

原神数值养成分析

前言：

该文档的目标：通过对角色基础属性的进行**数值分配** (数值之间有约束条件有公式) 计算出**不同定位角色在特定的环境下**使得其在**5min**的时间内的**伤害值达到最大**

需要考虑到问题：

- 1.如何进行元素属性的搭配 (元素属性有：雷、火、冰、水、风、岩. 即选择那几个元素反应作为输出补充)
- 2.选择哪一属性为主C
- 3.队伍是需要物理输出还是元素输出
- 4.是否需要频繁切换角色(队伍里有4个角色，角色只能在场一个)

一、基础属性

- 生命 (HP)
- 攻击力(ATK)
- 防御力(DEF)
- 元素精通(ELemental Mastery)
- 体力(Max Stamina)
- 暴击率(crit rate)
- 暴击伤害(critical damage)
- 治疗加成
- 元素充能(Elemental Enegy)
- 元素伤害加成(Elemental damage addition)
- 物理伤害加成(Physical damage addition)
- 其他辅助功能

1.属性分类

伤害属性

攻击力、暴击伤害、暴击率、元素伤害加成、元素充能、元素精通

防御属性

生命值、防御力、体力

辅助属性

治疗加成、元素精通、元素充能、其他辅助功能

2.角色定位

主C角色：队伍的主要伤害提供点. 尽可能提高**伤害属性**.

副C角色：队伍第二强力输出点. 当主C角色还未成型时，可以作为补充输出的点. 可以选择 **伤害+其他** (具体搭配需要按角色、现有武器、刷到的圣遗物、现有升**级材料而定**)

尽可能提高元素精通(因为风属

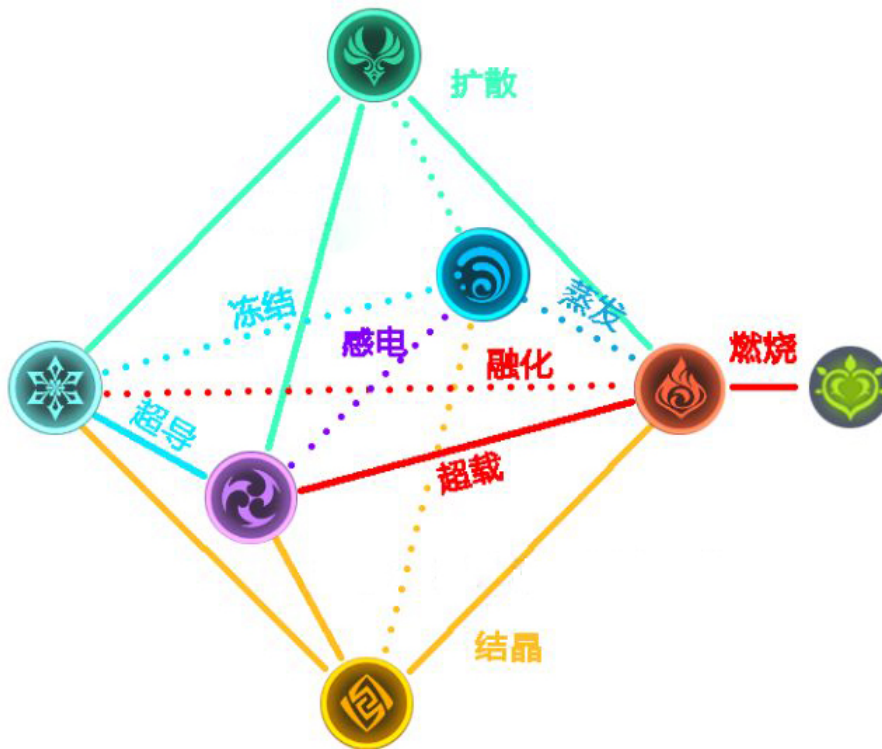
式不同)

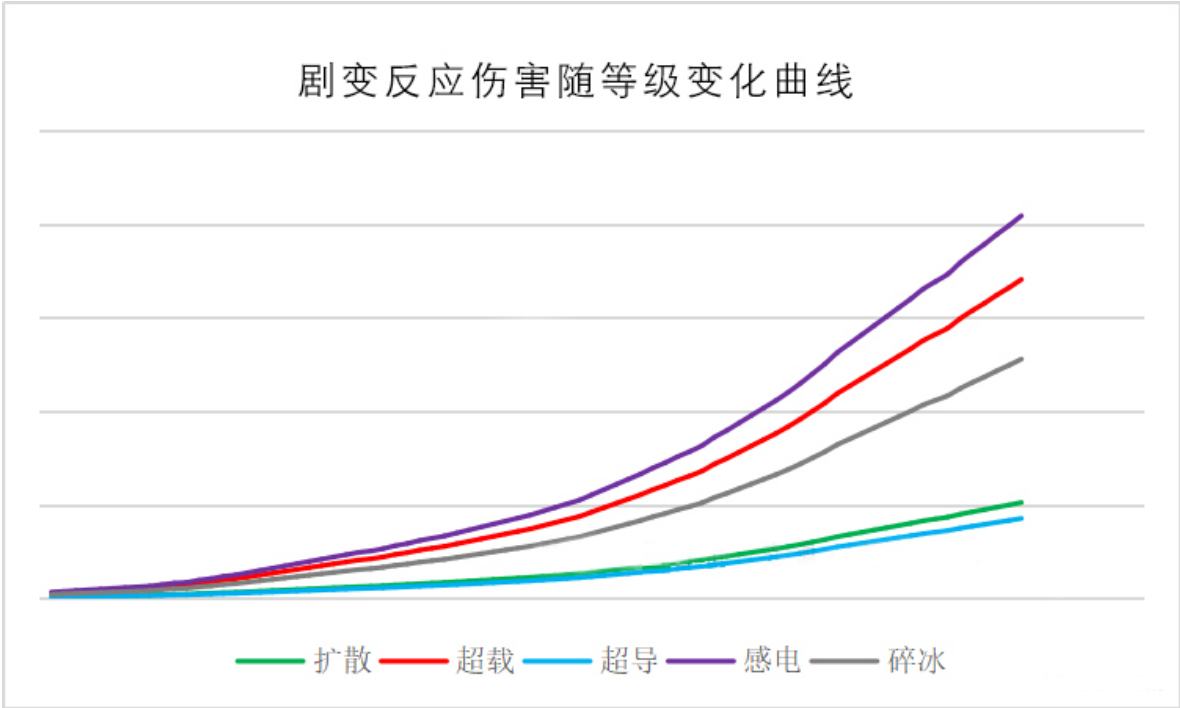
3.伤害分类

C七七

老爷(迪卢克)、主C公子、主C刻晴、副C皇女)

元素反应伤害：超载、感电、超导、融化、蒸发、冰冻、结晶、扩散





伤害比例：超导：扩散：碎冰：超载：感电 = 1：1.2：3：4：4.8

sp：

- 感电会造成二次伤害
- 超载会产生AOE和爆炸
- 碎冰会解除冰冻效果
- 超导会降低目标40%物理防御
- 扩散产生的元素结晶最多
- 结晶提供护盾和元素充能

增幅反应

- 水--火： 2.0
- 火--水： 1.5
- 火--冰： 2.0
- 冰--火： 1.5

4.武器分类

表1 武器类别和武器属性类别

武器\属性 加成 类别	攻击力 +暴击	攻击力 +爆伤	攻击力 +被伤	攻击力 +元充	攻击力 +元精	攻击力+物 伤+回复	攻击力 +生命
双手剑			古华	祭礼			
单手剑				祭礼、 西风		风鹰	
长枪							
弓箭							
法典	四风				万国 图、祭 礼		讨龙

5.圣遗物分类

表2 圣遗物套装类别和属性类别

套 装 \ 属 性	套装描述	攻击 力 (ATK)	暴击 (CR)	爆伤 (CD)	元伤 (EDA)	元 充	元 精	生命 (HP)	防御 (DEF)
角斗	加攻击力和近战角色普攻伤害								
乐团	加元素精通和远程角色普攻伤害								
宗室	加元素爆发和全体属性加成								
火套 (魔女)	加火属性伤害和火类元素反应伤害								
雷套 (雷怒)	加雷属性伤害和火类元素反应伤害								
风套 (翠绿)	加风属性伤害和火类元素反应伤害								
岩套 (古岩)	加岩伤和护盾状态时的元素伤害								
染血	物理伤害加成								

二、伤害计算公式

1.物理伤害

伤害期望

$$S_{物} = ATK * (1 + CR * CD) * (1 + PDA)$$

ATK为攻击力，S为伤害，CR为暴击率，CD为暴击额外伤害，PDA为物理伤害加成

2.元素伤害

伤害期望

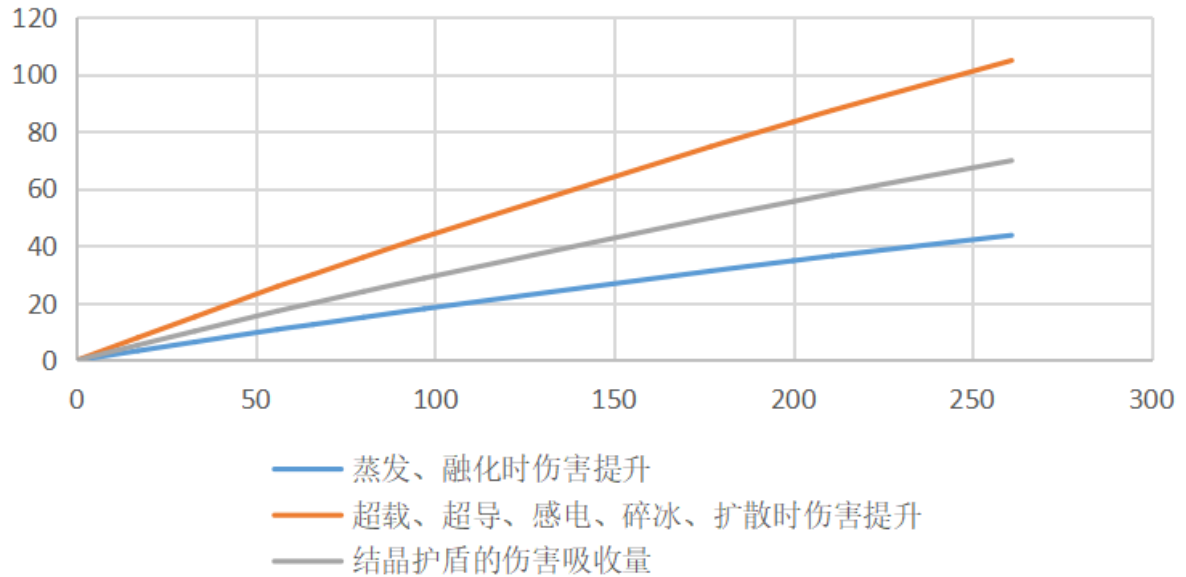
$$S_{元} = ATK * (1 + CR * CD) * (1 + EDA)$$

EDA为元素伤害加成

3.元素反应伤害

元素精通对元素反应具有加成

元素精通加成随元素精通数值变化曲线



三、圣遗物属性分析

- 攻击力(ATK)
- 暴击率(Crit Rate)
- 暴击伤害(Critical Damage)
- 元素伤害加成(Elemental Damage Addition)
- 元素精通(Elemental Mastery)
- 生命 (HP)
- 治疗加成
- 元素充能(Elemental Energy)
- 防御力(DEF)

1.圣遗物属性分为1个主属性4条副属性

表3 五星圣遗物+20主属性和+0副属性

属性类别	主属性
攻击力/ATK	311/46.6%
暴击率/CR	31.1%
暴击伤害/CD	62.2%
元素伤害加成/EDA	46.6%
元素精通/EM	
元素充能/EE	
生命值/HP	
防御力/DEF	

表4 +20五星圣遗物副属性磁条

ATK	ATK_1	CR	CD	EM	EE	HP	HP_1	DEF	DEF_1
	11.10%	7.00%			9.70%	478			
	9.30%					418		42	11.70%
	13.40%					299	5.80%		19.70%
18	15.70%			40					13.10%
37			12.40%		9.70%	538			
53				42	17.50%			23	
19		13.60%	14.80%		11.00%				
	5.30%			21	28.50%	508			
64		11.70%		47		239			

圣遗物分为花、羽、沙、杯、冠

花：主磁条必为生命值，无需考虑主属性

羽：主磁条必为攻击力，无需考虑主属性

沙：主磁条为攻击力、元素精通

杯：主磁条为攻击力、元素伤害加成

冠：主磁条为攻击力、暴击率、暴击伤害、元素精通

四、武器属性分析

sp: 不考虑武器被动效果，只分析武器基本属性

表5 常用80级未突破1精炼武器属性

	攻击力	暴击率	暴击伤害	物理伤害	元素充能	元素精通
狼的末路(五星)	532+65.3%					
试做古华	497+25.1%					
风鹰剑(五星)	590+20%			37.7%		
祭礼单手剑	401				55.9%	
祭礼双手剑	497				27.5%	
西风弓	401				55.9%	
西风大剑	401				55.9%	
西风剑						
绝弦弓	449					151
万国图谱	497					101
流浪乐章	449		50.3%			
祭礼法典	401					201
四风原典	532	30.2%				

五、角色属性数值约束条件推导

1.圣遗物属性约束分析

假设：为了推导简化，只考虑+20五星圣遗物的属性

+20的圣遗物主属性磁条都是固定数值，可以先分析副属性，再根据花、羽、沙、杯、冠的主属性选择加入到约束中即可

圣遗物在+8、+12、+16、+20时会随机对其中一个副属性进行加成

根据表5中的信息我们需要推导出10个副属性之间的约束条件，空白的表示为零

圣遗物副属性有10个属性 $X = [ATK, ATK_1, CR, CD, EM, EE, HP, HP_1, DEF, DEF_1]$

10个属性满足：

$$\sum(X./A) = 4$$

其中

$$A = [a_{ATK}, a_{ATK_1}, a_{CR}, a_{CD}, a_{EM}, a_{EE}, a_{HP}, a_{HP_1}, a_{DEF}, a_{DEF_1}]$$

表示对应属性每次强化时增加的大小

由于数据中空缺数据较多直接拟合造成一下结果

```
1 import numpy as np
2 import pandas as pd
3 from sklearn.linear_model import LinearRegression
4
5 RelicsData = pd.read_excel('圣遗物副属性.xlsx')
```



```

6 RelicsData.fillna(0, inplace=True )
7 features = ['ATK', 'ATK_1', 'CR', 'CD', 'EM', 'EE', 'HP', 'HP_1', 'DEF',
  'DEF_1']
8 # print(features[:10])
9 X_train = RelicsData[features]
10 len_X = len(X_train)
11 Y_train = np.ones((len_X,), dtype = np.int)*4
12 # X_train.info()
13 linreg = LinearRegression()
14 linreg.fit(X_train,Y_train)
15 y_predict = linreg.predict(X_train)
16 # print(linreg.score(X_train, Y_train))
17 print(linreg.coef_, linreg.intercept_)

```

结果:

```

1 A = [0 0 0 0 0 0 0 0 0 0] 截距=4.0

```

将截距设置为零

```

1 import numpy as np
2 import pandas as pd
3 from sklearn.linear_model import LinearRegression
4
5 RelicsData = pd.read_excel('圣遗物副属性.xlsx')
6 RelicsData.fillna(0, inplace=True )
7 features = ['ATK', 'ATK_1', 'CR', 'CD', 'EM', 'EE', 'HP', 'HP_1', 'DEF',
  'DEF_1']
8 # print(features[:10])
9 X_train = RelicsData[features]
10 len_X = len(X_train)
11 Y_train = np.ones((len_X,), dtype = np.int)*4
12 # X_train.info()
13 linreg = LinearRegression(fit_intercept = False)#将截距设置为0
14 linreg.fit(X_train,Y_train)
15 y_predict = linreg.predict(X_train)
16 print(linreg.score(X_train, Y_train))
17 print(linreg.coef_, '截距:%.3f' %linreg.intercept_)

```

得到一下结果

```

1 A=[2.78246314e-02 1.65053149e+01 8.87584962e+00 1.05442944e+01
2 1.52892833e-02 6.39391423e+00 1.93598324e-03 1.31545587e+01
3 3.31416855e-02 2.26286026e+00] 截距:0.000

```

与实际相差很大，均不合理