Higher order ode's

To be able to solve higher order ode's in MATLAB, they must be written in terms of a system of first order ordinary differential equations. An ordinary differential equation can be written in the form:

$$\frac{d^n y}{dx^n} f(x, y, y', y'', y''', \ldots\ldots, y^{n-1})$$

it can also be written as a system of first order differential equations such that

$$y_1 = y, y_2 = y', y_3 = y'', \ldots\ldots y_n = y^{n-1}$$

This system of equations can then be represented as an arrangement such that

$$y'_1 = y_2, y'_2 = y_3, \ldots\ldots\ldots y'_{n-1} = y_n, where\, y'_n = f(x, y_1, y_2, \ldots y_n).$$

Exercise 5: Solve the following differential equation by converting it to a system of first order differential equations, then using a numeric solver to solve the system. Plot the results. $T'' + T' + T = 0, \quad T(0) = 1\, and\, T'(0) = 0$

To convert this 2nd order ode to a system of 1st order ode's, the following assignment is made: $T = y_1 = y(1) \quad and \quad T' = y_2 = y(2)$

Then the ode can be written as the first-order system:

$$T = y(1)$$

$$\frac{dT}{dt} = T' = y(2)$$

$$\frac{d^2T}{dt^2} = T'' = -(y(2) - y(1))$$

The function file containing this system can now be created and solved. The function file named Exercise 5 IS shown below:

```
function dydt=example5(t,y)
dydt(1,1)=y(2);
dydt(2,1)=-(y(2)-y(1));
```

And run file are shown below:

```
clear all,clc
tspan=[0 10]
To=[1 0]
[t,T]=ode45('example5',tspan,To)
plot(t,T(:,1),'k-+',t,T(:,2),'k:*')
legend('T','dTdt'), xlabel('t'), ylabel('T and dT/dt')
```

The following curve is produced

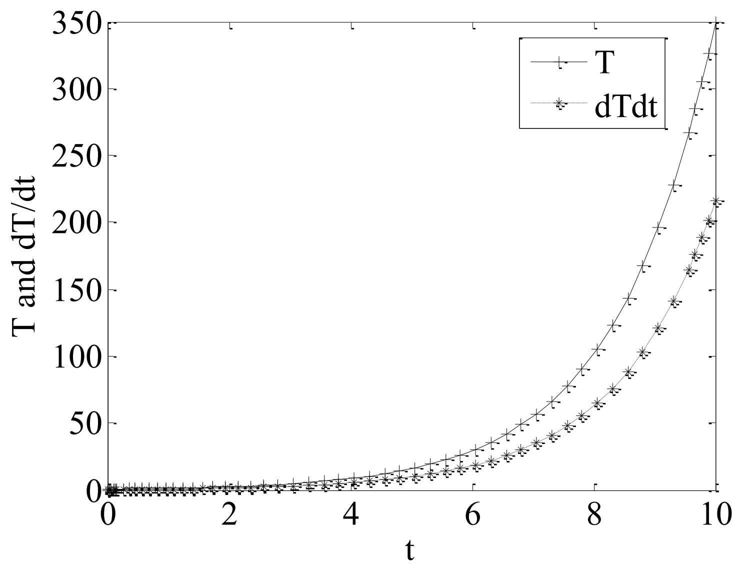

Figure (1) Solution of 2nd order differential equation using runge-kutta integration method

Note: the exact solution of higher order ordinary nonlinear differential equations can be found by **dsolve** command.

Non-Linear Equation Solving
MATLABs built-in function: (fsolve)

fsolve finds a root (zero) of a system of nonlinear equations. It uses a modification of Newton-Raphson method combined with Gaussian elimination. More information about this can be found on the MATLAB help.

You need to express your equation as F(x) =0

Syntax

x = fsolve(fun,x0)

Description

x = fsolve(fun,x0) starts at x0 and tries to solve the equations described in fun. You need to define your equation in a function file.

Exercise 1:

Given that the vapor pressure of methyl chloride at 60 ^oC is 13.76 bar, use the Redlich-kwong equation to calculate the molar volume of saturated vapor at these conditions. Where Redlich-kwong equation is given as follows:

$$P = \frac{RT}{V-b} - \frac{a}{T^{1/2}V(V+b)}$$

Where:

$$a = \frac{0.42748 R^2 T_c^{2.5}}{p_c} \quad \text{and} \quad b = \frac{R T_c}{8\, p_c}$$

The variables are defined as:

P = pressure in bar

V = molar volume in cm^3/mol

T = temperature in K

R = gas constant (83.14 = bar.cm^3/mol.K)

Tc = critical temperature in K

Pc = critical pressure in bar

Knowing: Pc=66.8 bar, Tc=416.3 °K

Since the above equation is nonlinear in the molar volume (V), it can be solved by a proper numerical technique.

Write MATLAB code to solve this exercise using fsolve method:

Function file named EX

```
function F= EX(V)
global R T P a b
F = (R*T)/(V-b)-a/(T^.5*V*(V+b))-P;
```

Main file for fsolve method

```
global R T P a b
T=60+273.15; % in Kelvin
Tc=416.3; % in Kelvin
P=13.76; % in bar
Pc=66.8;% in bar
R=0.08314; % in bar.m3/kmol.K
a=(0.42748*R^2*Tc^2.5)/Pc;
b=R*Tc/(8*Pc);
x0 = 2;   % Make a starting guess at the solution
[V] = fsolve('EX',x0);
Z=P*V/(R*T);
disp('Molar volume (m3/kmol)')
disp(V)
disp('Compressibility factor')
disp(Z)
```

The above code will find the value of the variable V which makes the function F equal to zero. The results will be:-

Molar volume (m3/kmol)

1.7466

Compressibility factor

0.8677

Practice Problems

1) Solve the following ordinary differential equation initial value problems for y(x):
y”+y’-6y=0, y(0)=0, y’(0)=1
And compare your results with the exact solution:
$y=(1/5)e^{2x}-(1/5)e^{-3x}$

2) Solve the following ordinary differential equation initial value problems for y(x):
y”+3y=e^{-x}, y(0)=1, y’(0)=1
And compare your results with the exact solution:
$y=(3/4)\cos(\sqrt{3}x)+\frac{\sqrt{3}}{4}\sin(\sqrt{3}x)+(1/4)e^{-x}$

3) Given
$$\frac{d^2y}{dt^2}+2\frac{dy}{dt}+y=e^{-t},\ y(0)=1,\ \frac{dy}{dt}(0)=2$$
Find $y(0.75)$ by using Fourth order Runge-Kutta

4) Plot y with respect to x in the range *0<x< 10* for the following system of equations:-
$$\frac{d^2y}{dx^2}+2\frac{dy}{dx}+y=0,\quad \frac{dy}{dx}(0)=2,\ y(0)=4,$$
$$\frac{d^3y}{dx^3}+3\frac{d^2y}{dx^2}+5\frac{dy}{dx}+y=\sin x,\quad \frac{d^2y}{dx^2}(0)=12,\ \frac{dy}{dx}(0)=2,\quad y(0)=4$$

5) Consider the following set of coupled first order, nonlinear ODEs
$$\frac{dx}{dt}=x+y-x(x^2+y^2)$$
$$\frac{dy}{dt}=-x+y-y(x^2+y^2)$$
Solve the set of equations with initial conditions x(0) = 2 and y(0) = 2 over the time interval 0 ≤ t ≤ 20. Plot x vs t and y vs t in two different figures. Use **hold on** to keep the plots and graphs of subsequent solutions as overlay plots.

1) The equation to be solved is

$$0.028 = \frac{(6 + x)}{(41 - 3x)^2(32 - x)}$$

Write required code to determine a value for x that satisfies this equation, by **fsolve-function** method.