Table of Contents


```
clear; close all;
```

Set up parameters

```
xd = 1.32; % m
k = 2.40; % 1
g = 9.81; % ms^-2
m = 200; % kg
x0 = [2, 0]';
Nx = length(x0);
dt = 0.01; % s
t0 = 0;
tf = 10;
Nt = (tf - t0) / dt;
T = linspace(t0, tf, Nt);
```

Setup process model and its Jacobian

```
syms f t real
x = sym('x', [2 1]);

f = [x(2); -g.*(1-(xd./x(1)).^k)];
dfdx = jacobian(f, x);
E = m*g/(k - 1)*xd^k./(x(1).^(k - 1)) + m*g*x(1) + 1/2*m*x(2).^2;

f = matlabFunction(f, 'Vars', {t, x});
E = matlabFunction(E, 'Vars', {x});
dfdx = matlabFunction(dfdx, 'Vars', {t, x});
```

Solve with explicit Euler

```
A = 0;
c = 0;
b = 1;

ButcherArray.A = A;
ButcherArray.b = b;
ButcherArray.c = c;

x_expeul = ERKTemplate(ButcherArray, f, T, dt, x0);
E_expeul = E(x_expeul);
```

Solve with Implicit Euler

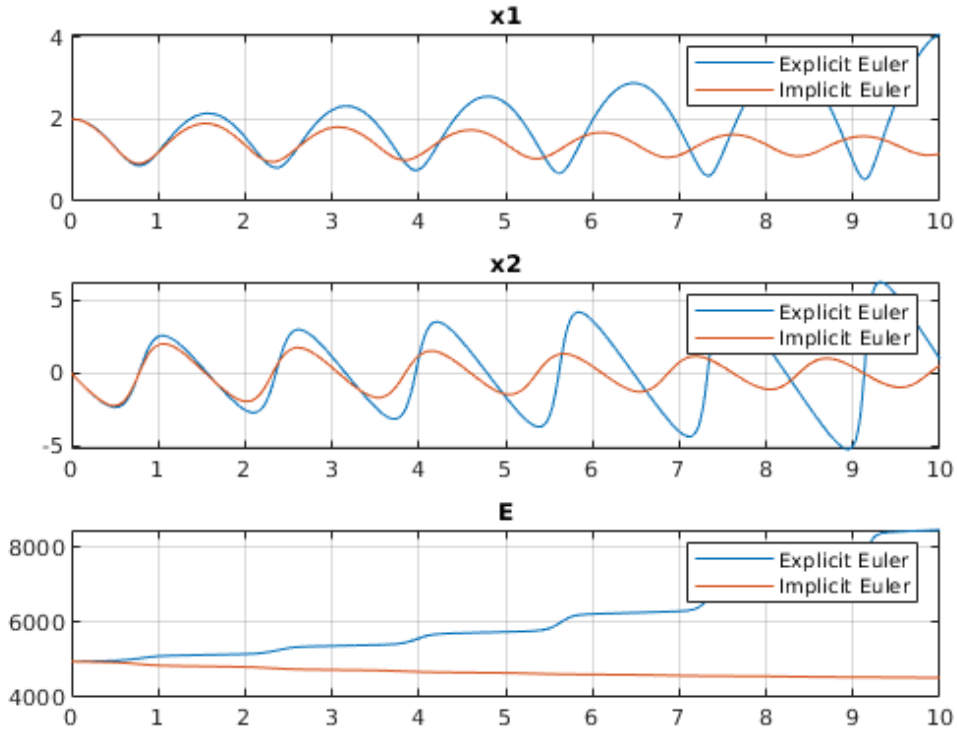
```
x_impleul = ImplicitEulerTemplate(f, dfdx, T, x0);
E_impleul = E(x_impleul);
```

Plot results

```
figure(1);
subplot(3, 1, 1);
    plot(T, x_expeul(1,:), T, x_impleul(1,:));
    legend('Explicit Euler', 'Implicit Euler');
    grid on;
    title('x1');

subplot(3, 1, 2);
    plot(T, x_expeul(2,:), T, x_impleul(2,:));
    legend('Explicit Euler', 'Implicit Euler');
    grid on;
    title('x2');

subplot(3, 1, 3);
    plot(T, E_expeul, T, E_impleul);
    legend('Explicit Euler', 'Implicit Euler');
    grid on;
    title('E');
```



Published with MATLAB® R2019a