Projekt 3: RRZ - kontrola kroku czasowego w problemach sztywnych.

Tomasz Chwiej

4 listopada 2020

1 Wstęp

Na zajęciach rozwiążemy równanie różniczkowe 2 rzędu oscylatora Van der Pola

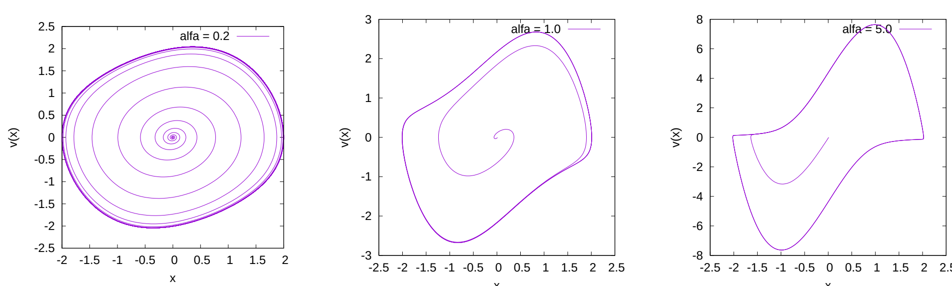
$$\frac{d^2x}{dt^2} - \alpha(1-x^2)\frac{dx}{dt} + x = 0 \tag{1}$$

RRZ 2 rzędu zamieniamy na układ 2 RRZ 1 rzędu (wprowadzając nową zmienną $dx/dt = v$):

$$\frac{dx}{dt} = f(t,x,v) = v \tag{2}$$

$$\frac{dv}{dt} = g(t,x,v) = \alpha(1-x^2)v - x \tag{3}$$

Taki układ RRZ 1 rzędu rozwiążemy stosując metodę jawną RK2 i metodę niejawną trapezów.
Dla $\alpha = 0$ problem sprowadza się do problemu oscylatora harmonicznego, natomiast dla $\alpha >> 1$ problem staje się sztywny. Z tego powodu problem rozwiążemy kontrolując krok czasowy.



Rysunek 1: Rozwiązanie w przestrzeni fazowej $V(x)$ dla parametru $\alpha = 0.2;\ 1; 5$. Warunki początkowe: $x = 0.01$ i $V = 0$. Niezależnie od WP, rozwiązanie zawsze będzie dążyć do ustalonej trajektorii (istnieje atraktor).

2 Kontrola kroku czasowego

Jeśli znamy rozwiązanie w chwili t_n to rozwiązanie w chwili t_{n+2} możemy uzyskać na dwa sposoby. Pierwszy sposób to oczywiście wykonanie jednego długiego kroku o długości $2\Delta t$ które daje nam rozwiązanie $x_{n+2}^{(1)}$ różniące od dokładnego o pewien błąd lokalny:

$$x_{dok}(t_{n+2}) = x_{n+2}^{(1)} + C_x(2\Delta t)^{p+1} + O(\Delta t^{p+2}) \tag{4}$$

gdzie: p oznacza rząd metody (dla RK2 $p = 2$, dla trapezów $p = 2$). Możemy też wykonać dwa krótsze kroki co Δt i otrzymać lepsze rozwiązanie $x_{n+2}^{(2)}$ co da nam inny błąd lokalny:

$$x_{dok}(t_{n+2}) = x_{n+2}^{(2)} + 2C_x(\Delta t)^{p+1} + O(\Delta t^{p+2}) \tag{5}$$

Stałą błędu C_x wyznaczamy odejmując od (5) równanie (4). Następnie możemy określić błąd rozwiązania uzyskanego w dwóch krokach (dla x i V):

$$E_x = 2C_x \Delta t^{p+1} = \frac{x_{n+2}^{(2)} - x_{n+2}^{(1)}}{2^p - 1} \tag{6}$$

$$E_v = 2C_v \Delta t^{p+1} = \frac{v_{n+2}^{(2)} - v_{n+2}^{(1)}}{2^p - 1} \tag{7}$$

i wykorzystać go do modyfikacji kroku czasowego:

$$\Delta t_{nowy} = \left(\frac{S \cdot tol}{\max\left(|E_x|, |E_v|\right)} \right)^{\frac{1}{p+1}} \cdot \Delta t_{stary} \tag{8}$$

gdzie funkcja $\max(a, b)$ zwraca większą z dwóch porównywanych liczb.
Ogólny algorytm numerycznego rozwiązywania równania różniczkowego z doborem kroku czasowego:

inicjalizacja: t=0, $\Delta t = \Delta t_0$, $x_n = x_0$, $v_n = v_0$, t_{max}
DO
    //stawiamy dwa kroki Δt
        $(x_{n+1}^{(2)}, v_{n+1}^{(2)}) \leftarrow schemat\_numeryczny(x_n, v_n, \Delta t, \alpha)$
        $(x_{n+2}^{(2)}, v_{n+2}^{(2)}) \leftarrow schemat\_numeryczny(x_{n+1}^{(2)}, v_{n+1}^{(2)}, \Delta t, \alpha)$

    //stawiamy jeden krok $2 \cdot \Delta t$
        $(x_{n+2}^{(1)}, v_{n+2}^{(1)}) \leftarrow schemat\_numeryczny(x_n, v_n, 2 \cdot \Delta t, \alpha)$

    liczymy: E_x, E_v

    //sprawdzamy czy wynik jest akceptowany
    IF($\max(|E_x|, |E_v|)$<TOL)THEN
        $t = t + 2 \cdot \Delta t$
        $x_n = x_{n+2}^{(2)}$
        $v_n = v_{n+2}^{(2)}$
        zapis danych do pliku: t, Δt, x_n, v_n
    END IF

    //zmiana kroku następuje zawsze
        $\Delta t = \left(\frac{S \cdot TOL}{max(|E_x|, |E_v|)} \right)^{\frac{1}{p+1}} \cdot \Delta t$
ENDDO WHILE(t<t_{max})

2.1 schemat_numeryczny: metoda trapezów

Chcemy wyznaczyć 2 rozwiązania tj. x_{n+1} i v_{n+1} dla t_{n+1}:

$$x_{n+1} = x_n + \frac{\Delta t}{2}[f(x_n, v_n) + f(x_{n+1}, v_{n+1})] \tag{9}$$

$$v_{n+1} = v_n + \frac{\Delta t}{2}[g(x_n, v_n) + g(x_{n+1}, v_{n+1})] \tag{10}$$

Problem jest dwuwymiarowy więc definiujemy dwie funkcje nieliniowe:

$$F = x_{n+1} - x_n - \frac{\Delta t}{2}[f(x_n, v_n) + f(x_{n+1}, v_{n+1})] \tag{11}$$

$$G = v_{n+1} - v_n - \frac{\Delta t}{2}[g(x_n, v_n) + g(x_{n+1}, v_{n+1})] \tag{12}$$

Rozwijając funkcje $F(x_{n+1}+\Delta x, v_{n+1}+\Delta v)$ i $G(x_{n+1}+\Delta x, v_{n+1}+\Delta v)$ w szereg Taylora z dokładnością do wyrazów liniowych uzyskamy:

$$F(x_{n+1} + \Delta x, v_{n+1} + \Delta v) = F(x_{n+1}, v_{n+1}) + \frac{\partial F}{\partial x}\Delta x + \frac{\partial F}{\partial v}\Delta v \tag{13}$$

$$G(x_{n+1} + \Delta x, v_{n+1} + \Delta v) = G(x_{n+1}, v_{n+1}) + \frac{\partial G}{\partial x}\Delta x + \frac{\partial G}{\partial v}\Delta v \tag{14}$$

Z założenia metody Newtona wynika że $F(x_{n+1}+\Delta x, v_{n+1}+\Delta v) = 0$ oraz $G(x_{n+1}+\Delta x, v_{n+1}+\Delta v) = 0$, więc rozwinięcie możemy zapisać w postaci macierzowej

$$\begin{bmatrix} -F \\ -G \end{bmatrix} = \begin{bmatrix} \frac{\partial F}{\partial x_{n+1}} & \frac{\partial F}{\partial v_{n+1}} \\ \frac{\partial G}{\partial x_{n+1}} & \frac{\partial G}{\partial v_{n+1}} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta v \end{bmatrix} \tag{15}$$

Elementy macierzy układu A (górny indeks k to numer iteracji):

$$a_{11} = \frac{\partial F}{\partial x_{n+1}} = 1 \tag{16}$$

$$a_{12} = \frac{\partial F}{\partial v_{n+1}} = -\frac{\Delta t}{2} \tag{17}$$

$$a_{21} = \frac{\partial G}{\partial x_{n+1}} = -\frac{\Delta t}{2}(-2\alpha x_{n+1}^k v_{n+1}^k - 1) \tag{18}$$

$$a_{22} = \frac{\partial G}{\partial v_{n+1}} = 1 - \frac{\Delta t}{2}\alpha \left[1 - (x_{n+1}^k)^2\right] \tag{19}$$

$$\tag{20}$$

Rozwiązanie układu 2×2 znajdujemy metodą wyznacznikową:

$$\Delta x = \frac{(-F)a_{22} - (-G)a_{12}}{a_{11}a_{22} - a_{12}a_{21}} \tag{21}$$

$$\Delta v = \frac{a_{11}(-G) - a_{21}(-F)}{a_{11}a_{22} - a_{12}a_{21}} \tag{22}$$

a następnie poprawiamy rozwiązanie

$$x_{n+1}^{k+1} = x_{n+1}^k + \Delta x \tag{23}$$

$$v_{n+1}^{k+1} = v_{n+1}^k + \Delta v \tag{24}$$

Na starcie przyjmujemy: $x_{n+1}^0 = x_n$ i $v_{n+1}^0 = v_n$, a następnie iterację prowadzimy aż do spełnienia warunku: $|\Delta x| < \delta$ i $|\Delta v| < \delta$ dla $\delta = 10^{-10}$.

2.2 schemat_numeryczny: metoda RK2

Obliczenia prowadzimy sekwencyjnie

$$k_{1x} = f(t_n, x_n, v_n) = v_n \tag{25}$$
$$k_{1v} = g(t_n, x_n, v_n) = \alpha(1 - x_n^2)v_n - x_n \tag{26}$$

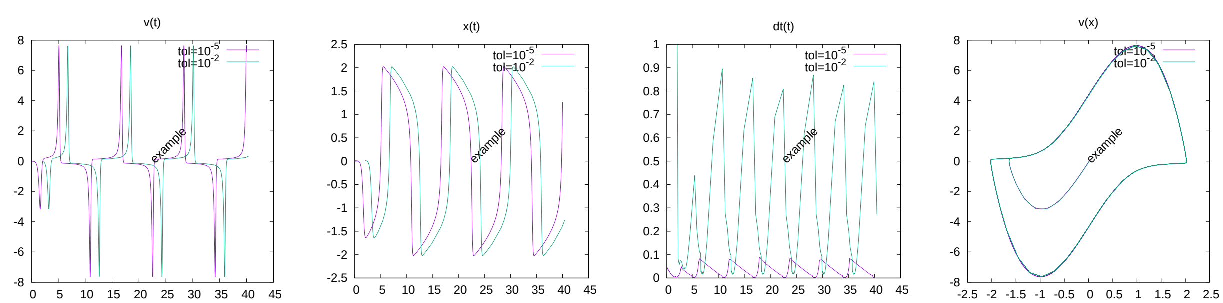
$$k_{2x} = f(t_n + \Delta t, x_n + \Delta t k_{1x}, v_n + \Delta t k_{1v}) = v_n + \Delta t k_{1v} \tag{27}$$
$$k_{2v} = g(t_n + \Delta t, x_n + \Delta t k_{1x}, v_n + \Delta t k_{1v}) \tag{28}$$
$$= \alpha \left[1 - (x_n + \Delta t k_{1x})^2\right] (v_n + \Delta t k_{1v}) - (x_n + \Delta t k_{1x}) \tag{29}$$

$$x_{n+1} = x_n + \frac{\Delta t}{2}(k_{1x} + k_{2x}) \tag{30}$$
$$v_{n+1} = v_n + \frac{\Delta t}{2}(k_{1v} + k_{2v}) \tag{31}$$

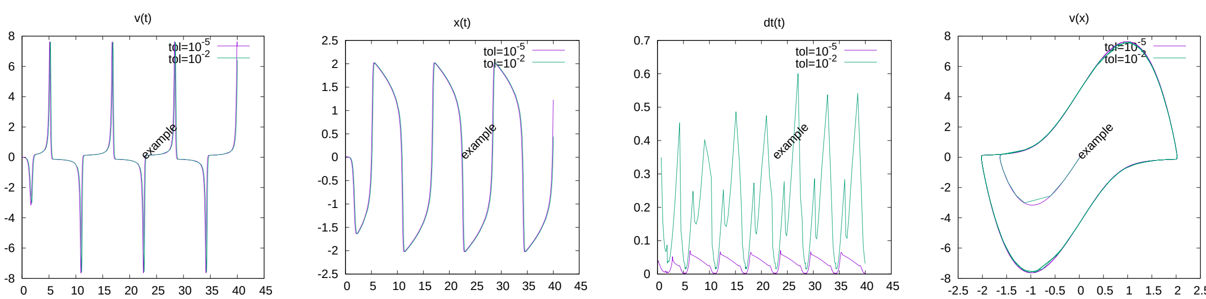
3 Zadania do wykonania

1. Zaprogramować metody: trapezów i RK2 jako dwie osobne procedury. Do procedur przekazujemy: x_n, v_n, $\Delta t, \alpha$; a mają zwracać: x_{n+1}, v_{n+1}.
2. Zaimplementować algorytm kontroli kroku czasowego.
3. Przyjąć parametry startowe: $x_0 = 0.01$, $v_0 = 0$, $\Delta t_0 = 1$, $S = 0.75$, $p = 2$ (rząd dokładności obu metod), $t_{max} = 40$, $\alpha = 5$.
4. Rozwiązać równanie (1) metodą RK2 z kontrolą kroku czasowego dla parametru $TOL = 10^{-2}; 10^{-5}$. Wykonać rysunki: $x(t)$, $v(t)$, $\Delta t(t)$ i $v(x)$. Wykresy tej samej wielkości dla obu wartości TOL umieścić na jednym rysunku (będą 4 rysunki). (50 pkt)
5. Rozwiązać równanie (1) metodą trapezów z kontrolą kroku czasowego dla parametru $TOL = 10^{-2}; 10^{-5}$. Wykonać rysunki: $x(t)$, $v(t)$, $\Delta t(t)$ i $v(x)$. Wykresy tej samej wielkości dla obu wartości TOL umieścić na jednym rysunku (będą 4 rysunki). (50 pkt)

4 Przykładowe wyniki



Rysunek 2: Wyniki dla metody trapezów



Rysunek 3: Wyniki dla metody RK2