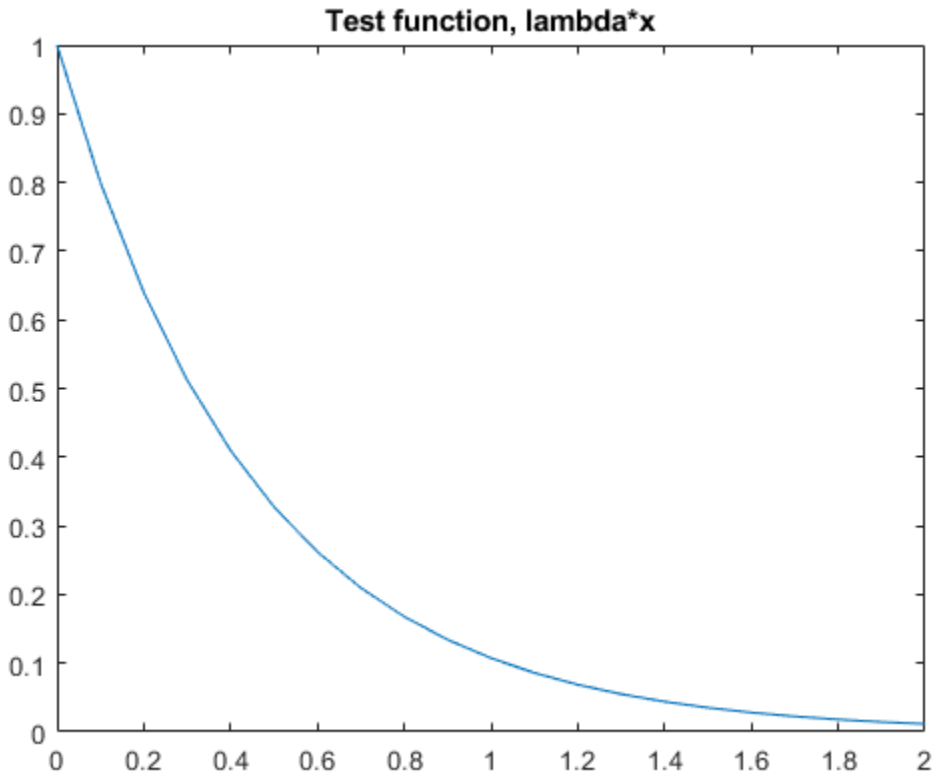
Table of Contents

..... 1
Explicit Euler trajectory: test function 1
Explicit Euler, pneumatic spring: 2
Implicit Euler, pneumatic spring: 2
Implicit RK , midpoint 2
Energy actual 3
Plotting 3


```
close all;
clear;
clc;
```

Explicit Euler trajectory: test function

```
lambda = -2;
f = @(t, x) lambda*x;
T = 0:0.1:2;
x0 = 1;
f_test = figure();
x = explicit_euler(f, T, x0);
plot(T, x);
title('Test function, lambda*x');
```



Explicit Euler, pneumatic spring:

```
dt = 0.01;
tf = 10;
T = 0:dt:tf;
x0 = [2;0];
x_d = 1.32;
k = 2.40;
g = 9.81;
m = 200;

x_ee = explicit_euler(@(t, x) pneumatic_spring_dynamics(t, x, g, k,
 x_d), T, x0);

E_ee = energy(x_ee);
```

Implicit Euler, pneumatic spring:

```
dt = 0.01;
tf = 10;
T = 0:dt:tf;
x0 = [2;0];
x_d = 1.32;
k = 2.40;
g = 9.81;
m = 200;

a = 1;
b = 1;
c = 1;
butcherTableau = struct('A', a, 'b', b, 'c', c);


x_ie = IRKTemplate(butcherTableau, @(t, x)
 pneumatic_spring_dynamics(t, x, g, k, x_d), @(t,x)
 pneumatic_spring_dfdx(t, x, g, k, x_d), T, x0);
E_ie = energy(x_ie);
```

Implicit RK , midpoint

```
dt = 0.01;
tf = 10;
T = 0:dt:tf;
x0 = [2;0];
x_d = 1.32;
k = 2.40;
g = 9.81;
m = 200;

a = 1/2;
b = 1;
```

```
c = 1/2;
butcherTableau = struct('A', a, 'b', b, 'c', c);


x_i_midpoint = IRKTemplate(butcherTableau, @(t, x)
 pneumatic_spring_dynamics(t, x, g, k, x_d), @(t,x)
 pneumatic_spring_dfdx(t, x, g, k, x_d), T, x0);

E_midpoint = energy(x_i_midpoint);
```

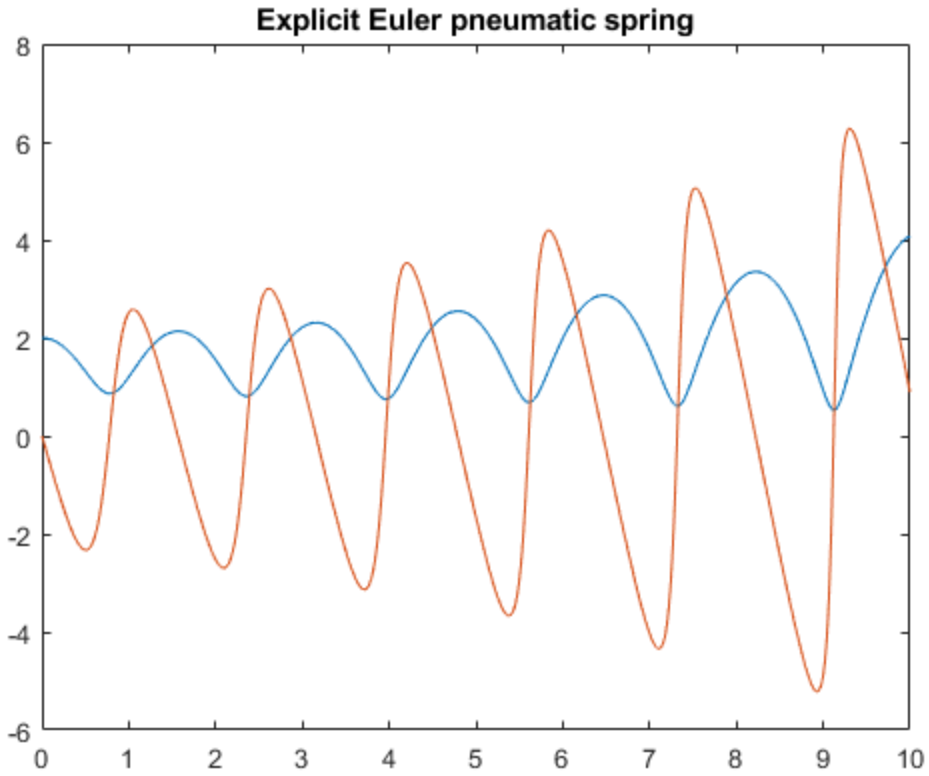
Energy actual

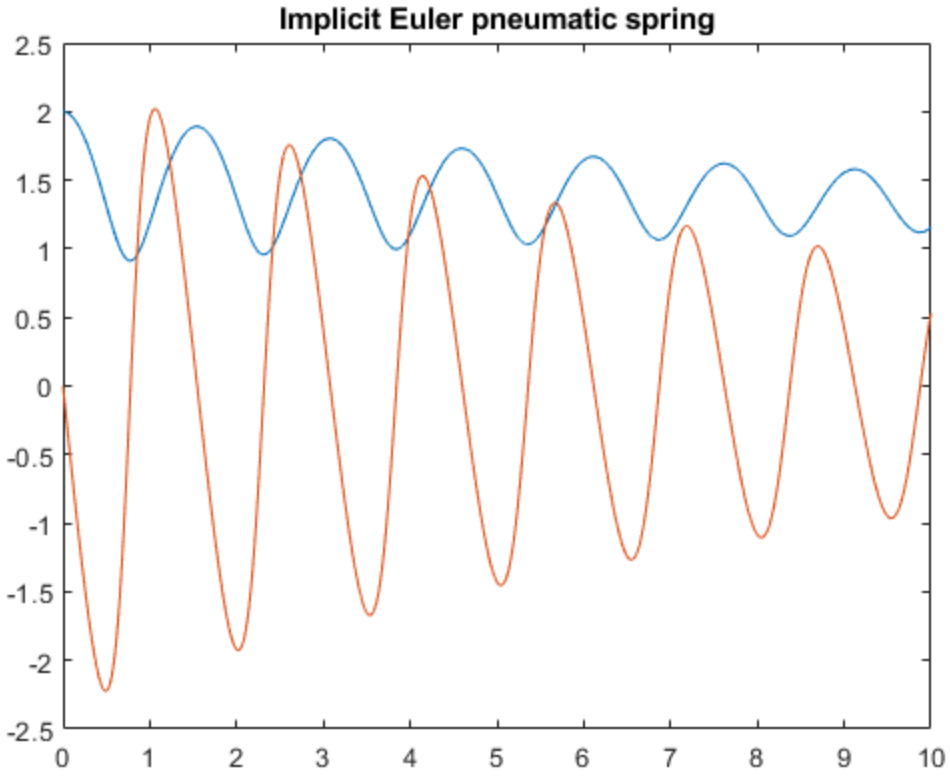
```
dt = 0.01;
tf = 10;
T = 0:dt:tf;
x0 = [2;0];
x_d = 1.32;
k = 2.40;
g = 9.81;
m = 200;
E = ((m*g)/(k-1)*x_d^k/x0(1)^(k-1)+m*g*x0(1)+1/2*m*x0(2));
```

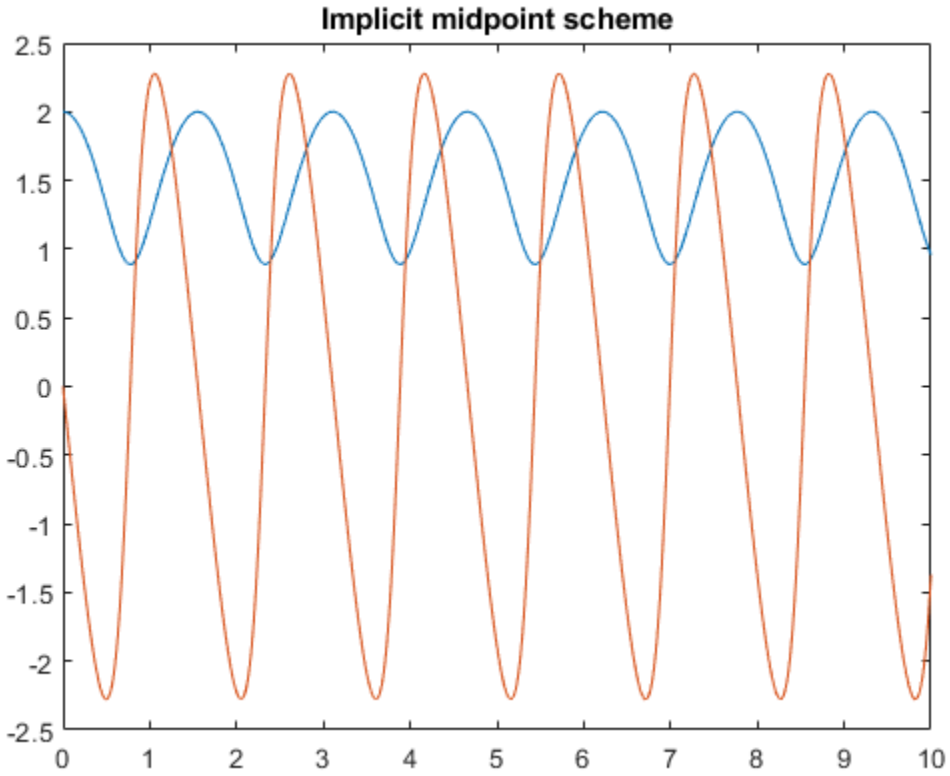
Plotting

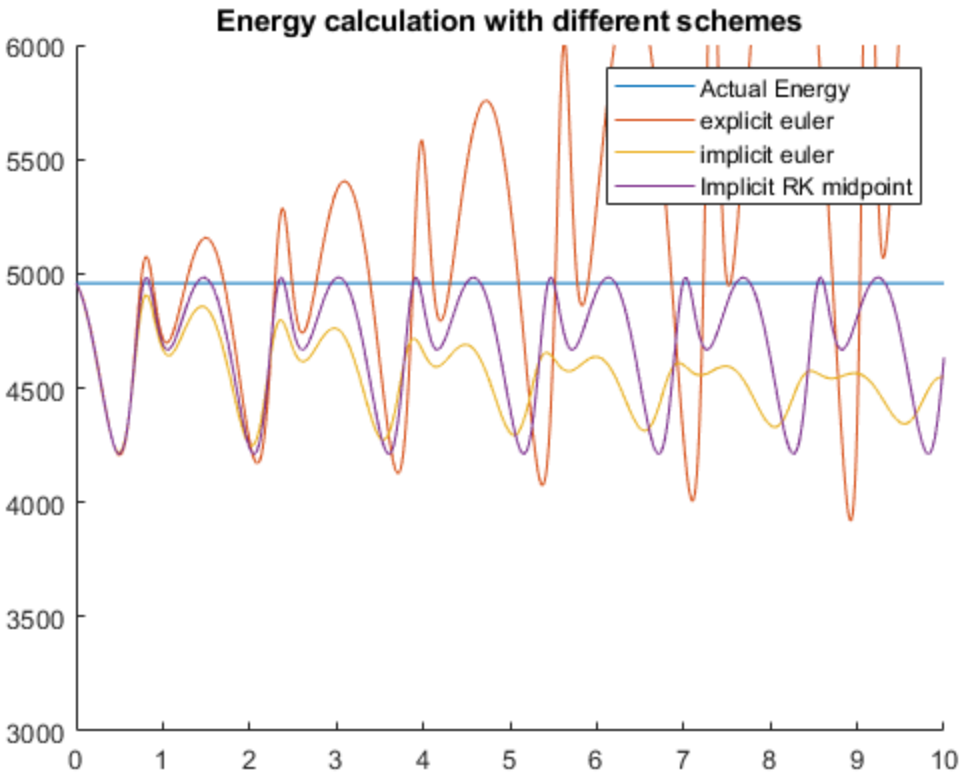
```
f_ee = figure();
plot(T, x_ee);
hold on;
title('Explicit Euler pneumatic spring');
f_ie = figure();
plot(T, x_ie);
hold on;
title('Implicit Euler pneumatic spring');
f_i_midpoint = figure();
plot(T, x_i_midpoint);
hold on;
title('Implicit midpoint scheme');

f_energy = figure();
hold on;
plot(T, E*ones(1,length(T)));
hold on;
plot(T, E_ee);
hold on;
plot(T, E_ie);
hold on;
plot(T, E_midpoint);
ylim([3 6]*1e3)
legend('Actual Energy', 'explicit euler', 'implicit euler', 'Implicit
 RK midpoint');
title('Energy calculation with different schemes');
movegui(f_ee, 'northwest');
movegui(f_ie, 'southwest');
```

```
movegui(f_i_midpoint, 'south');
movegui(f_energy, 'northeast');
```




Implicit Euler pneumatic spring
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
1
2
3
4
5
6
7
8
9
10


Implicit midpoint scheme
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
1
2
3
4
5
6
7
8
9
10




```
syms x1 x2 t g k x_d;
jac = jacobian(pneumatic_spring_dynamics(t, [x1;x2], g, k, x_d), [x1,
 x2]);
simplify(jac)
```

```
ans =

[                                                0, 1]
[ -(conj(g)*conj(k)*conj((x_d/x1)^k))/conj(x1), 0]
```

Published with MATLAB® R2019b