Table of Contents


```
close all
clear
clc
```

Mass-damper-spring parameters

```
m = 1;
d = 0.1;
k = 1;
A = [ 0      1;
     -k/m -d/m];
% Mass-damper-spring vector field
fMassDamperSpring = @(t,x) A*x;
lambda = -0.05;
f = @(t, x) lambda*x;

% Heun's third-order method
A_3 = [0  0  0;
    1/3 0  0;
    0  2/3 0];
c_3 = sum(A,2);
b_3 = [1/4; 0; 3/4];

A_1 = 0;
b_1 = 1;
c_1 = 0;

A_2 = [0 0;
       0.5 0];
c_2 = [0;1/2];
b_2 = [0;1];

A_4 = [0 0 0 0;
       0.5 0 0 0;
       0 0.5 0 0 ;
       0 0 1 0];
b_4 = [1/6; 1/3; 1/3; 1/6];
c_4 = [0; 1/2; 1/2; 1];
```

```
HeunArray_1 = struct('A',A_1,'b',b_1,'c',c_1);
HeunArray_2 = struct('A',A_2,'b',b_2,'c',c_2);
HeunArray_3 = struct('A',A_3,'b',b_3,'c',c_3);
HeunArray_4 = struct('A',A_4,'b',b_4,'c',c_4);

% Simulation parameters
dt = 0.01;

x0 = 1;
number_of_deltas = 7;
dts = zeros(number_of_deltas);
X_ns_1 = zeros(number_of_deltas,1);
X_ns_2 = zeros(number_of_deltas,1);
X_ns_3 = zeros(number_of_deltas,1);
X_ns_4 = zeros(number_of_deltas,1);
final_time = 50;
for i = 1:number_of_deltas
    T = 0:dt:final_time;
    steps = size(T, 2);
    dts(i) = dt;
    dt = dt*2;

    X_RK1 =  ERKTemplate(HeunArray_1,f,T,x0, dt);
    X_RK2 =  ERKTemplate(HeunArray_2,f,T,x0, dt);
    X_RK3 =  ERKTemplate(HeunArray_3,f,T,x0, dt);
    X_RK4 =  ERKTemplate(HeunArray_4,f,T,x0, dt);
    X_ns_1(i) = abs(X_RK1(steps)-exp(lambda*final_time));
    X_ns_2(i) = abs(X_RK2(steps)-exp(lambda*final_time));
    X_ns_3(i) = abs(X_RK3(steps)-exp(lambda*final_time));
    X_ns_4(i) = abs(X_RK4(steps)-exp(lambda*final_time));


end
```

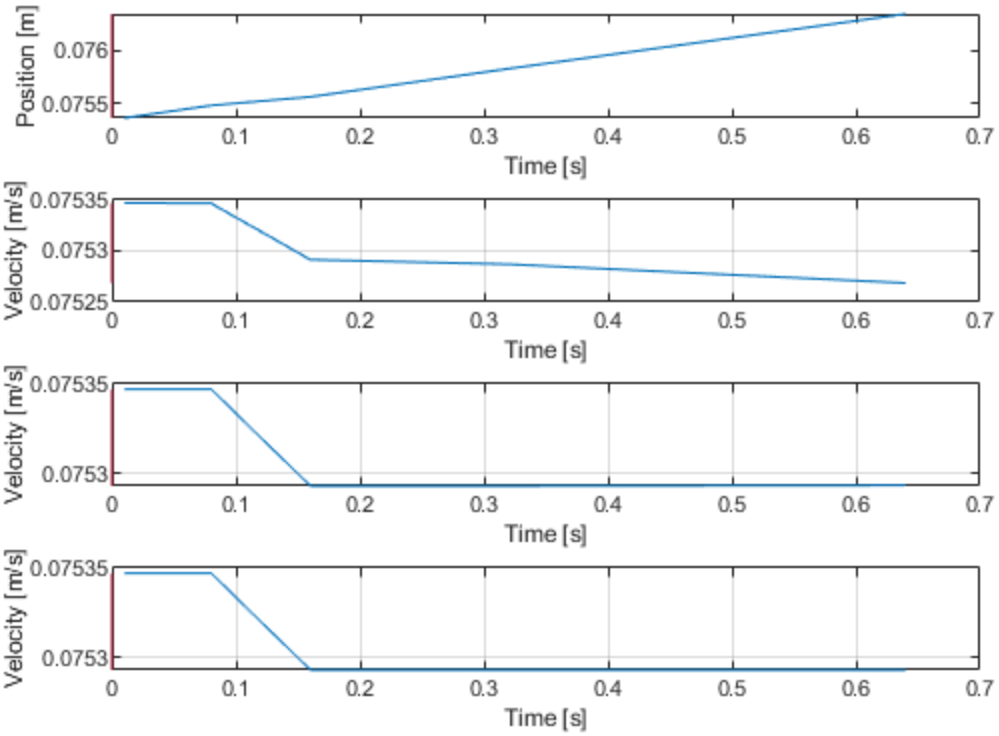
Simulate

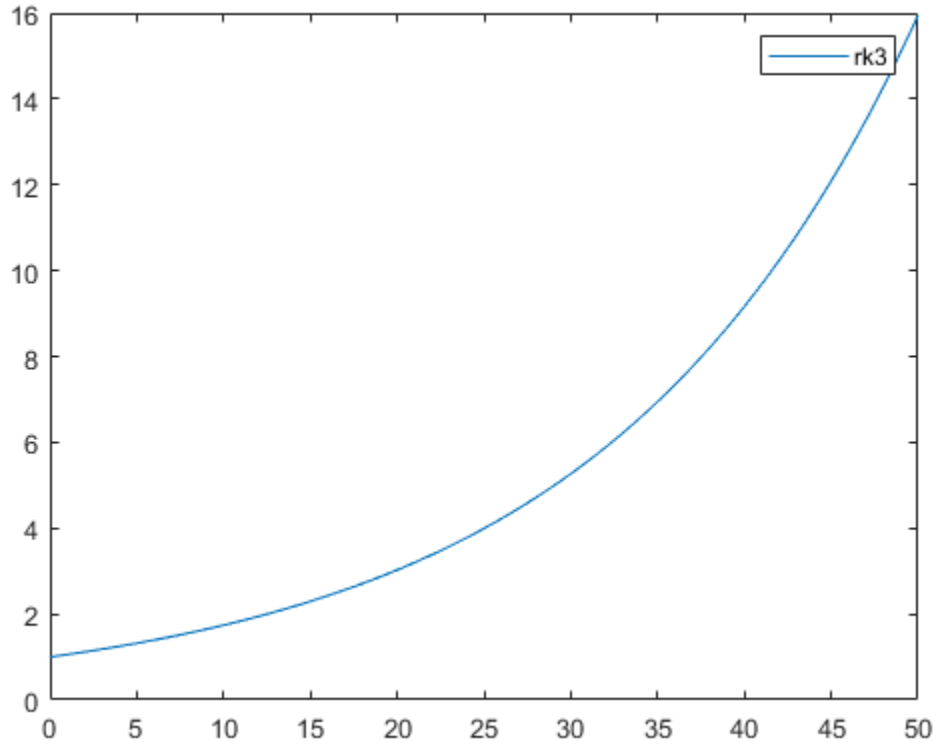
```
%X = ERKTemplate(HeunArray_2,fMassDamperSpring,T,x0);
```

Plot

```
figure
subplot(4,1,1)
plot(dts,X_ns_1(:)')
ylabel('Position [m]')
xlabel('Time [s]')

subplot(4,1,2)
plot(dts,X_ns_2(:)')
grid on
ylabel('Velocity [m/s]')
```

```
xlabel('Time [s]')


subplot(4,1,3)
plot(dts,X_ns_3(:)')
grid on
ylabel('Velocity [m/s]')
xlabel('Time [s]')

subplot(4,1,4)
plot(dts,X_ns_4(:)')
grid on
ylabel('Velocity [m/s]')
xlabel('Time [s]')
figure(2)

dt = 0.4
lambda = -7

T = 0:dt:final_time;
    steps = size(T, 2);
    dts(i) = dt;

f = @(t, x) lambda*x;

    X_RK1 =  ERKTemplate(HeunArray_1,f,T,x0, dt);
    X_RK2 =  ERKTemplate(HeunArray_2,f,T,x0, dt);
    X_RK3 =  ERKTemplate(HeunArray_3,f,T,x0, dt);
    X_RK4 =  ERKTemplate(HeunArray_4,f,T,x0, dt);
%
% plot(T,X_RK1(:)')
% hold on
% plot(T,X_RK2(:)')
% hold on;
% plot(T,X_RK3(:)')
% hold on;
plot(T,X_RK4(:)')
legend('rk3','rk4')


dt =

    0.4000


lambda =

    -7

Warning: Ignoring extra legend entries.
```


Position [m]
0.076
0.0755
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Time [s]
Velocity [m/s]
0.07535
0.0753
0.07525
Time [s]
Velocity [m/s]
0.07535
0.0753
Time [s]
Velocity [m/s]
0.07535
0.0753
Time [s]


rk3
16
14
12
10
8
6
4
2
0
0
5
10
15
20
25
30
35
40
45
50

Published with MATLAB® R2019b