

带权区间调度问题 - 贪心算法与动态规划对比

项目简介

本项目实现了带权区间调度问题的两种求解算法：**贪心算法**和**动态规划算法**，并通过高质量可视化动画清晰对比两种算法的求解过程，特别展示了贪心算法的局部最优缺陷。

核心特点：

- 25个活动，满足多样性要求（短低权、中中权、长高权混合）
- 最终解包含10+个活动
- 设计“贪心陷阱”：权重60的长活动阻挡总权重239的10个高价值短活动
- 贪心最优率 $\leq 95\%$ ，清晰展示局部最优与全局最优的差异

文件结构

```
├── weighted_scheduling_code.py # 主程序 (算法实现+动画生成)
├── test_data_enhanced.json # 测试数据 (自动生成)
├── weighted_scheduling.gif # 生成的动画文件
├── README.md # 使用说明
├── README.pdf # 使用说明
├── requirements.txt # Python依赖列表
├── report.pdf # 报告
└── report.md # 报告
```

环境要求

- **Python:** 3.8 或更高版本
- **操作系统:** Windows / Linux / macOS
- **内存:** $\geq 2\text{GB}$
- **磁盘空间:** $\geq 100\text{MB}$

快速开始

1. 安装依赖

```
pip install -r requirements.txt
```

依赖包括：

- numpy >= 1.19.0
- matplotlib >= 3.3.0
- pillow >= 8.0.0

2. 一键运行

```
python weighted_scheduling.py
```

程序将自动完成：

1. 生成测试数据（25个活动）
2. 贪心算法求解
3. 动态规划算法求解
4. 结果验证（无冲突检查）
5. 生成动画 `weighted_scheduling.gif`

运行时间：约 40-60 秒

输出文件

1. `test_data_enhanced.json`

包含完整的测试数据和结果：

```
{  
  "seed": 42,  
  "n": 25,  
  "activities": [...],  
  "greedy_weight": 152,  
  "dp_weight": 268,  
  "diff": 116,  
  "ratio": 0.567  
}
```

2. weighted_scheduling.gif

规格：

- 分辨率：1920×1080 (1080p)
- 帧率：15 FPS
- 时长：约 35-45 秒
- 文件大小：约 25-30 MB

动画结构：

- **开场** (2-3秒)：活动逐个淡入显示
- **求解过程** (30-35秒)：左右分屏同步展示决策过程
- **结尾对比** (3-4秒)：最终结果对比，高亮显示差异

数据设计

核心"贪心陷阱"

活动0：[5, 35), weight=60
- 结束时间早（35），贪心优先选择
- 但会阻挡后续黄金区间

黄金区间（活动4-13）：[35, 100), 10个活动
- 总权重：239
- 与活动0冲突
- DP会跳过活动0，选择这10个活动

数据质量保证

程序自动验证以下指标：

- DP最终解 \geq 10个活动
- 权重差 (DP - 贪心) \geq 15
- 贪心最优率 \leq 95%

不满足条件会自动更换随机种子重新生成。

中文字体配置

Windows

自动使用 SimHei 或 Microsoft YaHei，无需配置。

运行示例

```
$ python weighted_scheduling.py
```

```
=====带权区间调度问题 - 可视化演示=====
```

[1/5] 生成测试数据（确保最优解 ≥ 8 个活动）...

验证：DP权重=268，贪心权重=152，差值=116，比例=56.72%

✓ 数据生成成功：DP选择11个活动，贪心选择4个活动

✓ 权重差值：116，贪心最优率：56.7%

✓ 已生成测试数据

[2/5] 贪心算法求解...

✓ 选择：[0, 14, 18, 23]

✓ 权值：152

✓ 验证无冲突... 通过

[3/5] 动态规划求解...

✓ 选择：[4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 24]

✓ 权值：268

✓ 验证无冲突... 通过

[4/5] 结果对比：

贪心：152 (活动数：4)

动态规划：268 (活动数：11)

差值：+116

✓ 动态规划更优！最优解包含11个活动（要求 ≥ 8 个）

[5/5] 生成动画...

开始生成动画，总帧数：620，预计时长：41.3秒

动画已保存至 weighted_scheduling.gif，文件大小：26.84 MB

```
=====完成!=====
```

生成文件：

- test_data_enhanced.json
- weighted_scheduling.gif

大模型使用说明

本项目使用 **Claude Sonnet 4.5** 辅助开发。

使用的关键词提示词

请帮我实现带权区间调度问题的贪心算法和动态规划算法，并生成可视化动画。

要求：

1. 设计"贪心陷阱": 权重60的活动阻挡总权重239的10个活动
2. 生成25个活动，最终解 \geq 10个活动，贪心最优率 \leq 95%
3. 左右分屏展示，右侧包含智能警告系统
4. 修复所有可能的错误（AttributeError、RecursionError）
5. 使用专业配色，统一高度时间线
6. 能离线一键运行