

问题求解(二)课程项目-第二阶段项目报告

林朗

完成情况：

2.1 使用 git 管理代码

共有 12 次 commit 记录，其中还有一些较小的更改与 commit。Commit history 将在之后的视频中展示。

2.2 文件分离

已实现文件分离。

2.3 收集素材

通过在淘宝购买素材与和同学互换素材，收集到了足够的图片、动画与音效素材。

2.4 开始菜单

开始菜单为一张图片与开始游戏/游戏说明/退出游戏/打赏四个选项，设置了循环播放的 bgm。

2.5 地图

按照项目要求创建了地图与 15*20 个单元；游戏场景设置了循环播放的 bgm。

2.6 玩家

完成了所有要求。还增加了人物死亡动画与音效。

2.7 机器人

机器人算法：

放炸弹：满足以下两个条件时机器人会放置炸弹：a.机器人在当前位置放置炸弹可以炸到软墙或者玩家/其它的机器人；b.机器人距离上次放炸弹已经过去了 3s；

后续会考虑优化解决机器人把自己炸死的问题；

移动：尝试向四个方向走和不动五种策略，利用赋权值的方法找出最优解

(保命为先，锁敌攻击为后)，并且考虑以下两种特殊情况：

- 待在原地不动马上就会死，抛弃原地不动的策略；
- 有多种策略的权值相同且最小（设定最小最优），那么就先锁定离自己曼哈顿距离最小的玩家/其它机器人，用 bfs 跑一遍最短路，如果最短路的

第一步移动对应的权值最小，那么就走这一步；

具体算法会在视频中展示；

上图为一阶段时的机器人算法。二阶段主要更改了如下内容：

1. 为最短路移动目标设置了权值，解除了其在特定情况下必走的判断；
 2. 优化了炸弹权值算法；
 3. 对即将被炸到的情况做了优化，使其不再成为必须移动的情况；
 4. 对放炸弹的算法进行了优化，只炸玩家和目前判断的最短路的障碍箱子，防止无意义的躲避和冒险；另外取消了 3s 的放炸弹固定间隔，让机器人也可以享受道具效果；
 5. 增加了寻找最短路的随机性，让机器人更难陷入死局；
- 值得一提的是，虽然对机器人进行了优化，但二阶段的地图箱子密度也设置的更大，因此整体表现并不会明显比在一阶段优秀（因为地图本身变难了）（但一阶段的机器人在多箱子地图中就很不智能）。

2.8 炸弹

炸弹释放后有动态的特效，爆炸时会生成光束（水波）并播放音效；

2.9 道具

软墙后有 1/5 的概率存在道具，相关内容均已完成。

2.10 其他显示

游戏界面右侧显示了实时得分、道具情况与倒计时。炸到一个箱子得 2 分，吃到一个道具得 8 分。此外，设置了游戏结束后的胜利/平局图片展示与音效，并锁定屏幕（但可以按返回）。

遇到的困难：

- 1.obj 在挂载与 destroy 之后，要等全部 update 完成后才能进行相关访问，否则会 crashed。被这个问题困扰了三天，de 了十数个 bug；
- 2.人物移动图片较多，而我不擅长用 ps，一张一张截图很痛苦；
- 3.项目框架理解耗费了较长时间。说实话直到大致完成项目第二阶段我才意识到框架的含义与经典而巧妙的地方。

实验心得：

- 1.学到了一些针对 crashed 的 debug 方法与心得；
- 2.养成经常 commit 的好习惯；
- 3.不要拖到最后一周再写。

致谢

李晗学长，邓振霄同学，胡皓明同学，鞠哲同学，王远博同学。