Oppgave 5

Matlab-program som beregner van der Waals koeksistenstrykk for luft ved temperatur 123 K. I denne versjonen leses det inn en gjetning på væskevolumet, hvoretter det beregnes hvilket gassvolum som gir samme trykk. Forskjellen mellom arealet under $p(V)$ og arealet under den rette linjen bestemmes og skrives ut. Deretter må programmet startes på nytt dersom man vil foreta en ny og bedre gjetning på væskevolumet.

Med en temperatur på 123 K blir koeksistenstrykket lik 25.021 bar med en nøyaktighet på 1 mbar. Tilsvarende molare væske- og gassvolum er hhv 0.0672 og 0.250 L.

```
%%FY1005/TFY4165, Oving 9, Oppgave 5: van der Waals koeksistenstrykk
%%
%%Programmet tegner opp p(V) og ber om initialverdi for vaeskevolumet.
%%Deretter beregnes gassvolumet som gir samme trykk.
%%Endelig beregnes integralet av p(V) numerisk og sammenlignes med
%%arealet under den rette linjen.
%%
%%Velg utskrift med flere gjeldende siffer enn det som er standard
format long;
%%Temperatur i Kelvin
T=123;
%%V = molart volum (L/mol)
Vmin=0.05;
Vmax=0.5;
NV=500;
%%V = vektor med verdier mellom Vmin og Vmax, i alt NV verdier
V=linspace(Vmin,Vmax,NV);
%%Verdier for a og b for luft: 1.368  0.0367
%%Enheter: [a] = bar*(L/mol)^2 og [b] = L/mol
R=8.134;
a=1.368;
b=0.0367;

%%van der Waals tilstandsligning:
p = (R*T./(V-b))/100 - a./(V.*V);
%%Plott av isotermen
fig = plot(V,p);
axis([0 Vmax 0 70]);
%%Tekst p figuren
title('Isoterm: van der Waals gass','fontsize',18);
xlabel('Molart volum (L/mol)','fontsize',18);
ylabel('Trykk (bar)','fontsize',18);
%%Med hold on; vil plottene "overleve", saa vi kan sammenlikne.
hold on;
%%Les inn V1:
```

```
V1 = input('Gjett p V1:  ')
%%Riktig verdi er omtrent V1=0.06724129;
%%f.eks. 0.06724 gir 4*E-4 i feil for T=123K.
%%Bestemmer tilsvarende P1:
P1 = (R*T/(V1-b))/100 - a/(V1*V1);
%%Her har vi et volum som er mye storre enn V2:
V_stor=0.50;
%%Setter steglengden dV "liten nok"
dV=0.000001;
%%Bestemmer trykket for volumet V_stor
p_find=(R*T/(V_stor-b))/100 - a/(V_stor*V_stor);
%%Betingelsen V_stor > 0 gjor for at programmet stopper hvis noe gaar galt
while ( p_find < P1 && V_stor > 0)
V_stor = V_stor-dV;
p_find=(R*T/(V_stor-b))/100 - a/(V_stor*V_stor);
end;
%%Naa har vi funnet P2, V2, i den siste verdien:
V2=V_stor;
P2=p_find;
%%Skriver ut trykk- og volumverdier:
disp('V1: ')
disp(V1)
disp('V2: ')
disp(V2)
disp('P1: ')
disp(P1)
disp('P2: ')
disp(P2)
%%Estimat av usikkerhet i trykket P2:
dp = (- (R*T/((V2-b)*(V2-b)))/100 + 2.0*a/(V2*V2*V2))*dV;
disp('Estimert usikkerhet i trykket P2: ')
disp(abs(dp))
disp('Faktisk feil i P2: ')
disp(abs(P2-P1))
%%Tegner en linje paa plottet:
plot([V1,V2],[P1,P2]);
%%Saa skal vi integrere trykket fra V1 til V2, og sammenlikne med
%%P*(V2-V1), dvs. arealet under linja. For aa integrere med quad() maa
%%vi forst definere en symbolsk funksjon F:
%%"@(v)" sier at v er variabelen vaar, resten er likningen for P(v).
F = @(v) ((R*T)./(v-b))/100-a./(v.*v);
%%Numerisk integrasjon av F fra V1 til V2 med saakalt adaptiv
%%Simpson-kvadratur.
I = quad(F, V1, V2);
%%
```

```
disp('Feilen er:')
disp(I-P1*(V2-V1))
%%Tegner en linje paa plottet:
plot([V1,V2],[P1,P2]);
disp('Kjoer programmet paa nytt med ny gjetning for V1 dersom')
disp('feilen er for stor.')
```

I versjonen nedenfor leses det inn en gjetning på væskevolumet og gassvolumet, samt ønsket nøyaktighet i koeksistenstrykket. Programmet beregner deretter korrekt væske- og gassvolum samt koeksistenstrykk med den innleste nøyaktigheten.

```
%%FY1005/TFY4165, Oving 9, Oppgave 5: van der Waals koeksistenstrykk
%%
%%Programmet tegner opp p(V) og ber om initialverdi for vaeske- og
%%gassvolumet, samt onsket noyaktighet. Koeksistenstrykket regnes
%%deretter ut automatisk.
%%
%%Velg utskrift med flere gjeldende siffer enn det som er standard
format long;
%%Temperatur i Kelvin
T=123;
%%V = molart volum (L/mol)
Vmin=0.05;
Vmax=0.5;
NV=500;
%%V = vektor med verdier mellom Vmin og Vmax, i alt NV verdier
V=linspace(Vmin,Vmax,NV);
%%Verdier for a og b for luft: 1.368  0.0367
%%Enheter: [a] = bar*(L/mol)^2 og [b] = L/mol
R=8.134;
a=1.368;
b=0.0367;
%%van der Waals tilstandsligning
p = (R*T./(V-b))/100 - a./(V.*V);
%%Plott av isotermen
fig = plot(V,p);
axis([0 Vmax 0 70]);
%%Tekst paa figuren
title('Isoterm: van der Waals gass','fontsize',18);
xlabel('Molart volum (L/mol)','fontsize',18);
ylabel('Trykk (bar)','fontsize',18);
%%Saa vi kan tegne en rett linje i samme figur til slutt:
hold on;

%%Oppgi V1:
```

```
V1 = input('Gjett vaeskevolum V1: ')
epsilon=input('Angi tillatt usikkerhet i koeksistenstrykket: ')
%%Vi setter variabelen err storre enn epsilon,
%%for aa komme i gang med while-lokka nedenfor:
err=2*epsilon;
%%Vi teller hvor mange ganger den ytterste lokken har kjort:
n=0;
%%Oppgi en startverdi for V_stor som sikkert er storre enn gassvolumet V2:
V_stor = input('Startverdi for gassvolumbestemmelse: ')
%%Lagrer innlest V_stor som V_stornok for senere bruk
V_stornok = V_stor;
%%Saa lenge feilen er storre enn epsilon prover vi paa nytt.
%%Vi stopper ogsaa hvis vi har iterert mange ganger (mer enn 50).
while (abs(err) > epsilon && n < 50)
   disp('--------------------')
   disp('Iterasjon nr.:  ')
   n = n + 1;
   disp(n)
   P1 = (R*T/(V1-b))/100 - a/(V1*V1);
   %%Siden err dimensjonsmessig tilsvarer et trykk, settes steglengden
   %%dV for volumet lik 0.05*V1*err/P1. Da haaper vi at feil i V1 og V2 ikke
   %%vil dominere den totale numeriske feilen. Faktoren 0.05 er basert
   %%paa proving og feiling.
   dV=0.05*V1*err/P1
   p_find=(R*T/(V_stor-b))/100 - a/(V_stor*V_stor);
   %%Betingelsen V_stor > 0 sorger for at vi stopper hvis noe gaar galt.
   while (p_find < P1 && V_stor > 0)
V_stor = V_stor - abs(dV);
p_find=(R*T/(V_stor-b))/100 - a/(V_stor*V_stor);
   end
   %%Naa vil V_stor og p_find med god tilnaermelse vaere lik hhv V2 og P2,
   %%dvs gassvolum og tilsvarende trykk:
   V2=V_stor
   P2=p_find;
   %%Saa skal vi integrere trykket fra V1 til V2, og sammenlikne med P*(V2-V1):
   %%For aa integrere med quad() maa vi forst definere en symbolsk funksjon F:
   F = @(v) ((R*T)./(v-b))/100-a./(v.*v);
   %%Numerisk integrasjon av F fra V1 til V2 med adaptiv Simpson-kvadratur:
   I = quad(F, V1, V2);
   %%Regner ut forskjellen mellom rektangelareal og areal under p(V)-kurven,
   %%og deretter et estimat err for feilen i trykket, ved aa dividere med V2-V1:
   err = (I-P1*(V2-V1))/(V2-V1);
   disp('Feilen er:  ')
   disp(err)
   %%Foreta endring i V1. Hvis err > 0, maa V1 reduseres, ergo
```

```
    %%subtraherer vi dV (som ble regnet ut lenger opp):
    V1 = V1 - dV
    %%Ny gjetning paa V_stor settes lik V_stornok
    V_stor = V_stornok;
    %%Skriv ut P- og V-verdiene ved aa fjerne %% i de neste linjene:
    %%V1=V1
    %%V2=V2
    %%P1=P1
    %%P2=P2
end

%Tegner linjen som gir like arealer
plot([V1,V2],[P1,P2]);
%Skriver ut verdier for losningen
disp('Vaeskevolum V1: ')
disp(V1)
disp('Gassvolum V2: ')
disp(V2)
disp('Trykket P1: ')
disp(P1)
disp('Trykket P2: ')
disp(P2)
hold off;
```

Figur: