

人工智能实验报告

实验名称：马尔可夫决策过程在机器人导航问题中的应用

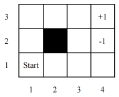
|  |  |  |  |
| --- | --- | --- | --- |
| 学生学号 | 202110120401 | 202110120420 | 202110120408 |
| 学生姓名 | 刘广润 | 赵威 | 喻慧超 |
| 分工 | 算法实现,测试算法性能，撰写报告 | 算法速度测试，绘制表并进行对比，撰写报告 | 优化算法、撰写报告 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **实验评价结果** | | | | |
| **占比** | 10% | 40% | 50% | 总分 |
| **评分标准** | 报告的规范性。报告中的术语、格式、图表、数据、公式、标注及参考文献是否符合规范要求。 | 报告的严谨性。结构是否严谨，论述的层次是否清晰，逻辑是否合理，语言是否准确。 | 实验的充分性。实验是否完备，数据是否合理，是否有创新性成果或独立见解。 |
| 评分 |  |  |  |  |

报告提交时间： 2023 年 12 月 23 日

# 试验任务

图1.1(a)是一个机器人导航问题的地图，黑色格子是障碍物。机器人从起点Start出发进行连续移动，移动过程中机器人知道所在的格子。机器人每次移动一格，移动前必须在上下左右中选择一个方向，但是由于地板打滑的原因，实际移动的结果并不一定是在所选择的方向上。如图1.1(b) 所示，机器人每次移动的实际结果是机器人以0.8的概率移向所选择方向，也可能是以0.1的概率移向垂直于所选方向。如果实际移动的方向上有障碍物，则机器人会停在原地，继续进行移动决策。如果机器人进入标有+1和-1的格子，则终止移动。机器人移动到图中每个格子，会获得一份报酬，图1.1(a) 中标有+1 和-1 的格子中标记的就是该格子的报酬，其他格子的报酬是-0.04，报酬会随着时间打折。用Markov决策的知识计算问题的价值函数，以及机器人的最佳策略。



# 实验平台

Windows操作系统，Python编译环境，math，random，matplotlib等程序库。

# 实验原理

## 马尔可夫决策过程

马尔可夫决策过程（MDP）是一种数学模型，用于描述在不确定性环境中的决策问题。它以一种结构化的方式模拟决策者如何通过一系列的决策达到目标，特别适用于那些决策的结果不仅取决于当前的行动选择，而且还受到某种随机因素影响的情形。

MDP由几个关键元素组成：状态、动作、转移概率和奖励。在MDP框架中，决策者在每个时刻都处于某种状态，并在可选的动作集合中选择动作。选择的动作决定了系统转移到下一个状态的可能性，这种转移通常具有一定的随机性，这就是转移概率。同时，每次状态转移都伴随着一定的奖励或惩罚，反映了每个状态或状态转移的价值。

MDP的核心问题是找到一个最优策略，这个策略定义了在每个状态下应该选择哪个动作，以便最大化长期累积的奖励。这通常涉及到在立即奖励和长期收益之间做出权衡，特别是在面对复杂的、多步骤的决策过程时。

在实际应用中，MDP被广泛用于各种领域，如机器人路径规划、经济决策模型、生态系统管理等。通过构建MDP模型，可以更好地理解和预测复杂环境中的决策后果，并设计出更加智能和高效的决策策略。

## 价值迭代

价值迭代是求解马尔可夫决策过程（MDP）的一种算法，它通过迭代地更新状态价值来找到最优策略。在价值迭代中，每个状态的价值代表了从该状态出发，遵循最优策略所能获得的最大期望回报。算法的核心是贝尔曼最优性方程，它为每个状态定义了其价值，即该状态的即时奖励加上后续状态的折现价值的最大值。

价值迭代的过程是这样的：初始化所有状态的价值为任意值（通常为0），然后反复更新每个状态的价值。在每次迭代中，对于每个状态，算法计算执行所有可能动作后的期望回报，选择最大的一个作为该状态的新价值。这个过程重复进行，直到所有状态的价值不再发生大的变化，即达到收敛。

价值迭代的关键在于，它不直接寻找最优策略，而是通过逐渐逼近每个状态的最优价值来推导出最优策略。一旦价值函数收敛，最优策略可以通过选择每个状态下能够带来最大期望回报的动作来得到。

价值迭代的优点在于其实现简单直接，适用于各种MDP问题。但其缺点也很明显：在状态空间很大时，计算效率可能会很低，因为它需要在每次迭代中更新每个状态的价值。此外，如何确定价值函数是否已经收敛也是一个挑战，因为这通常需要设定一个阈值，而这个阈值的选择可能会影响到算法的效果。尽管如此，价值迭代仍是强化学习和决策理论中一种非常重要的工具。

## 策略迭代

策略迭代是解决马尔可夫决策过程（MDP）中最优策略问题的一种有效方法，它通过迭代改进策略来逐渐逼近最优解。与价值迭代不同的是，策略迭代在每个迭代步骤中显式地维护和更新一个策略。

策略迭代包含两个主要步骤：策略评估和策略改进。在策略评估阶段，算法对当前策略进行评估，计算在该策略下从每个状态出发所能获得的期望回报。这通常通过解决一系列线性方程来完成，每个方程对应一个状态，其解给出了该状态的价值。这个价值反映了在当前策略下，长期从该状态出发可以获得的累积奖励。

一旦评估完成，算法进入策略改进阶段。在这个阶段，算法会检查每个状态，看看是否有其他动作能带来比当前策略更高的期望回报。如果有，那么在这个状态下选择这个更好的动作，从而改进当前策略。换句话说，策略改进是基于当前策略价值的最新估计来进行的。

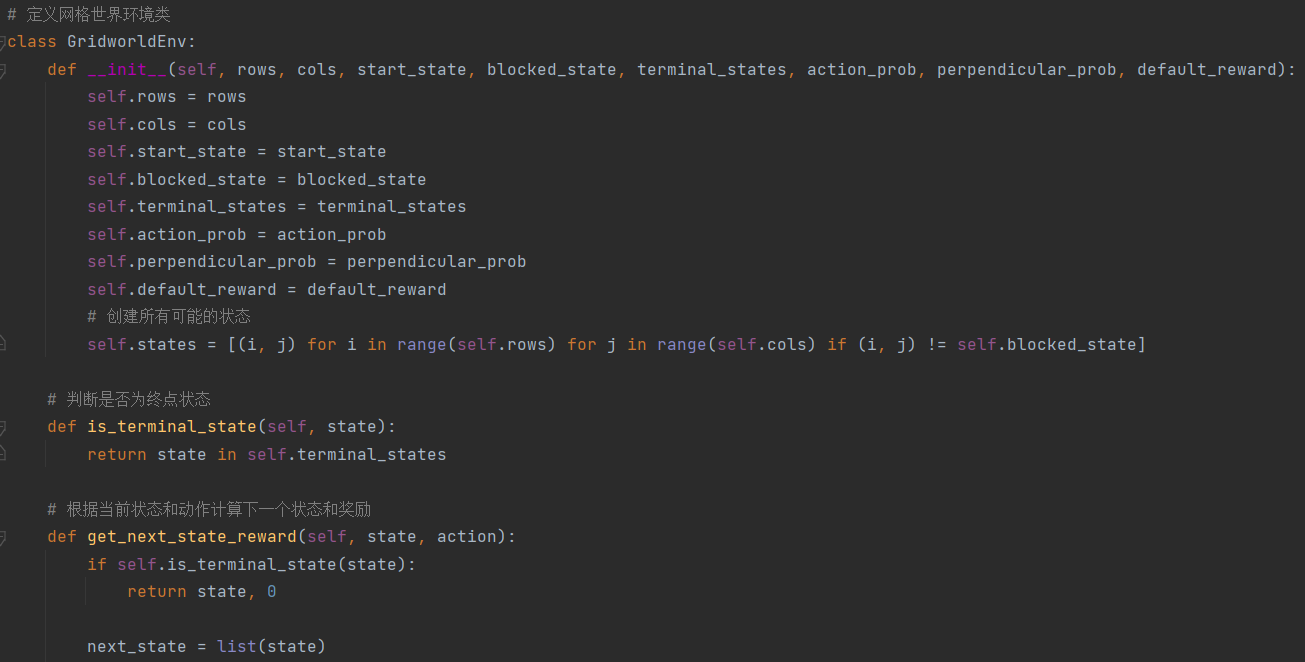
这两个步骤交替进行，每次策略评估后都紧接着进行一次策略改进。这个过程重复进行，直到策略不再发生变化，此时算法收敛，找到了最优策略。

策略迭代的优点在于通常收敛速度比价值迭代快，因为每次策略改进都是基于最新的价值估计。此外，由于在每次迭代中都维护一个显式的策略，所以在任何时刻都可以提供一个有效的策略。但其缺点在于每一次策略评估都可能相对耗时，特别是在状态空间较大时，因为它涉及到解决一组线性方程。

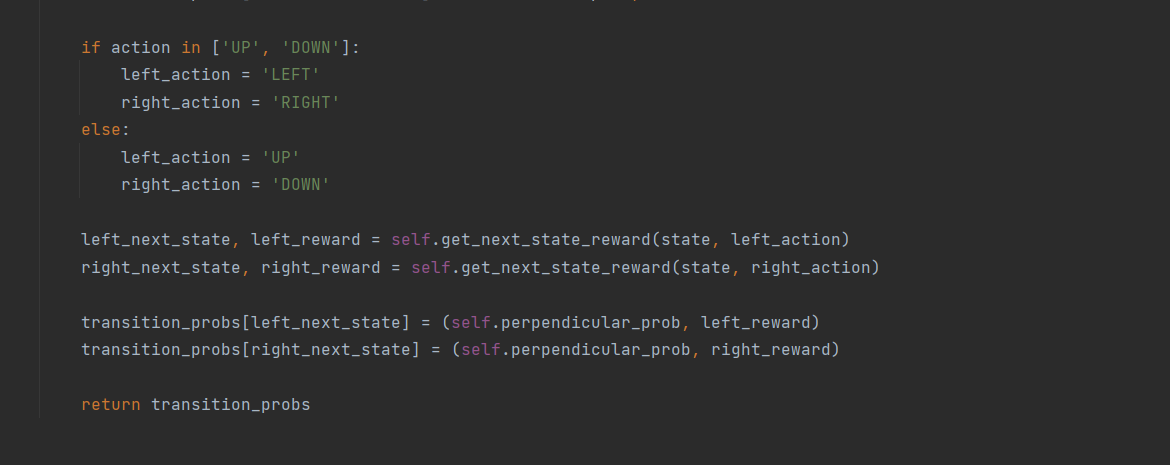
总的来说，策略迭代是一种在理论上和实践中都非常重要的方法，适用于各种MDP问题，尤其是在那些状态空间不是特别大的情况下。

# 实验步骤

## 建立ENV模块



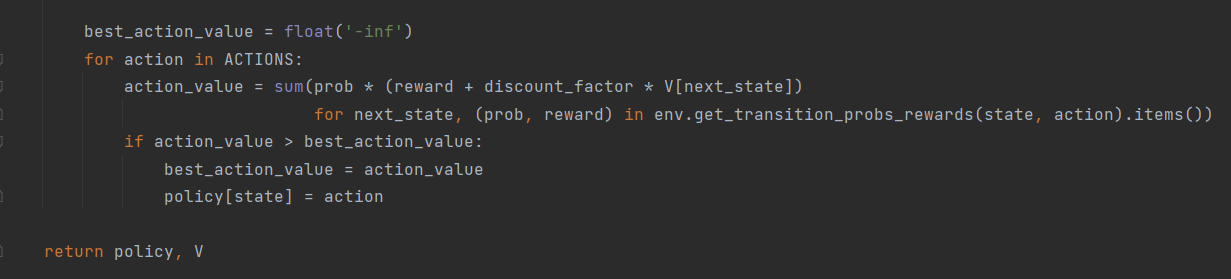




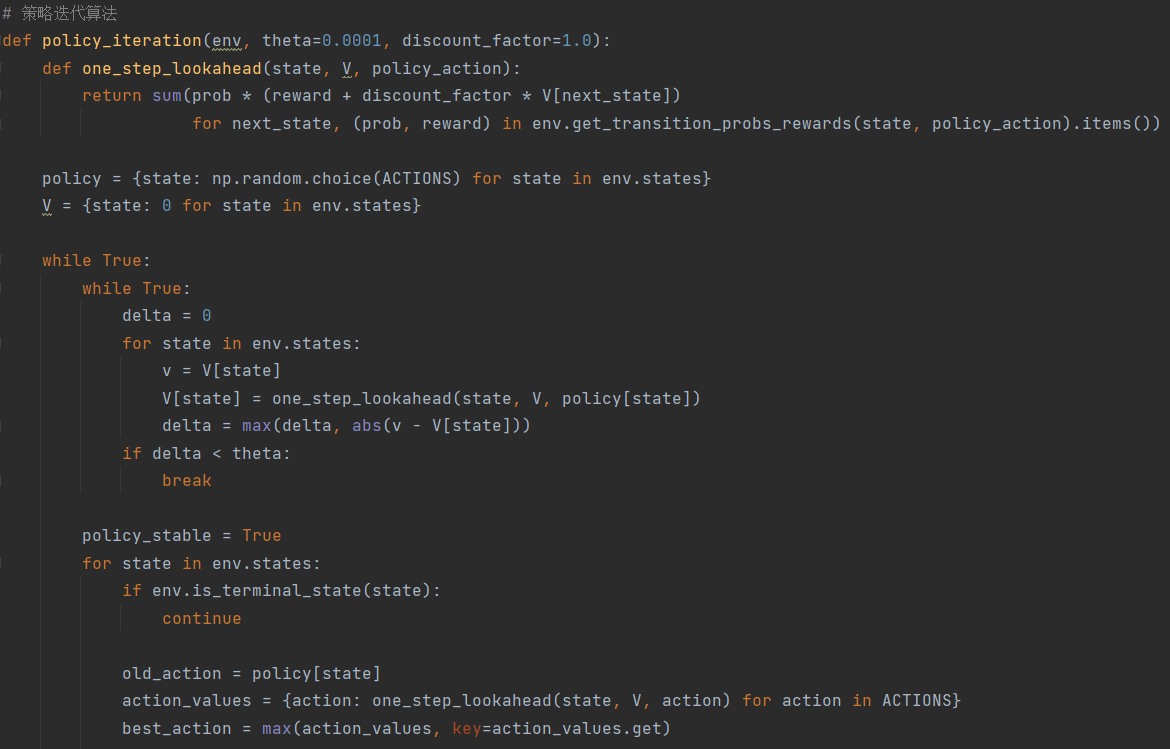
## 实现算法

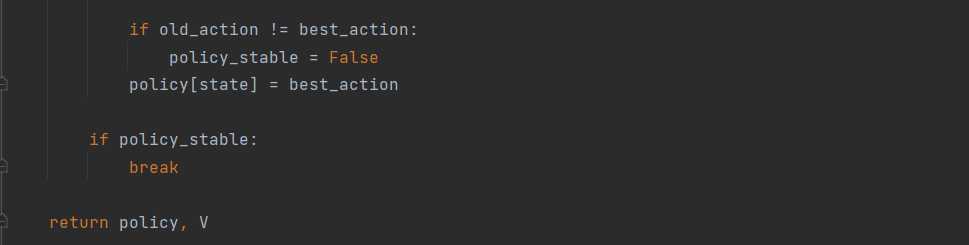
## 价值迭代





## 策略迭代





# 实验结果分析

## 参数分析

在分析实验结果的参数时，我们通常会综合考虑报酬定义、折扣值和转移概率等因素对最优策略的影响。实验结果显示，不同的报酬定义会直接影响算法寻找到达目标状态的路径，高报酬值通常鼓励算法快速寻找到目标，而低或负报酬值则使算法更加谨慎。折扣因子影响了未来奖励的价值，高折扣因子使算法更重视长期回报，而低折扣因子则让算法更专注于短期回报。转移概率的变化影响了算法对行动结果的确定性预期，高转移概率意味着行动结果更加可预测，而低转移概率会使算法在选择动作时更加谨慎，以避免不利的结果。通过对这些参数的综合分析，我们能够更好地理解不同环境设置下的最优策略选择，以及如何根据具体情况调整算法参数以达到更好的性能。

## 算法比较

价值迭代和策略迭代是两种在马尔可夫决策过程（MDP）中寻找最优策略的主要算法。它们各有特点和适用场景。

价值迭代算法直接关注于找到每个状态的最大价值，而不是直接寻找最优策略。它通过迭代更新状态的价值，直至收敛。每次迭代中，价值迭代算法计算并更新所有状态的价值，基于当前价值估计来决定最佳动作。这种方法的主要优点是实现简单、直接，并且总是能保证找到最优策略。然而，它的缺点是在大型状态空间中可能需要很多迭代才能收敛，且每次迭代都需要对所有状态进行更新，这可能导致计算量很大。

策略迭代算法在每次迭代中交替进行两个步骤：策略评估和策略改进。在策略评估步骤中，算法评估当前策略下每个状态的期望回报；在策略改进步骤中，它尝试找到能提高这些期望回报的新策略。策略迭代的优点是通常能更快收敛到最优策略，因为每次策略改进都基于最新的价值估计。其缺点是算法实现相对复杂，特别是在策略评估阶段可能需要求解大型线性方程组，这在状态空间很大时可能非常耗时。

总的来说，价值迭代在实现上更简单，适合于那些对收敛速度要求不高的应用场景。而策略迭代虽然在计算上可能更复杂，但它在许多情况下能更快地收敛，尤其适合于状态空间较小，或者能够有效解决策略评估中的线性方程组的情形。

# 代码

https://github.com/SonicAge1/Markov