

Real Gas Effect in Reacting Flow

This code repository is used to evaluate the advantages of the proposed real gas equation of state (EoS)

1. Code Environment

```
# Cantera used in /opt/cantera_libs/cantera, one can modified code in `src` and  
build & install by  
conda activate cantera  
python3 /usr/bin/scons build  
python3 /usr/bin/scons install
```

```
# The code in this repo can be run by  
source /usr/local/bin/setup_cantera  
python3 src/xxx.py
```

2. Basic Theory

1.0 Generalized EoS

Helmholtz Free Energy: $F \equiv U - TS = A(T, v, n)$, where U is the internal energy of the system, T is the absolute temperature, S is the entropy of the system.

- 1st law of thermodynamics: $dU = \delta Q + \delta W$
- 2nd law of thermodynamics: $\delta Q = TdS$
- pressure work: $\delta W = -pdV$

Thus we have:

- $dU = TdS - pdV$
- $\Rightarrow dU = d(TS) - SdT - pdV$
- $\Rightarrow d(U - TS) = -SdT - pdV$
- $\Rightarrow dF = -SdT - pdV$

Therefore,

$$\begin{aligned} S &= -\left(\frac{\partial F}{\partial T}\right)_V \\ p &= -\left(\frac{\partial F}{\partial V}\right)_T \\ U &= F + TS = F - T\left(\frac{\partial F}{\partial T}\right)_V \\ H &= U + pV = F - T\left(\frac{\partial F}{\partial T}\right)_V - V\left(\frac{\partial F}{\partial V}\right)_T \\ C_p &= \left(\frac{\partial H}{\partial T}\right)_p = T \left[\left(\frac{\partial^2 F}{\partial V \partial T}\right)_{V,T}^2 / \left(\frac{\partial^2 F}{\partial V^2}\right)_T - \left(\frac{\partial^2 F}{\partial T^2}\right)_V \right] \\ C_V &= \left(\frac{\partial U}{\partial T}\right)_V = -T\left(\frac{\partial^2 F}{\partial T^2}\right)_V \end{aligned}$$

1.1 Ideal Gas

$$p = \frac{nRT}{V_m}$$

where V_m is the molar volume of the gas .

1.2 Van der Waals (vdW) real gas equation 1783

$$p = \frac{RT}{V_m - b} - \frac{a}{V_m^2}$$

where b is the volume that is occupied by one mole of the molecules. vdW is a cubic type EoS.

1.4 Redlich-Kwong (RK) real gas equation 1948

$$p = \frac{RT}{V_m - b} - \frac{a}{\sqrt{T}V_m(V_m + b)}$$

where a is a constant that corrects for attractive potential of molecules and b is a constant that corrects for volume. The constants are different depending on which gas is being analyzed and can be calculated from the critical point data of corresponding gas:

$$a = \frac{1}{9(\sqrt[3]{2} - 1)} \frac{R^2 T_c^{2.5}}{P_c}$$
$$b = \frac{\sqrt[3]{2} - 1}{3} \frac{RT_c}{P_c}$$

Where T_c is the temperature at the critical point and P_c is the pressure at the critical point. Noted that a and b are solved analytically from the thermodynamic criteria for the critical point:

$$\left(\frac{\partial P}{\partial V} \right)_T = 0, \left(\frac{\partial^2 P}{\partial V^2} \right)_T = 0.$$

For more details, please refer to original paper or wikipedia of Redlich Kwong equation of state.

1.5 Soave modified Redlich-Kwong (SRK) real gas equation 1972

In 1966, Barner noted that the acentric factor ω can be used to improve RK EoS. SRK is proposed by Soave as:

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{V_m(V_m + b)}$$

where

$$\alpha = \left[1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - \sqrt{T_r}) \right]^2$$
$$a = \frac{1}{9(\sqrt[3]{2} - 1)} \frac{R^2 T_c^2}{P_c}$$
$$b = \frac{\sqrt[3]{2} - 1}{3} \frac{RT_c}{P_c}$$

where $T_r = T/T_c$ is the reduced temperature of the compound. Noted that $\frac{1}{\sqrt{T}}$ term in RK is vanished and its effect is put into α in SRK.

1.6 Peng-Robinson (PR) real gas equation 1976

The Peng-Robinson EoS further modified the RK EoS by modifying the attractive term, giving

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{V_m(V_m + b) + b(V_m - b)}$$

where

$$\alpha = \left[1 + (0.37464 + 1.54226\omega - 0.26992\omega^2)(1 - \sqrt{T_r}) \right]^2$$

$$a = 0.457235 \frac{R^2 T_c^2}{P_c}$$

$$b = 0.077796 \frac{RT_c}{P_c}$$

1.7 PR-RK real gas equation 2005

The limited accuracy of the two-parameter cubic EoS is caused by the inconsistency in the density dependence of its equation rather than the empirical nature of model constants a and b . Many researchers proposed three parameters EoS. In 2005, Cismonti and Mollerup have proposed a three-parameter cubic EoS (hereafter RK-PR EoS) by introducing an additional parameter, δ_1 .

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{(V_m + \delta_1 b)(V_m + \delta_2 b)}$$

where δ_2 is not independent and $\delta_2 = (1 - \delta_1)(1 + \delta_1)$. And it will degenerate to SRK when $\delta_1 = 1, \delta_2 = 0$; and degenerate to PR when $\delta_1 = 1 + \sqrt{2}, \delta_2 = 1 - \sqrt{2}$.

1.7 Mixture properties

When handling mixture, to obtain the aggregate parameters a_m, b_m of the cubic EoS, mixing rules from the van der Waals one-fluid theory is applied:

$$a_m = \sum_i^{N_s} \sum_j^{N_s} X_i X_j a_{ij}$$

$$b_m = \sum_i^{N_s} X_i b_i$$

where X_i is the mole fraction of the i^{th} components of the mixture, X_j is the mole fraction of the j^{th} components of the mixture. Besides, a_{ij} is the attractive term between a molecule of species i and species j , which is usually calculated by $a_{ij} = (a_i a_j)^{1/2}$

2. The calibration model of alpha

The α is previously calculated by Sovae-type functions like

- In SRK: $\alpha = \left[1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - \sqrt{T_r}) \right]^2$
- In PR: $\alpha = \left[1 + (0.37464 + 1.54226\omega - 0.26992\omega^2)(1 - \sqrt{T_r}) \right]^2$

But those are empirical approximations and would lead to notable effects in particular thermodynamically sensitive simulations. Here a **Gaussian Process (GP) based semi-parametric method** is used to utilize the information of experimental data as most as possible.

2.1 The GP model

Here we're trying to fit a function that can map the inputs including T_r to the output α . Using the concept of Gaussian Process (GP) model, given the observations (\mathbf{X}, y) , when applying GP to predict the target y_* at new input \mathbf{X}_* , one have

$$\begin{aligned}\mu(y_*) &= \mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y} \\ \sigma(y_*) &= \mathbf{K}_{**} - \mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{K}_*\end{aligned}$$

where σ_n^2 is the covariance of estimated Gaussian noise from observed data (\mathbf{X}, y) , \mathbf{K} related terms are the covariance matrices as

$$\begin{aligned}\mathbf{K} &= \text{cov}(\mathbf{X}, \mathbf{X}) \\ \mathbf{K}_* &= \text{cov}(\mathbf{X}, \mathbf{X}_*) \\ \mathbf{K}_{**} &= \text{cov}(\mathbf{X}_*, \mathbf{X}_*)\end{aligned}$$

2.2 Kernel functions

The covariance are usually obtained via kernel functions, which are very important for GP's data training. The most famous ones are:

- Squared exponential (or say Radial Basis Function, RBF):

$$\text{cov}(x_i, x_j) = \exp \left(-\frac{d(x_i, x_j)^2}{2l^2} \right)$$

where $d(x_i, x_j)$ is the Euclidean distance of x_i and x_j , l is the length scale of kernel size.

- Matern:

$$k(x_i, x_j) = \frac{1}{\Gamma(\nu) 2^{\nu-1}} \left(\frac{\sqrt{2\nu}}{l} d(x_i, x_j) \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}}{l} d(x_i, x_j) \right),$$

where Γ is the Gamma function, K_ν is the modified Bessel function of order ν .

2.3 Basis functions

The basis function provides an important ability to model the mean of y to zero.

$$y(\mathbf{x}) = f(\mathbf{x}; \boldsymbol{\theta}) + b_{\boldsymbol{\theta}}(\mathbf{x}) + \epsilon$$

where ϵ is the (Gaussian) stochastic error in the database, $f(\mathbf{x}; \boldsymbol{\theta})$ is a basis function that can model $y(\mathbf{x})$ as much as possible, and $b_{\boldsymbol{\theta}}(\mathbf{x})$ is the bias term that can compensate the error between $f(\mathbf{x}; \boldsymbol{\theta})$ and $y(\mathbf{x})$.

To solve this optimization problem, let $f(\mathbf{x}; \boldsymbol{\theta}) = \theta_3 x_2 + \theta_2 x_1 + \theta_1$ as a linear combination of the input variables \mathbf{x} , where the input variable are $\mathbf{x} = [x_1, x_2]$, x_1 and x_2 are the input from each dimension; and $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]$ are corresponding parameters in the linear combination function f .

The corresponding kernel function then be formed as:

$$k(x_i, x_j) = k_0(x_i, x_j) - h_{\boldsymbol{\theta}}(x_i)^T \mathbf{H}_{\boldsymbol{\theta}} h_{\boldsymbol{\theta}}(x_j)$$

where $h_{\boldsymbol{\theta}}(x)$ is the 1st order global sensitivity obtained from:

$$h_{\boldsymbol{\theta}}(x) = \int_{\mathbf{X}} \nabla_{\boldsymbol{\theta}} f(\mathbf{x}) k_0(x, \xi) d\xi$$

the 2nd order global sensitivity, i.e., Hessian matrix is

$$\mathbf{H}_\theta = \int_{\mathbf{X}} \int_{\mathbf{X}} (\nabla_\theta f(\xi)) (\nabla_\theta f(\zeta))^T k_0(\xi, \zeta) d\xi d\zeta$$

The parameter gradient of f shows

$$\nabla_\theta f(\mathbf{x}) = \frac{\partial}{\partial \theta} f(\mathbf{x}; \theta) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$$

The matrix constructed by gradients' outer product of randomly samples ξ, ζ

$$(\nabla_\theta f(\xi)) (\nabla_\theta f(\zeta))^T = \begin{bmatrix} 1 \\ \xi_1 \\ \xi_2 \end{bmatrix} \begin{bmatrix} 1 & \zeta_1 & \zeta_2 \end{bmatrix} = \begin{bmatrix} 1 & \zeta_1 & \zeta_2 \\ \xi_1 & \xi_1 \zeta_1 & \xi_1 \zeta_2 \\ \xi_2 & \xi_2 \zeta_1 & \xi_2 \zeta_2 \end{bmatrix}$$

The kernel function k_0 is:

$$k_0(\xi, \zeta) = \exp \left(- \sum_i \frac{(\xi_i - \zeta_i)^2}{\gamma_i} \right) = \exp \left(- \frac{(\xi_1 - \zeta_1)^2}{\gamma_1} \right) \exp \left(- \frac{(\xi_2 - \zeta_2)^2}{\gamma_2} \right)$$

where γ_1, γ_2 are the parameters of kernel function in dimension x_1 and x_2 .

The corresponding distribution of the evaluated output y_* at new x_* should be:

$$\begin{aligned} \mu(y_*) &= f(x_*; \theta) + \mathbf{K}_*^T \left(\mathbf{K} + \frac{\mathbf{K}_{**}}{\nu} \mathbf{I} \right)^{-1} [y - f(\mathbf{X}; \theta)] \\ \sigma(y_*) &= \nu + \mathbf{K}_{**} - \nu \mathbf{K}_*^T \left(\mathbf{K} + \frac{\mathbf{K}_{**}}{\nu} \mathbf{I} \right)^{-1} \mathbf{K}_* \end{aligned}$$

where ν is a measurement of possible noises, its maximum likelihood estimation (MLE) $\hat{\nu}$ is

$$\hat{\nu} = \frac{1}{N} [y - f(\mathbf{X}; \theta)]^T (\mathbf{K} + g\mathbf{I})^{-1} [y - f(\mathbf{X}; \theta)]$$

where g is the original possible noise level, which still should be estimated by optimization program.

2.4 Implementation

2.5 Validation

3. Appendix

Using the chain rule from Table 5, Page 4 of [Partial derivatives of thermodynamic state properties for dynamic simulation](#),

$$C_p = \left(\frac{\partial H}{\partial T} \right)_p = \left(\frac{\partial H}{\partial T} \right)_V - \left(\frac{\partial H}{\partial V} \right)_T \left(\frac{\partial p}{\partial T} \right)_V \left(\frac{\partial p}{\partial V} \right)_T^{-1}$$

where

$$\begin{aligned}
\left(\frac{\partial H}{\partial T}\right)_V &= \left(\frac{\partial F}{\partial T}\right)_V - \left(\frac{\partial F}{\partial T}\right)_V - T\left(\frac{\partial^2 F}{\partial T^2}\right)_V - V\left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} \\
&= -T\left(\frac{\partial^2 F}{\partial T^2}\right)_V - V\left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} \\
\left(\frac{\partial H}{\partial V}\right)_T &= \left(\frac{\partial F}{\partial V}\right)_T - T\left(\frac{\partial^2 F}{\partial T\partial V}\right)_{V,T} - \left(\frac{\partial F}{\partial V}\right)_T - V\left(\frac{\partial^2 F}{\partial V^2}\right)_T \\
&= -T\left(\frac{\partial^2 F}{\partial T\partial V}\right)_{V,T} - V\left(\frac{\partial^2 F}{\partial V^2}\right)_T \\
\left(\frac{\partial p}{\partial T}\right)_V &= -\left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} \\
\left(\frac{\partial p}{\partial V}\right)_T &= -\left(\frac{\partial^2 F}{\partial V^2}\right)_T
\end{aligned}$$

therefore,

$$\begin{aligned}
C_p &= \left(\frac{\partial H}{\partial T}\right)_V - \left(\frac{\partial H}{\partial V}\right)_T \left(\frac{\partial p}{\partial T}\right)_V \left(\frac{\partial p}{\partial V}\right)_T^{-1} \\
&= -T\left(\frac{\partial^2 F}{\partial T^2}\right)_V - V\left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} + \left[T\left(\frac{\partial^2 F}{\partial T\partial V}\right)_{V,T} + V\left(\frac{\partial^2 F}{\partial V^2}\right)_T\right] \left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} / \left(\frac{\partial^2 F}{\partial V^2}\right)_T \\
&= -T\left(\frac{\partial^2 F}{\partial T^2}\right)_V + T\left(\frac{\partial^2 F}{\partial T\partial V}\right)_{V,T} \left(\frac{\partial^2 F}{\partial V\partial T}\right)_{T,V} / \left(\frac{\partial^2 F}{\partial V^2}\right)_T \\
&= T \left[\left(\frac{\partial^2 F}{\partial V\partial T}\right)_{V,T}^2 / \left(\frac{\partial^2 F}{\partial V^2}\right)_T - \left(\frac{\partial^2 F}{\partial T^2}\right)_V \right]
\end{aligned}$$