Rozwiązywanie ODE w Matlabie

Sprawozdanie

Akademia Górniczo-Hutnicza im. Stanisława Staszica
Modelowanie Systemów Dynamicznych 2022
WEAIiIB, Automatyka i Robotyka

Data wykonania ćwiczenia:
30.11.2022 r.

Data oddania sprawozdania:
06.12.2022 r.

Jakub Górski

Grupa dziekańska nr 3

Spis Treści

1. Cel ćwiczeń

Tematem laboratorium jest rozwiązywanie zagadnień początkowych dla równań różniczkowych zwyczajnych przy użyciu narzędzi zwanych solverami.

2. Wstęp teoretyczny

Solver – metoda numeryczna rozwiązująca zagadnienie początkowe dla równań różniczkowych zwyczajnych. W momencie dostarczenia temu narzędziu warunków początkowych y_0 oraz przedziału czasu, dla którego szukana jest odpowiedź (t_0, t_f) algorytm (w sposób iteracyjny) wylicza rozwiązania. Solver jako rezultat przekazuje wektor chwil czasu, a także wektor rozwiązań dla wskazanych chwil.

Przykładowe solvery:

- ode45
- ode23
- ode113
- ode15s
- ode23tb

Składnia funkcji *ode*:

[*T*,*Y*] = *solver*(*odefun*, *tspan*, y_0, *options*),

gdzie:

solver – nazwa solvera,

odefun – uchwyt do funkcji określającej prawą stronę równania różniczkowego $y' = f(t, y)$,

tspan – zakres całkowania $[t_0, t_f]$,

y_0 – wektor warunków początkowych,

options – opcjonalne parametry.

Estymata błędu rozwiązania zaproponowanego przez dane *ode* jest określana na podstawie porównania rezultatów dwóch osobnych metod (np. *ode23* wyznacza błąd rozwiązania wyznaczonego przez algorytm Rungego–Kutty rzędu 2–go porównując wynik z wartościami, które można otrzymać stosując algorytm Rungego–Kutty rzędu 3–go). Na podstawie tego stwierdzenia można zauważyć, że im większe liczby towarzyszą nazwie solvera, tym narzędzie to jest dokładniejsze.

Schemat metody Eulera:

$y_{n+1} = y_n + hf(t_n, y_n)$, $(y' = f(t, y))$

$t_{n+1} = t_n + h$,

gdzie *h* to krok.

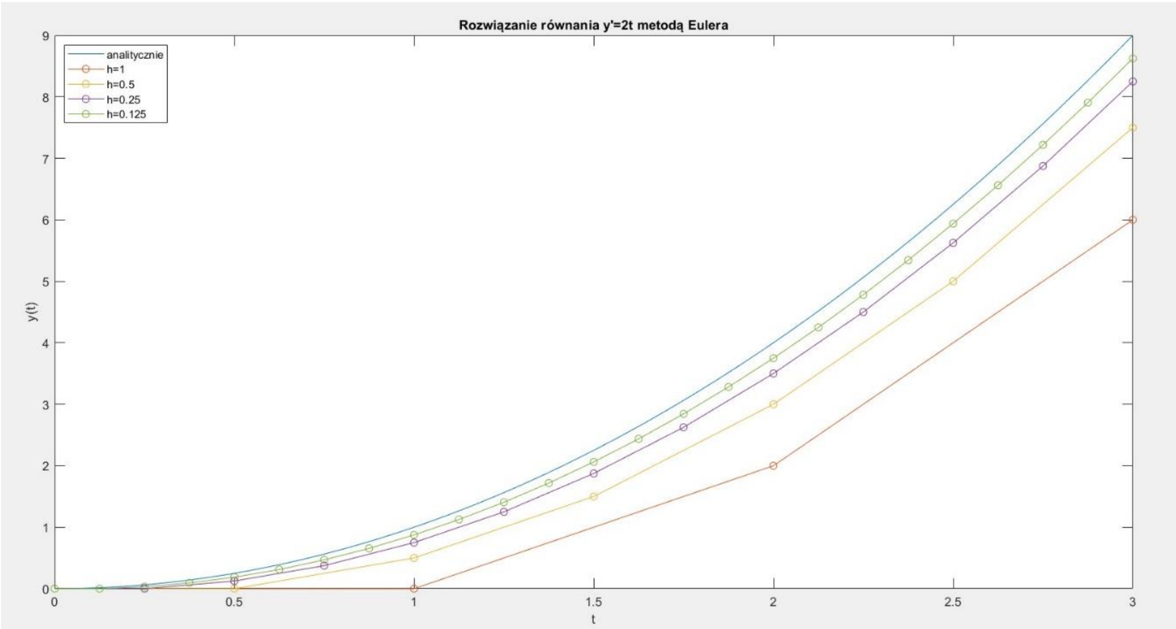
3. Wykonanie zadań

3.1. Zadanie 1 – Rozwiązywanie równań różniczkowych metodą Eulera

Ćwiczenie polega na znalezieniu rozwiązania równania $y' = 2t$ korzystając z metody Eulera na przedziale czasowym [0, 3], przy warunku początkowym $y_0 = 0$ oraz kroku h należącego do zbioru {1, 0.5, 0.25, 0.125}. Otrzymane wyniki porównano z analitycznym rozwiązaniem problemu ($y = t^2$).

```
%% zadanie 1 - metoda Eulera
clear all; close all;

figure;
fplot(@(t) t.^2, [0 3]);
hold on;

N = 3;

for h = [1 0.5 0.25 0.125]
    t = 0:h:N;
    y(1) = 0;
    for i = 1:N/h
        dy = 2*t(i);
        y(i+1) = y(i) + dy*h;
    end
    plot(t, y, '-o');
end

title('Rozwiązanie równania y''=2t metodą Eulera');
xlabel('t'); ylabel('y(t)');
legend('analitycznie', 'h=1', 'h=0.5', 'h=0.25', 'h=0.125', 'Location', 'northwest');
hold off;
```

Kod do zadania 1



Otrzymane wykresy w zadaniu 1

Jak można zauważyć, im większy krok, tym różnica między wartościami otrzymanymi, a oczekiwanym wynikiem jest coraz większa. Na wykresach oznaczono okręgami punkty wyznaczone przedstawioną metodą.

3.2. Wahadło – przykład

Równanie ruchu wahadła z tłumieniem:

$$\ddot{\theta} = -\frac{b}{m}\dot{\theta} - \frac{mg}{L(m-2b)} sin\theta,$$

gdzie:

Θ - kąt wychylenia wahadła,

b - współczynnik tłumienia,

m - masa,

g - przyspieszenie ziemskie,

L - długość linki.

Warunki początkowe:

$$\begin{cases} \theta = \frac{\pi}{4} \\ \dot{\theta} = 0 \end{cases}$$

Zmienne stanu:

$$\begin{cases} y_1 = \theta \\ y_2 = \dot{\theta} \end{cases}$$

Przestrzeń stanów:

$$\begin{cases} \dot{y_1} = y_2 \\ \dot{y_2} = -\frac{b}{m} y_2 - \frac{mg}{L(m-2b)} \sin{(y_1)} \end{cases}$$

W ramach zadania utworzono funkcje, której celem jest reprezentacja równań stanu.

```
function d2ydt2 = wahadlo(t, y)
    g = 9.8; % m/s^2
    m = 1;   % kg
    L = 2;   % m
    b = 0.2;
    d2ydt2 = [y(2); -b/m*y(2)-m*g/L/(m-2*b)*sin(y(1))];
end
```

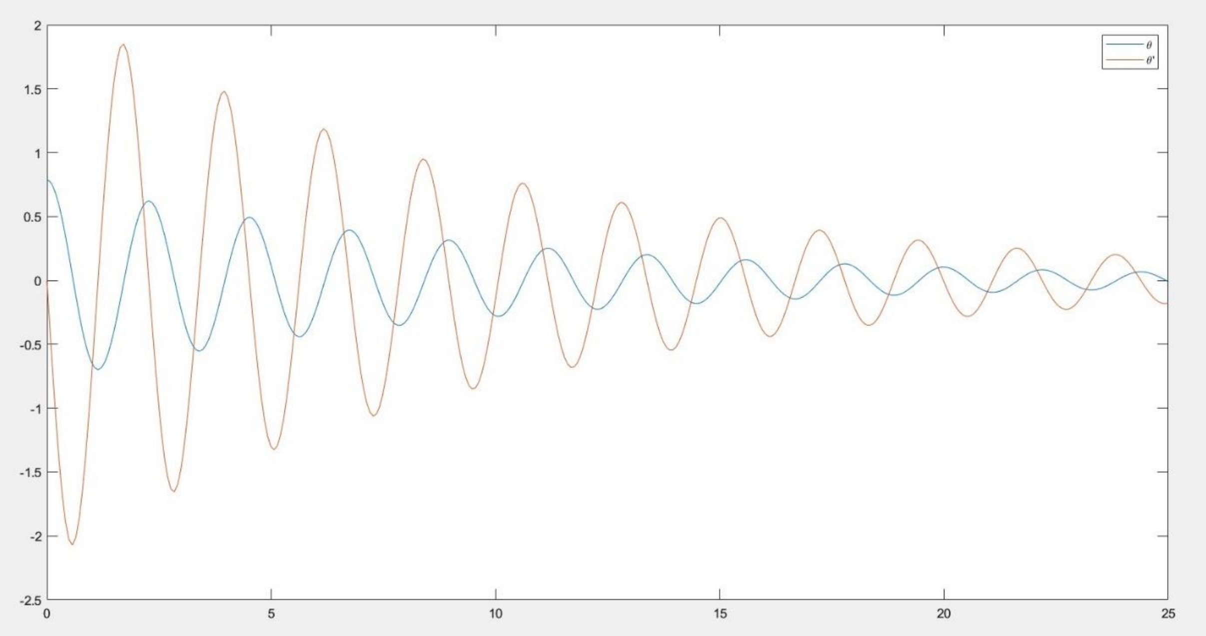
Funkcja *wahadlo*

Następnie wykorzystano solver *ode45* przy zadanych parametrach do rozwiązania przedstawionej sytuacji.

```
%% wahadło
clear all; close all;

opts = odeset('stats', 'on');
tspan = [0 25];
y0 = [pi/4, 0];

[t, y] = ode45(@wahadlo, tspan, y0, opts);

figure;
plot(t, y(:, 1), t, y(:, 2));
legend('\theta', '\theta''');
```

Kod do przykładu

Efektem końcowym jest zaprezentowany poniżej wykres.



Wykres położenia kątowego i prędkości kątowej wahadła w dziedzinie czasu

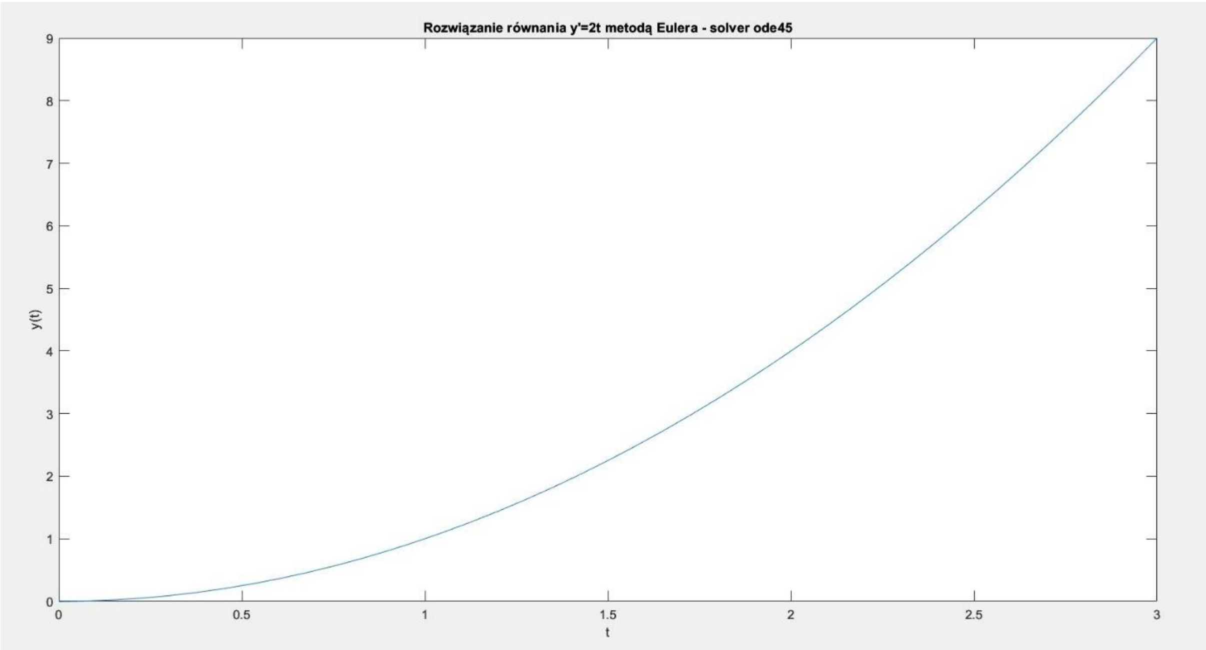
3.3. Zadanie 2 - Rozwiązywanie równań różniczkowych za pomocą solverów

Polecenie polega na ponownym rozwiązaniu równania różniczkowego $y' = 2t$, jednak przy tym stosując wiedzę dotyczącą solverów.

Program realizujący żądaną logikę prezentuje się następująco.

```
%% zadanie 2
clear all; close all;

N = 3;

opts = odeset('stats', 'on');
tspan = [0 N];
y0 = 0;
fun = @(t, y) 2*t;

[t, y] = ode45(fun, tspan, y0, opts);

figure;
plot(t, y);
title('Rozwiązanie równania y''=2t metodą Eulera - solver ode45');
xlabel('t'); ylabel('y(t)');
```

Kod do zadania 2



Wykres otrzymany w ramach programu przedstawionego powyżej

Porównując uzyskaną zależność z wynikiem analitycznego rozwiązania omawianego równania można stwierdzić, że różnica między rezultatami jest niewidoczna.

3.4. Zadanie 3 – Urządzenie do hamowania lądujących samolotów

Zadanie to skupia się na rozwiązaniu równania różniczkowego opisującego dynamikę urządzenia, które służy do hamowania lądującego samolotu.

Zmienne stanu:

$$\begin{cases} x_1 = x \\ x_2 = \dot{x} \\ x_3 = y_1 \\ x_4 = \dot{y_1} \\ x_5 = y_2 \\ x_6 = \dot{y_2} \end{cases}$$

Przestrzeń stanów:

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = -2f_{k1}sin\theta/m_1 \\ \dot{x_3} = x_4 \\ \dot{x_4} = (2f_{k1} - f_{k2})/m_2 \\ \dot{x_5} = x_6 \\ \dot{x_6} = (f_{k2} - f_b)/m_3 \end{cases}$$

Wszystkie potrzebne parametry i zależności zostały wzięte z poprzednich laboratoriów i zostały zaimplementowane w ramach zadanych funkcji.

```
function Dx = hamownik(t, x)
    h = 42;         %[m]
    m1 = 14000;     %[kg]
    m2 = 450.28;    %[kg]
    m3 = 200;       %[kg]
    k1 = 54700;     %[N/m]
    k2 = 303600;    %[N/m]

    % interpolacja
    wezlyF3 = [0 10 20 30 40 50 60 70 80 90 94 98 102 104 107 120];
    wartosciF3 = [833 400 160 320 520 520 660 830 1070 1600 2100 2800 4100 5000 9000 9000];
    funF3 = interp1(wezlyF3, wartosciF3, x(5), 'pchip');
    Fb = funF3*x(6)^2;

    % zmienne stanu
    y1 = sqrt(x(1)^2 + h^2) - h;
    sin_theta = x(1)/(h + y1);

    if y1 >= 2*x(3)
        Fk1 = k1*(y1 - 2*x(3));
    else
        Fk1 = 0;
    end

    if x(3) >= x(5)
        Fk2 = k2*(x(3) - x(5));
    else
        Fk2 = 0;
    end

    dx(1) = x(2);
    dx(2) = -2*Fk1*sin_theta/m1;
    dx(3) = x(4);
    dx(4) = (2*Fk1 - Fk2)/m2;
    dx(5) = x(6);
    dx(6) = (Fk2 - Fb)/m3;

    Dx = [dx(1); dx(2); dx(3); dx(4); dx(5); dx(6)];
end
```

Funkcja *hamownik*

Z uwagi na to, że przyspieszenie hamującego samolotu nie jest zmienną stanu, w celu ukazania zależności przyspieszenia od czasu należy utworzyć dodatkową funkcję, która będzie realizować zadanie wyliczenia przyspieszenia w danych chwilach czasowych.

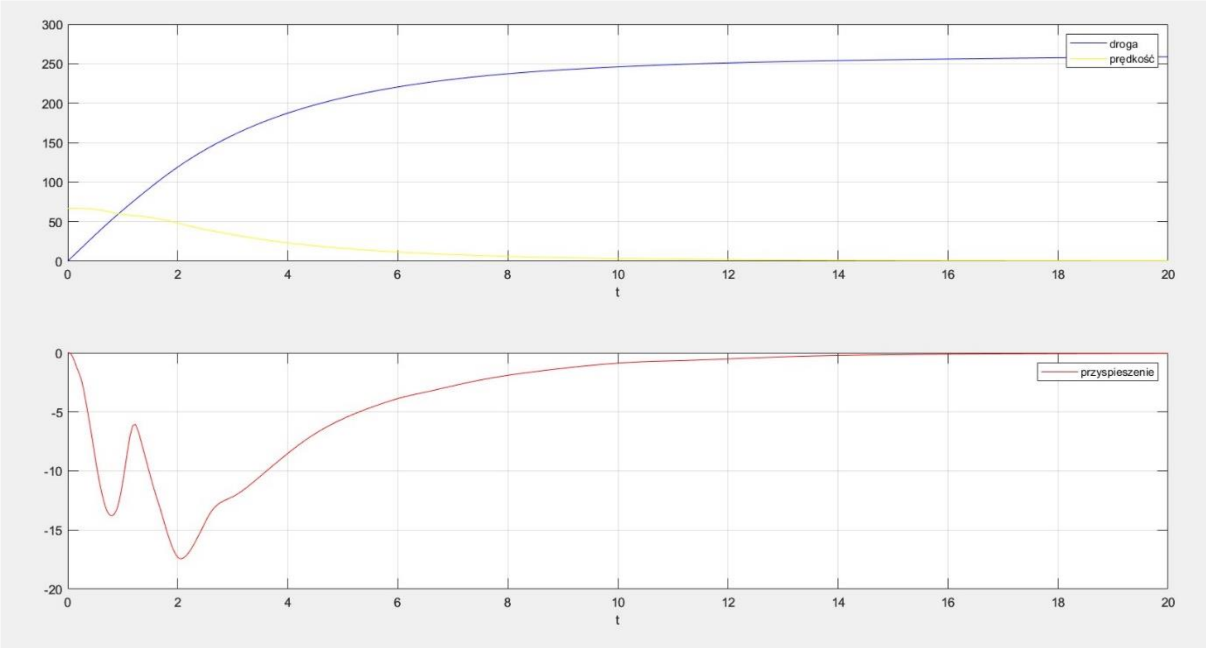
```
function status = hamownik_out(t ,x, flag)
    global w3

    h = 42;        %[m]
    k1 = 54.7e3;   %[N/m]
    m1 = 14000;    %[kg]

    if strcmp(flag, 'init')
        w3 = 0;
    elseif isempty(flag)
        y1 = sqrt(x(1)^2+h^2)-h;
        sin_alfa = x(1)/(h+y1);

        if y1 >= 2*x(3)
            Fk1 = k1*(y1-2*x(3));
        else
            Fk1 = 0;
        end

        w3 = [w3; -2*Fk1*sin_alfa/m1];
    end

    status = 0;
end
```

Funkcja *hamownik_out*

Biorąc pod uwagę fakt, że zmienna *w3* nie jest zwracana przez funkcję oraz jest widziana tylko w obrębie powyższego kodu, trzeba ustawić wskazaną zmienną jako zmienną globalną.

```
%% zadanie 3
clear all; close all;

global w3;

options = odeset('OutputFcn', @hamownik_out, 'Refine', 1);
[T, Y] = ode45(@hamownik, [0 20], [0 67 0 0 0 0], options);

figure;
subplot(2, 1, 1);
plot(T, Y(:, 1), 'b');
hold on;
plot(T, Y(:, 2), 'y');
xlabel('t');
grid on;
legend('droga', 'prędkość');
hold off;

subplot(2, 1, 2);
plot(T, w3, 'r');
xlabel('t');
grid on;
legend('przyspieszenie');
```

Kod do zadania 3



Wykresy zależności przemieszczenia, prędkości i przyspieszenia hamującego samolotu od czasu

Otrzymane wyniki zgadzają się z tymi, które zostały przedstawione na poprzednich zajęciach.

4. Wnioski

Solvery są pożytecznymi narzędziami zawartymi w oprogramowaniu *Matlab*. Dzięki ich pomocy użytkownik jest w stanie rozwiązać zagadnienie początkowe równań różniczkowych zwyczajnych, co okazuje się być użyteczne w momencie wykonywania zadań poruszających tematykę przestrzeni stanów wskazanych układów. W *Matlabie* istnieją różne solvery posiadające odmienne estymaty błędu wyznaczonego rozwiązania.

5. Bibliografia

- Konspekt do zajęć „Rozwiązywanie ODE w Matlabie”