EE202 Lab#8 Part 2 Numerical Methods II

[12.2] Matlab includes a number of functions that solve ordinary differential equations of the form

$$\frac{dy}{dx} = f(t, y)$$

numerically. In order to solve higher-order differential equations (and systems of differential equations) they must be reformatted into a system of first-order expressions. Matlab includes a wide variety of differential equation solvers (Table 12.1). However, all these solvers have the same format. This makes it easy to try different techniques by just changing the function name. Each solver needs the following three inputs as a minimum;

1. A function handle to a function that describes the first-order differential equation or system of differential equations in terms of t and y.
2. The time span of interest
3. An initial condition for each equation in the system.

The solvers all return an array t- and y-values. If you do not specify an output array, the function creates a plot of the results.

1. Make a new M-File named **`ode_exp`** as define in Appendix A (points will be deducted if you do not include the header instructions and comments). Add the following function definition and comments at the beginning of your script.

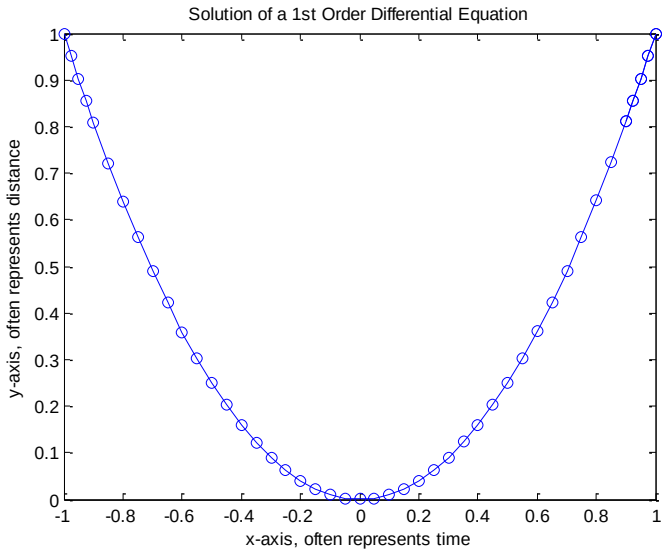
```
function  ode_exp
% Chapters 12 Numerical Methods: Part II

%% Lab 8 Part 2 Question 1 Simple First Order ODE.
```

Next, add the following comment at the end of your function. Always keep this as the last instruction in the M-File.

```
end
```

In part 1 of this lab sequence (Part 1 Problem 5) you used Matlab's ODE solver(s) to find the solution to a first order differential equations. Let's review by using ode45 to solve the following linear first order equation, for time -1 to 1 with initial conditions $y(-1) = 1$

$$\frac{dy}{dx} = 2t$$



In this second part of the lab we will be looking at how to solve ODEs of higher power and which are in some cases may be non-linear. The solution of these problems is a three-step process:

Step 1. Convert your n[th]-order differential equation $\frac{d^n}{dt^n}y_n(t) + \frac{d^{n-1}}{dt^{n-1}}y_{n-1}(t) \ldots y(t) = 0$ into state space form; specifically, a series of simultaneous first-order differential equations

$$\frac{d}{dx}y(t) = f\big(t, y(t)\big), \qquad y(t_0) = y_0$$

Step 2. Create a Matlab function which defines your differential equation in state space form.

In Matlab you may define the function (1) as an M-file function, (2) as a handler to a function that evaluates *f*, and (3) as a nested function.

Step 3. Call the ODE solving function with the function defined in step 1. This will typically be ODE45.

2. Working with the M-File from problem 1, add the following comment at the beginning of your script. %% Lab 8 Part 2 Question 2 Second Order ODE. Now let's follow the above steps in order to solve the 2[nd] order differential equation

$$\frac{d^2y}{dt^2} + \frac{dy}{dt} - y(t) - t = 0$$

over the interval -1 to +1 and initial conditions defined as $y = 0$ and $z = 0$ (which is the same as $y = 0$ and $dy/dt = 0$) .

Step 1. Convert your n[th]-order differential equation into a series of simultaneous first-order differential equations .

Let $y_2 = \frac{dy}{dt}$ eq. 1

where $y_1 = y$

It directly follows that $\frac{dy_2}{dt} = \frac{d^2y}{dt^2}$

Substituting into the original equation and solving for $\frac{dy_2}{dt}$ we have

$\frac{dy_2}{dt} = y_1 + t - \frac{dy_1}{dt}$ eq. 2

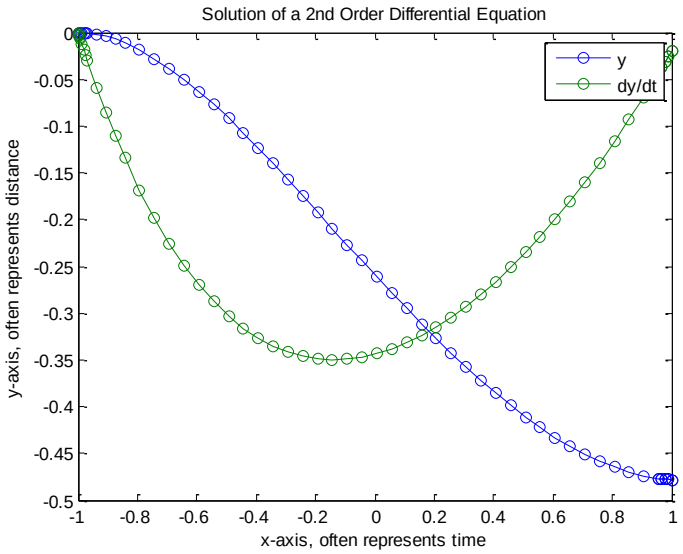
We now have two simultaneous first-order differential equations (equations 1 and 2).

Step 2. Create a Matlab function which defines your differential equation in state space form. Add the following nest function to your M-file.

```
% ------------------------------------------------------------------------
% Nested function
% % 2nd Order Differential Equation (Matlab for Engineers page 530)
    function dydt = twoeq(t,y)
        dydt(1) = y(2);
        dydt(2) = y(1) + t - dydt(1);
        dydt = dydt';
    end
% ------------------------------------------------------------------------
```

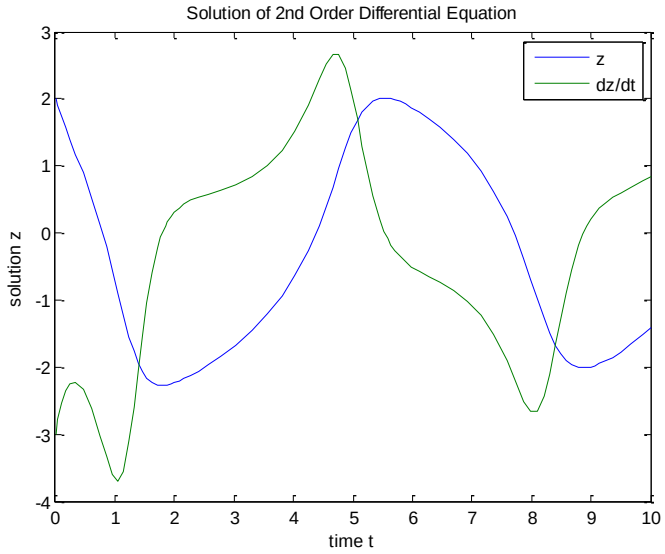
Step 3. Call the ODE solving function with the function defined in step 1. This will typically be ODE45. Add the following Matlab script just before your nest function definition.

```
% 2nd Order Linear Differential Equation (Matlab for Engineers page 530)
figure;     %create a new figure 2
ode45(@twoeq,[-1,1],[0,0])
title('Solution of a 2nd Order Differential Equation');
xlabel('x-axis, often represents time');
ylabel('y-axis, often represents distance');
legend('y','dy/dt')
```



3. Working with the M-File from problem 1, add the following comment at the beginning of your script. %% Lab 8 Part 2 Question 3 Control Problem. Applying what you have learned solve the following second order ODF, over the time interval 0 to 10 seconds.

$$\ddot{Z} + (Z^2 - 1)\dot{Z} + Z = 0, \quad Z(0) = 2, \quad \dot{Z}(0) = -3$$

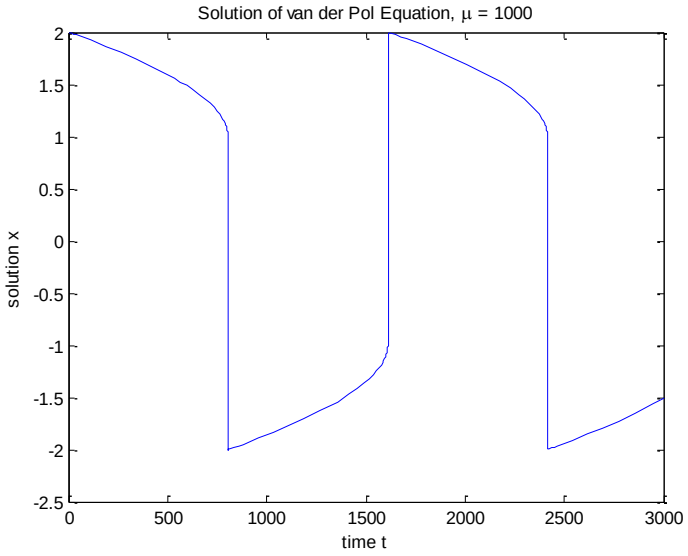


4. Working with the M-File from problem 1, add the following comment at the beginning of your script. %% Lab 8 Part 2 Question 4 Van der Pol oscillator. Applying what you have learned solve the following second order ODF over the interval 0 to 3μ or 20, whichever is greater.

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x = 0$$

and initial conditions defined as $x(0) = 2$ and $dx/dt = 0$.

You can find out more about the Van der Pol oscillator here. Try following the steps above to solve the Van der Pol oscillator equation. If you need help, look-up `vdpode` in the Matlab library folder on your computer. The plot below was done with μ = 1000.



Appendix A Creating Your Lab Report

How to Organize and Clean up Your Work

To clean up past work, place the following line at the beginning of your M-File(s).

```
format compact
clear,clc, close all  %  The close all command closes all figure windows
```

To make more than one figure in an M file, use the function `figure(n)`, where **n** is the next figure to be drawn.

How to Publish your Lab

Matlab can format your M-file and the resulting outputs for publishing (i.e., a format you can turn in with your lab) by selecting File → Publish To →Word Document. If the conversion fails then try File → Publish To HTML, open in IE or Firefox and convert to PDF or simply print.