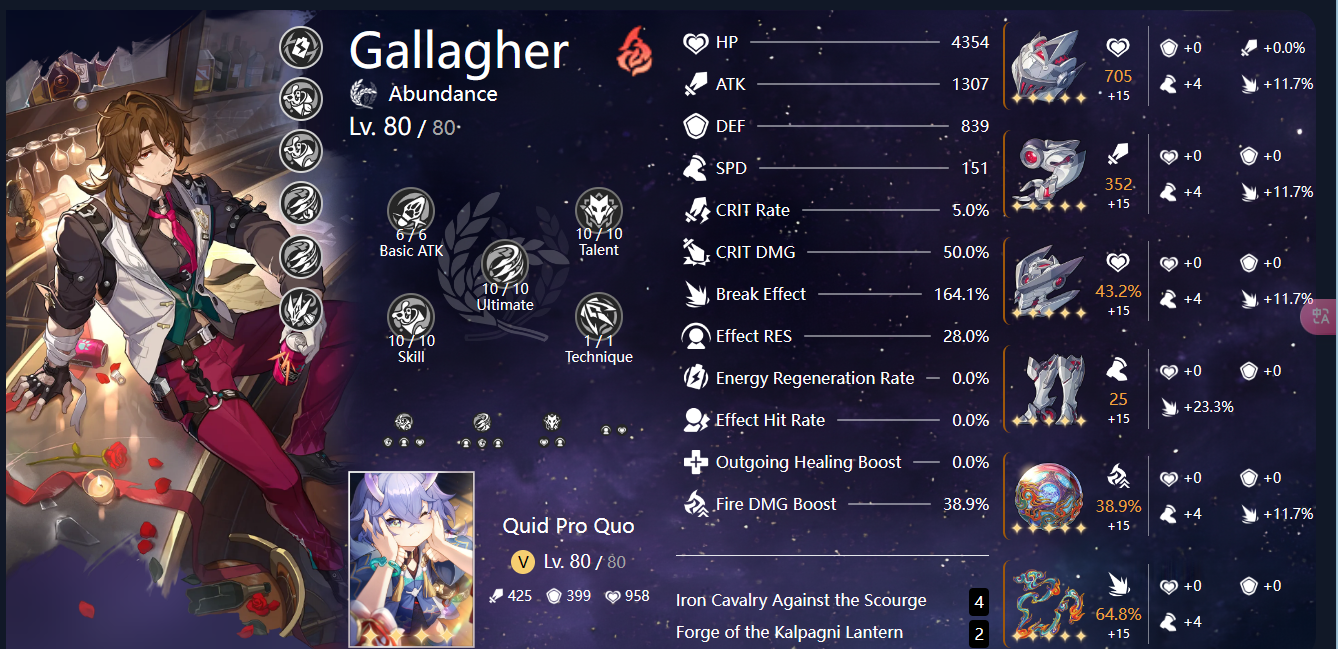
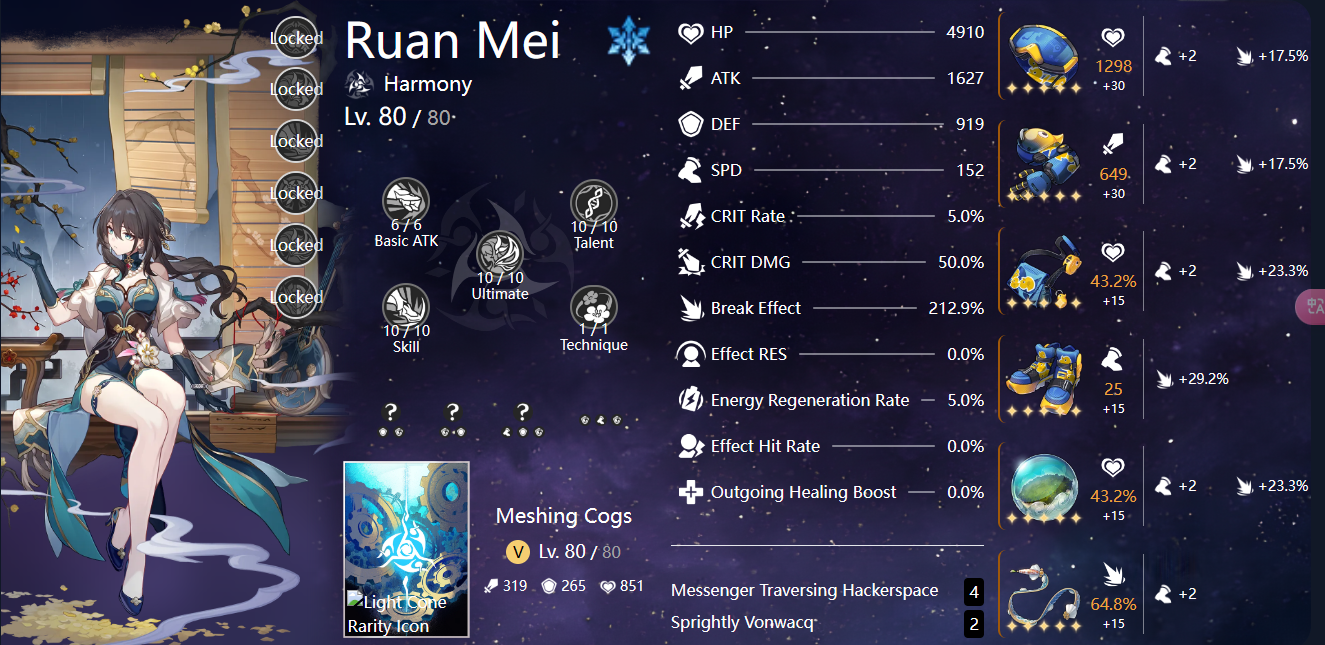
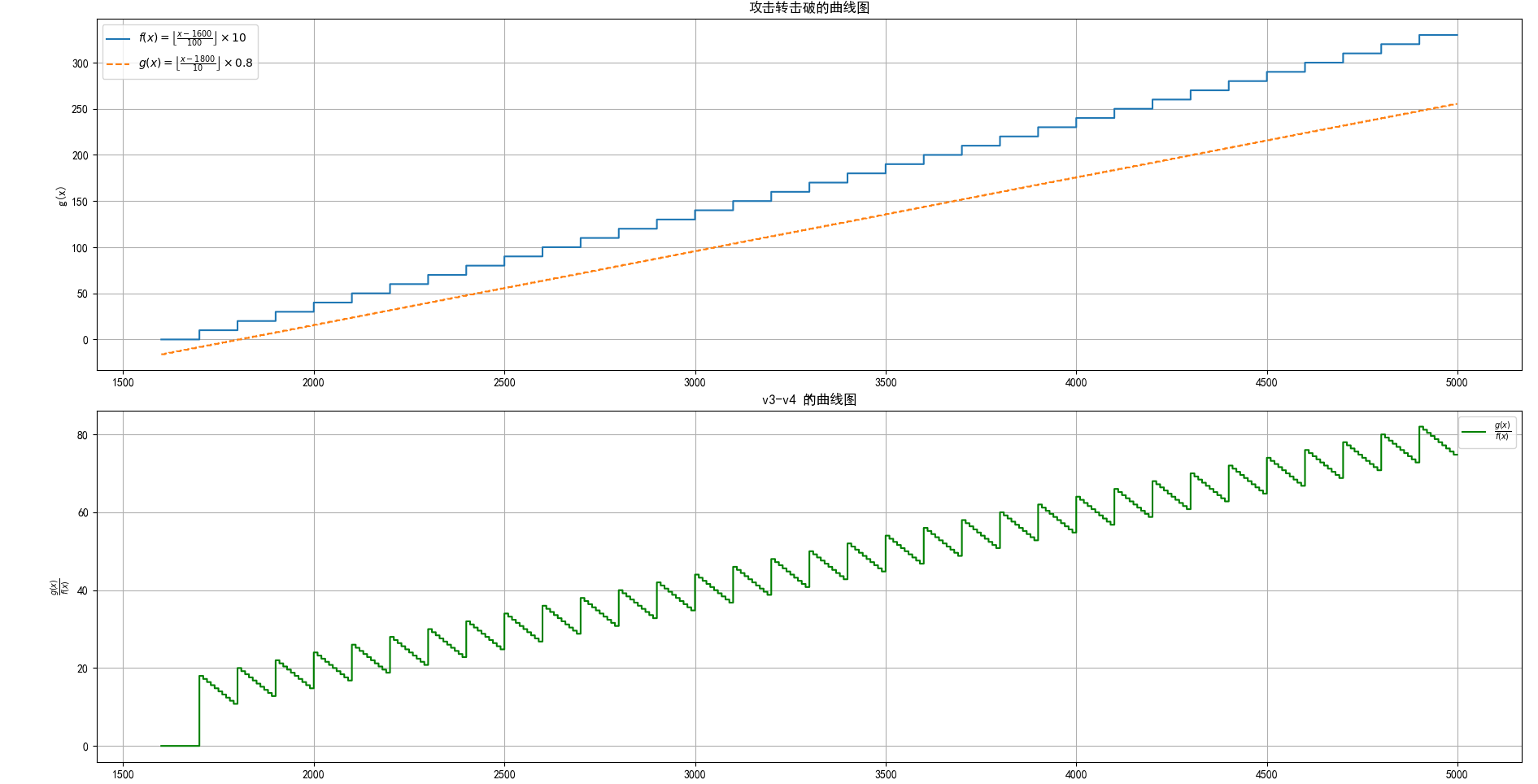
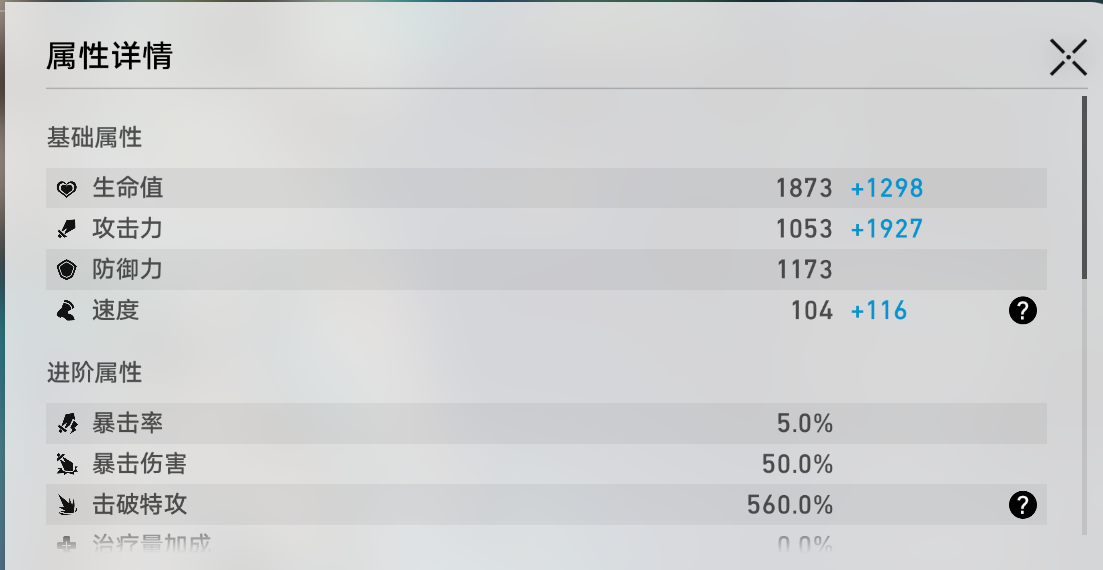
流萤v3->v4击破研究

实验结果前置省流：

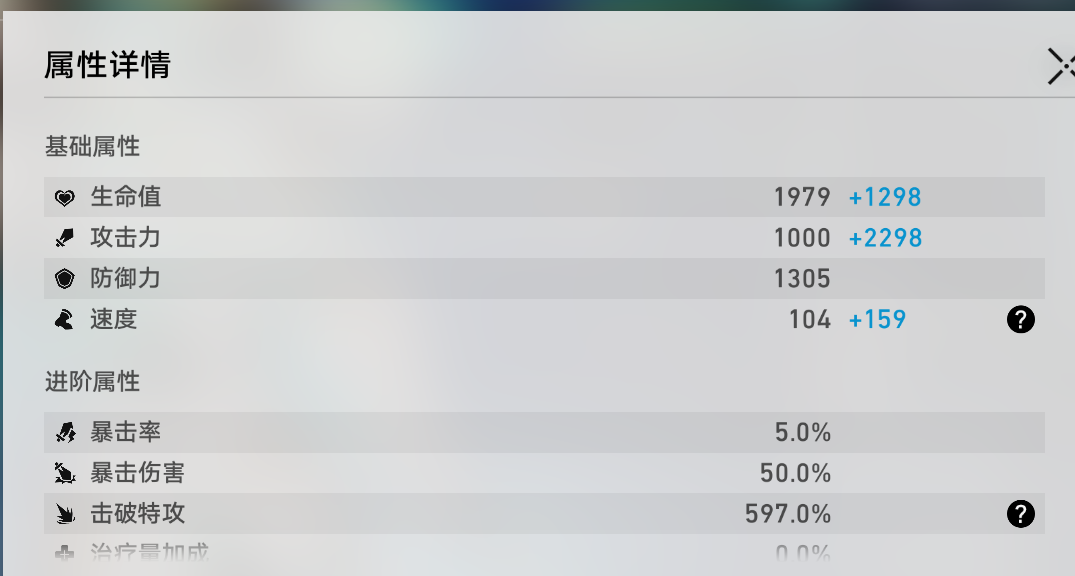
  
先放配置：  
  
实验过程：  
1.艾主软流（实战很差，非必要不推荐。但我一直坚信，没有哪个角色说缺哪个角色就不能玩的，所以还是想测）：  
 如图：击破617，攻击不到3800.星神一层,4层击破和攻击均会更高，但主角秘技buff会掉，实测攻击4125，击破588.平均攻击4000.  
 v4-v3见图1可知，大概是少60，占10%。

  
图1（后面有py代码）

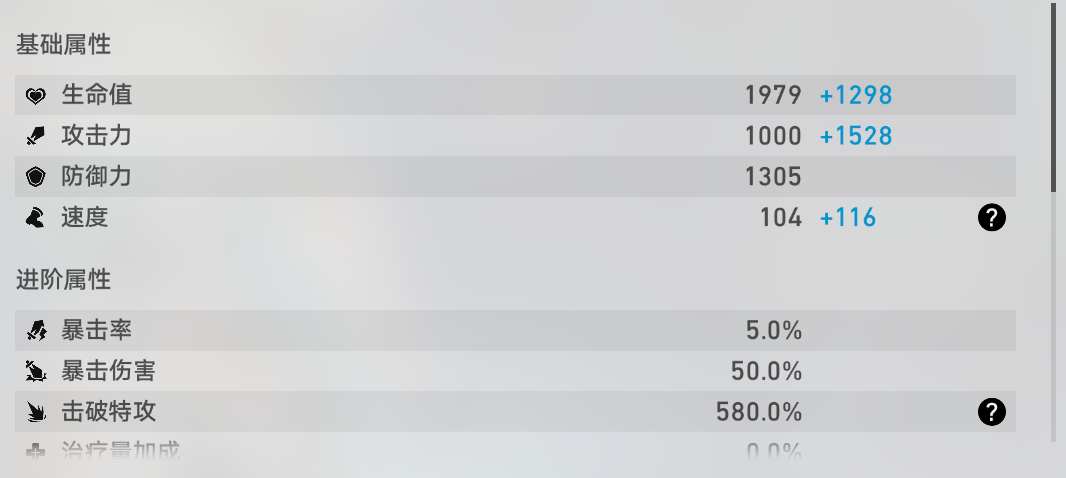
2.软主软流：   
 如图：击破560，攻击3000.星神一层,4层击破和攻击均会更高，但主角秘技buff会掉，实测攻击3300，击破540.平均攻击3200. v4-v3见图1可知，大概是少43，占7.8%



流萤换专武：  
  
重复上述：  
实战：  
1.艾主软流：  
 如图：击破597，攻击不到3300. 但主角秘技buff会掉.v4-v3见图1可知，大概是少45，占8%。



2.软主软流：   
 如图：击破580，攻击2528但主角秘技buff会掉， v4-v3见图1可知，大概是少30，占5.3%

  
击破曲线绘制py代码(gpt辅助)  
import numpy as np

import matplotlib.pyplot as plt

from matplotlib import rcParams

# 设置字体，确保支持中文和数学符号

rcParams['font.sans-serif'] = ['SimHei'] # 使用黑体（SimHei）

rcParams['axes.unicode\_minus'] = False # 解决负号显示为方块的问题

# 定义函数

def f(x):

return np.floor((x - 1600) / 100) \* 10

def g(x):

return np.floor((x - 1800) / 10) \* 0.8

# 生成 x 数据

x = np.arange(1600, 5000, 1)

# 计算 y 数据

y\_f = f(x)

y\_g = g(x)

y\_ratio = np.zeros\_like(y\_f)

nonzero\_indices = y\_f != 0

y\_ratio[nonzero\_indices] =y\_f[nonzero\_indices] - y\_g[nonzero\_indices]

# 创建图形

fig, axs = plt.subplots(2, 1, figsize=(10, 15))

# 绘制 f(x) 曲线

axs[0].step(x, y\_f, where='post', label=r'$f(x) = \left\lfloor \frac{x-1600}{100} \right\rfloor \times 10$')

axs[0].set\_xlabel('x')

axs[0].set\_ylabel('f(x)')

axs[0].set\_title('攻击转击破的曲线图')

axs[0].legend()

axs[0].grid(True)

# 绘制 g(x) 曲线

axs[0].step(x, y\_g, where='post', label=r'$g(x) = \left\lfloor \frac{x-1800}{10} \right\rfloor \times 0.8$', linestyle='--')

axs[0].set\_xlabel('x')

axs[0].set\_ylabel('g(x)')

axs[0].set\_title('攻击转击破的曲线图')

axs[0].legend()

axs[0].grid(True)

# 绘制 g(x)/f(x) 曲线

axs[1].step(x, y\_ratio, where='post', label=r'$\frac{g(x)}{f(x)}$', color='green')

axs[1].set\_xlabel('x')

axs[1].set\_ylabel(r'$\frac{g(x)}{f(x)}$')

axs[1].set\_title('v3-v4 的曲线图')

axs[1].legend()

axs[1].grid(True)

# 显示图形

plt.tight\_layout()

plt.show()