Matlab Code:

Tuesday, October 16, 2018 3:37 PM

My beattie bridgeman:

```
format long

T=293; %kelvin
Ru=8.314;%J/mol-K
%constants from table
Ao=507.2836;
a=0.07132;
Bo=0.10476;
b=0.07235;
C=660000;


P=100:6000/i:6000;


V = ones(1,6000/i);
Vi=Ru*T./P(1:i); %Initial Value from ideal gas
er = 1;
for i = 1:6000/i

  while er >= 0.001
  vi=Ru*T./P(1,i);
  A=Ao*(1-(a./Vi));
  B=Bo*(1-(b./Vi));
  f=-(A/Vi)-((B*C*Ru)/((T^2)*(Vi^3)))+((B*Ru*T)/(Vi^2))-((C*Ru)/((T^2)*(Vi^2)))+((Ru*T)/(Vi))-P(i);
  df=-A*(T^2)+B*Ru*(T^3)-C*Ru+0.5*Ru*(T^3)*Vi-0.33*P(i)*(T^2)*(Vi^2);
  V(i)= (Ru*T/100) -(f/df);
  er = abs(V(i) - Vi);
  Vi = V(i);
  end
  V(i) = Vi/44.01;
  Vi = 1;
  er = 1;
end




plot(P,V(i),'ob');
  xlabel('Pressure (KPa)');
  ylabel('Specific Volume (m^3/Kg)');

  hold on
  grid on
```

Trystian's van der waals:

```
format long
P = 100:100:6000; %Range of Pressure values used
v0 = 1; %Initial guess used for Newton's method
vv=ones(1,size(P,2));%Array that has all value of 1 for specific volume
error = 1; %Initial value of error
for i = 1:size(P,2)%for loop that iterates over every value of pressure
    while error >= 0.000000005 %While loop used for newton's method
     f0 =  Thermof(P(i),v0);%VDW equation evaluated at a value of pressure and specific volume
     f0p = ThermoDev(P(i),v0);%VDW derivative evaluated at a value of pressure and specific volume
%      if f0p == 0 %If statement that checks if derivative is 0
%         disp('Derivative is 0')
%         disp('Cannot find specific volume');
%      return
%      end
     vv(i) = v0 - f0/f0p;%Newtons method
     error = abs(vv(i) - v0); %Error used for comparison to original error in while loop
     v0 = vv(i); %Resets initial guess for specific volume to previous iteration from Newton's method
    end
    v0 = 1;%Resets guess for specific volume and error
    error = 1;
end
disp(P)%Displays values for pressure
disp(vv)%Displays corresponding values for specific volume
plot(vv,P)%Commands used to graph pressure vs. specific volume and labels for graph
title('Pressure vs. specific volume for CO2 based on the Van Der Waals EOS')
legend('Temperature of 298 Kelvin')
xlabel('v (m^3/kg)')
```

My Van Der Waals:

```
%Van Der Waals using Newton Method

P1=100; % K-pascals
P2=6000;
T=293; %kelvin
R=0.1889; % Kj/Kmol-Kelvin
Tcr=304; % Kelvin
Pcr=7380; %K Pa
a=(27*(R^2)*(Tcr^2))/(64*Pcr);
b=(R*Tcr)/(8*Pcr);
i=500;
P=P1;%allowable error
P2=P2+i; %estimate of distance to root
v=zeros(1,6000/i);
v(1:1)= 0.55347; % use ideal gas law for first estimate [ (R*T/Pinitial) ]
n=1;
while P<P2
   n=n+1;
   V=v(1,n-1);
   f=(R*T)/(V-b)-(a/(V^2))-P;
   df=-(R*T)/(V-b)^2+(2*a)/(V^3);
   v(1,n)=V-(f/df);

  P=P+1;
end
disp(v)
plot(v);
```

```
disp(vv)%Displays corresponding values for specific volume
plot(vv,P)%Commands used to graph pressure vs. specific volume and labels for graph
title('Pressure vs. specific volume for CO2 based on the Van Der Waals EOS')
legend('Temperature of 298 Kelvin')
xlabel('v (m^3/kg)')
ylabel('P (kPa)')
```

My newest beattie-bridgeman

```
format long
%problem constants:
T=293; %kelv(1,n)n
Ru=8.314;%J/mol-K

%constants from table:
Ao=507.2836;
a=0.07132;
Bo=0.10476;
b=0.07235;
C=660000;
%end of constants

%initializations
error=1;
P=100:100:6000;%Range of Pressure values used
v=zeros(size(P));
vi=0.55347; % use ideal gas law for first estimate [ (R*T/Pinitial) ]
%end of initializations
for n=3:size(P,2)

while error>=0.001 %error = allowable tollerance =0.001


% v=v(1,n-1);   %v is previous guess or Xn, v is current estimate Xn+1
A=Ao*(1-(a/vi));
B=Bo*(1-(b/vi));
f=-(A/vi)-((B*C*Ru)/((T^2)*(vi^3)))+((B*Ru*T)/(vi^2))-((C*Ru)/((T^2)*(vi^2)))+((Ru*T)/(vi))-P(n);
df=-A*(T^2)+B*Ru*(T^3)-C*Ru+0.5*Ru*(T^3)*vi-0.33*P(n)*(T^2)*(vi^2);
v(1,n)=vi-(f/df);%
error=abs(v(1,n)-vi);
vi=v(1,n);

end
error=1;
v(1:1)=0.55347;

end
```

```
i=i+1;
end
disp(v)
plot(v);
```

Merge of mine and trystans (not functional)

```
format long
T=293; %kelvin
R=0.1889; % Kj/Kmol-Kelvin
Tcr=304; % Kelvin
Pcr=7380; %K Pa
P = 100:100:6000; %Range of Pressures
v0 = 0.55347; %Initial guess based on ideal gas law
V=ones(1,size(P,2));%Array that has value initialized at 1 for specific volume
error = 1; %Initialized
a=(27*(R^2)*(Tcr^2))/(64*Pcr);
b=(R*Tcr)/(8*Pcr);

for i = 1:size(P,2)%    for loop that iterates through every value of pressure

    while error >= 0.001 %While loop to determine when estimated root is within tollerable distance from actual
    f=(R*T)./(V-b)-(a./(V.^2))-P;
    df=-(R*T)/(V-b)^2+(2*a)/(V^3);

     V(i) = v0 - f/df;%Newtons method
     error = abs(V(i) - v0); %Error used for comparison
     v0 = V(i); %Resets initial guess for specific volume to previous iteration from Newton's method
    end
    v0 = 0.55347;%Resets guess for specific volume and error
    error = 1;
end
disp(P)%Displays values for pressure
disp(V)%Displays corresponding values for specific volume
plot(V,P)%Commands used to graph pressure vs. specific volume and labels for graph
title('Pressure vs. specific volume for CO2 based on the Van Der Waals EOS')
legend('Temperature of 298 Kelvin')
xlabel('v (m^3/kg)')
ylabel('P (kPa)')
```