Drexel University

Office of the Dean of the College of Engineering

ENGR 232 – Dynamic Engineering Systems

Week 4 – Pre Lab

Decoupling Second-Order Linear Differential Equations for ode45

Consider a 2nd-order differential equation of the form:

$$ay'' + by' + cy = g(t) \qquad y(0) = y_0,\ \ y'(0) = v_0$$

We know we can solve this using method of undetermined coefficients which will be completed next week.

These differential equations can be **decoupled** into a system of first-order differential equations by considering the following variables:

$$x_1(t) = y(t)$$

$$x_2(t) = y'(t)$$

Notice, based on this definition:

$$x_1'(t) = y'(t) = x_2(t)$$

$$x_2'(t) = y''(t)$$

From the original differential equation, we have:

$$y''(t) = \frac{1}{a}(g(t) - by' - cy)$$

$$\rightarrow x_2'(t) = \frac{1}{a}g(t) - \frac{b}{a}x_2(t) - \frac{c}{a}x_1(t)$$

Now we have 2 coupled differential equations that represent the same system, now in terms of variables x_1 and x_2. Solving for variable $x_1 = y$ gives the solution to the system.

$$x_1'(t) = x_2$$

$$x_2'(t) = -\frac{b}{a}x_2 - \frac{c}{a}x_1 + \frac{1}{a}g(t)$$

The initial conditions will apply to each equation separately, with $x_1(0) = y(0) = x_0$, and $x_2(0) = y'(0) = v_0$.

The variables x_1 and x_2 are called **states** of the system.

Now we have a system of coupled differential equations, which we can find the solution to using **ode45** in MATLAB. Remember, in matrix notation this can be expressed by:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -c/a & -b/a \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/a \end{bmatrix} g(t)$$

This formulation has the representation:

$$\dot{X} = AX + BU$$

Here, $U \equiv g(t)$

Example:

$$3y'' - 2y' + 6y = \sin(t) \qquad y(0) = 2,\ \ y'(0) = 1$$

Let $x_1 = y$ and $x_2 = y'$

Thus, we get:

$$x_1' = x_2$$

$$x_2' = \frac{2}{3}x_2 - 2x_1 + \frac{\sin(t)}{3}$$

$$x_1(0) = 2, \quad x_2(0) = 1$$

Matrix notation:

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},\ \ A = \begin{bmatrix} 0 & 1 \\ -2 & 2/3 \end{bmatrix},\ \ B = \begin{bmatrix} 0 \\ 1/3 \end{bmatrix}, \quad U = \sin\,(t)$$

We can make use of this formulation to implement a solution in MATLAB using ode45. Remember, ode45 is only for first order equations, well it can also handle **systems** of first order equations.

1. Define the function in MATLAB in the same way we did for 1st-order systems:

```
function    [x_dot]  = diff_sys(t,x)
    x1 = x(1,1);
    x2 = x(2,1);

    x_dot(1,1) = x2;
    x_dot(2,1) = (2/3)*x2 - 2*x1 + sin(t)/3;
end
```

Alternatively use an anonymous function:

```
% Below, x represents the state vector, so x has two components.
f = @(t, x) [x(2); (2/3)*x(2) - 2*x(1) + sin(t)/3]
```

2. Our initial conditions go in a **column** vector, just like the states go in a column vector as well.

```
% Initial conditions
x10 = 2;
x20 = 1;
x_0 = [x10; x20];
```

3. Now we use the same syntax as we did for solving first-order differential equations as follows:

Use this syntax if you defined the function file **diff_sys.m**

```
% Solve the DE over the interval from 0 to 10.
tSpan = [0 10];
[t_out, y_out] = ode45(@diff_sys, tSpan, x_0)
```

Note that the variable y_out now will have 2 columns of data. The first column will contain the numerical solution to the variable x_1 which corresponds to the solution $y(t)$ of the original differential equations. The second column will contain the data from the numerical solution of the variable x_2 which corresponds to $y'(t)$.

Use this syntax <u>instead</u> if you defined the anonymous function **f** above.

```
% Solve the DE over the interval from 0 to 10.
tSpan = [0 10];
[t_out, y_out] = ode45(f, tSpan, x_0);
```

No @ symbol here!

4. Plot the solution and its derivative. Adjust the plot options.

```
% Plot the solution and its derivative.
figure
plot(t_out, y_out, 'LineWidth', 3)
grid on
set(gca, 'FontSize', 20)
xlabel('time')
ylabel('y and y''')
legend('y','y''', 'Location', 'Best')
```

You should see a resulting figure similar to this.