

TermProject 第三阶段报告

姓名: 邵钰乾 学号: 211240036

July 3, 2022

1 完成进度

本项目第三阶段的必做部分在 7 月 2 日全部完成, 详细说明如下:

- **对第二阶段的继续完善:**
 - **画面呈现:** 在第二阶段, 我已经实现了基于玩家和机器人运动状况 (坐标) 实现的动画、弹幕提示、状态栏等功能, 也添加了音乐和音效。所以在画面表现方面, 我没有做出改动;
 - **bug 修复:** 在第二阶段, 存在“当炸弹在道具处爆炸时, 爆炸范围不会延伸”的 bug, 现在已经修复;
 - **更好的炸弹:** 在第二阶段, 炸弹没有碰撞体积。当炸弹被踢飞时, 它们在第一道墙前停止。经过优化后, 现在它们在第一个炸弹前停止, 并把动量传递到下一个炸弹, 牛顿欣慰地流下了眼泪。
- **机器人:** 改进后的机器人使用 A* 算法进行寻路, 使用普通 bfs 躲避炸弹。详情将在下一板块进行说明。

2 过程记录

在这里主要介绍遇到的问题以及解决方案:

1. 机器人寻路算法

如果要寻找最短路, bfs 确实可以解决问题 (此处在讲解视频中可能有一些口误)。但是在寻路终点明确确定的情况下, 我使用了效率更高的 A* 算法进行优化。

A* 优化的灵魂是: $F = G + H$, 其中 G 是路径当前消费, H 是后续预期消费, F 是总消费。 G 的计算十分简单, 在我的算法中是每格加 10 花费; H 的计算则是基于机器人可以八向移动的前提 (尽管实际上它们并不能八向移动..... 这是为了避免向曼哈顿距离的退化), 算式为 $10 * |x_{distance} - y_{distance}| + 14 * \min\{x_{distance}, y_{distance}\}$ 。

相比于传统的 bfs, A* 使用优先级队列, 总花费最小的路径永远在顶部优先处理。这样就实现了有明确倾向的寻路, 较无目的地向四周广搜提升了性能。

但是由于在躲避炸弹时, 并没有一个明确的终点, 所以仍采用 bfs 进行寻路。

在游戏中可以使用组合键 shift + g 切换机器人难度, 使用组合键 shift + h 显示或隐藏寻路结果——这是一个调试功能, 我进行了一个保留。

运行效果还是可以接受的。

3 实验心得

做完了就好

4 致谢

老样子，首先要感谢李晗学长的框架！太好用了！同时也要感谢朱宇博学长的项目文档。其次要感谢张行健同学，我们进行了很多有益的交流。