

## Exercício de aplicação para o capítulo 8

O Potencial de Helmholtz para o etanol, de acordo com o artigo “*A Fundamental Equation of State for Ethanol*” é obtido pela expressão

$$\frac{a(\rho, T)}{RT} = \alpha = \alpha^0(\delta, \tau) + \alpha^r(\delta, \tau) \quad (1)$$

Em que  $a$  é energia de Helmholtz,  $\alpha$  a energia de Helmholtz adimensional,  $\alpha^0$  a contribuição de gás ideal da energia adimensional de Helmholtz e  $\alpha^r$  a porção residual da energia. Onde

$$\delta = \rho/\rho_c \quad (2)$$

$$\tau = T_c/T \quad (3)$$

Sendo  $\rho_c = 5.93 \text{ mol. dm}^{-3}$  a densidade crítica e  $T_c = 514,71 \text{ K}$  a temperatura crítica. Os termos  $\alpha^0$  e  $\alpha^r$  são escritos como

$$\alpha^0 = \ln \delta + x_1 \ln \tau + x_2 + x_3 \tau + \sum_{i=4}^7 x_i \ln[1 - \exp(-y_i \tau)] \quad (4)$$

$$\begin{aligned} \alpha^r = & \sum_{i=1}^6 N_i \delta^{d_i} \tau^{t_i} + \sum_{i=7}^{16} N_i \delta^{d_i} \tau^{t_i} \exp(\delta^{l_i}) \\ & + \sum_{i=17}^{25} N_i \delta^{d_i} \tau^{t_i} \exp[-\eta_i(\delta - \epsilon_i)^2 - \beta_i(\tau - \gamma_i)^2] \end{aligned} \quad (5)$$

As constantes  $x_i$  e  $y_i$  são dadas pela tabela 18.

TABLE 18. Constants of the ideal gas Helmholtz energy equation, Eq. (19)

$i$	$x_i$	$y_i$
1	3.430 69	—
2	−12.7531	—
3	9.390 94	—
4	2.143 26	0.816 771
5	5.092 06	2.591 75
6	6.601 38	3.804 08
7	5.707 77	8.587 36

E  $N_i, t_i, d_i, l_i, \eta_i, \beta_i, \gamma_i$  e  $\epsilon_i$  pela tabela 19.

TABLE 19. Constants of the residual portion of the fundamental equation of state, Eq. (21)

$i$	$N_i$	$l_i$	$d_i$	$l_i$	$\eta_i$	$\beta_i$	$\gamma_i$	$\epsilon_i$
1	0.058200796	1.00	4					
2	0.94391227	1.04	1					
3	-0.80941908	2.72	1					
4	0.55359038	1.174	2					
5	-1.4269032	1.329	2					
6	0.13448717	0.195	3					
7	0.42671978	2.43	1	1				
8	-1.1700261	1.274	1	1				
9	-0.92405872	4.16	1	2				
10	0.34891808	3.30	3	1				
11	-0.91327720	4.177	3	2				
12	0.022629481	2.50	2	1				
13	-0.15513423	0.81	2	2				
14	0.21055146	2.02	6	1				
15	-0.21997690	1.606	6	1				
16	-0.0065857238	0.86	8	1				
17	0.75564749	2.50	1		1.075	1.207	1.194	0.779
18	0.10694110	3.72	1		0.463	0.0895	1.986	0.805
19	-0.069533844	1.19	2		0.876	0.581	1.583	1.869
20	-0.24947395	3.25	3		1.108	0.947	0.756	0.694
21	0.027177891	3.00	3		0.741	2.356	0.495	1.312
22	-0.00090539530	2.00	2		4.032	27.01	1.002	2.054
23	-0.12310953	2.00	2		2.453	4.542	1.077	0.441
24	-0.089779710	1.00	2		2.300	1.287	1.493	0.793
25	-0.39512601	1.00	1		3.143	3.090	1.542	0.313

A condição de estabilidade para o potencial de Helmholtz é uma função côncava da temperatura de acordo com a equação

$$\frac{\partial^2 a}{\partial T^2} \leq 0 \quad (6)$$

O desenvolvimento analítico foi realizado até a primeira derivada.

Utilizando a regra da cadeia

$$\frac{\partial a}{\partial T} = \frac{\partial a}{\partial \alpha} \frac{\partial \alpha}{\partial \tau} \frac{\partial \tau}{\partial T} \quad (7)$$

$$\frac{\partial a}{\partial \alpha} = RT \quad (8)$$

$$\frac{\partial \alpha}{\partial \tau} = \frac{\partial(\alpha^0 + \alpha^r)}{\partial \tau} \quad (9)$$

$$\frac{\partial \tau}{\partial T} = -\frac{T_c}{T^2} \quad (10)$$

Aplicando em  $\alpha^0$  e  $\alpha^r$

$$\frac{\partial \alpha^0}{\partial \tau} = \frac{x_1}{\tau} + x_3 + \sum_{i=4}^7 \frac{x_i y_i \exp(-y_i \tau)}{[1 - \exp(-y_i \tau)]} \quad (11)$$

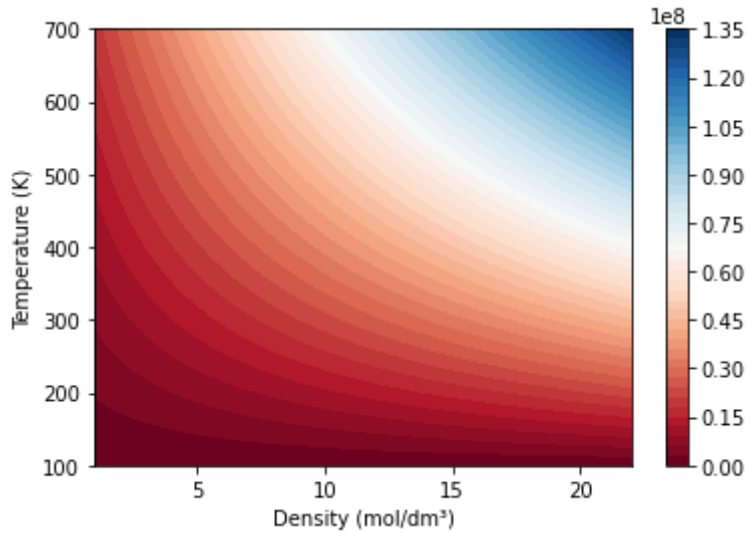
$$\begin{aligned} \frac{\partial \alpha^r}{\partial \tau} = & \sum_{i=1}^6 N_i \delta^{d_i} t_i \tau^{t_i-1} + \sum_{i=7}^{16} N_i \delta^{d_i} t_i \tau^{t_i-1} \exp(-\delta^{l_i}) \\ & + \sum_{i=17}^{25} N_i \delta^{d_i} (-2\tau^{t_i} \exp(-\eta_i(\delta - \epsilon_i)^2 - \beta_i(\tau - \gamma_i)^2) \beta_i(\tau - \gamma_i) \\ & + t_i \tau^{t_i-1} \exp(-\eta_i(\delta - \epsilon_i)^2 - \beta_i(\tau - \gamma_i)^2) \end{aligned} \quad (12)$$

Substituindo na equação 12

$$\begin{aligned} \frac{\partial a}{\partial T} = & \frac{x_1}{\tau} + x_3 + \sum_{i=4}^7 \frac{x_i y_i \exp(-y_i \tau)}{[1 - \exp(-y_i \tau)]} + \sum_{i=1}^6 N_i \delta^{d_i} t_i \tau^{t_i-1} + \sum_{i=7}^{16} N_i \delta^{d_i} t_i \tau^{t_i-1} \exp(-\delta^{l_i}) \\ & + \sum_{i=17}^{25} N_i \delta^{d_i} (-2\tau^{t_i} \exp(-\eta_i(\delta - \epsilon_i)^2 - \beta_i(\tau - \gamma_i)^2) \beta_i(\tau - \gamma_i) \\ & + t_i \tau^{t_i-1} \exp(-\eta_i(\delta - \epsilon_i)^2 - \beta_i(\tau - \gamma_i)^2) \end{aligned} \quad (13)$$

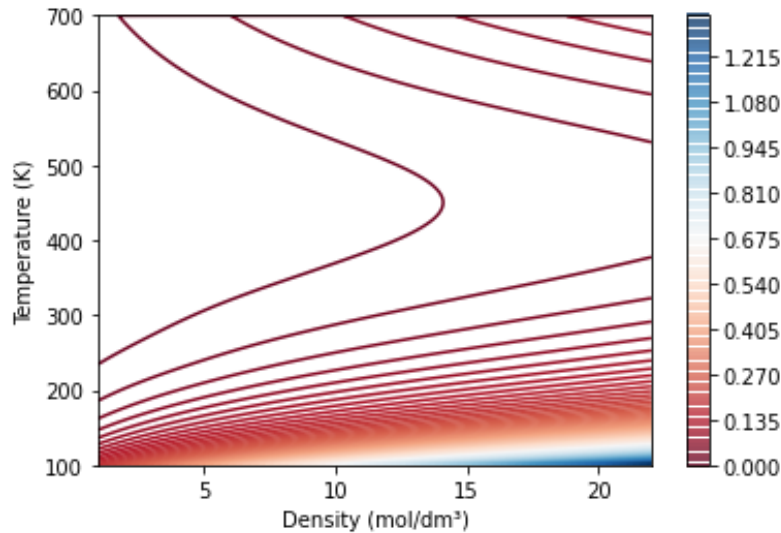
Devido à complexidade dos termos, a tentativa de estimação das regiões de estabilidade foi realizada por meio da utilização de ferramentas numéricas. Para o problema, fez-se o uso de um algoritmo em Python.

A função na formulação do potencial de Helmholtz dada pela equação 1 descreve a superfície abaixo, com unidade de  $J \cdot mol^{-1}$ .



A equação 6 determina a condição para que determinado sistema termodinâmico seja estável. Nas demais regiões, o sistema apresenta inomogeneidades internas, que são uma característica básica das transições de fase e identificam os estados de não equilíbrio.

A região limitada pela equação 4 obtida é demonstrada abaixo.



Os resultados foram obtidos numericamente utilizando uma discretização de 4000 elementos para cada variável independente. Nota-se que quanto maior o número de elementos utilizados, mais as curvas de nível se aproximam do ponto crítico nas coordenadas  $(\rho_c, T_c) = (5.93, 514.71)$ .

Código utilizado:

```
###
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# %%

table18 = pd.read_csv('table18.txt', index_col = 'i', sep=' ', dtype={'xi': 'float', 'yi': 'float'})
table19 = pd.read_csv('table19.txt', index_col = 'i', sep=' ', dtype={'Ni': 'float', 'ti': 'float', 'di': 'float',
'li': 'float', 'hi': 'float', 'bi': 'float', 'gi': 'float', 'ei': 'float'})

table19.columns = ['Ni', 'ti', 'di', 'li', 'hi', 'bi', 'gi', 'ei']

# %%

# Constants

Tc = 514.71 # K

rho_c = 5.93 # mol/dm³
```

```

R = 8.31446262 #J/(mol.K)

# %%

T = np.linspace(100,700,4000)
rho = np.linspace(1, 22, 4000)
tau = Tc/T
delta = rho/rho_c

# %%

xdelta, xtau = np.meshgrid(delta,tau)

# %%

def alpha_0(delta, tau): # J/mol

sum_alpha0 = np.cumsum([table18['xi'][i]*(1-np.log(1-np.exp(-table18['yi'][i]*tau))) for i in
range(4,8)], axis=1)[-1]

other_terms0 = np.log(delta) + table18['xi'][1]*np.log(tau) + table18['xi'][2] + table18['xi'][3]*tau

return (other_terms0 + sum_alpha0)*T*R


def alpha_r(delta, tau):

sum_r1 = np.cumsum([table19['Ni'][i]*delta**(table19['di'][i])*tau**(table19['ti'][i]) for i in
range(1,7)], axis=1)[-1]

sum_r2 = np.cumsum([table19['Ni'][i]*delta**(table19['di'][i])*tau**(table19['ti'][i])*np.exp(-
delta**(table19['li'][i])) for i in range(7,17)], axis=1)[-1]

sum_r3 = np.cumsum([table19['Ni'][i]*delta**(table19['di'][i])*tau**(table19['ti'][i])*np.exp((-
table19['hi'][i]*(delta-table19['ei'][i])**2-table19['bi'][i]*(tau-table19['gi'][i])**2)) for i in
range(17,26)], axis=1)[-1]

return sum_r1 + sum_r2 + sum_r3


def helm(delta, tau):

return alpha_0(delta, tau) + alpha_r(delta,tau)

```