Peng_Robinson_EOS

November 25, 2025

1 Generalized Peng-Robinson Equation of State

Routines to calcualte the Generalized Peng-Robinson Equation of State

SIS is Stanley I. Sandler, *Chemical, Biochemcial and Engineering Thermodynamics*, 5th ed.

Eric Furst
November 2025

The Generalized Peng-Robinson equation of state is

$$P = \frac{RT}{\underline{V} - b} - \frac{a(T)}{\underline{V}(\underline{V} + b) + b(\underline{V} - b)} \qquad \text{(Eq. 6.4-2)}$$

with

$$b = 0.07780\frac{RT_c}{P_c} \qquad \text{(Eq. 6.7-2)}$$

$$a(T) = a(T_c)\alpha(T) = 0.45724\frac{R^2T_c^2}{P_c}\alpha(T) \qquad \text{(Eq. 6.7-1)}$$

$$\sqrt{\alpha} = 1 + \kappa\left(1 - \sqrt{\frac{T}{T_c}}\right) \qquad \text{(Eq. 6.7-3)}$$

$$\kappa = 0.37464 + 1.54226\omega - 0.26992\omega^2 \qquad \text{(Eq. 6.7-4)}$$

The acentric factor ω and the crticial temperatures and pressures are given in SIS table 6.6-1.

Calculating the pressure P given $\underline{V}$ and T is straightforward, but to calculate the molar volume given P and T, we need to solve the cubic equation of state of the form

$$Z^3 + \alpha Z^2 + \beta Z + \gamma = 0 \qquad \text{(Eq. 6.4-4)}$$

where Z is the compressibility factor

$$Z = \frac{P\underline{V}}{RT}$$

For the Peng-Robinson EOS (see SIS Table 6.4-3),

$$\alpha = -1 + B$$
$$\beta = A - 3B^2 - 2B$$
$$\gamma = -AB + B^2 + B^3$$

and

$$A = \frac{aP}{(RT)^2}$$
$$B = \frac{bP}{RT}$$

[22]:
```
"""
Generalized Peng-Robinson EOS
PR_pressure returns the pressure given V, T, Pc, Tc, omega
PR_volume returns all molar volumes (real roots of EOS) given P, T, Pc, Tc,␣
 ↪omega
"""
import numpy as np
from scipy import constants
from numpy.polynomial import Polynomial

R = constants.R # Set the gas constant to R

def calc_b(Pc,Tc):
    return 0.07780*R*Tc/Pc

def calc_a(T,Pc,Tc,omega):
    kappa = 0.37464 + 1.54226*omega - 0.26992*omega**2
    sqrtalpha = 1 + kappa*(1-np.sqrt(T/Tc))
    return 0.45724*R**2*Tc**2/Pc*sqrtalpha**2

# Calculate the presure given V, T for PR EOS
def PR_pressure(V,T,Pc,Tc,omega):
    a = calc_a(T,Pc,Tc,omega)
    b = calc_b(Pc,Tc)

    P = R*T/(V-b) - a/(V*(V+b)+b*(V-b))
    return P

# Calculate the molar volume given P, T for PR EOS in m^3/mol
# Note that we can return multiple real roots (up to three)
# The largest and smallest will be the vapor and liquid, respectively
def PR_volume(P, T, Pc, Tc, omega):
    # Calculate a, b, A, and B
    a = calc_a(T, Pc, Tc, omega)
    b = calc_b(Pc, Tc)
```

```
    A = a*P/R**2/T**2
    B = b*P/R/T

    # Definitions of alpha, beta, gamma in SIS Table 6.4-3 for PR EOS
    alpha = -1 + B
    beta = A - 3*B**2 -2*B
    gamma = -A*B + B**2 + B**3

    # polynomial with coefficients in increasing order: c0 + c1 x + c2 x**2 + ..
 ↪.
    p = Polynomial([ gamma, beta, alpha, 1 ])

    roots = p.roots()        # returns all (possibly complex) roots of Z
    real_roots = roots.real[abs(roots.imag) < 1e-12]  # filter real ones

    # Convert real values of Z to molar volume
    V = real_roots*R*T/P

    return V
```

1.1 Example calculation

Given the PR EOS, calculate the molar volume(s) of ethane given different temperature and pressures for ethane.

[24]:
```
# Given the PR EOS, calculate the molar volume(s) of ethane given different T␣
 ↪and P:

# Data for ethane
Pc = 4.884e6
Tc = 305.4
omega = 0.098
MW = 30.07

P = 1e6
T = -33+273.15

# molar volume m^3/mol
print((PR_volume(P, T, Pc, Tc, omega)))

# specific density kg/m^3
print(1/(PR_volume(P, T, Pc, Tc, omega)/MW*1000))
```

```
[6.08349837e-05 2.38749240e-04 1.65668503e-03]
[494.28796012 125.94804501  18.15070421]
```

[ ]:
```

```