

平台后台性能测试

方案描述

本地同时发起n个连接，每个连接对应一场对战。对于一场对战，进行m局比赛。计算所有连接完成对战的总时间。

在二人局中，添加的AI均为CallAgent，一场比赛将在 4*2=6 次动作后结束，若每次动作的消息延迟小于0.1s，则一场比赛的时间应小于0.8s。

在二人局中，添加的AI均为CallAgent，一场比赛将在 4*6=24 次动作后结束，若每次动作的消息延迟小于0.1s，则一场比赛的时间应小于2.4s。

测试代码

- 执行命令 `python battle_n_pais_m_matches.py`
- 命令参数
 - `--num_players` 对战人数 2或6
 - `--connections` 同时发起的对战数
 - `--num_matches` 每场对战进行的局数
- 示例

```
python battle_n_pais_m_matches.py --num_players=2 --connections=10 --num_matches=2
# 同时发起10个连接，每个连接为2个AI对打2局
```

- 批量测试脚本
 - `bash run_test_2.sh`
 - `bash run_test_6.sh`

测试结果

2人

| 连接数----对战局数----总时间 | 1 | 10 | 20 | 50 |
|--------------------|-------|--------|--------|--------|
| 1 | 0.457 | 4.445 | 8.905 | 22.264 |
| 50 | 0.570 | 4.568 | 9.067 | 22.483 |
| 100 | 1.453 | 4.759 | 9.471 | 23.074 |
| 500 | 1.942 | 6.501 | 12.255 | 26.973 |
| 1000 | 2.038 | 9.468 | 16.478 | 34.934 |
| 3000 | 5.134 | 22.395 | 39.293 | 93.364 |

6人

| 连接数----对战局数----总时间 | 1 | 10 | 20 | 50 |
|--------------------|-------|--------|--------|---------|
| 1 | 1.218 | 10.938 | 21.724 | 53.913 |
| 50 | 1.221 | 11.251 | 22.065 | 55.632 |
| 100 | 3.131 | 13.859 | 25.568 | 59.243 |
| 500 | 4.372 | 21,908 | 42.362 | 101.920 |
| 1000 | 6.650 | 39.625 | 77.219 | 190.520 |