Rozwiązanie problemu początkowego z automatyczną kontrolą kroku czasowego. *

17 listopada 2008

Równanie różniczkowe pierwszego rzędu

$$\frac{dy}{dt} = f(t, y), \tag{1}$$

z warunkiem początkowym $y(t = 0) = y_0$, rozwiązujemy przy pomocy wybranego jednokrokowego schematu różnicowego, który pozwala na wyliczenie $y(t + \Delta t)$, jeśli znamy $y(t)$

$$y(t + \Delta t) = y(t) + \Delta t F(t, y(t), \Delta t, f) + O(\Delta t^n), \tag{2}$$

gdzie ostatni wyraz oznacza błąd lokalny rzędu n (dla jawnego Eulera $n = 2$, dla schematu trapezów $n = 3$, dla metody RK4 $n = 5$). Wykorzystamy znajomość n do 1) podniesienia dokładności schematu 2) do opracowania algorytmu automatycznego doboru kroku czasowego. Obydwa zabiegi wykorzystują porównanie wyniku uzyskanego w pojedynczym i długim kroku $2\Delta t$, z dokładniejszym wynikiem uzyskanym w dwóch krótszych krokach Δt.

Rozwiązanie dokładne w chwili $t+2\Delta t$, różni się od y_1 numerycznego uzyskanego z krokiem $2\Delta t$ o błąd lokalny

$$y(t + 2\Delta t) = y_1 + O([2\Delta t]^n) = y_1 + C(2\Delta t)^n + O([\Delta t]^{n+1}). \tag{3}$$

Bardziej precyzyjne przybliżenie uzyskamy w dwóch krótszych krokach: licząc z krokiem Δt najpierw $y(t + \Delta t)$ a następnie $y_2 = y(t + 2\Delta t)$ popełniamy dwukrotnie błąd $C(\Delta t)^n$

$$y(t + 2\Delta t) = y_2 + 2C(\Delta t)^n + O([\Delta t]^{n+1}).^1 \tag{4}$$

Odejmując stronami (3) i (4) wyliczymy stałą błędu

$$C = \frac{y_2 - y_1}{2(2^{n-1} - 1)\Delta t^n}. \tag{5}$$

*Laboratorium z inżynierskich metod numerycznych, Wydział Fizyki i Informatyki Stosowanej AGH 2008/2009. bszafran@agh.edu.pl

[1] Stosujemy tutaj przybliżenie zakładające, że stała błędu C jest identyczna w obydwu krokach - patrz wykład.

Zgodnie z (5) szacujemy, że błąd obcięcia wykonany w dwóch krokach Δt wyniósł

$$E = 2C(\Delta t)^n = \frac{y_2 - y_1}{2^{n-1} - 1}. \tag{6}$$

Poprawiając y_2 z równania (4) dostaniemy przybliżenie poprawione

$$y_2' = y_2 + \frac{y_2 - y_1}{2^{n-1} - 1} + O(\Delta t^{n+1}). \tag{7}$$

Wartość y_2' wykorzystujemy w rachunkach dla późniejszych chwil czasowych. Równanie (7) daje nam schemat różnicowy o podniesionym o jeden rzędzie błędu lokalnego. Zabieg szacowania błędu przy pomocy rachunków z różnymi krokami całkowania nazywany jest ekstrapolacją Richardsona.[2]

Zadanie 1. Wzór (7) znacząco poprawia precyzję rozwiązania numerycznego. Weźmy $f(t, y) = y(t)$ i warunek brzegowy $y(0) = 1$ [dokładne rozwiązanie $y(t) = \exp(t)$]. Przyjmijmy $\Delta t = 1/32$, i rozwiążmy numerycznie równanie (1) przy pomocy schematu Eulera ($n = 2$) i RK2 ($n = 3$) dla $t \in [0, 5/4]$. Policzyć i narysować różnicę między rozwiązaniem numerycznym i analitycznym w funkcji t. **1.1:10 pkt**. Następnie wprowadzić eliminację błędu przez ekstrapolację Richardsona (7). Powtórzyć rachunek dla błędu globalnego i narysować go w funkcji t dla obydwu metod. **1.2:20 pkt**.

Zadanie 2. Ekstrapolacji Richardsona można użyć również do kontroli (zmiany w trakcie obliczeń) kroku czasowego Δt, tak żeby błędy obcięcia nie przekraczały pewnej zadanej wartości *tol*. (Uwaga: w tym zadaniu **nie** eliminujemy błędu, to jest **nie** stosujemy wzoru (7). Błąd tylko monitorujemy wg wzoru (6)).

W trakcie obliczeń sprawdzamy na bieżąco oszacowanie błędu $E = \frac{y_2 - y_1}{2^{n-1} - 1}$. Jeśli $E < tol$ rozwiązanie $y(t + \Delta 2t)$ akceptujemy, ale zwiększamy krok czasowy do dalszego całkowania. Jeśli natomiast $E > tol$ zmniejszamy Δt i powtarzamy obliczenia. Zmianę kroku czasowego w obydwu przypadkach obsłuży nam wzór

$$\Delta t_{\text{nowy}} = \left(\frac{S \times tol}{|E|} \right)^{1/n} \Delta t, \tag{8}$$

gdzie czynnik $S < 1$ wprowadzamy dla bezpieczeństwa.

Jako przykład weźmy oscylator harmoniczny, którego położenie i prędkość dane są układem równań

$$\begin{aligned} \frac{dx}{dt} &= V(t) \\ \frac{dV}{dt} &= -x(t). \end{aligned} \tag{9}$$

Rozwiążemy równanie z warunkiem początkowym $V(0) = 0$ oraz $x(0) = 1$, przy pomocy metody RK2. Będziemy sprawdzać błędy obcięcia dla prędkości

[2] ekstrapolujemy oszacowanie błędu do $\Delta t = 0$.

i położenia stosując dla obydwu wielkości tą samą tolerancję *tol*. W każdej iteracji krok czasowy przyjmiemy jako mniejszy z kroków Δt danych przez wzór (8). Przyjąć $S = 0.75$. Rachunki przeprowadzimy dla czterech okresów wahadła [$t \in (0, 8\pi)$]. Narysować krzywą $V(x)$ dla tolerancji błędu obcięcia $tol = 0.1, 0.01, 0.001$ oraz 0.0001. **2.1:20 pkt**. Sprawdzić zależność energii $E(t) = \frac{1}{2}\left(x^2(t) + V^2(t)\right)$ dla każdej tolerancji **2.2:10 pkt**. Dla $tol = 0.0001$ zbadać jak algorytm ustawia krok czasowy w zależności od x^2 i V^2 **2.3:10 pkt**.

Zadanie 3. Zakodować RK4 (wersja klasyczna) ($n = 5$) z kontrolowanym krokiem czasowym i przetestować go w problemie z Zadania 2. Narysować: $V(x)$ **3.1:10 pkt** dla $tol = 1e-1, 1e-2, 1e-3, 1e-4, 1e-5, 1e-6$. Porównać kroki czasowe dane przez RK2 i RK4 **3.2:10 pkt** oraz $E(t)$ **3.3:10 pkt** przy tej samej tolerancji błędu obcięcia.

RK2 dla układu równań

$$\frac{du^1}{dt} = f^1(t, u^1, u^2) \tag{10}$$

$$\frac{du^2}{dt} = f^2(t, u^1, u^2). \tag{11}$$

Liczymy kolejno:

$$k_1^1 = f^1(t_{n-1}, u_{n-1}^1, u_{n-1}^2) \tag{12}$$

$$k_1^2 = f^2(t_{n-1}, u_{n-1}^1, u_{n-1}^2) \tag{13}$$

następnie

$$k_2^1 = f^1(t_{n-1} + \Delta t, u_{n-1}^1 + \Delta t k_1^1, u_{n-1}^2 + \Delta t k_1^2) \tag{14}$$

$$k_2^2 = f^2(t_{n-1} + \Delta t, u_{n-1}^1 + \Delta t k_1^1, u_{n-1}^2 + \Delta t k_1^2) \tag{15}$$

i ostatecznie:

$$u_n^1 = u_{n-1}^1 + \frac{\Delta t}{2} k_1^1 + \frac{\Delta t}{2} k_2^1 \tag{16}$$

$$u_n^2 = u_{n-1}^2 + \frac{\Delta t}{2} k_1^2 + \frac{\Delta t}{2} k_2^2 \tag{17}$$

RK4 dla układu równań

$$\frac{du^1}{dt} = f^1(t, u^1, u^2) \tag{18}$$

$$\frac{du^2}{dt} = f^2(t, u^1, u^2) \tag{19}$$

Liczymy kolejno dla $i = 1, 2$

$$k_1^i = f^i(t_{n-1}, u_{n-1}^1, u_{n-1}^2) \tag{20}$$

następnie dla $i = 1, 2$

$$k_2^i = f^i(t_{n-1} + \frac{\Delta t}{2}, u_{n-1}^1 + \frac{\Delta t k_1^1}{2}, u_{n-1}^2 + \frac{\Delta t k_1^2}{2}) \tag{21}$$

potem

$$k_3^i = f^i(t_{n-1} + \frac{\Delta t}{2}, u_{n-1}^1 + \frac{\Delta t k_2^1}{2}, u_{n-1}^2 + \frac{\Delta t k_2^2}{2}) \tag{22}$$

później

$$k_4^i = f^i(t_{n-1} + \Delta t, u_{n-1}^1 + \Delta t k_3^1, u_{n-1}^2 + \Delta t k_3^2) \tag{23}$$

w końcu

$$u_n^i = u_{n-1}^i + \frac{\Delta t}{6}\left(k_1^i + 2k_2^i + 2k_3^i + k_4^i\right). \tag{24}$$