



二元汽液平衡数据测定实验报告

作者：非非

日期：2024/12/22

一 实验目的

1. 从实验测得的 T-P-X-Y 数据计算各组分的活度系数
2. 学会二元汽液平衡相图的绘制
3. 绘制 t-x-y 曲线图和 x-y 曲线图

二 实验原理

2.1 活度系数计算

活度系数的计算公式如下：

$$\gamma_i = \frac{y_i P}{x_i P_i^0} \quad (2.1)$$

其中， γ_i 是组分 i 的活度系数， y_i 和 x_i 分别是气相和液相中组分 i 的摩尔分数， P 是系统总压， P_i^0 是纯组分 i 在系统温度下的饱和蒸气压。

2.2 相图绘制

t-x-y 曲线图反映了系统温度与液相和气相组成的关系，x-y 曲线图反映了液相和气相组成的关系。这两种图都是描述二元系统气液平衡性质的重要工具。

三 实验数据

原始实验数据如下：

表 3.1 原始实验数据

温度 (°C)	液相折光指数	气相折光指数
92	1.3772	1.3766
90.5	1.3759	1.3732
89.2	1.3735	1.3707
88	1.3719	1.3686
87	1.3711	1.3674
86	1.3690	1.3662
85	1.3672	1.3645
84.8	1.3665	1.3637
84	1.3624	1.3633
83.5	1.3610	1.3615

四 数据处理

4.1 计算公式

4.1.1 质量分数计算

使用以下回归方程计算乙醇的质量分数：

$$w_1 = 58.844116 - 42.61325n_D \quad (4.1)$$

其中， w_1 为乙醇的质量分数， n_D 为折光仪读数。

4.1.2 摩尔分数计算

使用以下公式将质量分数转换为摩尔分数：

$$x_1 = \frac{\frac{w_1}{M_1}}{\left(\frac{w_1}{M_1}\right) + \left(\frac{w_2}{M_2}\right)} \quad (4.2)$$

其中， x_1 为乙醇的摩尔分数， $M_1 = 46 \text{ g/mol}$ 为乙醇的摩尔质量， $M_2 = 60 \text{ g/mol}$ 为正丙醇的摩尔质量。

4.1.3 饱和蒸气压计算

使用安托因方程计算饱和蒸气压：

$$\log_{10} p^0 = A - \left(\frac{B}{T + C} \right) \quad (4.3)$$

其中， p^0 的单位为 mmHg， T 的单位为 $^{\circ}\text{C}$ 。

乙醇的安托因方程参数：A = 8.04494, B = 1554.3, C = 222.65 正丙醇的安托因方程参数：A = 7.99733, B = 1569.7, C = 209.5

4.1.4 活度系数计算

使用以下公式计算活度系数：

$$\gamma_i = \frac{y_i P}{x_i p_i^0} \quad (4.4)$$

其中， γ_i 是组分 i 的活度系数， y_i 和 x_i 分别是气相和液相中组分 i 的摩尔分数， P 是系统总压（101325 Pa）， p_i^0 是纯组分 i 在系统温度下的饱和蒸气压（单位：mmHg）。

4.2 计算示例

以数据表中第一行（温度 92°C ）为例：

4.2.1 液相组成计算

液相折光指数 $n_D = 1.3772$

计算质量分数：

$$w_1 = 58.844116 - 42.61325 \times 1.3772 = 0.1571 \quad (4.5)$$

计算摩尔分数：

$$x_1 = \frac{\frac{0.1571}{46}}{\left(\frac{0.1571}{46}\right) + \left(\frac{1-0.1571}{60}\right)} = 0.1956 \quad (4.6)$$

4.2.2 气相组成计算

气相折光指数 $n_D = 1.3766$

计算质量分数：

$$w_1 = 58.844116 - 42.61325 \times 1.3766 = 0.1827 \quad (4.7)$$

计算摩尔分数：

$$y_1 = \frac{\frac{0.1827}{46}}{\left(\frac{0.1827}{46}\right) + \left(\frac{1-0.1827}{60}\right)} = 0.2258 \quad (4.8)$$

4.2.3 饱和蒸气压计算

对于乙醇（ $T = 92^{\circ}\text{C}$ ）：

$$\log_{10} p_1^0 = 8.04494 - \left(\frac{1554.3}{92 + 222.65} \right) = 2.1052 \quad (4.9)$$

$$p_1^0 = 10^{2.1052} = 1273.9889 \text{ mmHg} \quad (4.10)$$

对于正丙醇（ $T = 92^{\circ}\text{C}$ ）：

$$\log_{10} p_2^0 = 7.99733 - \left(\frac{1569.7}{92 + 209.5} \right) = 1.7911 \quad (4.11)$$

$$p_2^0 = 10^{1.7911} = 618.0565 \text{ mmHg} \quad (4.12)$$

4.2.4 活度系数计算

乙醇的活度系数:

$$\gamma_1 = \frac{0.2258 \times 101325}{0.1956 \times 1273.9889 \times 133.322} = 0.6890 \quad (4.13)$$

正丙醇的活度系数:

$$\gamma_2 = \frac{(1 - 0.2258) \times 101325}{(1 - 0.1956) \times 618.0565 \times 133.322} = 1.1845 \quad (4.14)$$

五 实验结果

5.1 数据处理结果

表 5.1 更新后的数据处理结果

T (°C)	液相质量分数	气相质量分数	液相摩尔分数	气相摩尔分数	γ_1	γ_2
92.0000	0.1571	0.1827	0.1956	0.2258	0.6890	1.1845
90.5000	0.2125	0.3276	0.2604	0.3886	0.9408	1.0802
89.2000	0.3148	0.4341	0.3747	0.5002	0.8826	1.1007
88.0000	0.3830	0.5236	0.4474	0.5891	0.9101	1.0751
87.0000	0.4171	0.5748	0.4827	0.6381	0.9482	1.0540
86.0000	0.5066	0.6259	0.5725	0.6858	0.8921	1.1539
85.0000	0.5833	0.6983	0.6461	0.7512	0.8992	1.1503
84.8000	0.6131	0.7324	0.6740	0.7812	0.9032	1.1073
84.0000	0.7878	0.7495	0.8289	0.7960	0.7714	2.0337
83.5000	0.8475	0.8262	0.8788	0.8611	0.8023	1.9961

5.2 t—x—y 曲线图

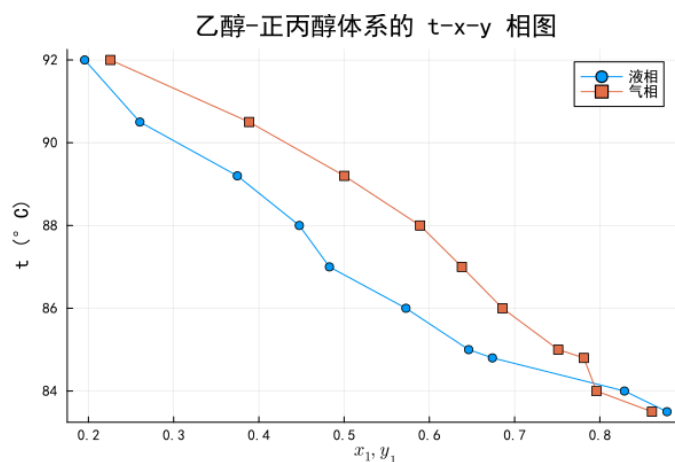


图 5.1 t—x—y 曲线图

5.3 x—y 曲线图

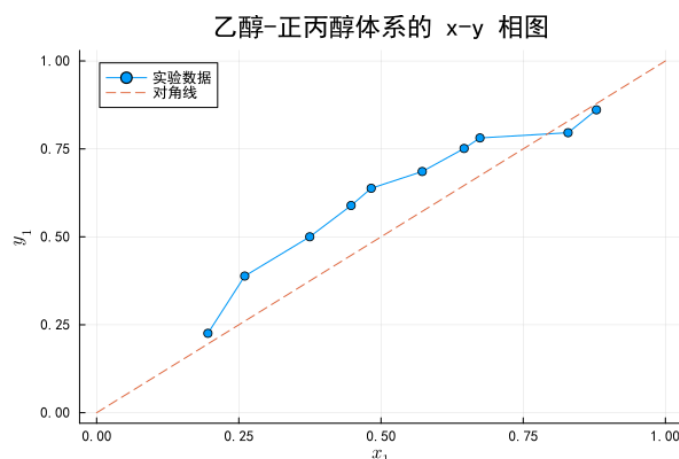


图 5.2 x—y 曲线图

六 结果讨论

1. 从 t—x—y 曲线图可以看出，随着乙醇浓度的增加，系统的沸点逐渐降低。这是因为乙醇的沸点比正丙醇低。
2. x—y 曲线图显示，在整个浓度范围内，气相中乙醇的浓度都高于液相，这表明乙醇比正丙醇更容易挥发。
3. 活度系数的计算结果显示，两种组分的活度系数都大于 1，说明该系统存在正偏差，表现出非理想性。这可能是由于乙醇和正丙醇分子间的相互作用力与纯组分分子间的相互作用力不同造成的。
4. 在接近共沸点时（约 84°C 附近），活度系数出现了较大的波动。这可能是由于实验误差或者系统在接近共沸点时的特殊行为造成的。
5. 总体来说，实验数据反映了乙醇-正丙醇体系的基本热力学特性，但在某些温度点（特别是接近共沸点时）可能存在一些异常值，需要进一步验证和分析。

七 误差分析

7.1 实验过程中可能存在的误差来源：

- 温度测量误差
- 折光指数测量误差
- 平衡状态判断误差
- 样品采集和处理过程中的误差
- 环境因素（如大气压变化）的影响

7.2 减少误差的方法：

- 使用更精确的温度计和折光仪
- 多次测量取平均值
- 仔细观察平衡状态，确保系统达到真正的平衡
- 改进样品采集和处理方法，减少污染和蒸发
- 保持实验环境稳定，记录大气压等环境参数
- 严格控制实验条件，包括加热和冷却速率
- 重复实验，增加数据的可靠性
- 与理论值和文献数据对比，验证实验结果的合理性
- 使用统计方法分析数据，识别和处理异常值

A 附录

```
1  ```julia
2  using DataFrames
3  using CSV
4  using Plots
5  using LaTeXStrings
6  using Printf
7
8  # 设置中文字体
9  default(fontfamily="SimHei");
10
11 # 读取数据
12 data = CSV.read("experiment1_data.csv", DataFrame);
13
14 # 定义计算质量分数的函数
15 function calculate_mass_fractions(n_D)
16     w1 = 58.844116 - 42.61325 * n_D # 乙醇质量分数
17     w2 = 1 - w1 # 正丙醇质量分数
18     return w1, w2
19 end;
20
21 # 定义计算摩尔分数的函数
22 function calculate_mole_fraction(w1, w2)
23     M1 = 46 # 乙醇分子量
24     M2 = 60 # 正丙醇分子量
25     n1 = w1 / M1
26     n2 = w2 / M2
27     return n1 / (n1 + n2)
28 end;
29
30 # 计算质量分数和摩尔分数
31 data.w1_L = zeros(nrow(data)); # 液相乙醇质量分数
32 data.w2_L = zeros(nrow(data)); # 液相正丙醇质量分数
33 data.w1_V = zeros(nrow(data)); # 气相乙醇质量分数
34 data.w2_V = zeros(nrow(data)); # 气相正丙醇质量分数
35 data.x1 = zeros(nrow(data)); # 液相乙醇摩尔分数
36 data.y1 = zeros(nrow(data)); # 气相乙醇摩尔分数
37 data.x2 = zeros(nrow(data)); # 液相正丙醇摩尔分数
38 data.y2 = zeros(nrow(data)); # 气相正丙醇摩尔分数
39
40 for i in 1:nrow(data)
41     data.w1_L[i], data.w2_L[i] = calculate_mass_fractions(data.L[i])
42     data.w1_V[i], data.w2_V[i] = calculate_mass_fractions(data.V[i])
43     data.x1[i] = calculate_mole_fraction(data.w1_L[i], data.w2_L[i])
44     data.y1[i] = calculate_mole_fraction(data.w1_V[i], data.w2_V[i])
45     data.x2 = 1 .- data.x1
46     data.y2 = 1 .- data.y1
47 end;
48
49 # 绘制 T-x-y 曲线图
50 p1 = plot(data.x1, data.t, label="液相", marker=:circle);
51 plot!(p1, data.y1, data.t, label="气相", marker=:square);
52 xlabel!(p1, L"x1, y1");
53 ylabel!(p1, "t (°C)");
```

```

54 title!(p1, "乙醇-正丙醇体系的 t-x-y 相图")
55 savefig(p1, "experiment1_res/t-x-y_plot.png")
56
57 # 绘制 x-y 曲线图
58 p2 = plot(data.x1, data.y1, label="实验数据", marker=:circle);
59 plot!(p2, [0, 1], [0, 1], label="对角线", linestyle=:dash);
60 xlabel!(p2, L"x1");
61 ylabel!(p2, L"y1");
62 title!(p2, "乙醇-正丙醇体系的 x-y 相图")
63 savefig(p2, "experiment1_res/x-y_plot.png")
64
65 # 计算 lg(p0) 的函数
66 function calculate_lg_p0(A, B, C, t)
67     return A - B / (t + C)
68 end;
69
70 # 计算 p0 的函数 (单位: # mmHg)
71 function calculate_p0(lg_p0)
72     return 10^(lg_p0)
73 end;
74
75 # 计算活度系数的函数
76 function calculate_activity_coefficient(x1, y1, p10, p20)
77     P = 101325 / 133.22 # (单位: # mmHg)
78     γ1 = y1 * P / (x1 * p10)
79     γ2 = (1 - y1) * P / ((1 - x1) * p20)
80     return γ1, γ2
81 end;
82
83 # 安托因方程参数
84 A1 = 8.04494;
85 B1 = 1554.3;
86 C1 = 222.65;
87
88 A2 = 7.99733;
89 B2 = 1569.7;
90 C2 = 209.5;
91
92 # 计算每个数据点的活度系数
93 data.p10 = zeros(nrow(data));
94 data.p20 = zeros(nrow(data));
95 data.γ1 = zeros(nrow(data));
96 data.γ2 = zeros(nrow(data));
97
98 for i in 1:nrow(data)
99     t = data.t[i]
100
101     lg_p10 = calculate_lg_p0(A1, B1, C1, t)
102     lg_p20 = calculate_lg_p0(A2, B2, C2, t)
103
104     p10 = calculate_p0(lg_p10)
105     p20 = calculate_p0(lg_p20)
106
107     data.p10[i] = p10
108     data.p20[i] = p20
109

```

```

110     data.y1[i], data.y2[i] = calculate_activity_coefficient(
111         data.x1[i],
112         data.y1[i],
113         p1^0,
114         p2^0
115     )
116 end;
117
118 # 定义一个函数来格式化数字为四位小数的字符串
119 function four_decimals(x)
120     return @sprintf("%.4f", x)
121 end;
122
123 # 创建一个新的 DataFrame，按照所需的顺序排列并保存为四位小数的字符串
124 result_data = DataFrame(
125     t      = four_decimals.(data.t),
126     w1_L= four_decimals.(data.w1_L),
127     w2_L= four_decimals.(data.w2_L),
128     w1_V= four_decimals.(data.w1_V),
129     w2_V= four_decimals.(data.w2_V),
130     x1    = four_decimals.(data.x1),
131     y1    = four_decimals.(data.y1),
132     x2    = four_decimals.(data.x2),
133     y2    = four_decimals.(data.y2),
134     p1^0 = four_decimals.(data.p1^0),
135     p2^0 = four_decimals.(data.p2^0),
136     y1    = four_decimals.(data.y1),
137     y2    = four_decimals.(data.y2)
138 );
139
140 # 打印结果
141 println("结果:")
142 println(result_data)
143
144 # 将结果保存到 CSV 文件
145 CSV.write("experiment1_res/results.csv", result_data)
146 ```

```