Numerical Integration - Lecture 2

ME3001 - Mechanical Engineering Analysis

April 16, 2020

Euler's Method for Higher-Order Models

Lecture 2 - Euler's Method for Higher-Order Models:

- Review Euler's Method
- A More Exciting Model
- Equation Decomposition
- MATLAB Solution

Forward Integration

Last time we solved a **first order** ODE with Euler's method.

ODE and IC: $m\dot{v} + cv = 0 \qquad v(t = 0) = v_0$

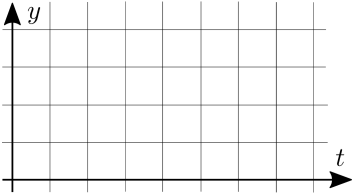
Slope Function:

$$v(t_{i+1}) = v(t_i) + f(t_i, v_i)\Delta t$$

$$v(\quad) = v(\quad) + f(\quad, \quad)\Delta t$$

$$v(\quad) = v(\quad) + f(\quad, \quad)\Delta t$$

$$v(\quad) = v(\quad) + f(\quad, \quad)\Delta t$$



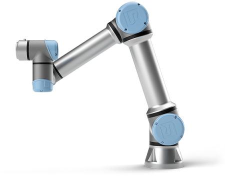
Euler's Method in MATLAB

```
% approximate with Euler's forward integration
v_eu(1)=v0;
for j=1:length(time)-1
    v_eu(j+1)=v_eu(j)+(f(time(j),v_eu(j),m,c))*dt;
end

% If this is an 'Inline Definition' of the function
% it MUST go at the bottom of the script
function [dvdt]=f(t,v,M,C)
    dvdt=-C/M*v;
end
```

A More Exciting Model

- First and second order linear models are frequently used in science and engineering
- However the world is __________-________________________
- Many exciting and important engineering problems invlove **more complex models** involving rotational motion.



Non-Linear Swinging Pendulum

An example of a non-linear system is an *inverted pendulum metronome*.

How will this system behave?

$$I_o\ddot{\theta} + k_T - (m \cdot g \cdot l)\, sin(\theta) = 0$$

Finding an analytical solution is **very involved** and only mathematicians have time for all that...but you can look here

As we have seen using Euler's method is not hard, but we have to setup the problem correctly. This is a reoccuring theme!

$$I_o\ddot{\theta} + k_T - (m \cdot g \cdot l)\, sin(\theta) = 0$$

To solve a second order system with an integration method like Euler's you must write the **slope function(s)**.

There are two derivatives so there are two ________ __________.

x2 First Order from x1 Second Order

One second order ODE can be **decomposed** into *two* first order ODEs through a simple change of variables. This step can be confusing, but remember it is just an algebraic substitution!

$$I_o\ddot{\theta} + k_T - (m \cdot g \cdot l)\, sin(\theta) = 0$$

Execute Euler's Method

$f_1 =$

$f_2 =$

Use Euler's method just as before with both slope functions. Compute the values of the solution one-by-one **forward in time** *for each variable*.

i	$z_1(t_{i+1}) = z_1(t_i) + f_1(t_i, z_{1i}, z_{2i})\Delta t$	$z_2(t_{i+1}) = z_2(t_i) + f_2(t_i, z_{1i}, z_{2i})\Delta t$
1	$z_1(\quad) = z_1(\quad) + f_1(\quad,\quad,\quad)\Delta t$	$z_2(\quad) = z_2(\quad) + f_2(\quad,\quad,\quad)\Delta t$
2	$z_1(\quad) = z_1(\quad) + f_1(\quad,\quad,\quad)\Delta t$	$z_2(\quad) = z_2(\quad) + f_2(\quad,\quad,\quad)\Delta t$
3	$z_1(\quad) = z_1(\quad) + f_1(\quad,\quad,\quad)\Delta t$	$z_2(\quad) = z_2(\quad) + f_3(\quad,\quad,\quad)\Delta t$

As you can see this can get messy quickly.

Part 1 - Setup and Analytical Solution

```
clear variables;close all;clc

% define the constant parameters
global m g l kt;
m=2;g=9.8;
l=42*(1/100);kt=6;

% initial conditions
theta0=15;
omega0=0;

% create an array of time values
dt=.001;tstop=10;
time=0:dt:tstop;
```

Part 2 - Euler's Method

```
% approximate with Euler's forward integration
z1_eu(1)=theta0*pi/180; z2_eu(1)=0;  % initial condit

for j=1:length(time)-1
    z1_eu(j+1)=z1_eu(j)+f1(time(j),z1_eu(j),z2_eu(j))*
    z2_eu(j+1)=z2_eu(j)+f2(time(j),z1_eu(j),z2_eu(j))*
end

function [dz1dt]=f1(t,z1,z2)
    global m g l kt;
    dz1dt=z2;
end
function [dz2dt]=f2(t,z1,z2)
    global m g l kt;
    dz2dt=(m*g*l*sin(z1)-kt*z1)/(m*l^2);
end
```

Part 3 - Graph the Solutions

```
% plot the results of both methods
figure(1);hold on
plot(time,v_exact,'r-','LineWidth',2)
plot(time,v_eu,'b*')

% add some labels
title('Radio Flyer: mdv/dt+cv=0, v(t=0)=v0')
legend('Exact','Euler''s')
xlabel('Time (s)')
ylabel('Velocity')
grid on
```

Do you believe the results?