

# 说明文档

## 代码功能概述

代码实现了基于深度强化学习（DQN）的无人机（UAV）与路边单元（RSU）协同混合缓存与卸载模型，该系统主要用于优化边缘车联网中车辆用户任务的处理过程，通过合理地卸载任务至 UAV、RSU 或云端，并对混合缓存策略进行管理，从而降低任务响应时延。

## 模块组成说明

### A. 主程序模块 `main_test_DQN_to_cloud.py`

初始化系统环境并协调环境模块更新系统状态并计算奖励，可以设置迭代次数，车辆用户数量，每个用户产生的任务数等参数。

### B. DQN 模块 `DQN_multi_action.py`

实现深度 Q 网络，用于学习任务卸载决策和缓存操作的策略。可以设置学习率，奖励折现因子，经验回放缓冲区容量等

### C. 环境模块 `UAV_copy.py`、`RSU_copy.py`、`Users.py`、`Provider.py`

模拟实际的系统环境，`UAV_copy.py` 中可以设置无人机的相关属性设置等，`Users.py` 设置车辆用户的移动速度、任务生成方式、任务类型分布等，`RSU_copy.py` 定义路边单元相关参数设置，`Provider.py` 可以设置服务和内容的类型、大小、访问频率等

### D. 可视化模块 `Draw.py`

绘制训练过程中的性能指标曲线图并输出对应数据表格。

重要输入参数设置

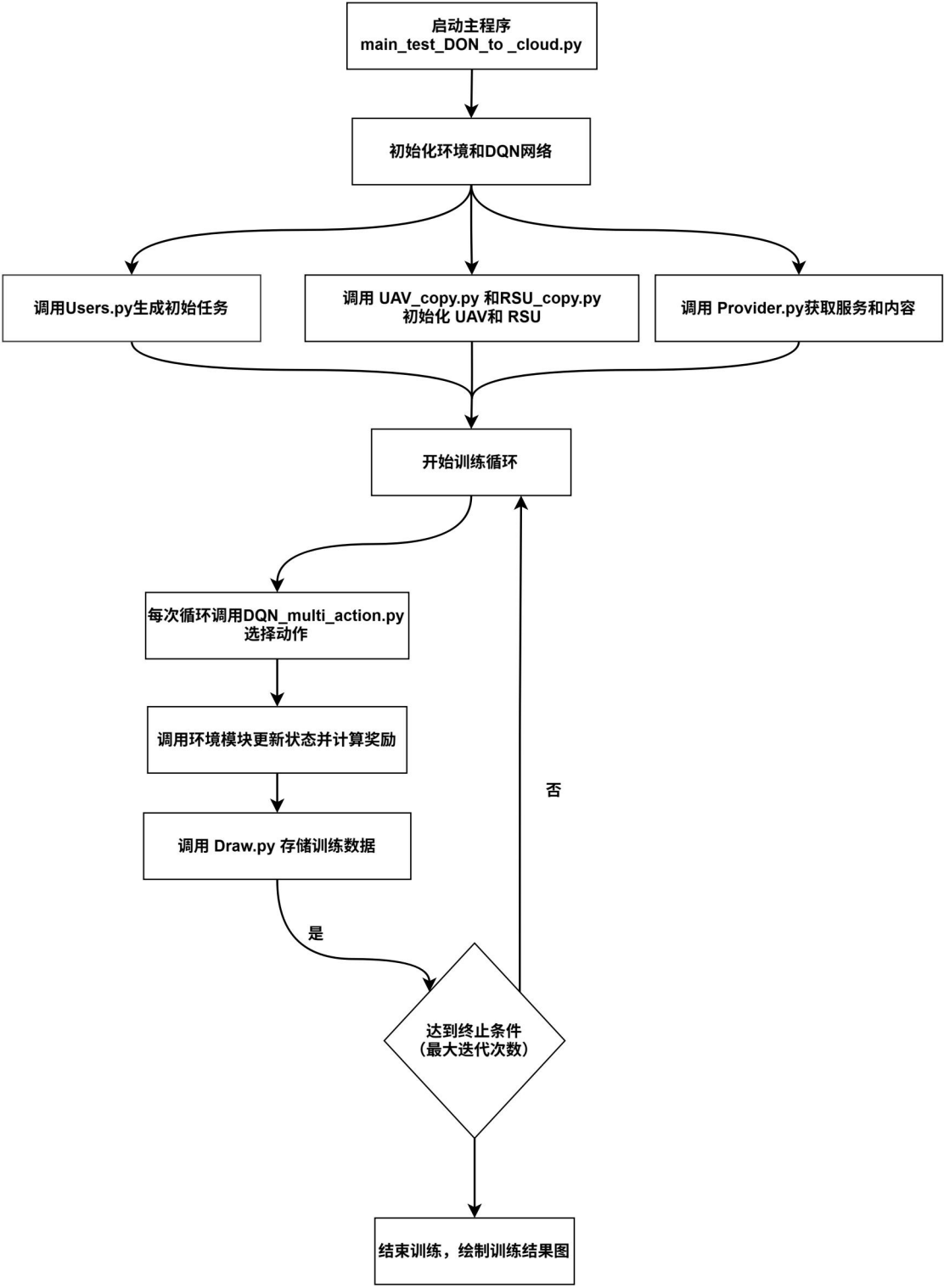
参数	描述	参考值	对应文件
episode_num	迭代次数	1000	main_test_DQN_to_cloud.py
users	用户数	6	main_test_DQN_to_cloud.py
user_tasks_num	生成任务数	10	main_test_DQN_to_cloud.py
LR	学习率	0.005	DQN_multi_action.py
GAMMA	折扣因子	0.9	DQN_multi_action.py

输出

决策结果：优化缓存策略得到的无人机上最优的缓存决策向量 C 和无人机中服务缓存和内容缓存的比例 $\alpha$ 。

系统性能指标：包括任务完成的总时延、奖励值、缓存配比参数以及损失值等。

系统工作流程



## 代码运行说明

### 1. 环境配置参考

Python==3.10.0, Numpy==1.23.5, Pytorch==1.13.0

注意 Pytorch, Numpy, Python 的版本兼容问题

### 2. 在终端或者命令行中启动训练过程

```
python main_test_DQN_to_cloud.py
```

### 3. 监控运行和输出

程序运行后, 在终端或命令行界面中会显示每个 episode 的进度信息, 包括状态更新, 动作选择, alpha 值等内容。

当训练达到预设的迭代 episode 数时, 程序会自动结束训练过程, 并输出指标图以及 xlsx 文件记录每个 episode 的信息, 例如:

