

# Numerical Integration - Lecture 1

ME3001 - Mechanical Engineering Analysis

April 13, 2020

**Euler's Forward Integration**

## Lecture 1 - Euler's Forward Integration:

- Review and Motivation
- Euler's Forward Integration
- Example Problem
- MATLAB Solution



Leonard Euler (1707-1783)

## What is a Differential Equation? Solution?

A **differential equation** is an equation which describes a function and one or more of its \_\_\_\_\_ of the

\_\_\_\_\_ with respect to the \_\_\_\_\_.

The **solution** to a differential equation describes the \_\_\_\_\_ as a function of the \_\_\_\_\_.

# Analytical vs. Numerical Solutions

## Analytical

- solution to a problem that can be written in **closed form**
- solution in terms of known functions, constants, etc.
- gives an **exact answer**

## Numerical

- an **approximation** to the solution of a mathematical equation
- known as **numerical integration**
- numerical integration is more than *the computation of integrals*

## Which one should you choose?

? ? ? ? ?

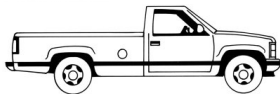


It depends on the problem. It also depends on how you intend to use the solution.

# The Initial Value Problem

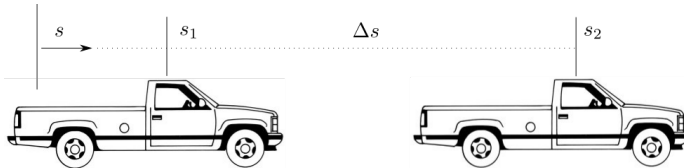
You learned about the **initial value problem** in differential equations class. Do you remember?

You have probably been thinking about this idea for much longer than that. Consider riding in a *truck* waiting to arrive at you destination...



## Integrating a Rate

You may not have known it but you were **integrating** when performing these mental calculations. You can math.



# The Taylor Series



James Gregory (1638-1675)



Brook Taylor (1685-1731)

Consider the Taylor Series. How does this apply to our problem?

$$y(x) \approx$$

$$y(a) + y'(a)(x - a) + \frac{y''(a)}{2!}(x - a)^2 + \frac{y^{(3)}(a)}{3!}(x - a)^3 + \dots + \frac{y^{(n)}(a)}{n!}(x - a)^n$$

What does this even mean?



## Euler's Method

Given a *function describing the slope* and an *initial condition*, discretized values of the solution can be approximated.

This is known as **Euler's method**.

$$y(x + \Delta x) = y(x) + \frac{dy}{dx} \Delta x = y(x) + f(x, y(x)) \Delta x$$

It is commonly shown with subscript notation.

$$y(x_{i+1}) = y(x_i) + f(x_i, y(x_i)) \Delta x = y_i + f(x_i, y_i) \Delta x$$

Careful: This is not the same as Euler's formula which is an essential trigonometric identity also used in differential equations.

# The Slope Function

The differential equation must be written as a function describing the first derivative or **the slope** of the dependent variable.

$$f(x, y) = \frac{\text{rise}}{\text{run}} = \frac{dy}{dx} \neq y(x)$$

or with subscript notation shown below

$$f(x_i, y_i)$$

Careful: The first argument  $x$  is not always used and is often left out. However it is an important placeholder (ODE45()) and shows this method can be used for *non-linear* equations with generalized input functions.

## Forward Integration

Using this concept to solve the initial value problem is called **Euler's forward integration** or **Euler's Method**. Most of the time, the independent variable is \_\_\_\_\_.

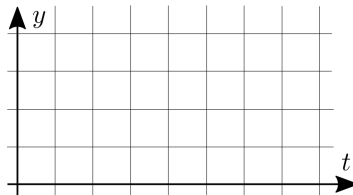
Compute the values of the solution one-by-one **forward in time**.

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

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

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

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



## The Previous Example - Radio Flyer

If this is a valid technique we should be able to solve the problem we solved in the previous lecture. Ferrari anyone? Let's do a Radio Flyer instead.



$$m\dot{v} + cv = 0 \quad \text{with} \quad v(t=0) = v_0$$

$$\implies v(t) = v_0 e^{-\frac{c}{m}t}$$

## The Problem Statement

This method is not difficult *if* we setup the problem correctly. Read the problem statement carefully.

Approximate a solution to the differential equation using Euler's Method. Graph the solution from 0 to 10 seconds and use a stepsize of  $\Delta t = 1.0$ ,  $1.0$ , and  $1.0$  seconds.

$$m\dot{v} + cv = 0 \quad \text{with} \quad v(t = 0) = v_0$$

$$m = 100(\text{kg}), \quad c = 0.5\left(\frac{\text{n-m}}{\text{s}}\right), \quad v_0 = 5.0\left(\frac{\text{m}}{\text{s}}\right)$$

## Breakdown The Problem Statement

ODE:

$$m\dot{v} + cv = 0$$

Initial Condition:

$$v(t = 0) = v_0$$

Parameters:

$$m = 100(\text{kg}), \quad c = 0.5\left(\frac{\text{N}\cdot\text{s}}{\text{m}}\right), \quad v_0 = 5\left(\frac{\text{m}}{\text{s}}\right)$$

Strategy:

Euler's Method,  $\Delta t = 1.0, 0.1, \text{ and } 0.01(\text{s})$

Look at the formula we derived. What goes where?

$$y_{i+1} = y_i + f(x_i, y_i)\Delta x$$

## Execute Euler's Method

First, write the **slope function**.

$$f(t, y(t)) = f(t, y) =$$

Then, start with the initial condition and compute the values of the solution *one by one, forward in time*.

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

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

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

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

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

This method is not suitable for manual computation.

## Part 1 - Setup and Analytical Solution

```
% ME 3001 - Mechanical Engineering Analysis  
% Tristan Hill - Spring 2020  
% Numerical Integration - Lecture 1  
clear variables;close all;clc  
  
% define the constant parameters  
m=100;c=1.5;v0=2.0;  
dt=1.0;tstop=60;  
  
% create an array of time values  
time=0:dt:tstop;  
% compute solution from derived equation  
v_exact=v0*exp(-c/m*time);
```



## Part 2 - Euler's Method

```
% 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
```

## 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:  $mdv/dt + cv = 0$ ,  $v(t=0) = v_0$ ')  
legend('Exact','Euler''s')  
xlabel('Time (s)')  
ylabel('Velocity')  
grid on
```

# Do you believe the results?

