黑色的仙人掌

中度可信度描述已自动生成

**计算机与信息学院**

**2023年秋季学期课程论文**

**《人工智能》**

|  |  |
| --- | --- |
| 课程类型： | 专业核心/必修 |
| 学 号： | 202110120401 |
| 姓 名： | 刘广润  赵威  喻惠超 |
| 专 业： | 计算机科学与技术 |
| 授课教师： | 盛冠群 |

完成日期：2024年1 月 10 日

基于强化学习的机器人导航问题

, ,

摘 要：本实验致力于实现强化学习在机器人导航问题中的应用，特别是在建立和实施马尔可夫决策模型的环境中。我们实现了用于模拟机器人在未知网格世界中的移动和决策过程的环境模块。然后应用时序差分（TD）学习方法来计算价值函数和行动价值函数。我们还采用了线性函数逼近拟合状态价值函数。实验的目标是利用RL学习到与实际意义相符的状态价值函数和行为价值函数，通过实验结果检验强化学习算法的效果。

**关键词：强化学习，TD学习，状态价值函数，行为价值函数，线性拟合**

# 1 引言

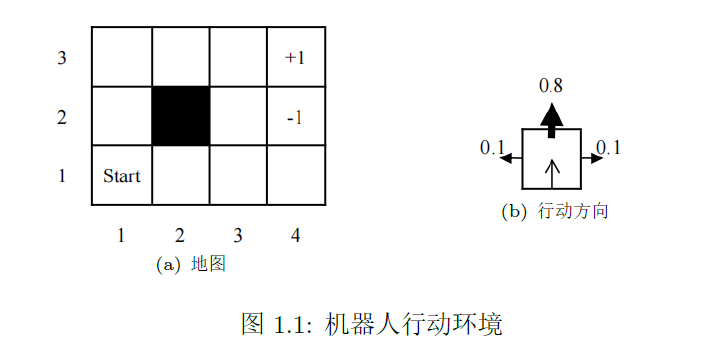
本实验聚焦于强化学习在机器人导航问题中的应用，具体涉及建立马尔可夫决策模型、实现环境模块、运用时序差分（TD）学习，并通过线性函数逼近来计算价值函数。研究核心在于模拟机器人在未知环境中的学习过程。

我们首先构建了一个环境模块，模拟机器人的状态转换和相应奖励，为学习算法提供了基础框架。然后应用TD学习方法评估状态和行动的价值，并采用线性函数逼近。最后，通过马尔可夫决策方法计算价值函数和最优策略通过可视化状态价值和行为价值、调整奖励策略来评估和提高算法输出与实际意义的契合程度。

# 2 问题描述

图1.1(a)是一个机器人导航问题的地图。机器人从起点Start出发，每一个时间点，它必须选择一个行动(上下左右)。在马尔可夫决策中实验中，机器人是根据环境模型中的转移矩阵 P (x, a, x) 来进行价值函数和最优策略的计算，但是本次实验中，机器人并不知道这个转移矩阵。已知机器人行动之后，环境会告知机器人两件事情–新的实际位置以及到达新位置所得到的报酬。因此，如果机器人有一个策略π(x)，那么在与环境的交换中，机器人会具有这样一个数据序列：(x0, a0, r0, x1, a1, r1, . . . , an−1, rn−1, xn)，其中xi, ri 是环境告知的状态和该状态的报酬，ai = π(xi)是机器人自己的决策，x0是Start, xn 是一个终止状态。这个数据序列也称为一个样本路径。强化学习的任务是让机器人在环境中运行多次，得到多条样本路径，通过这些样本路径，来求解最优策略。

样本路径是与环境的交互中产生的，你先要实现一个环境模型。假设实际位置由环境按图1.1(b) 的方式决定：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。机器人移动到图中每个格子，会获得一个报酬, 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，假设折扣是1。



# 3 算法说明

在解决机器人导航问题时，我们用到以下几种方法：

## 3.1 基于MDP的环境模块构建

在本次实验中，环境模块的构建基于用于模拟机器人导航问题马尔可夫决策过程（MDP）。这一模块是整个实验的核心，它定义了机器人在虚拟世界中的行为和规则。MDP提供了包括状态空间、行动空间、转移概率和奖励函数的决策过程的数学描述。

在这个环境中，每个网格点代表一个可能的状态，构成了状态空间。机器人的目标是从起点出发，通过一系列决策（上、下、左、右）来到达目标。行动空间由这四个方向的移动组成，每个行动可能会导致状态的转变。机器人在每个状态下需要做出决策，以导航到目标状态。

状态转移在这个环境中是随机的，有0.8的概率向前移动或者有0.2的概率向垂直于前方的方向移动。奖励机制根据机器人的当前状态和所采取的行动来分配奖励或惩罚。例如到达目标状态会获得正奖励，而进入不良状态或执行无效行动则会获得负奖励。这种奖励设置可以引导机器人学习有效的导航策略。

## 3.2 时序差分学习

时序差分（TD）学习是是一种介于蒙特卡洛方法和动态规划方法之间的学习算法。TD学习的核心特点是在学习过程中同时使用当前的估计来更新价值函数。

在本次实验中，TD学习用于估计状态价值函数（V）和行动价值函数（Q）。因为它们提供了每个状态或行动的期望收益的估计，所有这些函数对于机器人做出决策非常重要。

**状态价值函数（V）：**这个函数给出了从某个状态开始，遵循特定策略所能获得的预期回报。在每次机器人移动并接收到奖励后，根据TD学习方法，状态价值函数会根据新的观察值进行更新。

**行动价值函数（Q）：**类似于状态价值函数，但是它为每个状态和在该状态下可能采取的每个行动分配一个值。这个函数的更新同样遵循TD学习的原则，即基于实际收到的奖励和对下一个状态的价值估计。

TD学习的更新规则可以表述为：

其中， 是学习率， 是折扣因子，是收到的奖励， 和 分别是当前状态和下一个状态， 是在状态 下采取的行动。

通过这种方式，TD学习允许机器人从每一次与环境的交互中学习，逐渐逼近最优策略。

## 3.3 线性函数逼近

**线性函数逼近的原理**

* 线性函数逼近是一种简化方法，它假设价值函数可以作为状态特征的加权和来表示。这些特征通常是原始状态的某种变换或表示。
* 在实验中，状态特征通过独热编码表示，即每个状态由一个特征向量表示，其中一个元素为1，其余为0。这样的表示形式可以简化学习过程，并降低计算复杂性。

**权重的更新**

* 线性逼近中的权重是通过与实际奖励和预测奖励之间的差异（时序差分误差）相关的方式进行更新的。
* 在每一步中，算法根据当前状态和下一状态的特征向量，以及收到的奖励来调整权重。这一过程不断重复，旨在减小预测值和实际值之间的差异。

**应用于TD学习**

* 在TD学习中，线性函数逼近用于改进状态价值函数（V）和行动价值函数（Q）的估计。
* 状态价值函数的更新可以表示为：，其中 是权重向量， 是状态 的特征向量。

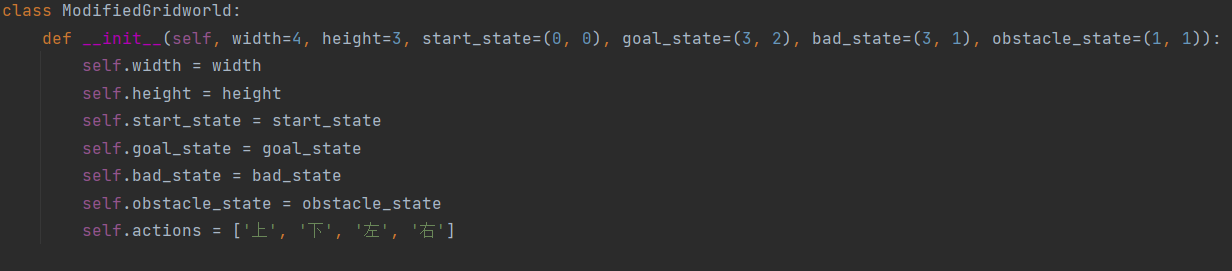
在实验中使用线性函数逼近能够更有效地处理大型状态空间，还能够通过逐步调整权重以更好地逼近实际的价值函数，从而提高机器人导航策略的效率和准确性。

# 3 问题解析

## 3.1. 环境模块（ModifiedGridworld 类）：

**类定义和初始化：**

* ModifiedGridworld 类定义了网格世界的环境。
* 初始化时，设定了网格的尺寸（宽度和高度）、起始状态、目标状态、不良状态（即负奖励状态）和障碍物状态。
* 参数决定了网格世界的基本结构，包括哪些区域可通行以及特殊区域的位置。



**状态和行动空间：**

* 每个网格点对应一个状态，表示为一个坐标（x, y）。
* 行动空间包括四个方向的移动：上、下、左、右。这些行动决定了机器人在网格中的移动路径。

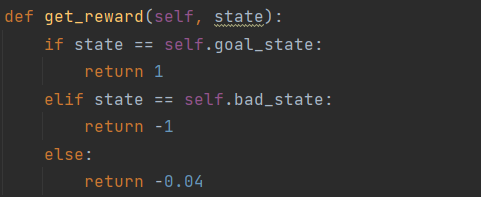
**move 函数：**

* move 函数是环境模块的核心，它根据机器人的当前状态和选择的行动来更新状态，并返回相应的奖励。
* 函数首先检查机器人是否已在目标状态或不良状态。如果是，机器人保持当前状态，并返回相应的奖励。
* 如果不是，则检查选择的行动是否会导致机器人移动到障碍物或越过边界。若是，则机器人保持当前状态并获得负奖励。否则，根据行动更新机器人的位置，并根据新位置计算奖励。



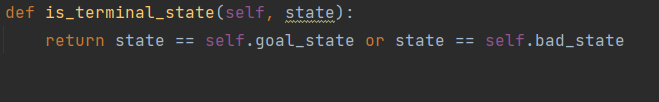
**奖励机制（get\_reward 函数）：**

* get\_reward 函数定义了不同状态下的奖励机制。到达目标状态获得正奖励，进入不良状态或进行无效行动（如撞到障碍物）获得负奖励。其他普通状态则有一个较小的负奖励，如-0.04，表示每步移动的成本。



**终止状态判断（is\_terminal\_state 函数）：**

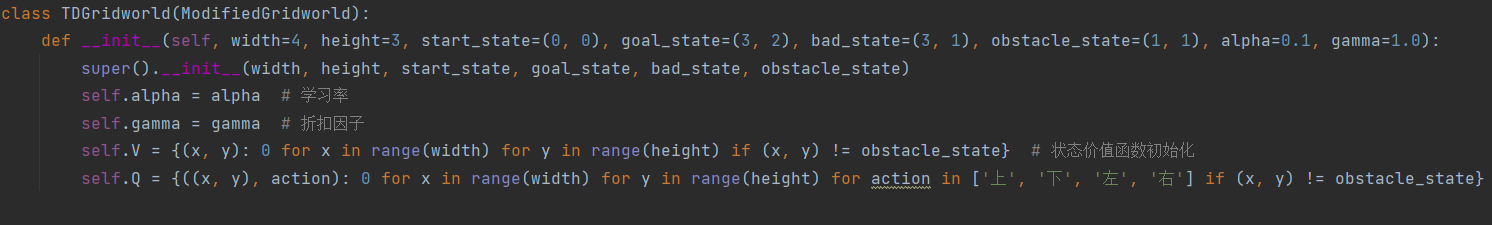
* is\_terminal\_state 函数用于判断当前状态是否为终止状态，即目标状态或不良状态。在这些状态下，机器人完成或终止其导航任务。



## 3.2. 时序差分学习（TDGridworld 类）：

**TDGridworld类:**

* 继承自ModifiedGridworld类，TDGridworld类实现了TD学习算法。该类利用环境（ModifiedGridworld）生成的状态转移和奖励来学习价值函数。
* 类中定义了状态价值函数V和行动价值函数Q。这些函数分别用于估计状态和特定行动的期望回报。



**学习参数:**

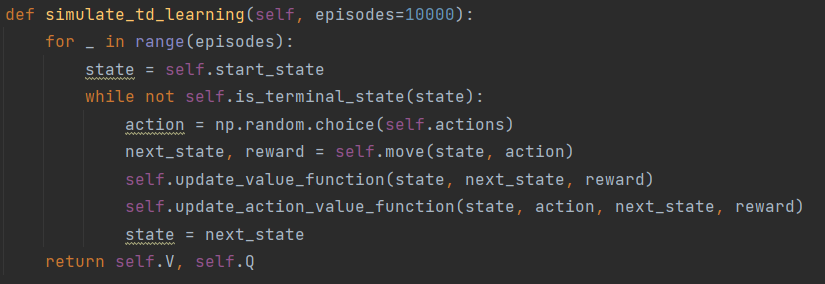
* 在类的初始化中设置了学习率alpha和折扣因子gamma。学习率控制着价值函数更新的步长，而折扣因子确定了奖励的重要程度。

**价值函数更新:**

* 状态价值函数V和行动价值函数Q通过TD学习的方法更新。在每一步的状态转移后，根据实际奖励和估计的未来回报来调整当前状态或行动的价值估计。
* 更新规则体现了TD学习的核心思想：当前的价值估计是基于之前的估计和最新的观察（即奖励和下一状态的价值）。

**simulate\_td\_learning:**

* 此函数模拟了与环境的多次交互，是学习过程的主体。
* 在每一轮模拟中，从起始状态开始，根据当前策略选择行动，环境返回下一状态和奖励。然后，根据这些信息更新价值函数。
* 这个过程重复进行，直到达到终止状态，从而完成一次完整的学习过程。通过多次重复这个过程，机器人能够逐渐改善其导航策略。



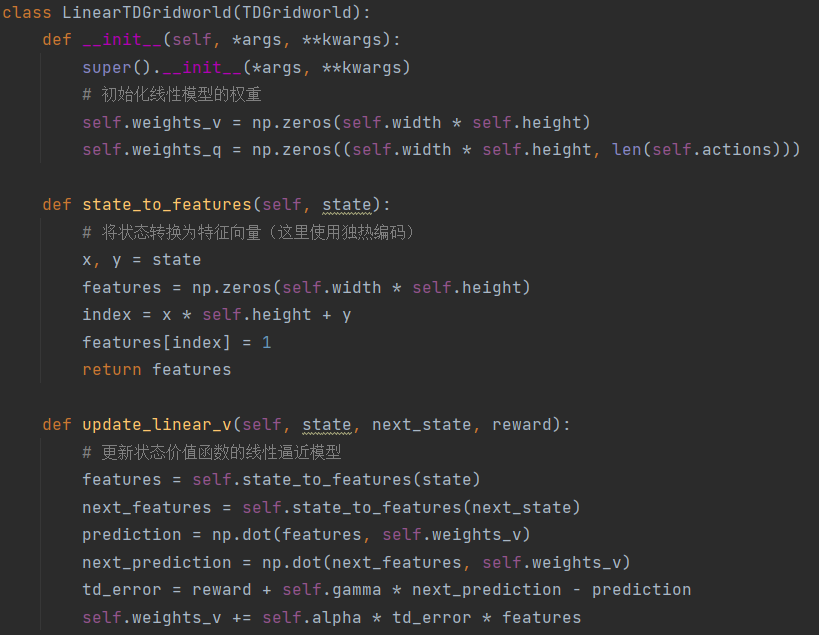
**策略优化:**

* TD学习使得机器人能够通过经验（即与环境的交互）来优化其策略。通过不断地更新价值函数，机器人学习如何在给定的环境中作出更好的行动选择。

## 3.3.线性函数逼近（LinearTDGridworld 类）：

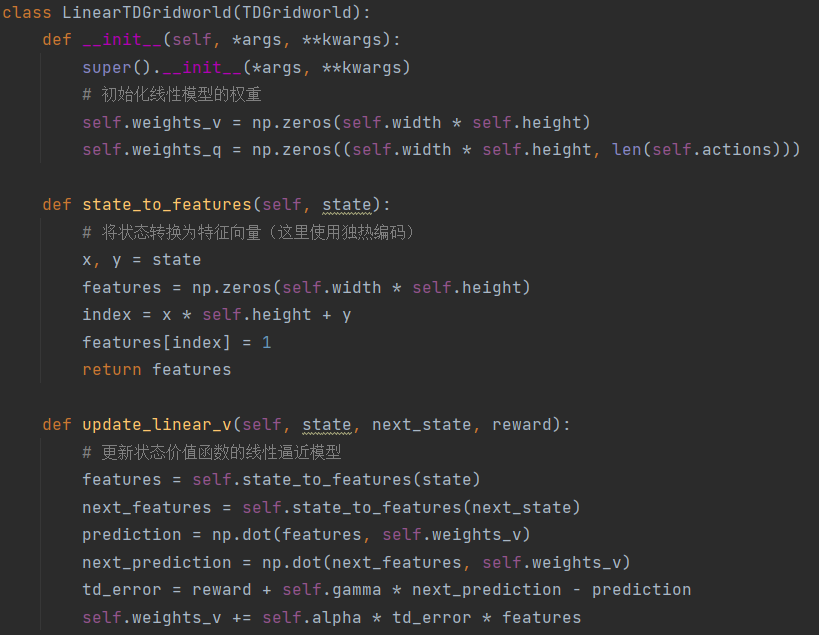
**特征表示（state\_to\_features 函数）**：

* 状态被转换成特征向量，这里采用的是独热编码方式。每个状态由一个长度等于状态空间大小的向量表示，向量中只有对应于该状态的索引处的元素为1，其余为0。
* 这种表示法简化了状态的复杂性，使得每个状态可以通过一个固定长度的向量唯一标识。



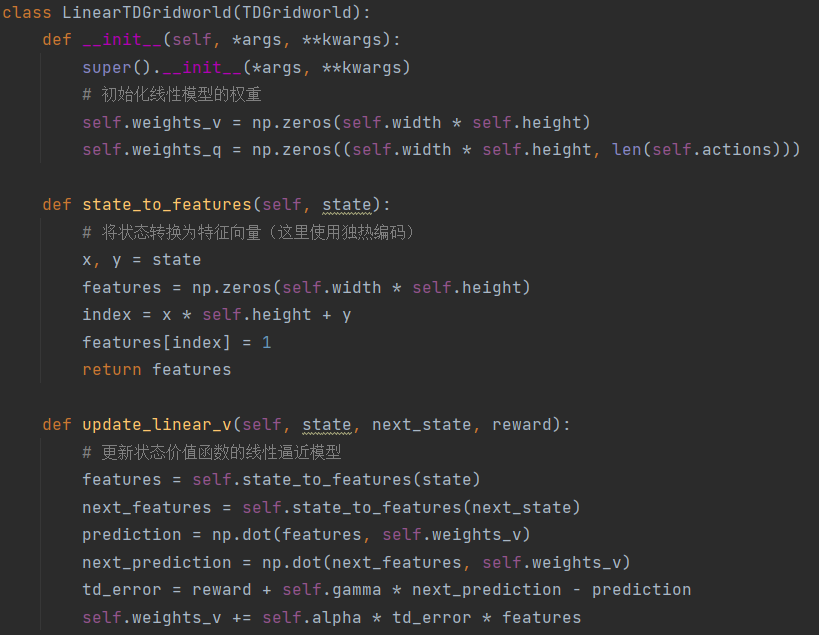
**线性权重初始化（LinearTDGridworld 类）**：

* 类中初始化了权重向量，weights\_v用于状态价值函数 V 的线性逼近。
* 权重向量初始设置为零，随后根据从环境中的数据进行调整。



**权重更新（update\_linear\_v函数）**：

* 这些函数使用时序差分误差来更新权重向量。通过计算实际收到的奖励和基于当前权重的价值估计之间的差异，权重被相应地调整以减少这种差异。
* 更新过程反映了线性逼近的核心思想：利用当前的观测数据来优化对未来状态价值的估计。



# 4 结果分析

## 4.1状态价值函数（V）

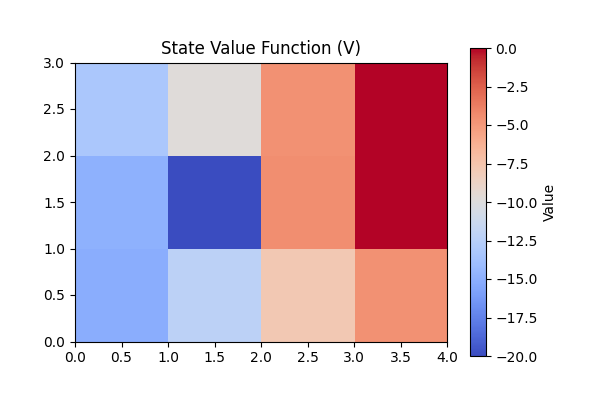


图 1状态价值函数热力图

状态价值函数V提供了在特定状态下，基于当前策略的预期回报的信息。以下是对状态价值函数V的详细分析：

**任务背景：**

* 任务中的网格世界代表了一个环境，其中机器人从起点出发，目标是达到一个特定的终点，同时避免不良状态或障碍物。
* 在这个过程中，每一步的行动会带来一定的奖励或惩罚，根据所处的状态不同而变化。

**状态价值函数V的解释：**

* 您提供的V值是通过时序差分学习得到的，反映了在各个状态下，基于当前策略的长期预期回报。
* 对于每个状态，V 值越高，意味着从该状态出发，最终获得正回报的可能性越大。

**分析具体数值：**

* 例如，状态(0, 0)的V值为-15，表明在此位置离终点最远，所以状态价值价值最低，相比之下其他距离终点更近的点则状态价值更高
* 状态(3, 2)和(3, 1)的V值为0，因为这些是终点状态，一旦达到即表示任务完成，没有进一步的行动。

**状态(1, 1)（障碍物位置）的特殊处理：**

* 状态(1, 1)的V值被设置为-20，这是一个特别的设计，用来表示障碍物位置，即不可通行的状态。这个值的设定反映了对机器人导航的额外约束。

**总体趋势：**

* 距离目标状态越近的位置，其V值应该越高，因为到达目标所需的步骤更少，预期回报更高。
* V值的负数表明机器人在到达目标之前预期会有一定的损失，这因为机器人在随机行进的过程中会持续受到负奖励。

## 4.2行为价值函数（Q）

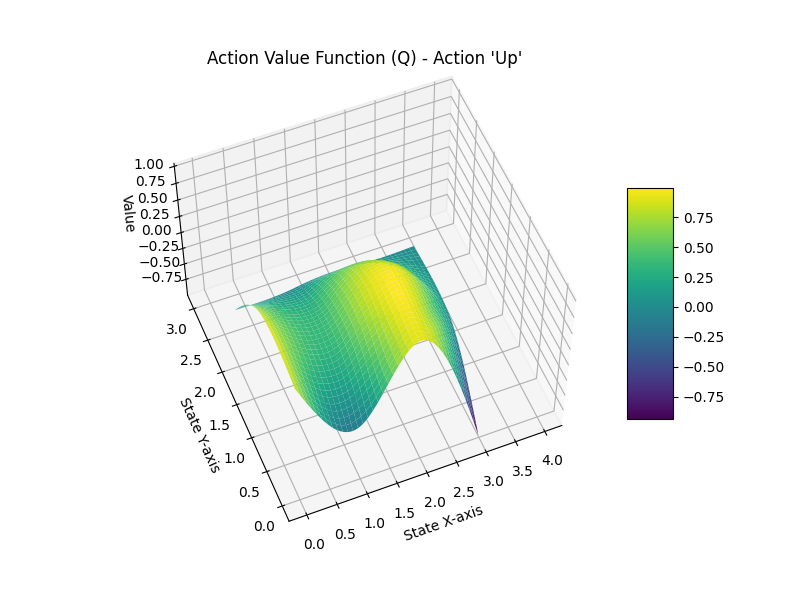


图 2 行动价值函数（上）

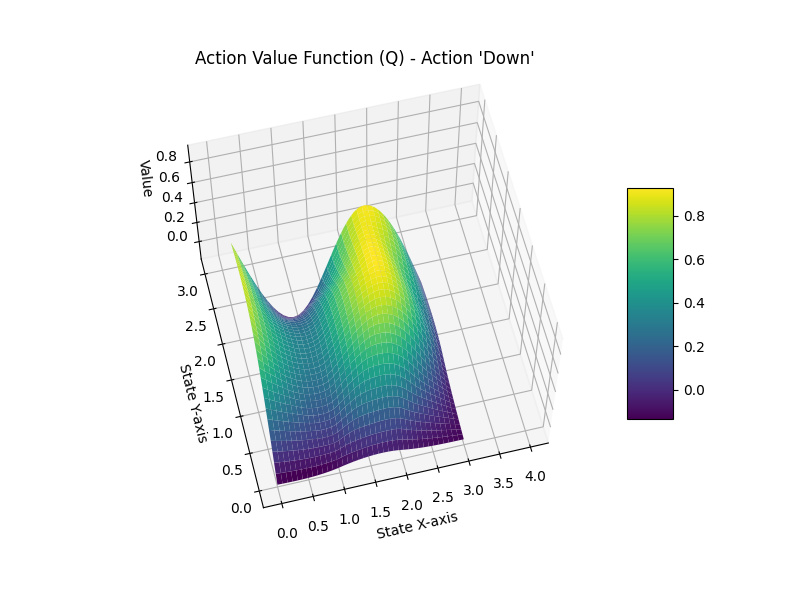


图 3 行为价值函数（下）

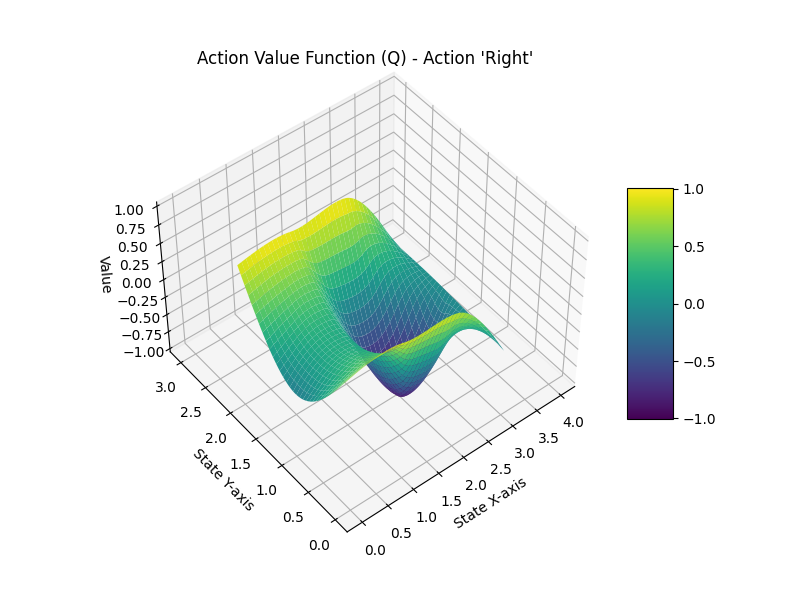


图 4 行为价值函数（右）

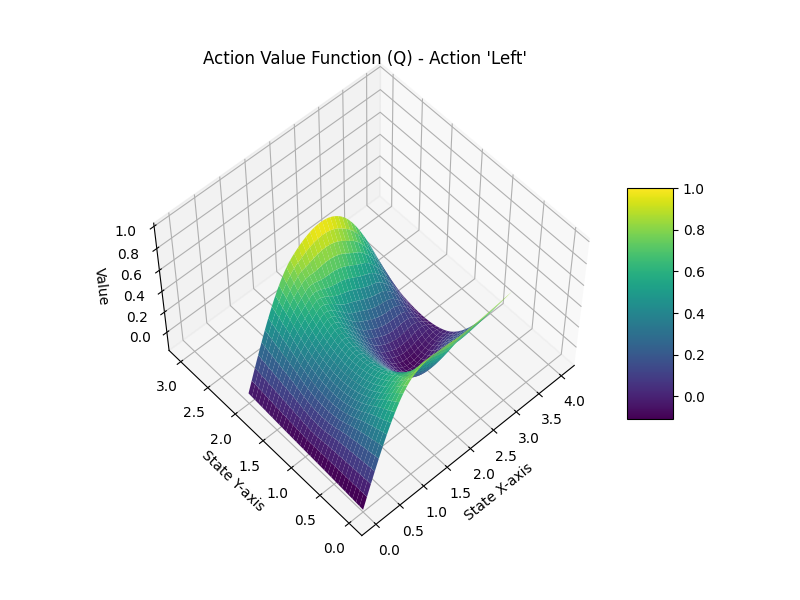


图 5 行为价值函数（左）

行动价值函数 Q为每个状态的不同行动对提供了估计的回报值。以下是针对行动价值函数Q的详细分析：

**行动价值函数Q的含义：**

* Q函数为每个状态-行动对（在特定状态下执行特定行动）分配一个值，该值代表执行该行动后所能获得的预期总回报。
* Q值是通过TD学习方法计算得到的，反映当前策略下的不同行动的行动效果。比如说在图5中，(2,1)点有一个明显的凹陷，这代表如果移动到(3,1)点会受到极大负影响，实际上(3,1)是不良状态，预测价值与实际价值相符。

**Q值的解释：**

* 例如，对于状态(0, 0)，向上移动的Q 值为0.8，而向下移动的Q值为负数。这表明从(0, 0)状态向上移动比向下移动预期回报更高，实际上在(0,0)处向下移动会撞墙，正确的选择确实应该是向上或向右，所以基本正确。

**终点状态的Q值：**

* 在目标状态(3, 2)和不良状态(3, 1)，所有行动的Q值都被设置为0，因为当机器人到达这些状态时任务就结束了，没有进一步行动的必要。

**行动价值函数的应用：**

* 通过比较在同一状态下不同行动的Q值，机器人可以选择预期回报最高的行动，从而优化导航路径。

## 4.3 线性拟合状态价值函数

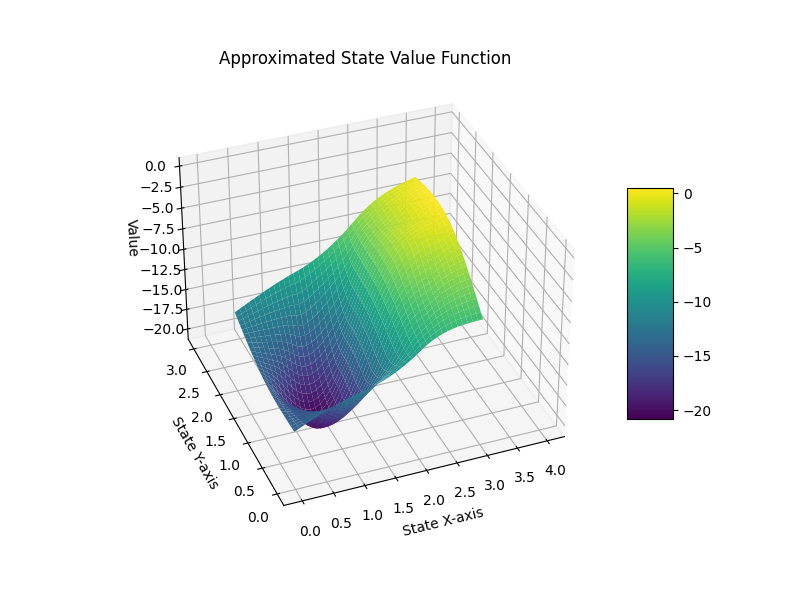


图 6 线性拟合状态价值函数

**权重向量 (weights\_v) 分析：**

* 输出中的weights\_v是线性拟合过程中学习到的权重。这些权重与状态特征的线性组合可以决定每个状态的估计价值。
* weights\_v中的每个值代表对应特征在价值函数中的重要性。正值表明特征对提高状态价值有正向影响，负值则表明有负向影响。

**状态价值函数V的拟合效果：**

* 图6显示每个状态的价值估计。例如状态(0, 0)的价值为-15、而(1,1)点因为是障碍物，所以被设置为了-20。
* 比较图1和图6数据可以看出，线性拟合方法可以在误差很小的近似原始的价值函数V。

## 4.4与马尔可夫决策方法得到价值函数和最优决策的对比

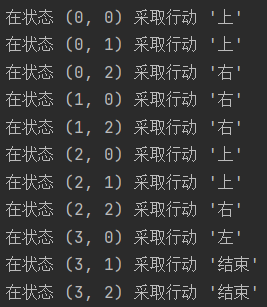


图 7 马尔可夫决策方法最优决策

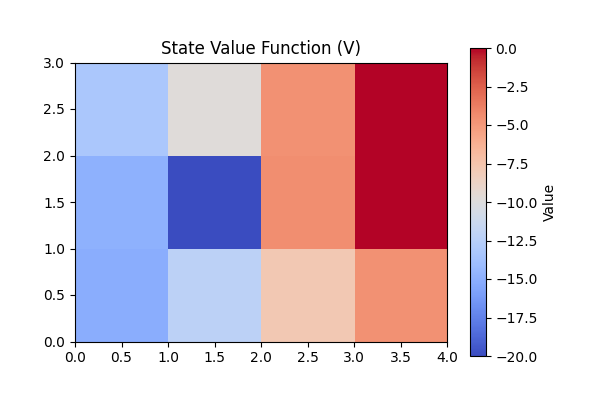
****

图 8 马尔可夫方法状态价值函数

由图7和图8可见，由马尔可夫方法得到的状态价值函数和最优决策与由TD学习得到的状态价值函数和最优决策基本相同，有效验证了强化学习结果的正确性。

# 5 总结

本实验在机器人导航问题中通过时序差分（TD）方法来计算价值函数和最优策略。实验中，我们首先构建了一个简单的网格世界环境，其中包括了起始点、目标点、障碍点和不良点，以及机器人可以执行的动作集。基于这个环境，我们实现了TD学习并用MDP的价值迭代算法检验了其有效性。

实验结果显示，尽管TD学习和MDP方法在理论上和实现方式上存在差异，但它们在这个特定问题中得出了高度相似的价值函数和策略。这一结果表明，在无法完全知晓环境动态的情况下，TD学习作为一种无模型的强化学习方法能够有效地逼近真实的价值函数和策略。这一点对于那些环境模型难以获取或不完全的实际应用场景具有重要意义。

此外，通过线性函数逼近价值函数的尝试进一步展示了TD学习在实际应用中的灵活性和泛化能力。尽管简化的线性逼近无法完全捕捉到复杂环境中的所有细节，但它提供了一种计算效率更高、实现更简单的替代方案。

# 6 代码地址

https://github.com/SonicAge1/RL

|  |  |  |  |
| --- | --- | --- | --- |
| **项目** | **分值** | **评价依据** | **评分** |
| 论文格式及书写 | 10 | 论文格式是否严格按照规定执行，论文中的术语、格式、图表、数据、公式、引用、标注及参考文献均符合规范要求，字体撰写是否工整认真。 |  |
| 论文结构与文字  表达 | 40 | 论文结构是否严谨，逻辑性如何，论述层次是否清晰，语言是否准确；内容主题是否鲜明，是否符合课程教学内容。 |  |
| 论文内容 | 50 | 论文内容是否紧扣结课论文主题，篇幅是否达到结课论文要求，参考文献充分、正文引用恰当，是否坚持课程论文原创，有无抄袭剽窃现象。所得结论是否属实，所列举实例是否完整，代表性如何。论文是否有创新性成果或独立见解。 |  |
| 总分 | | |  |
| 评语：  论文格式严格按照规定执行，结构严谨，逻辑性强，论述清晰。    　　　　　　　　　　　　教师签名： 日期： | | | |