



浙江大学物理化学实验

线性电势扫描方法表征缓蚀剂在金属表面的 吸附行为

实 验 报 告

参加学生：叶青杨（3210100360）

指导老师：方文军

浙江大学化学实验教学中心

2023年10月26日

线性电势扫描方法表征缓蚀剂在金属表面的吸附行为

叶青杨 (3210100360), 指导教师: 方文军

一、实验目的与原理

实验目的

1. 熟悉电化学工作站的使用方法;
2. 掌握如何由线性电势扫描获得的塔菲尔曲线斜率计算自腐蚀电流, 并由自腐蚀电流, 计算缓蚀剂在金属基底材料表面覆盖度;
3. 学习基于电化学测量技术研究固液界面吸附过程热力学和动力学的方法。

实验原理

当金属浸于腐蚀介质时, 如果金属的平衡电极电位低于介质中去极化剂 (如 H^+ 或氧分子) 的平衡电极电位, 则金属和介质构成一个腐蚀体系, 称为共轭体系。此时, 金属发生阳极溶解, 去极化剂发生还原。

阳极反应的速度 (电流密度) 以 i_a 表示, 阴极反应的速度 (电流密度) 以 i_k 表示, 当体系达到稳定时即金属处于自腐蚀状态时, $i_a = i_k = i_{corr}$ (i_{corr} 为腐蚀电流), 体系不会有净的电流积累, 体系处于稳定电位 ϕ_c , 称为自腐蚀电位或开路电位。在一定的极化电位控制下并且可以忽略浓差极化时, 金属的腐蚀速度可以表示为:

$$i = i_a - i_k = i_{corr} \{ \exp [(\phi - \phi_c)/\beta_a] - \exp [(\phi - \phi_c)/\beta_k] \}$$

当对电极进行阳极极化, 在强极化区 (一般 $\delta E \geq 100/nmV$, n 为摩尔阳极电极反应电子数), 阴极分支电流 $i_k \approx 0$,

$$\Delta E = \beta_a \ln (i/i_{corr}) = b_a \lg (i/i_{corr})$$

当对电极进行阴极极化, 在强极化区 (一般 $\delta E \leq 100/nmV$, n 为摩尔阴极电极反应电子数), 阳极分支电流 $i_a \approx 0$,

$$\Delta E = \beta_k \ln (|i|/i_{corr}) = b_k \lg (|i|/i_{corr})$$

因此在强极化区, 极化电位与外测电流密度的对数值之间具有线性关系, 称为塔菲尔关系式。将极化曲线上的塔菲尔区外推到自腐蚀电位处, 得到的交点坐标即为自腐蚀电流 i_{corr} 。

对于表面吸附类缓蚀剂, 通过线性电势扫描方法测量得到未添加缓蚀剂以及添加不同浓度缓蚀剂的酸性介质的极化曲线, 由塔菲尔外推法计算得到金属自腐蚀电流值, 进而计算得到对应浓度下缓蚀剂分子在金属表面覆盖率值:

$$\theta = 1 - i_{corr, inhi} / i_{corr, free}$$

如果缓蚀剂在金属表面为单层化学吸附，则覆盖率 θ 和缓蚀剂浓度 C 之间的关系满足 Langmuir 吸附等温式：

$$C/\theta = 1/K_{ads} + C$$

缓蚀剂分子在金属表面吸附的吸附热与温度之间的关系满足 Van' t Hoff 公式：

$$\ln K_{ads} = -\Delta H_{ads}^{\theta}/RT + Constant$$

$\ln K_{ads}$ 对 $1/T$ 作图得到直线，由直线的斜率可计算缓蚀剂分子在金属表面吸附的吸附热 ΔH_{ads}^{θ} 值。由 ΔG_{ads}^{θ} 和 ΔH_{ads}^{θ} 的值，可按式进一步计算缓蚀剂分子在金属表面吸附的熵变 ΔS_{ads}^{θ} 的值。

由不同温度下的 i_{corr} 值，使 $\ln i_{corr}$ 对 $1/T$ 作图得到直线，由直线斜率可以计算出表观活化能 E_a 值。

1 试剂与仪器

1.1 试剂

浓盐酸（36%~38%）（AR，国药）、NaCl（AR，国药）、3-氨基-1,2,4-三氮唑（ATA，AR，阿拉丁）、丙酮（AR，国药）、脱脂棉、去离子水

1.2 仪器

电化学工作站（CHI，上海）；计算机（含电化学工作站控制软件）；精密超级恒温槽（THX-05，宁波天恒仪器厂）；金相试样抛光机（PG-2T）；辅助电极（Pt 片）；参比电极（饱和甘汞电极）；H62 黄铜电极（自制， $\phi 4$ ，外衬聚四氟乙烯管，环氧树脂密封）；KCl 盐桥；500、1000 mL 玻璃烧杯；100 mL、2000 mL 容量瓶；100 mL 三口玻璃电解池；1200 目、2000 目和 3000 目金相砂纸等

二、实验

2 实验步骤

2.1 实验一

(1) 打开恒温槽，设置到所需温度。测试介质装入 500 mL 容量瓶置恒温槽中预热。经预热后的测试介质装入 100 mL 三口玻璃电解池中置恒温槽中，安装好辅助电极（铂片）、饱和甘汞电极参比电极（带饱和 KCl 盐桥），将电化学工作站测试线和相应的电极相连，在电化学工作站操作软件中选择 Tafel 曲线测试模式，设定电势扫描范围和扫描速率，勾选量程自动选择模式。Tafel 曲线测试参数：扫描范围：开路电压正负 0.25 V vs.SCE；扫描速率：0.001V/S；静止时间：0 s。

(2) 电极表面预处理：2000 目砂纸打磨 → 3000 目砂纸打磨 → 抛光 → 丙酮棉球擦洗 → 去离子水冲洗 → 1 M HCl 溶液浸蚀 1 min → 去离子水冲洗

(3) Tafel 曲线测试：经前处理的电极插入三口玻璃电解池中并开始计时，电极与电化学工作站相应的测试线相连，电极浸入测试介质 2 min. 后开始电势扫描，测试完毕后点击测试软件 Analysis→Special Analysis 菜单进行自腐蚀电流拟合，保存原始数据（和拟合结果）至相应文件夹。取出 H62 黄铜电极，插入另一已经前处理的电极重复前述步骤进行平行测量，合计至少进行 3 次平行测量。Tafel 曲线测试期间作电极表面抛光等前处理。

(4) 实验报告数据处理包括：腐蚀抑制率（ATA 在电极表面覆盖率）计算、Langmuir 吸附等温式拟合、一定温度下吸附平衡常数计算、表面吸附摩尔焓计算、表面吸附 Gibbs 能计算、表面吸附熵变计算以及吸附过程表观活化能计算。

使用 python 处理实验数据和绘图

3 实验结果与分析

室内气温：25-26 °C

室内气压：101.98 kPa

45°C 的数据来自别的小组

数据处理的方法和结果见 SI

表 1 Tafel 曲线外推 Cu 电极在 1 M HCl 溶液中的自腐蚀电流

温度 /°C	$i_c/10^{-6} A$		
25	2.838	3.459	3.467
平均值		3.255	
45	0.8375	3.062	2.979
平均值		2.293	

可以发现，在温度升高的条件下，铜电极在 1 M HCl 溶液中的自腐蚀电流有所下降，这意味一个负的表观活化能。类似地，我们可以通过绘制铜电极在 0.00 mM、0.30 mM、0.60 mM、1.00 mM 缓蚀液中的 Tafel 曲线，并使用外延法可以求得自腐蚀电流 i_c 、表面覆盖率 θ 及 ATA 浓度/表面覆盖率比 C/θ 。

表 2 Tafel 曲线外推自腐蚀电流 i_c 表面覆盖率 θ 及 ATA 浓度/表面覆盖率比 c/θ

C/mM	$i_c/10^{-6} A$	θ	$\frac{C}{\theta}/mM$	C/mM	$i_c/10^{-6} A$	θ	$\frac{C}{\theta}/mM$
0.00	2.723	0.0000	(25°C)	0.00	1.894	0.0000	(45°C)
0.30	0.5999	0.7797	0.3848	0.30	0.2149	0.8865	0.3384
0.60	0.3865	0.8581	0.6992	0.60	0.1963	0.8964	0.6693
1.00	0.2362	0.9133	1.095	1.00	0.08400	0.9556	1.046

由上表数据做 25 °C、45 °C 的 Langmuir 吸附等温曲线，线性拟合得到相应的 Langmuir 吸附等温式 $C/\theta = 1/K_{ads} + c$ 的参数

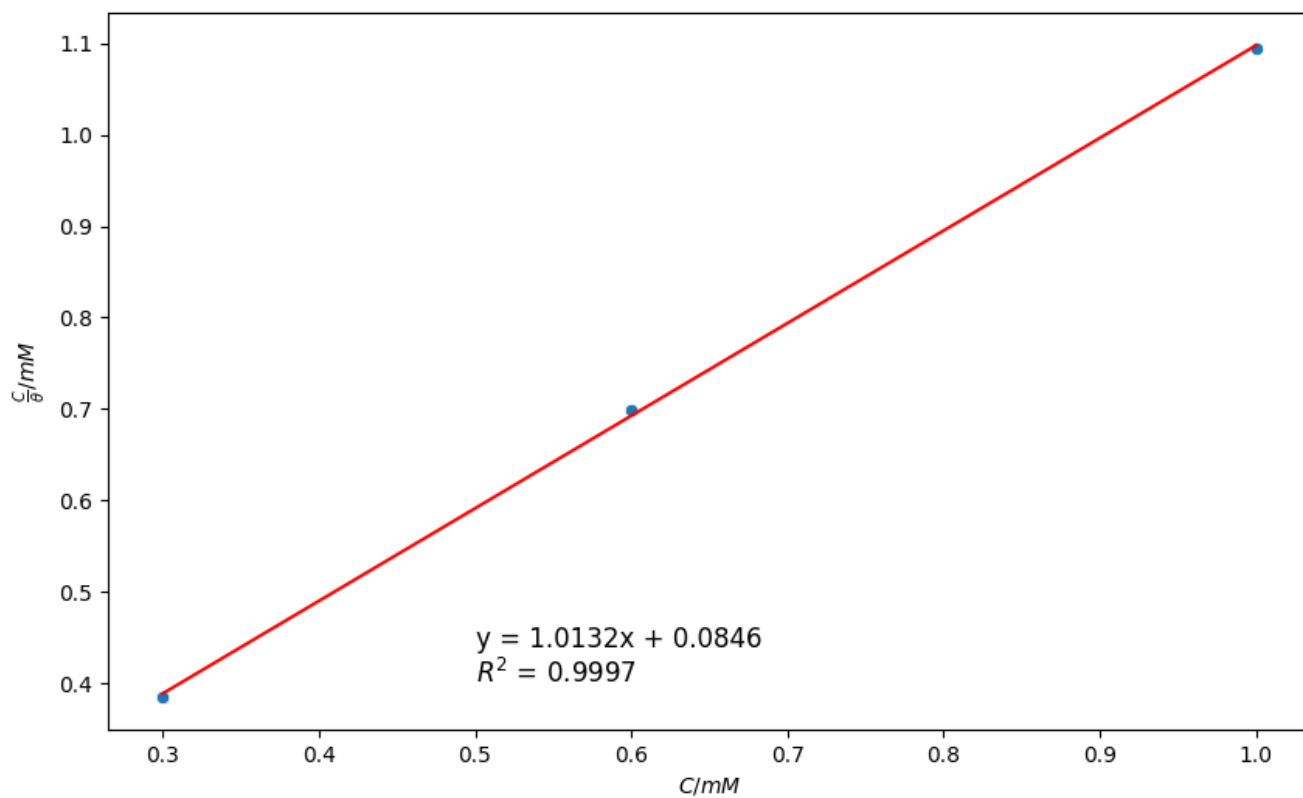


图 1 25°C 时表面吸附 Langmuir 线性拟合图像

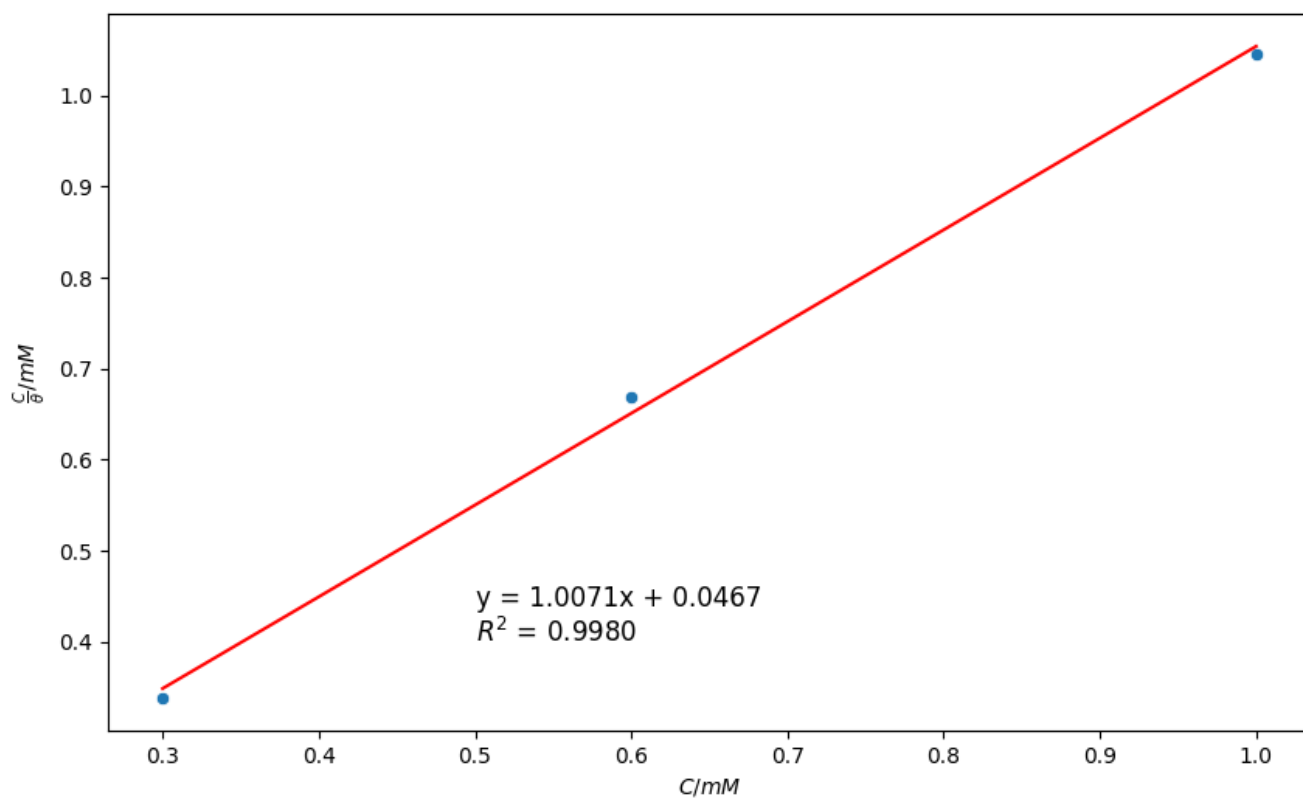


图 2 45°C 时表面吸附 Langmuir 线性拟合图像

表 3 计算得到的各项热力学数据

温度 /°C	$1/T$ /K ⁻¹	$1/K_{ads}$ /mol·L ⁻¹	斜率	相关系数 R^2	$\ln K_{ads}$	ΔH /kJ·mol ⁻¹	ΔG /kJ·mol ⁻¹	ΔS /J·mol ⁻¹ ·K ⁻¹	E_a /kJ·mol ⁻¹
25	0.003354	0.0846	1.0132	0.9997	2.47	23.25	-16.08	131.9	-13.81
45	0.003143	0.0467	1.0071	0.9980	3.06		-18.72	131.9	

对于一个吸附反应来说，拥有一个正的吸附焓、一个正的吸附熵以及一个负的表现活化能，是一个非常不正常的现象。由于我们使用的 45°C 的数据来自其他小组，和 25°C 的数据同一套仪器所测，故我们对实验计算结果的可靠性持保留态度。我们需要更多可靠的实验数据来得到热力学参数。

本实验测定的为微电流的体系，故本实验的实验数据对实验条件具有较高的敏感性，各组内部的数据的平行性一般。

可能的造成较大偏差的因素较多，电极表面打磨情况，原始电极表面积，打磨后暴露在空气中的时间甚至是搅拌对电极的振动都有可能造成一定程度的影响。在本次实验中隔壁小组的同学在调整反应体系时对我组的电极进行了触碰扰动，对该组数据造成了 63.1% 的电流偏差。

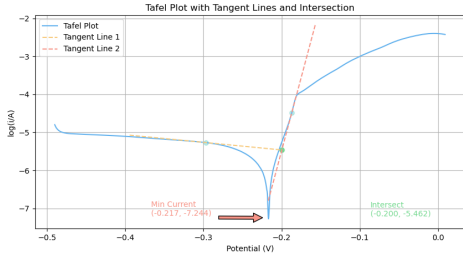
本次实验的 Tafel 区选择为大致在开路电压正负 0.06V 的位置，具体决定于斜率变缓区的具体位置。

四、参考文献

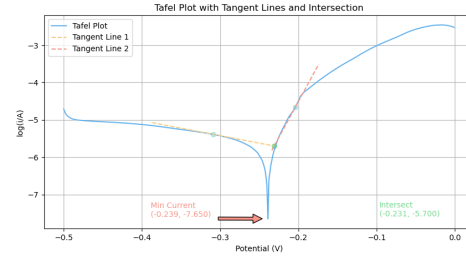
- [1] 王国平, 张培敏, 王永尧. 中级化学实验 [M]. 北京: 科学出版社, 2017.

Supporting Information

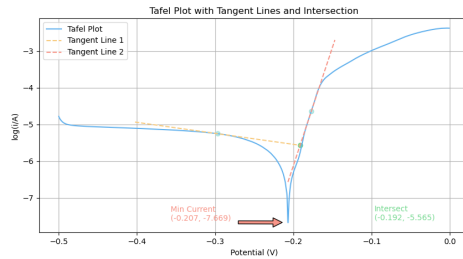
一、来自本组的实验数据绘图



(a) 第一次实验

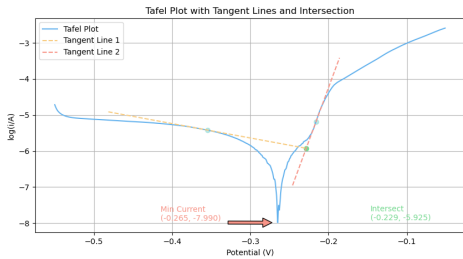


(b) 第二次实验

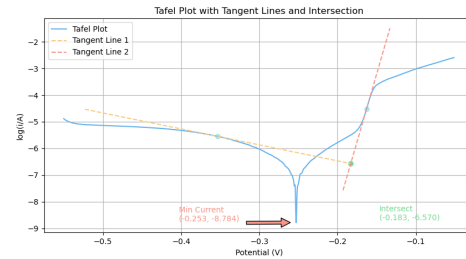


(c) 第三次实验

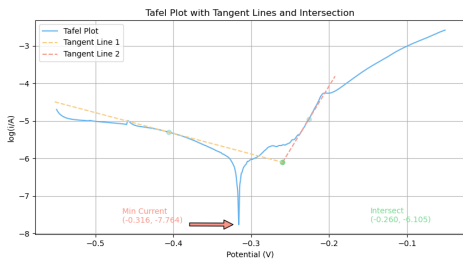
Figure 1: 0mM APA 的实验 (25°C)



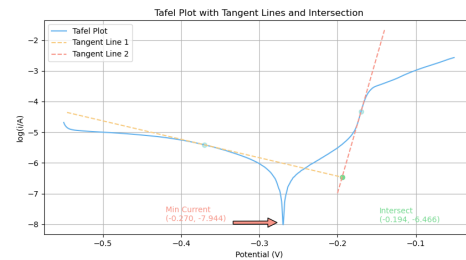
(a) 第一次实验



(b) 第二次实验

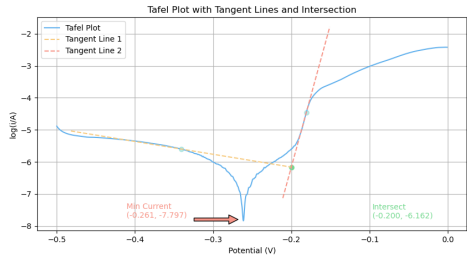


(c) 第三次实验 (被干扰)

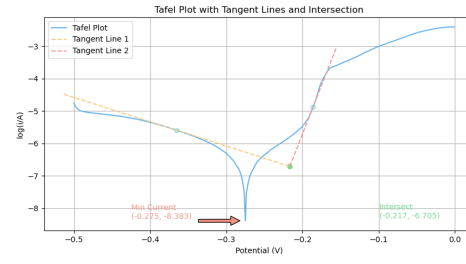


(d) 第四次实验

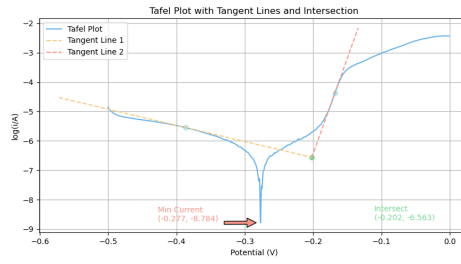
Figure 2: 0.3mM APA 的实验 (25°C)



(a) 第一次实验

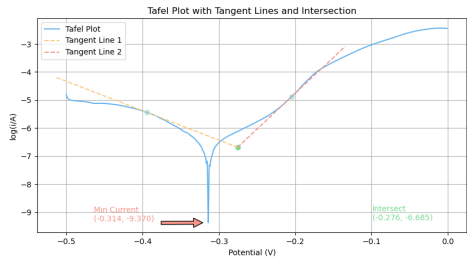


(b) 第二次实验

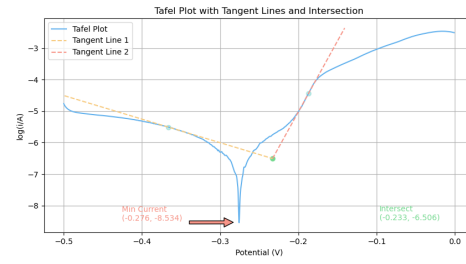


(c) 第三次实验

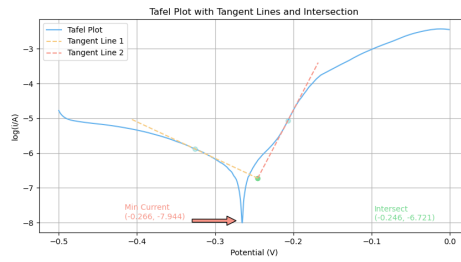
Figure 3: 0.6mM APA 的实验 (25°C)



(a) 第一次实验

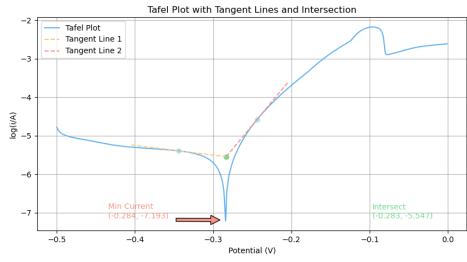


(b) 第二次实验

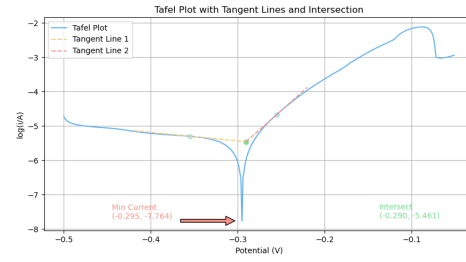


(c) 第三次实验

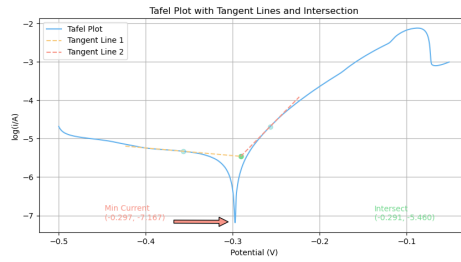
Figure 4: 1.0mM APA 的实验 (25°C)



(a) 第一次实验



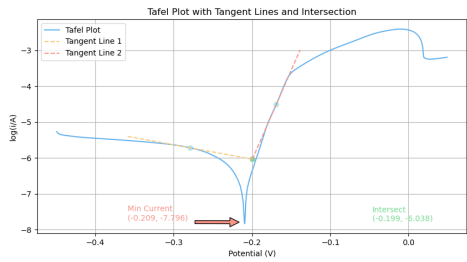
(b) 第二次实验



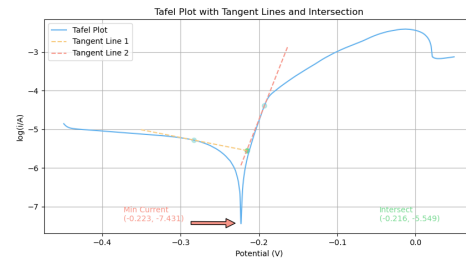
(c) 第三次实验

Figure 5: 1M HCl 环境的实验 (25°C)

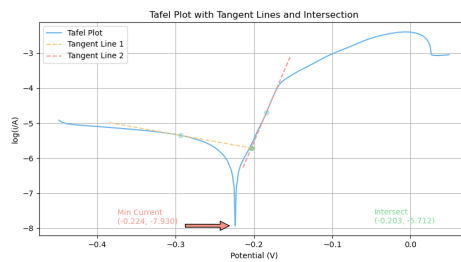
二、来自其他组的实验数据绘图



(a) 第一次实验

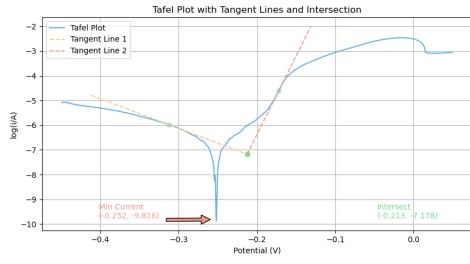


(b) 第二次实验

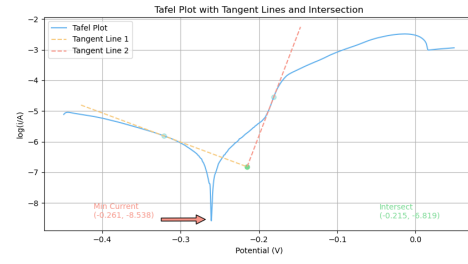


(c) 第三次实验

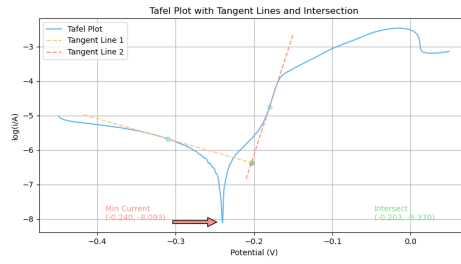
Figure 6: 0mM APA 的实验 (45°C)



(a) 第一次实验

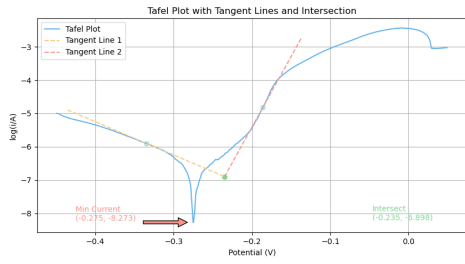


(b) 第二次实验

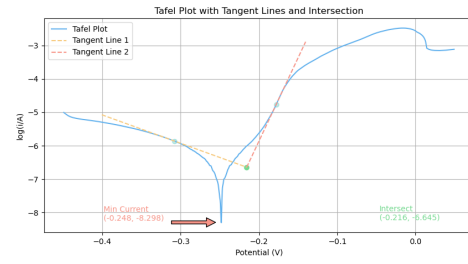


(c) 第三次实验

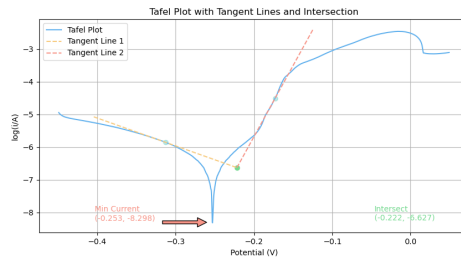
Figure 7: 0.3mM APA 的实验 (45°C)



(a) 第一次实验

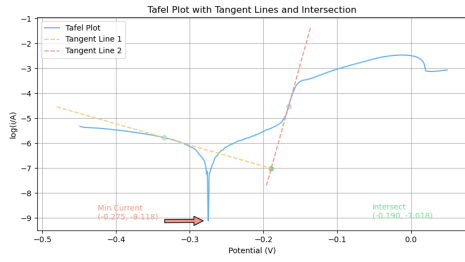


(b) 第二次实验

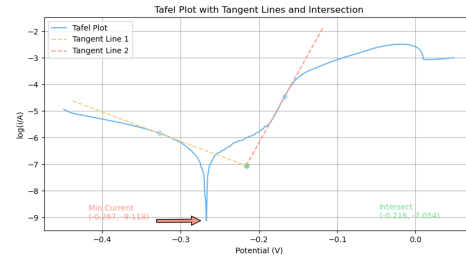


(c) 第三次实验

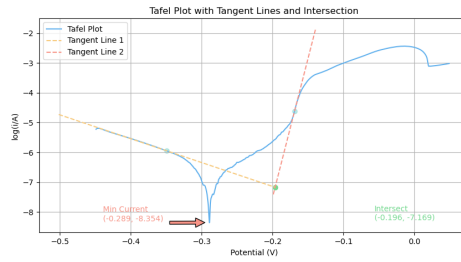
Figure 8: 0.6mM APA 的实验 (45°C)



(a) 第一次实验

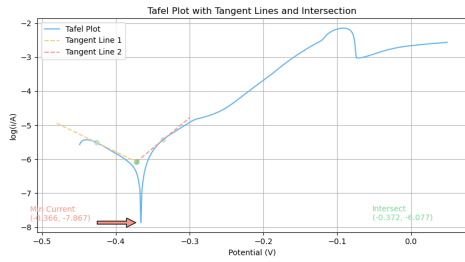


(b) 第二次实验

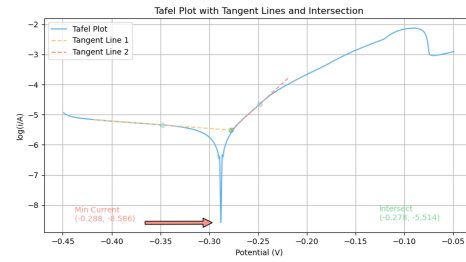


(c) 第三次实验

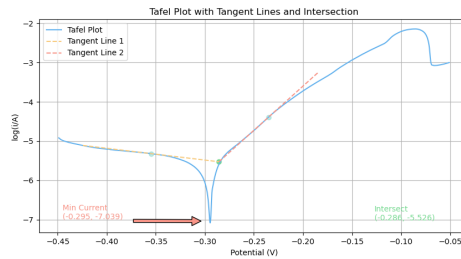
Figure 9: 1.0mM APA 的实验 (45°C)



(a) 第一次实验



(b) 第二次实验



(c) 第三次实验

Figure 10: 1M HCl 环境的实验 (45°C)

三、数据处理和绘图使用的程序

SI 中的数据处理与绘图使用 python 完成
 使用教程见: <https://populusathayana.github.io/>
 第一部分为 TXT 文件到 CSV 文件的批量转化
 第二部分的绘图在线演示程序: <http://47.122.18.217:8501/>

```
# extract.py 用于从目录下所有TXT文件中提取数据并保存到CSV文件中
import pandas as pd
import os

def extract_data_and_save_to_csv(txt_file_path):
```

```

# 加载数据
try:
    data = pd.read_csv(txt_file_path, skiprows=17, header=None, names=['
        Potential/V', 'Current/A', 'log(i/A)'])
except Exception as e:
    print(f"Failed to load data from {txt_file_path}: {str(e)}")
    return

# 构建输出CSV文件的路径和名称
output_file_path = os.path.splitext(txt_file_path)[0] + '.csv'

# 保存到CSV文件
try:
    data.to_csv(output_file_path, index=False)
    print(f"Data from {txt_file_path} has been successfully saved to {
        output_file_path}")
except Exception as e:
    print(f"Failed to save data to {output_file_path}: {str(e)}")

def main(directory_path):
    # 获取目录下所有的TXT文件
    txt_files = [f for f in os.listdir(directory_path) if os.path.isfile(os.
        path.join(directory_path, f)) and f.lower().endswith('.txt')]

    # 对每个TXT文件执行数据提取和保存操作
    for txt_file in txt_files:
        txt_file_path = os.path.join(directory_path, txt_file)
        extract_data_and_save_to_csv(txt_file_path)

if __name__ == "__main__":
    directory_path = input("Please enter the directory path: ")
    main(directory_path)

```

tafel.py 用于绘制Tafel曲线，并保存为PNG文件

```

import streamlit as st
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import linregress
from scipy.interpolate import interp1d
import tempfile
import shutil
import os

# 定义颜色方案
colors = {
    "color1": np.array([99, 178, 238]) / 255,
    "color2": np.array([118, 218, 145]) / 255,
    "color3": np.array([248, 203, 127]) / 255,
    "color4": np.array([248, 149, 136]) / 255,
    "color5": np.array([124, 214, 207]) / 255,
    "color6": np.array([145, 146, 171]) / 255,
    "color7": np.array([120, 152, 225]) / 255,
    "color8": np.array([239, 166, 102]) / 255,
    "color9": np.array([237, 221, 134]) / 255,
    "color10": np.array([153, 135, 206]) / 255
}

def plot_tafel(data, A_1, A_2):
    data['abs_Current/A'] = np.abs(data['Current/A'])

    # 1. 找到电流最接近0的点

```

```

min_current_row = data.loc[data['abs_Current/A'].idxmin()]

# 2. 在这个点的正负A_1, A_2范围内, 找到对应的两个点
mask = (data['Potential/V'] >= min_current_row['Potential/V'] - A_1) & (
    data['Potential/V'] <= min_current_row['Potential/V'] + A_2)
range_data = data[mask]

# 找到正负A_1, A_2对应的点
point1 = range_data.loc[range_data['Potential/V'].idxmin()]
point2 = range_data.loc[range_data['Potential/V'].idxmax()]

# 3. 在这两个位置的三个点上分别计算切线
def calculate_tangent_line(data, center_index):
    subset = data.loc[center_index-1:center_index+1]
    slope, intercept, _, _, _ = linregress(subset['Potential/V'], subset[
        'log(i/A)'])
    return slope, intercept

slope1, intercept1 = calculate_tangent_line(data, point1.name)
slope2, intercept2 = calculate_tangent_line(data, point2.name)

# 4. 求出这两条切线的交点
x_intersect = (intercept2 - intercept1) / (slope1 - slope2)
y_intersect = slope1 * x_intersect + intercept1

# 计算切线交点到切点的距离
distance_1 = x_intersect - point1['Potential/V']
distance_2 = x_intersect - point2['Potential/V']

# 对数据进行插值
interp_function = interp1d(data['Potential/V'], data['log(i/A)'], kind='
    cubic')
potential_interpolated = np.linspace(data['Potential/V'].min(), data['
    Potential/V'].max(), 1000000)
current_interpolated = interp_function(potential_interpolated)

# 5. 绘制塔菲尔曲线 (使用插值后的数据)、切线和交点
plt.figure(figsize=(10, 5))
plt.plot(potential_interpolated, current_interpolated, label='Tafel Plot'
    , color=colors["color1"])
# 在x坐标轴上标注电流最接近0的点
plt.annotate(f'Min Current\n({min_current_row["Potential/V"]:.3f}, {
    min_current_row["log(i/A)"]:.3f})',
    xy=(min_current_row['Potential/V'], min_current_row['log(i/A)
    ']),
    xytext=(min_current_row['Potential/V']-0.15, min_current_row[
    'log(i/A)']+0.05),
    arrowprops=dict(facecolor=colors["color4"], shrink=0.1),
    color=colors["color4"])

# 标记切线的交点
plt.scatter(x_intersect, y_intersect, color=colors["color2"])
# 确定切线最小范围
mm=0.03
# 绘制切线, 确保切线可见
x_values_1 = np.linspace(point1['Potential/V'] - max(abs(distance_1),mm),
    point1['Potential/V'] + max(abs(distance_1),mm), 1000)
y_values_1 = slope1 * x_values_1 + intercept1
x_values_2 = np.linspace(point2['Potential/V'] - max(abs(distance_2),mm),
    point2['Potential/V'] + max(abs(distance_2),mm), 1000)
y_values_2 = slope2 * x_values_2 + intercept2

plt.plot(x_values_1, y_values_1, label='Tangent Line 1', linestyle='--',
    color=colors["color3"])

```

```

plt.plot(x_values_2, y_values_2, label='Tangent Line 2', linestyle='--',
         color=colors["color4"])
# 标记切点
plt.scatter([point1['Potential/V'], point2['Potential/V']], [point1['log(
    i/A)'], point2['log(i/A)']], color=colors["color5"], alpha=0.5)
# 在图的右下角显示交点坐标
plt.text(0.78, 0.05, f'Intersect\n({x_intersect:.3f}, {y_intersect:.3f})',
         color=colors["color2"], ha='left', va='bottom', transform=plt.gca()
         .transAxes)

# 设置图表标题和坐标轴标签
plt.title('Tafel Plot with Tangent Lines and Intersection')
plt.xlabel('Potential (V)')
plt.ylabel('log(i/A)')

# 显示图例
plt.legend(loc='upper left')

# 显示图表
plt.grid(True)
st.pyplot(plt) # 使用st.pyplot()来显示图表

# Streamlit 应用
st.title("Tafel Plot Analysis")

#文件上传
uploaded_file = st.file_uploader("Choose a CSV file", type="csv")
if uploaded_file is not None:
    # 获取上传文件的文件名 (不包含扩展名)
    file_name, _ = os.path.splitext(uploaded_file.name)

    data = pd.read_csv(uploaded_file)

    # 自定义参数
    A_1 = st.number_input("输入左切点偏离值", value=0.06)
    A_2 = st.number_input("输入右切点偏离值", value=0.06)

    # 绘制图表
    plot_button = st.button("Plot Tafel Curve")
    if plot_button:
        plot_tafel(data, A_1, A_2)
        # 保存到临时文件
        temp_filename = tempfile.mktemp(suffix=".png")
        plt.savefig(temp_filename)
        st.session_state.temp_filename = temp_filename

    # 添加保存图片的按钮
    if "temp_filename" in st.session_state:
        save_button = st.button("Save Plot")
        if save_button:
            # 将临时文件移动到目标位置
            shutil.move(st.session_state.temp_filename, f"{file_name}.png")
            st.success("Plot saved successfully!")
            # 清除session状态中的临时文件名
            del st.session_state.temp_filename
    else:
        st.write("Please upload a CSV file to proceed.")

```