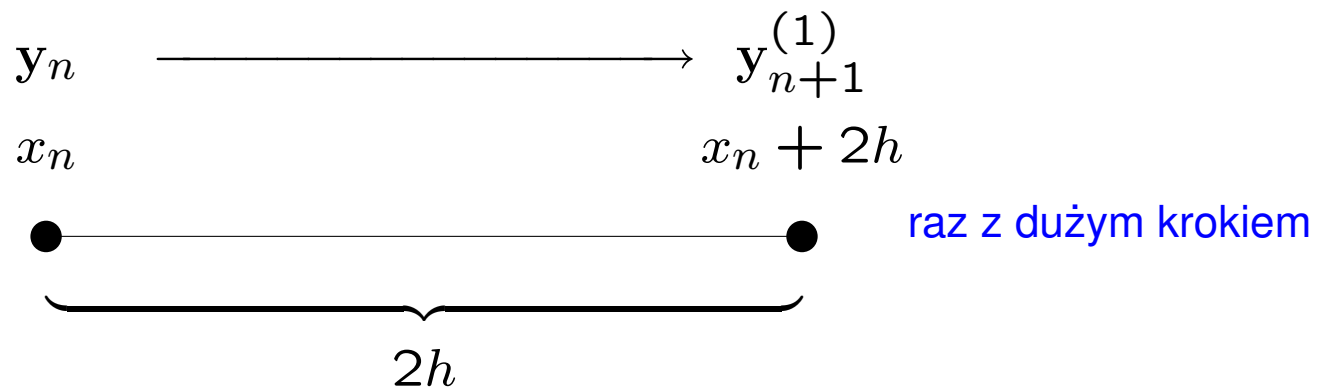
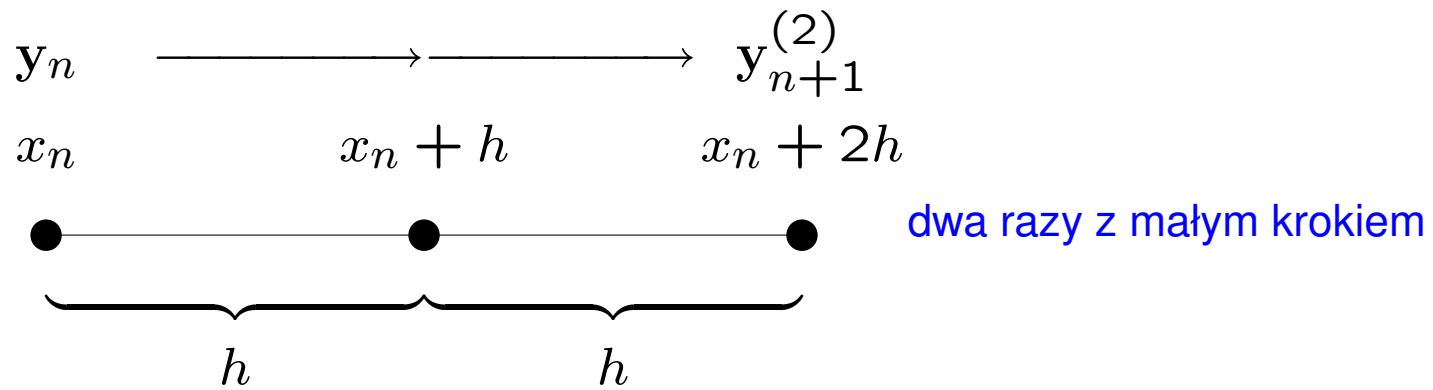
$$c_1^2 = \frac{3-\sqrt{3}}{6}, \quad \theta = \frac{1}{2} + \frac{2\sqrt{3}}{9}$$

Adaptacyjne podwajanie/połowienie kroku

Całkowanie ze stałym krokiem to numeryczne samobójstwo



Dobre algorytmy numerycznego całkowania ODE powinny, w miarę możliwości, same ustalać krok, z jakim przechodzą zadany przedział. W tym celu algorytm musi znać **oszacowanie błędu** popełnionego w ciągu jednego kroku. Dla metod Rungego-Kutty najprostszym tego typu algorytmem jest algorytm **adaptacyjnego podwajania/połowienia kroku**. Przypuśćmy, że żądam, aby błąd na jeden krok nie przekraczał Δ_{\max} . Niech aktualny krok całkowania wynosi h . Przechodzę przedział $[x_n, x_n + 2h]$ dwa razy: raz dwoma krokami o długości h i raz krokiem o długości $2h$.



W ten sposób otrzymuję *dwa* oszacowania wartości y_{n+1} . Dla metody rzędu p spełniają one

$$y(x_n + 2h) = y_{n+1}^{(2)} + 2h^{p+1}\phi + O(h^{p+2}), \quad (3a)$$

$$y(x_n + 2h) = y_{n+1}^{(1)} + (2h)^{p+1}\phi + O(h^{p+2}). \quad (3b)$$

Różnica

$$\Delta = y_{n+1}^{(1)} - y_{n+1}^{(2)} \quad (4)$$

stanowi oszacowanie błędu. Zachodzą dwa przypadki:

1. $\|\Delta\| \leq \Delta_{\max}$: Wówczas przechodzę do punktu $x_{n+1} = x_n + 2h$, jako rozwiązanie przyjmuję $y(x_n + 2h) = y_{n+1}^{(2)}$, zwiększam krok

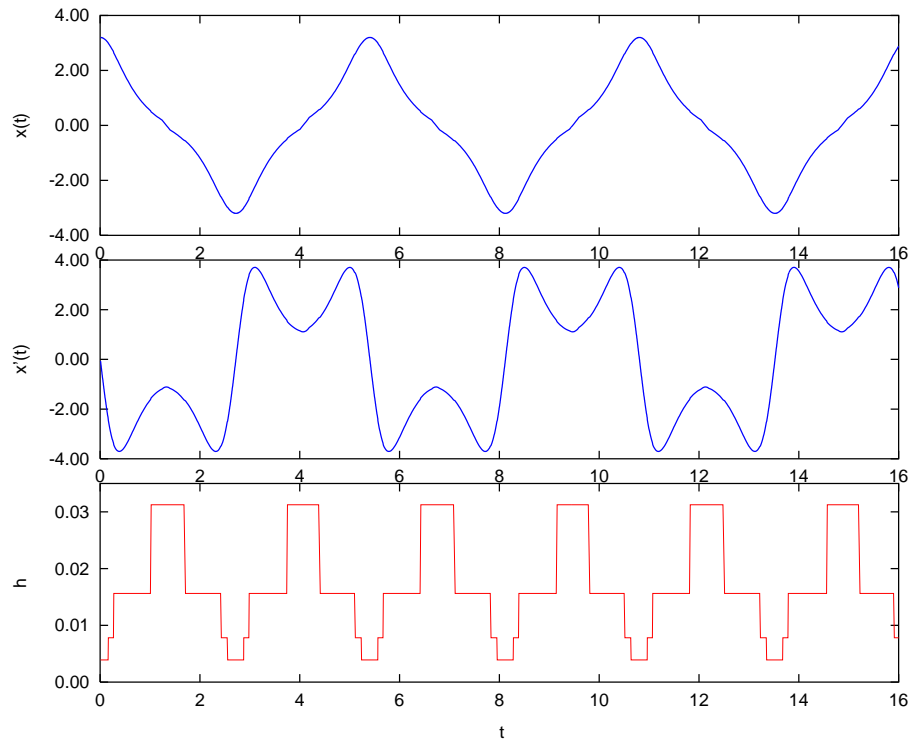
$h \rightarrow 2h$ i próbuję przejść następny przedział z dwa razy większym krokiem.

2. $\|\Delta\| > \Delta_{\max}$: Wówczas cofam się do punktu x_n , zmniejszam krok $h \rightarrow h/2$ i ponawiam całą procedurę. W tym wypadku należy się zabezpieczyć przed zmniejszeniem się kroku poniżej pewnego h_{\min} .

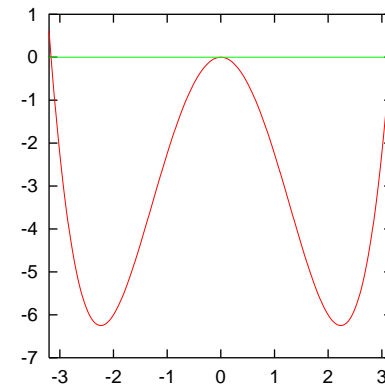
Istotne może być *jaka* normę weźmiemy.

Dla klasycznej metody czteroetapowej wymaga to $4 + 2 \cdot 4 - 1 = 11$ obliczeń prawej strony. *Metoda adaptacyjnego podwajania/połowienia kroku może być stosowana także przy obliczeniach za pomocą metod niejawnych.*

Przykład — oscylator Duffinga i podwajanie/półwienie kroku



$$\begin{cases} \ddot{x} + x^3 - 5x = 0 \\ x(0) = 3.2, \dot{x}(0) = 0 \end{cases}$$



Mały rysunek pokazuje potencjał Duffinga.

Lokalna ekstrapolacja

Jeśli popełniany błąd nie przekracza maksymalnego błędu dopuszczalnego, można jeszcze poprawić rozwiązanie za pomocą tak zwanej *lokalnej ekstrapolacji*: Przyjmijmy, że lewe strony obu równań (3) są sobie równe. Wówczas eliminując $h^{p+1}\phi$ otrzymujemy

$$y(x_n + 2h) = y_{n+1}^{(2)} + \frac{\Delta}{2^p - 1} + O(h^{p+2}). \quad (5)$$

Może to poprawić numeryczne własności rozwiązania, ale tak naprawdę zysk na rzędzie metody jest pozorny.

Zagnieżdżone (*embedded*) metody Rungego-Kutty

Rozważmy dwie jawne s -etapowe metody Rungego-Kutty:

$$\mathbf{y}_{n+1}^{(1)} = \mathbf{y}_n + h \sum_{i=1}^s w_i^{(1)} \mathbf{k}_i + O(h^{p+1}), \quad (6a)$$

$$\mathbf{y}_{n+1}^{(2)} = \mathbf{y}_n + h \sum_{i=1}^s w_i^{(2)} \mathbf{k}_i + O(h^{p+2}), \quad (6b)$$

$$\mathbf{k}_i = \mathbf{f} \left(x_n + \alpha_i h, \mathbf{y}_n + h \sum_{j=1}^{i-1} \beta_{ij} \mathbf{k}_j \right) \quad (6c)$$

Metody te są tak skonstruowane, że *różnią się jedynie wagami*, mają takie same punkty pośrednie, a więc taki sam zestaw wektorów $\{\mathbf{k}_i\}$, a ich rzędy różnią się o jeden.

Obliczając *jedyn* zestaw pochodnych w punktach pośrednich, mamy oszacowanie błędu:

$$\Delta = y_{n+1}^{(1)} - y_{n+1}^{(2)} = h \sum_{i=1}^s \left(w_i^{(1)} - w_i^{(2)} \right) \mathbf{k}_i, \quad \|\Delta\| \sim h^{p+1}. \quad (7)$$

Skoro $\|\Delta\| \sim h^{p+1}$, to $\Delta_{\max} \sim h_{\max}^{p+1}$. Mam więc oszacowanie

$$h_{\max} = h \left(\frac{\Delta_{\max}}{\|\Delta\|} \right)^{1/(p+1)}. \quad (8)$$

Jeśli $\|\Delta\| > \Delta_{\max}$, *zmniejszam* krok $h \rightarrow h_{\max}$ i powtarzam bieżący krok. Jeśli $\|\Delta\| < \Delta_{\max}$, *zwiększam* krok $h \rightarrow h_{\max}$ i z powiększonym krokiem próbuję iść dalej. Metody zagnieżdżone pozwalają zatem na bardziej precyzyjną kontrolę kroku od podwajania/półowienia, wymagają także mniej obliczeń (s obliczeń funkcji f).

Przykład: Metoda Bogackiego-Shampine'a — 3(2)

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{3}{4}$	0	$\frac{3}{4}$		
1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{4}{9}$	
	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{4}{9}$	0
	$\frac{7}{24}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{8}$

Puste miejsca oznaczają zera.

Uwaga: Dwa zestawy wag!

Metoda typu FSAL (First Same As Last).

Sprawdzamy rzędy tych metod:

pierwszy zestaw wag	drugi zestaw wag
$\frac{2}{9} \cdot 0 + \frac{1}{3} \cdot \frac{1}{2} + \frac{4}{9} \cdot \frac{3}{4} + 0 \cdot 1 = \frac{1}{6} + \frac{1}{3} = \frac{1}{2};$ $\frac{2}{9} \cdot 0 + \frac{1}{3} \cdot \frac{1}{2} + \frac{4}{9} \left(\frac{3}{4} + 0 \right) + 0 \cdot (\dots) =$ $\frac{1}{6} + \frac{1}{3} = \frac{1}{2}$ <p>jest rzędu drugiego</p>	$\frac{7}{24} \cdot 0 + \frac{1}{4} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{3}{4} + \frac{1}{8} \cdot 1 = \frac{1}{8} + \frac{1}{4} + \frac{1}{8} = \frac{1}{2};$ $\frac{7}{24} \cdot 0 + \frac{1}{4} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{4} + \frac{1}{8} \left(\frac{2}{9} + \frac{1}{3} + \frac{4}{9} \right) =$ $\frac{1}{8} + \frac{1}{4} + \frac{1}{8} = \frac{1}{2}$ <p>jest rzędu drugiego</p>
$\frac{2}{9} \cdot 0^2 + \frac{1}{3} \left(\frac{1}{2} \right)^2 + \frac{4}{9} \left(\frac{3}{4} \right)^2 + 0 \cdot 1^2 =$ $\frac{1}{12} + \frac{1}{4} = \frac{1}{3};$ $\frac{2}{9} \cdot 0 + \frac{1}{3} \cdot \frac{1}{2} \cdot 0 + \frac{4}{9} \left(0 \cdot 0 + \frac{3}{4} \cdot \frac{1}{2} \right) + 0 =$ $\frac{4}{9} \cdot \frac{3}{8} = \frac{1}{6}$ <p>jest rzędu trzeciego</p>	$\frac{7}{24} \cdot 0^2 + \frac{1}{4} \left(\frac{1}{2} \right)^2 + \frac{1}{3} \left(\frac{3}{4} \right)^2 + \frac{1}{8} \cdot 1^2 =$ $\frac{1}{16} + \frac{3}{16} + \frac{1}{8} = \frac{3}{8} \neq \frac{1}{3}$ <p>nie jest rzędu trzeciego</p>

Obszary stabilności metody Bogackiego-Shampine'a dane są przez następujące wyrażenia:

Dla metody rzędu trzeciego, służącej do obliczania następnych wartości poszukiwanej funkcji:

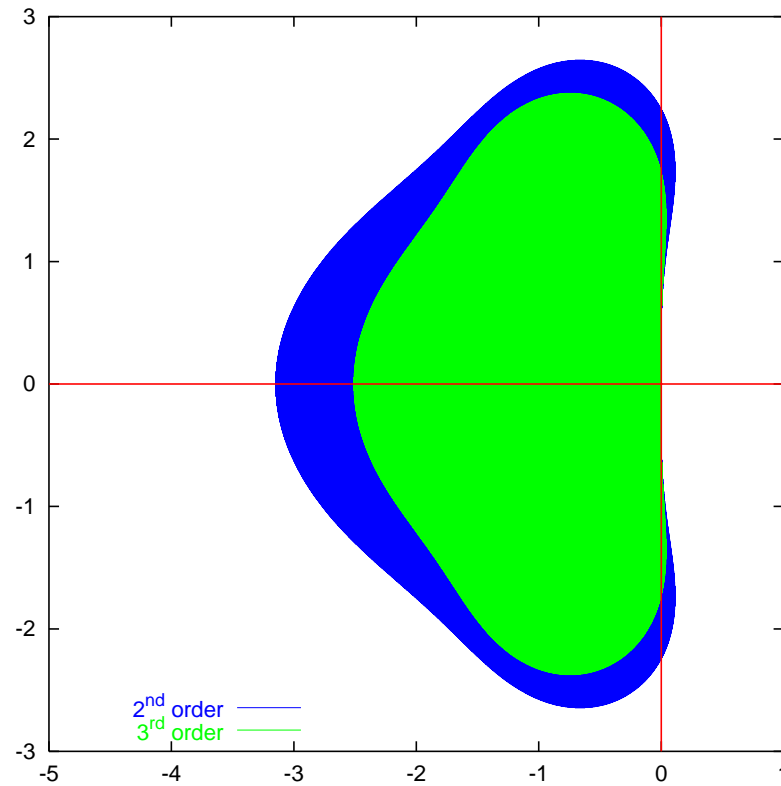
$$\left| \frac{1}{6} (z^3 + 3z^2 + 6z + 6) \right| < 1 \quad (9a)$$

Dla metody rzędu drugiego, służącej do kontroli błędów:

$$\left| \frac{1}{48} (z^4 + 9z^3 + 24z^2 + 48z + 48) \right| < 1 \quad (9b)$$

Krok powinien być tak dobrany, aby nie wyjść poza obszar stabilności żądanej z metod.

Obszary stabilności metody Bogackiego-Shampine'a



Jawnie rozpisana metoda Bogackiego-Shampine'a:

$$\mathbf{k}_1 = f(x_n, y_n) \quad (10a)$$

$$\mathbf{k}_2 = f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}h\mathbf{k}_1\right) \quad (10b)$$

$$\mathbf{k}_3 = f\left(x_n + \frac{3}{4}h, y_n + \frac{3}{4}h\mathbf{k}_2\right) \quad (10c)$$

$$y_{n+1} = y_n + h\left(\frac{2}{9}\mathbf{k}_1 + \frac{1}{3}\mathbf{k}_2 + \frac{4}{9}\mathbf{k}_3\right) \quad (10d)$$

$$\mathbf{k}_4 = f(x_n + h, y_{n+1}) \quad (10e)$$

$$\mathbf{z} = y_n + h\left(\frac{7}{24}\mathbf{k}_1 + \frac{1}{4}\mathbf{k}_2 + \frac{1}{3}\mathbf{k}_3 + \frac{1}{8}\mathbf{k}_4\right) \quad (10f)$$

$$\mathbf{k}_1 = \mathbf{k}_4 \quad (10g)$$

Wektor (10e) staje się wektorem (10a) w **następnym** kroku całkowania, a więc **oblicza się go tylko raz**. Wektor \mathbf{z} bierze udział tylko w szacowaniu błędów.

Metoda Fehlberga — 4(5)

0						
$\frac{1}{4}$	$\frac{1}{4}$					
$\frac{3}{8}$	$\frac{3}{32}$	$\frac{9}{32}$				
$\frac{12}{13}$	$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$			
1	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		
$\frac{1}{2}$	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	
	$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	0
	$\frac{16}{135}$	0	$\frac{6656}{12825}$	$\frac{28561}{56430}$	$-\frac{9}{50}$	$\frac{2}{55}$

Pierwszy zestaw wag daje rozwiązanie rzędu czwartego, drugi — rzędu piątego.

Metoda Casha-Karpa — 5(4)

0						
$\frac{1}{5}$	$\frac{1}{5}$					
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$				
$\frac{3}{5}$	$\frac{3}{10}$	$-\frac{9}{10}$	$\frac{6}{5}$			
1	$-\frac{11}{54}$	$\frac{5}{2}$	$-\frac{70}{27}$	$\frac{35}{27}$		
$\frac{7}{8}$	$\frac{1631}{55296}$	$\frac{175}{512}$	$\frac{575}{13824}$	$\frac{44275}{110592}$	$\frac{293}{4096}$	
	$\frac{37}{378}$	0	$\frac{250}{621}$	$\frac{125}{594}$	0	$\frac{512}{1771}$
	$\frac{2825}{27648}$	0	$\frac{18575}{48384}$	$\frac{13525}{55296}$	$\frac{277}{14336}$	$\frac{1}{4}$

Pierwszy zestaw wag daje rozwiązanie rzędu piątego, drugi — rzędu czwartego.
Mniejsze błędy, niż Fehlberg.

Metoda Dormanda-Prince'a — 5(4), FSAL

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$
	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0

Pierwszy zestaw wag daje rozwiązanie rzędu czwartego, drugi — rzędu piątego. Taka sama złożoność, jak Cash-Karp (sześć obliczeń funkcji na krok), ale mniejsze błędy.

State of the art

Zagnieżdżone metody Rungego-Kutty wysokiego rzędu, typu metoda Dormanda-Prince'a, pozwalające na adaptacyjną zmianę kroku, kontrolujące stabilność i wyprowadzające wyniki ze stałym krokiem, stanowią obecnie *state of the art* w numerycznym rozwiązywaniu zagadnień początkowych dla równań nie-sztywnych.

Sprawdzanie stabilności na ogół odbywa się “automatycznie” — narastający błąd jest oznaką niestabilności, a metoda i tak zmniejsza krok, gdy błąd jest za duży.

Pewien wariant metody punktu środkowego

Przypuśćmy, iż rozwiązując pewien problem Cauchy'ego decydujemy się wyprowadzać wyniki z krokiem H , krok ten jednak jest za duży do obliczeń (powodowałby zbyt duży błąd). Wykonujemy więc m małych kroków o długości

$$h = H/m . \quad (11)$$

Stosujemy metodę

$$\mathbf{z}_0 \equiv \mathbf{y}_n, \quad (12a)$$

$$\mathbf{z}_1 = \mathbf{z}_0 + hf(x_n, \mathbf{z}_0) \quad (\text{Euler}) \quad (12b)$$

dla $j = 2, 3, \dots, m$

$$\mathbf{z}_j = \mathbf{z}_{j-2} + 2hf(x_n + (j-1)h, \mathbf{z}_{j-1}) \quad (12c)$$

(pochodna w punkcie środkowym)

$$\mathbf{y}(x_n + H) \simeq \mathbf{y}_{n+1} = \frac{1}{2} (\mathbf{z}_m + \mathbf{z}_{m-1} + hf(x_n + H, \mathbf{z}_m)) \quad (12d)$$

Metoda ta wymaga $m + 1$ obliczeń prawej strony równania na kroku o długości H . Można pokazać, iż *błąd* tej metody zawiera *tylko parzyste* potęgi h :

$$y(x_n + H) - y_{n+1} = \sum_{j=1}^{\infty} a_j h^{2j}. \quad (13)$$

W szczególności, jeśli m jest parzyste i zastosujemy metodę (12) dwa razy, z m i $m/2$ kroków, dostaniemy

$$y(x_n + H) \simeq \frac{4y_m - y_{m/2}}{3} + O(h^5) \quad (14)$$

a więc jest to metoda rzędu czwartego. ($y_{m/2}$ oznacza końcowy wynik po zastosowaniu $m/2$ kroków o długości $2H/m$, nie zaś połowę kroków o długości H/m .) Wyrażenia (13) i (14) bardzo przypominają analogiczne wyrażenia dla całkowania metodami ekstrapolacji Richardsona i Romberga, a skoro tak, narzuca się zastosowanie jakiegoś „ekstrapolacyjnego” algorytmu całkowania ODE.

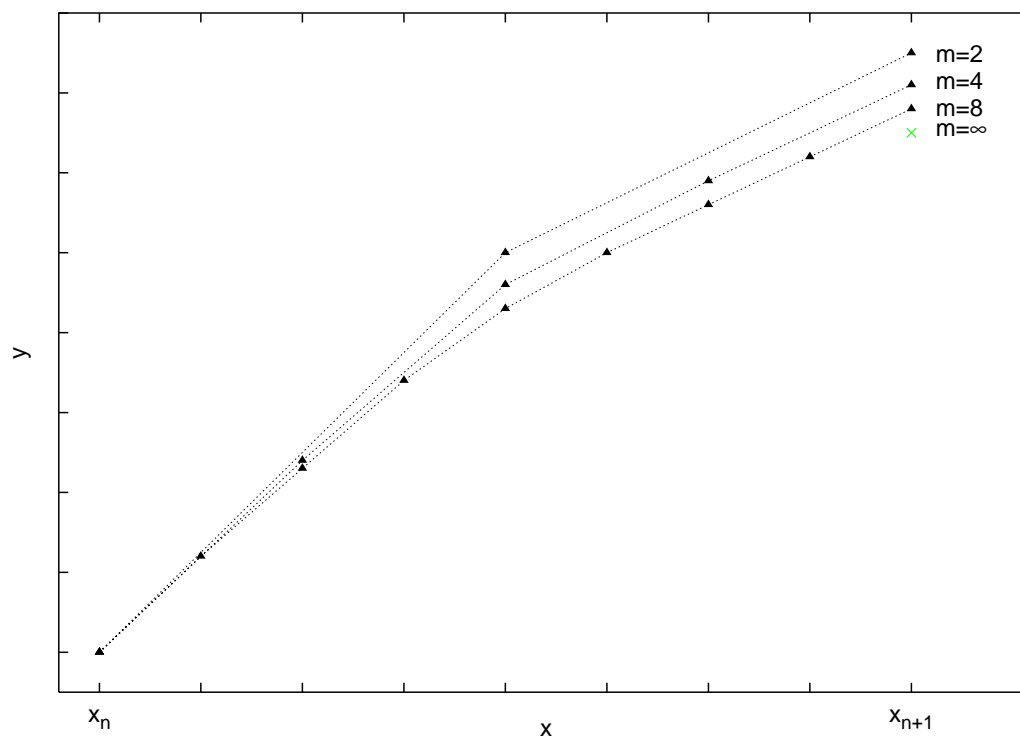
Metoda Bulirscha-Stoera

Przypuśćmy, iż przechodzimy przedział $[x_n, x_n + H]$ przy użyciu metody (12) kolejno z krokami $h_m = H/m$ dla $m = 2, 4, 8, \dots$. Otrzymujemy w ten sposób *ciąg kolejnych przybliżeń* wartości $y(x_n + H)$. Jeśli tak otrzymany ciąg *ekstrapolujemy* do nieskończonej liczby kroków pośrednich — lub też, w innym sformułowaniu, *do* $h_m \rightarrow 0$ — oszacujemy jak powinno wyglądać numeryczne rozwiązanie badanego ODE w granicy infinitezymalnie małych kroków.

Możemy użyć ekstrapolacji wielomianowej lub poprzez funkcje wymierne. Cała ta procedura jest dość (ale nie przesadnie) kosztowna i warto ją stosować, gdy

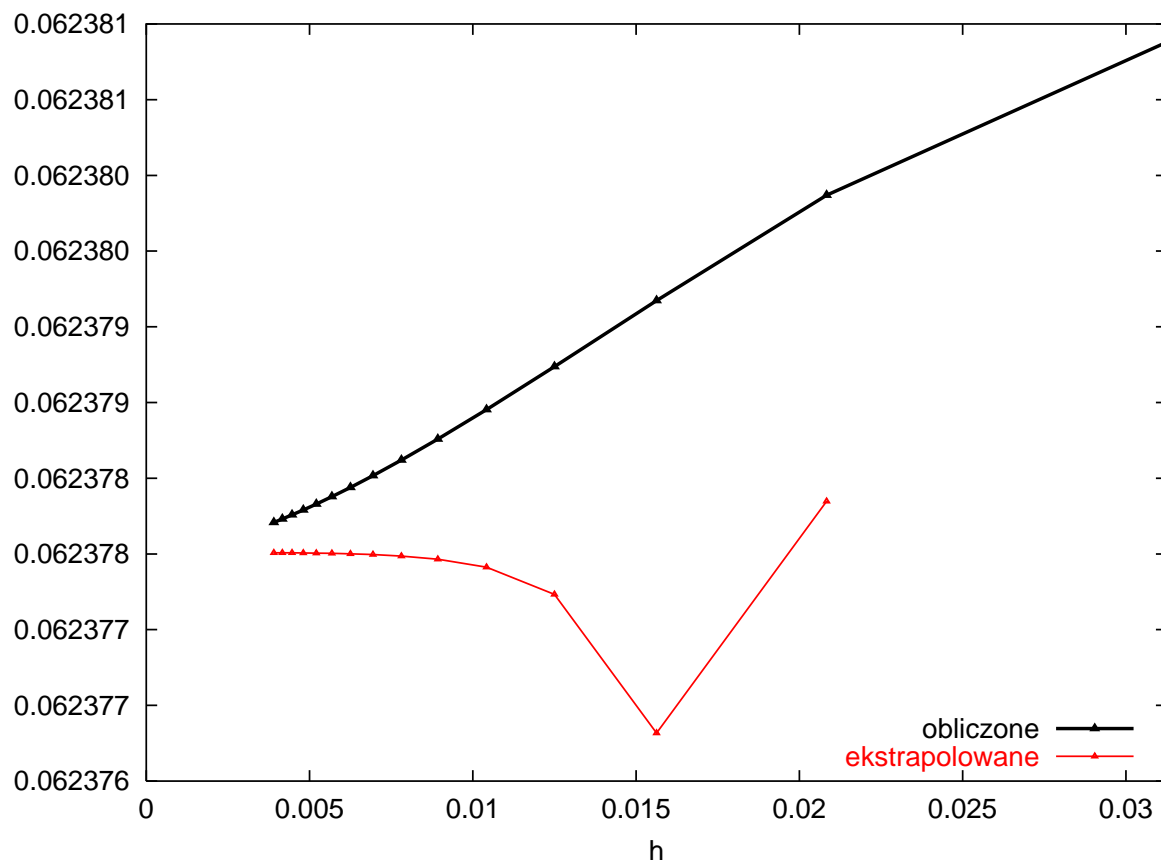
- zależy nam na szczególnie dokładnych rozwiązaniach,
- prawa strona równania zmienia się powoli.

Podobnie jak w ekstrapolacji Richardsona, w zasadzie podejście to można stosować tylko gdy ciąg kolejnych przybliżeń jest monotoniczny. Proces dzielenia odcinka kończymy gdy błąd ekstrapolacji jest mniejszy od zadanej tolerancji.

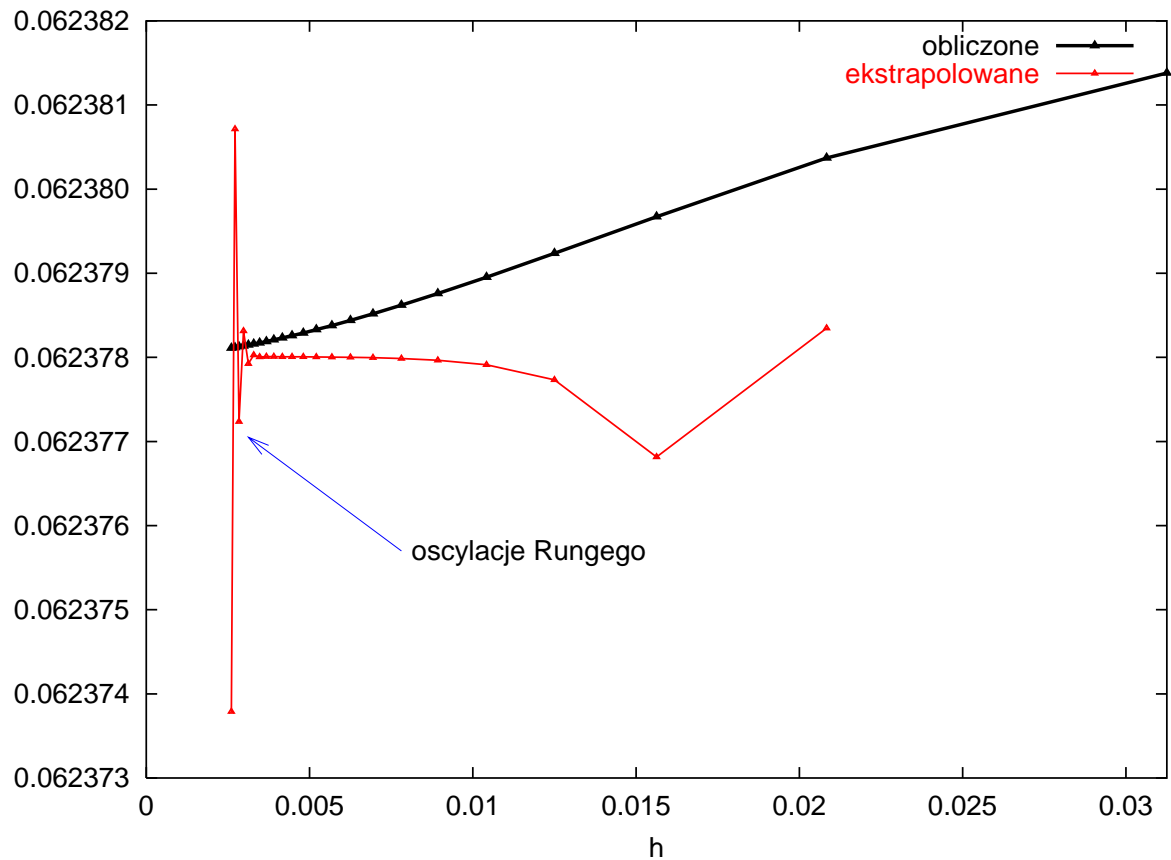


$\log_2 m$	obliczenie	ekstrapolacja
2	0.062381380134158664	
3	0.062380369335862672	0.062378347739270680
4	0.062379672556098376	0.062376816694340424
5	0.062379237240341216	0.062377733233895008
6	0.062378953354256184	0.062377912906545856
7	0.062378758903003136	0.062377965285744048
8	0.062378619973785528	0.062377985927438168
9	0.062378517198083888	0.062377995625316072
10	0.062378438952554864	0.062378000712959168
11	0.062378377939856608	0.062378003602874432
12	0.062378329395298192	0.062378005345019248
13	0.062378290100963920	0.062378006454798296
14	0.062378257819309512	0.062378007179444864
15	0.062378230954631280	0.062378007643411944
16	0.062378208343525576	0.062378007961218176

Graficzna ilustracja powyższych danych:

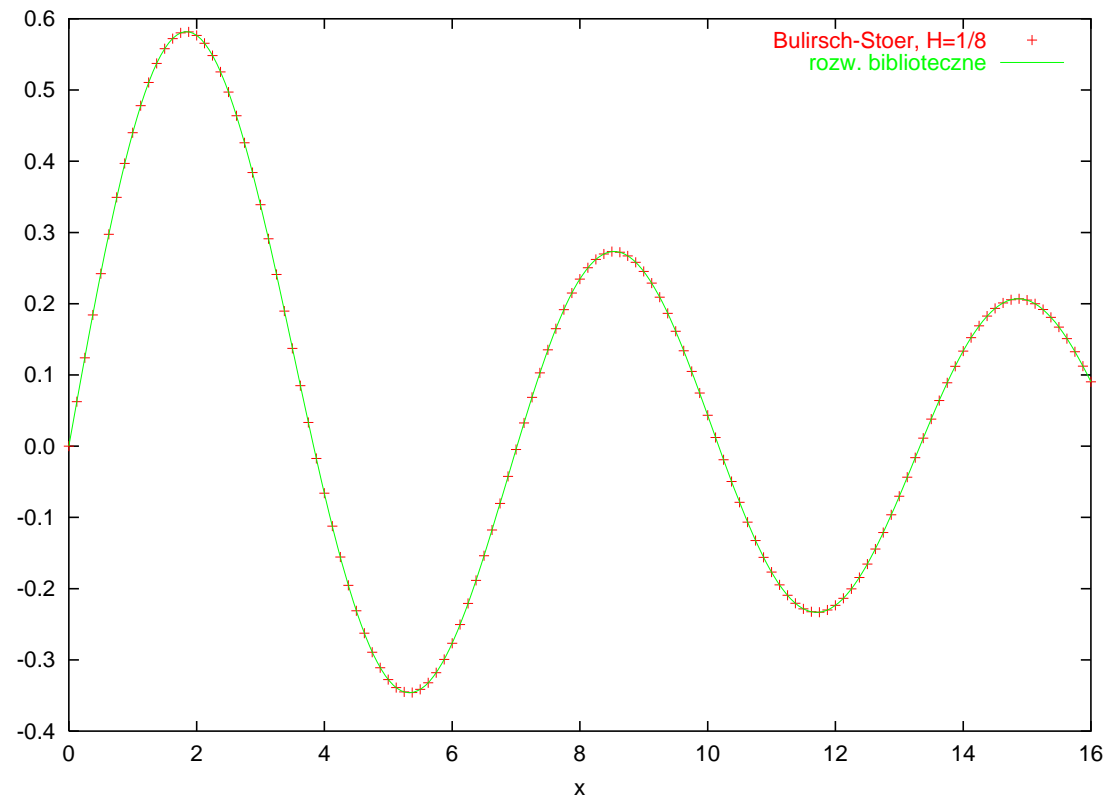


Zbyt duży stopień wielomianu ekstrapolacyjnego:



Ekstrapolacja funkcjami wymiernymi byłaby lepsza.

Przykład zastosowania metody Bulirscha-Stoera:



$$x^2 \frac{d^2 y}{dx^2} + x \frac{dy}{dx} + (x^2 - 1)y = 0$$

Odpowiednik metody Bulirscha-Stoera dla układów sztywnych

Jeśli spodziewamy się problemów ze stabilnością, używamy metod niejawnych, czasami jednak dokonujemy *pseudolinearyzacji* celem uproszczenia obliczeń. Punktem wyjścia jest następująca metoda niejawna:

$$\begin{aligned} \mathbf{y}_{n+1} - \mathbf{y}_{n-1} &= 2h\mathbf{f}\left(x_n, \frac{\mathbf{y}_{n+1} + \mathbf{y}_{n-1}}{2}\right) \\ &= 2h\mathbf{f}\left(x_n, \mathbf{y}_n + \underbrace{\frac{\mathbf{y}_{n+1} + \mathbf{y}_{n-1}}{2} - \mathbf{y}_n}_{\text{poprawka}}\right). \end{aligned} \quad (15)$$

Rozwijam prawą stronę (15) wokół (x_n, y_n) i porządkuję wyrazy:

$$\left[\mathbb{I} - h \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big|_{x_n, y_n} \right] \mathbf{y}_{n+1} = \left[\mathbb{I} + h \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big|_{x_n, y_n} \right] \mathbf{y}_{n-1} + 2h \left[\mathbf{f}(x_n, y_n) - h \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big|_{x_n, y_n} \mathbf{y}_n \right]. \quad (16)$$

Powyższe wyrażenie jest *liniowym* (w odróżnieniu od nieliniowego wyrażenia (15)) równaniem na nieznaną wielkość \mathbf{y}_{n+1} . Opierając się na (16), można skonstruować odpowiednik metody (12). Będą w nim występowały Jakobiany $\partial \mathbf{f} / \partial \mathbf{y}$ obliczane w kolejnych punktach pośrednich. **Dokonuję następnego uproszczenia: wszystkie Jakobiany wyliczam w lewym krańcu „dużego” przedziału.** Ostatecznie dostaję* ($h = H/m$):

*Dziękuję panu Piotrowi Kiełkowiczowi za wskazanie błędu.

$$\mathbf{H} = \left[\mathbb{I} - h \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big|_{x_n, \mathbf{y}_n} \right], \quad (17a)$$

$$\mathbf{z}_0 = \mathbf{y}_n, \quad (17b)$$

$$\Delta_0 = \mathbf{H}^{-1} h \mathbf{f}(x_n, \mathbf{z}_0), \quad (17c)$$

$$\mathbf{z}_1 = \mathbf{z}_0 + \Delta_0, \quad (17d)$$

$$\Delta_j = \Delta_{j-1} + 2\mathbf{H}^{-1} [h\mathbf{f}(x_n + jh, \mathbf{z}_j) - \Delta_{j-1}], \quad (17e)$$

$$\mathbf{z}_{j+1} = \mathbf{z}_j + \Delta_j, \quad (17f)$$

dla $j = 1, 2, \dots, m - 1$

$$\Delta_m = \mathbf{H}^{-1} [h\mathbf{f}(x_n + H, \mathbf{z}_m) - \Delta_{m-1}], \quad (17g)$$

$$\mathbf{y}(x_n + H) \simeq \mathbf{z}_m + \Delta_m. \quad (17h)$$

Równania $\mathbf{H}\Delta = \bullet$ można rozwiązać na przykład metodą rozkładu LU . Ponieważ zgodnie z przyjętym uproszczeniem macierz \mathbf{H} jest taka sama we wszystkich krokach iteracji (17), rozkładu tego można dokonać **tylko raz**.

Dalej postępuję tak, jak poprzednio: Zagęszczam podział i stosuję ekstrapolację do $h \rightarrow 0$.