Department of Chemical Engineering

ChE-101: Approaches to Chemical Engineering Problem Solving

MATLAB Tutorial IX

Solving Non Linear Algebraic Equations (last updated 6/01/05 by GGB)

Objectives:

These tutorials are designed to show the introductory elements for any of the topics discussed. In almost all cases there are other ways to accomplish the same objective, or higher level features that can be added to the commands below.

Any text below appearing after the double prompt (>>) can be entered in the Command Window directly or in an m-file.

The following topics are covered in this tutorial;

Introduction

Solution of a Single Non-Linear Equation Using the fsolve Command

Solution of a System of Non_Linear Equations Using the Numerical fsolve Command

Solved Problems (guided tour)

Introduction

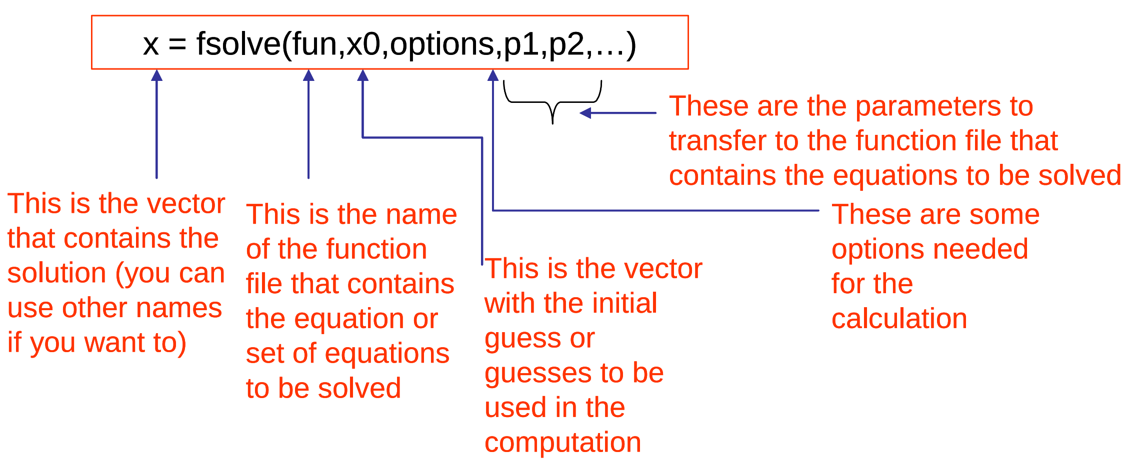
Solution of a single or system of non_linear equations is one of the most commonly encountered problems in chemical engineering.

We'll look at two cases, and show an example of each.

(1) The solution of a single non_linear equation

(2) The solution of a system of non_linear equations Note: If a single equation in a set is non_linear, the entire set is considered non_linear (review your class notes to distinguish linear and nonlinear algebraic equations).

The function available in Matlab that allows to solve a single or a system of nonlinear algebraic equations is fsolve. The general syntax for fsolve is:



Function files (fun):

fsolve finds the root of a nonlinear equation or the roots of a system of nonlinear equations:

$$f(x)=0 \qquad (1)$$

That is, the equation or equations given need to be modified so that they take the form of Eqn. (1), for example, if you are trying to solve the following equation:

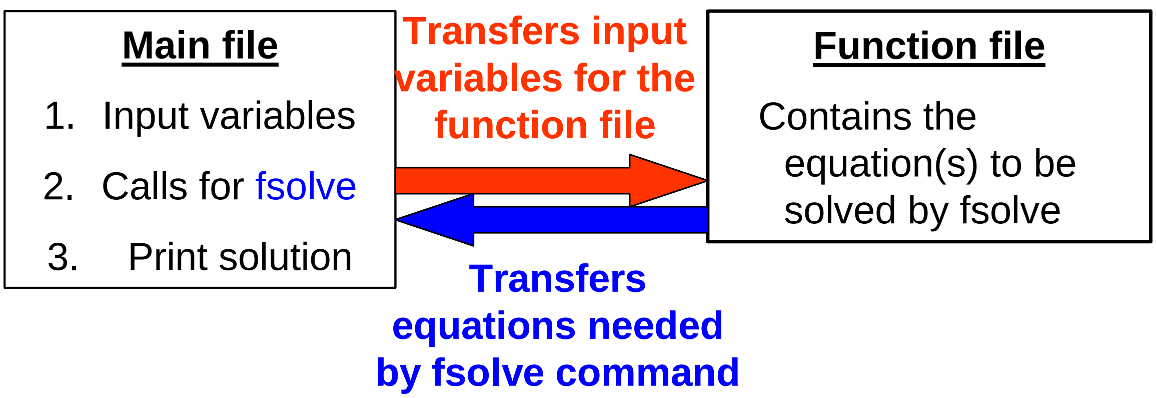
$$x^4+3x^3-x^2+2x=1 \qquad (2)$$

You need to rearrange it to make it take the form of Eqn (1), that is, the equation to be solved should look like:

$$x^4+3x^3-x^2+2x-1=0 \qquad (3)$$

Therefore, Eqn. (3) should be the one to use in the function file to be created by the user. Review Tutorial IV to learn about function files in Matlab.

The figure given below illustrates the way fsolve (which is called in the main program) is related to the user defined function file that contains the equations to be solved by fsolve:



Initial guess vector (x0):
The number of initial guesses to be used should be equal to the number of equations that need to be solved (that is, equal to the number of dependent variables that you are trying to solve). If the initial guesses are not close the root(s) you may not find a solution for the equation(s). In advance courses in chemical engineering (e.g., ChE-400) you will learn how to use chemical engineering principles to define appropriate guesses for a particular system. For this course (ChE-101), the instructor will provide you with a range for the initial guesses. If you don't achieve a solution, you need to change your initial guesses.

Options

There are many options available for the fsolve command. To learn more about all the options use the Matlab help. Most of the time the default options will allow achieving a solution for the problem. These are the minimum options that you need to include in your program:

LargeScale: Use large-scale algorithm if possible when set to 'on'. Use medium-scale algorithm when set to 'off'. This is related to the algorithm used for the solution. The recommended option is to set this to 'off'.
MaxIter: Maximum number of iterations allowed (remember that the solution implies a trial an error procedure, this indicates the number of times the procedure will be repeated). Usually the default is enough, so you don't need to adjust this number.
Display: Level of display. 'off' displays no output; 'iter' displays output at each iteration; 'final' (default) displays just the final output. This option is really helpful when you have problems that are difficult to converge (that is, a solution is not achieved easily). This option will allow you to look at intermediate values calculated by the fsolve during the iteration procedure.

Examples of the syntax to specify the options for fsolve are given below:

1. Use this option must of the time:

```
options = optimset('Largescale','off','Display','off')
```

If you don't set this to off you will have a message display when you find the solution

2. Use this option when you have difficult problems to converge (you will learn about this in advance ChE courses, e.g., ChE-400):

```
options=optimset('Largescale','off','Display','iter');
```

This allows to see the value of the function (how close it is to zero) at different iterations

==

Solution of a single nonlinear algebraic equation

Solve the following equation from heat transfer for the unknown temperature T.

$$\beta T^4 = Tair - T$$

where β and Tair are parameters set by the programmer or program user.

Here is the function file radiation.m that contains the equation to be solved. Note the form of the function. Put everything on the right side so the equation is in the form f(T) = 0 as Eqn. (1). We are searching for the T where y = 0.

This is the function file that contains the equation to be solved

C:\MATLAB6p1\work\Che-101\homework\radiation.m

```
function y=radiation(T,beta,Tair)
%This function is used by the main program heat in the fsolve syntax
%   It contains the equation to be solved: beta*T^4+T-Tair=0
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%Call syntax: y=radiation(T,beta,Tair)
%Input:
%   1. T: Temperature (oK). This is the dependent variable that we are trying to solve using fsolve
%         The input for this variable is transferred from the main file through the initial guess vector
%   2. beta: constant. This is known and fixed. This variable is transferred as a parameter in the fsolve command
%   3. Tair: Temperature of air (oK). This is known and fixed. This variable is transferred as a parameter
%            in the fsolve command
%Ouput:
%   1. y: contains the value of beta*T^4+T-Tair. This will be transferred to the fsolve command in the main program
%_____________________________________________
y = beta*T^4 + T -Tair;
```

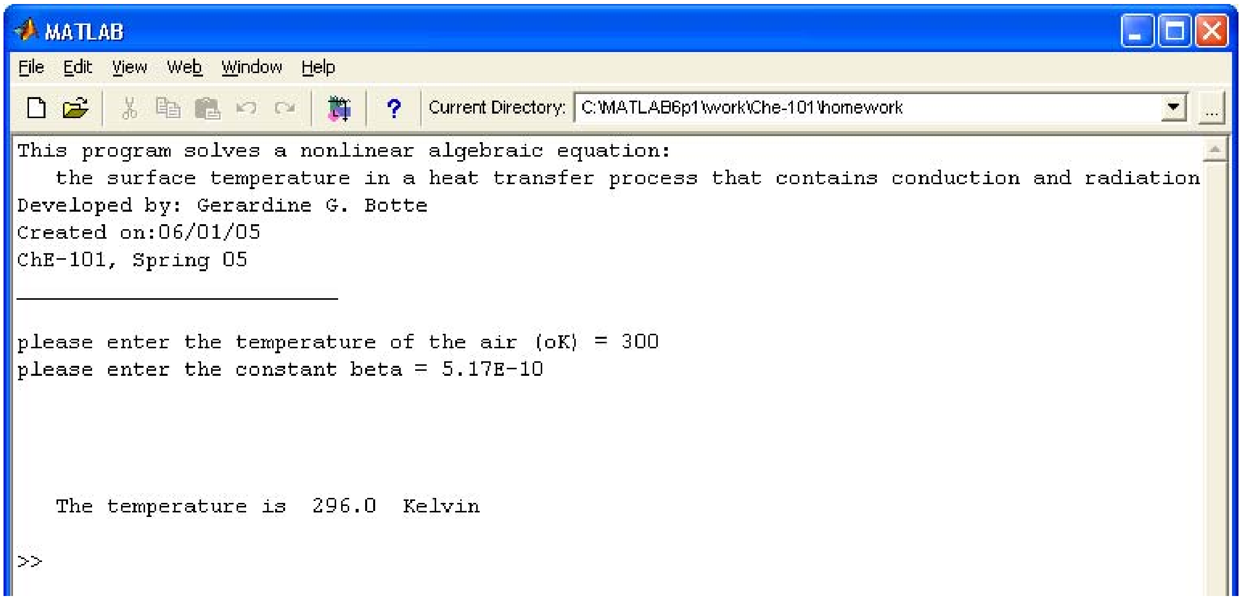
The file looks like the function files that we already learn how to build (see Tutorial IV). Additional descriptions are given to help you understand the connection with the fsolve command

This is the main file that calls for fsolve and prints the solution

```
%This program solves a nonlinear algebraic equation:
% the surface temperature in a heat transfer process that contains conduction and radiation
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%ChE-101, Spring 05
% This program calls for the user defined function radiation (see that function for details)
%__________
% Program starts
clc
clear
disp('This program solves a nonlinear algebraic equation:');
disp('   the surface temperature in a heat transfer process that contains conduction and radiation');
disp('Developed by: Gerardine G. Botte');
disp('Created on:06/01/05');
disp('ChE-101, Spring 05');
disp('_________________________');
fprintf('\n');

%Input Variables
Tair=input('please enter the temperature of the air (oK) = ');
beta=input('please enter the constant beta = ');
fprintf('\n');

%Define initial guesses, you do this in the code you don't ask the user for this
T0 = 300; %This is the initial guess to use for the calculation

% Calling for the fsolve command
options = optimset('Largescale','off','Display','off'); %These are the options
T = fsolve('radiation',T0, options,beta,Tair); %Notice that all the input needed for the function radiation
                                              % is being transferred: T0 takes care of T
                                              % beta: takes care of beta
                                              % Tair: takes care of Tair
%Printing the results
fprintf('\n\n\n   The temperature is % 5.1f  Kelvin \n\n',T)
```

Notice the way the input for the function 'radiation' are transferred. This is the syntax of the fsolve command

This is what will be displayed in the command window



Checking some of the option settings. Modify the options settings in your previous main file to:

```
options = optimset('Largescale','off')
```

and notice what will be displayed in the command window. Conclusion always set your Display option to 'off' to avoid displaying information that the user does not need to know.

Solution of a system of nonlinear algebraic equations

Solve the following system of equations:

$$x_1^2 + x_1 x_2 = 10$$

$$x_2 + 3x_1 x_2^2 = 57$$

Here is the function file equations.m that contains the equations to be solved. Note the form of the function. Put everything on the right side so the equations are in the form f(x) = 0 as Eqn. (1). We are searching for the x_1 and x_2 where z = 0.

This is the function file:

C:\MATLAB6p1\work\Che-101\homework\equations.m

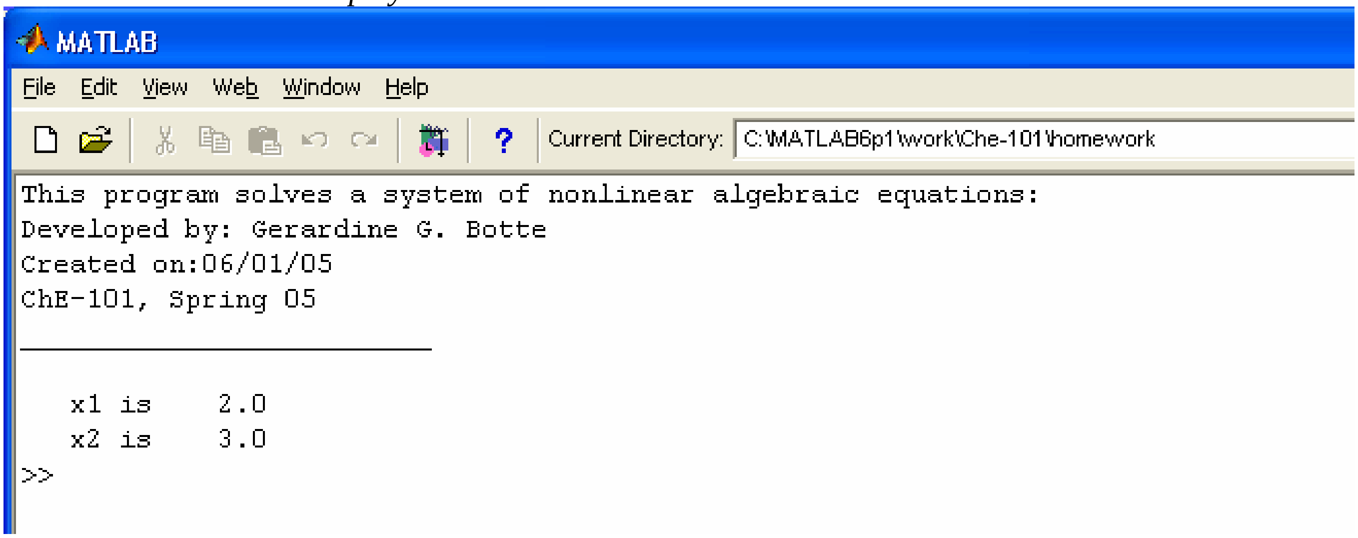
```
function z=equations(x)
%This function is used by the main program system1.m in the fsolve syntax
%   It contains the equations to be solved (notice that z is a vector and x is a vector):
%   z(1) = x(1)^2 + x(1)*x(2)-10; %This is equation #1, therefore element 1 of vector z
%   z(2) = x(2)+3*x(1)*x(2)^2-57; %This is equation # 2, therefore element 2 of vector x
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%Call syntax: z=equations(x)
%Input:
%   1. x: This is the vector of unknows that we are trying to solve using fsolve
%         The input for this variable is transferred from the main file through the initial guess vector
%Ouput:
%   1. z: vector that contains the value of the equations.
%         This will be transferred to the fsolve command in the main program
%________________________________________
z(1) = x(1)^2 + x(1)*x(2)-10; %This is equation #1, therefore element 1 of vector z
z(2) = x(2)+3*x(1)*x(2)^2-57;  %This is equation # 2, therefore element 2 of vector x
```

This is the main program

C:\MATLAB6p1\work\Che-101\homework\system1.m

```
%This program solves a system of nonlinear algebraic equations:
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%ChE-101, Spring 05
% This program calls for the user defined function equations (see that function for details)
%__________
% Program starts
clc
clear
disp('This program solves a system of nonlinear algebraic equations:');
disp('Developed by: Gerardine G. Botte');
disp('Created on:06/01/05');
disp('ChE-101, Spring 05');
disp('_________________________');
fprintf('\n');

%Define initial guesses, you do this in the code you don't ask the user for this
x0 = [10 1]; %This is the initial guess vector to use for the calculation

% Calling for the fsolve command
options = optimset('Largescale','off','Display','off'); %These are the options
x = fsolve('equations',x0, options); %Notice that all the input needed for the function equations
                                           % is being transferred: x0 takes care of x vector to start calculation

%Printing the results
fprintf('   x1 is %6.1f \n',x(1))
fprintf('   x2 is %6.1f \n',x(2))
```

This is what will be displayed on the command window



SOLVED PROBLEMS

1. The van der Waals equation of state is used as a mechanistic model to predict the volume of a gas at high pressures. The equation is given by:

$$\left(P+\frac{a}{v^2}\right)(v-b)=RT$$

where P is the pressure in atm, T is the temperature in K, *v* is the specific volume in l/mol, R is the universal gas constant 0.082 atm l/mol K, and "a" and "b" are known as the attraction and repulsion parameters, respectively. The attraction and repulsion parameters are a function of the critical temperature and critical pressure. For hydrogen the parameters needed are given by:

$$a = 0.245$$

$$b = 0.0266$$

Develop a program in Matlab that will build a table for the specific volume of hydrogen (with 3 decimals) at 350 K and the following pressures: 1 atm, 20 atm, 100 atm, 200 atm. The program should also be able to plot the volume as a function of pressure from 1 to 200 atm. Estimate your initial guesses using ideal gas law.

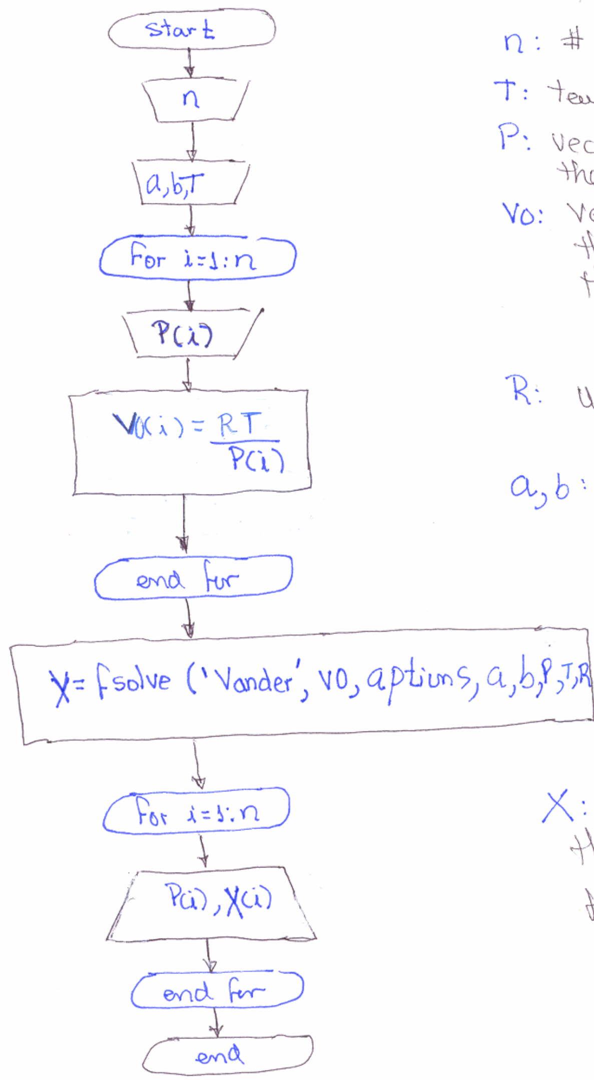
<u>Solution</u>:

1. The first step is to build a flowchart diagram for the program:

SEE NEXT PAGE

Flowchart diagram

1-. Main Program (Pressure.m)



this solves the nonlinear Eq.

Legend:

n: # of data points (pressures)

T: temperature, °K

P: vector that contains the pressure, atm

Vo: Vector that contains the initial guess for the volume, l/mol. Calculated using ideal gas law

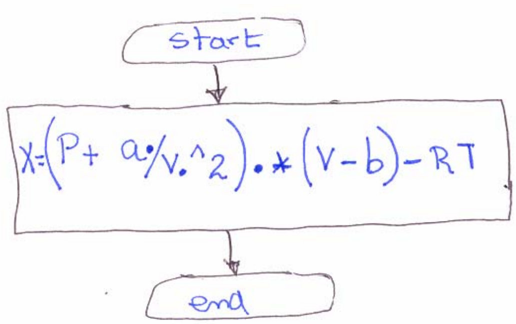
R: Universal gas constant, 0.082 atm l/K mol

a, b: Parameters of Vander Waals Eq for H_2 0.245 and 0.0266, respectively).

Vander: function file that contains the Eq to be solved

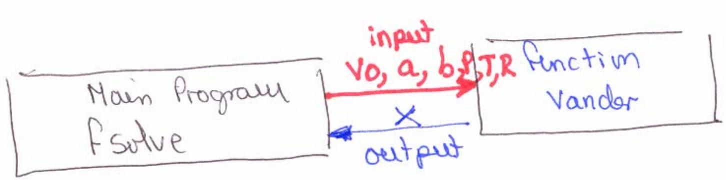
X: Vector that contains the solution for the volume, l/mol.

2.. Flowchart of function file 'Vander.m'



element by element Properties are use (.) because P and V are Vectors.

3. Relationship between main program and function file



2. Build the function file for the equation (Vander.m)

C:\MATLAB6p1\work\Che-101\homework\Vander.m

File Edit View Text Debug Breakpoints Web Window Help

Stack: Base

```
function x=Vander(V,a,b,P,T,R)
%This function is used by the main program Pressure.m in the fsolve syntax
%   It contains the equation to be solved (notice that x, V, and P are vectors):
%   x=(P+a./V.^2).*(V-b)-R*T
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%Call syntax: x=Vander(V,a,b,P,T,R)
%Input:
%   1. V: This is the vector of unknows that we are trying to solve (V in l/mol) using fsolve
%         The input for this variable is transferred from the main file through the initial guess vector (V0)
%   2. a: This is a parameter transferred from the fsolve syntax
%   3. b: This is a parameter transferred from the fsolve syntax
%   4. P: This is a vector that stores Pressure (atm) transferred from the fsolve syntax
%   5. T: This is the temperature (K) transferred from the fsolve syntax
%   6. R: This is the Universal gas constant, transferred from the fsolve syntax
%Ouput:
%   1. x: vector that contains the solution of the equation (l/mol)
%         This will be transferred to the fsolve command in the main program
%__________________________________________________
x=(P+a./V.^2).*(V-b)-R*T; %Element by element operations '.' are used because the equation involves arrays
                          %computations
```

3. Write the main file "Pressure.m"

C:\MATLAB6p1\work\Che-101\homework\Pressure.m

File Edit View Text Debug Breakpoints Web Window Help

Stack: Base

```
%This program calculates the specific volume using Van der Waals Equation of State
%Developed by: Gerardine G. Botte
%Created on:06/01/05
%Last modified on:06/01/05
%ChE-101, Spring 05
% This program calls for the user defined function Vander (see that function for details)
% Solution to Solved Problem 1, Tutorial VII
%___________
% Program starts
clc
clear
disp('This program calculates the specific volume using Van der Waals Equation of State');
disp('Developed by: Gerardine G. Botte');
disp('Created on:06/01/05');
disp('ChE-101, Spring 05');
disp('Solution to Solved Problem 1, Tutorial VII');
disp('___________________________');
fprintf('\n');

%Input for the program
a=input('Please enter the a constant of Van der Waals Eqn =');
b=input('Please enter the b constant of Van der Waals Eqn =');
T=input('Please enter the temperature in K =');
n=input('Please enter the number of data points for the calculation =');

fprintf('\n');

for i=1:n
    fprintf('Pressure # %3i\n',i);
    P(i)=input('Please enter the Pressure in atm = ');
end

%Define initial guesses, using ideal gas law
R=0.082; %Universal gas constant (atm l/mol K)
V0 = R*T./P; %This is the initial guess vector. Notice that the '.' is used because it involves arrays operations
```

```
% Calling for the fsolve command
options = optimset('Largescale','off','Display','off'); %These are the options
x = fsolve('Vander',V0, options,a,b,P,T,R); %Notice that all the input needed for the function Vander
                                            % is being transferred: V0 takes care of V vector to start calculation
                                            % a: takes care of the a constant
                                            % b: takes care of the b constant
                                            % P: takes care of the pressure vector
                                            % T: takes care of the temperature
                                            % R: takes care of the universal gas constant

%Printing the results. This is how the table is built
fprintf('\n');
fprintf('Table 1. Specific Volume of Hydrogen using Van der Waals Equation at %5.1f K\n',T);
fprintf('______________________________\n');
fprintf(' Pressure |  Specific Volume\n');
fprintf(' P, atm   |      V, l/mol  \n');
fprintf('______________________________\n');

for (i=1:n)
    fprintf(' %5.1f    |  %8.3f \n',P(i),x(i));
end
fprintf('______________________________\n');

%This will build the Plot (look at tutorial V.b to review how to make plots in Matlab
plot(P,x,'o-r'); %the format for the data will be red circles connected line

%Format for the plot
xlabel('\fontname{Arial}Pressure, P (atm)','Fontsize',12); %This is the format for the x axis
ylabel('\fontname{Arial}Specific Volume, V (l/mol)','Fontsize',12); %This is the format for the y axis

title('\fontname{Arial}Specific Volume of Hydrogen using Van der Waals Equation as a Function of Pressure','Fontsize',

legend('Temperature = 350 K',0);
```

Vander.m | Pressure.m | botte.m

Ready

This is what will be displayed in the command window:

MATLAB

File Edit View Web Window Help

Current Directory: C:\MATLAB6p1\work\Che-101\homework

```
This program calculates the specific volume using Van der Waals Equation of State
Developed by: Gerardine G. Botte
Created on:06/01/05
ChE-101, Spring 05
Solution to Solved Problem 1, Tutorial VII
_________________________

Please enter the a constant of Van der Waals Eqn =0.245
Please enter the b constant of Van der Waals Eqn =0.0266
Please enter the temperature in K =350
Please enter the number of data points for the calculation =4

Pressure #    1
Please enter the Pressure in atm = 1
Pressure #    2
Please enter the Pressure in atm = 20
Pressure #    3
Please enter the Pressure in atm = 100
Pressure #    4
Please enter the Pressure in atm = 200

Table 1. Specific Volume of Hydrogen using Van der Waals Equation at 350.0 K
______________________________
 Pressure |  Specific Volume
 P, atm   |      V, l/mol
______________________________
   1.0    |    28.718
  20.0    |     1.453
 100.0    |     0.306
 200.0    |     0.164
______________________________
>>
```

Ready

This is the plot: