Применение алгоритма Q-learning для обучения погрузочной машины на складе

Задача

Применение алгоритма Q-learning для обучения погрузочной машины на складе

Описание задачи:

- Робот представляет собой разгрузочно-погрузочную машину на складе
- Задача робота — перевезти груз в указанную часть склада
- Склад представлен набором коридоров, по которым может перемещаться робот
- На пути робота могут встречаться препятствия — стены склада

Модель задачи:

- Склад может быть представлен как лабиринт
- Робот — объект с физическими параметрами
- Возможности перемещения робота ограничены размерами склада

Инструменты: модель

Для решения задачи необходимо определить

- **Среду — модель склада**. Это можно сделать посредством имитационного моделирования
- **Способ выбора траектории робота**. Для этого предлагается использовать **алгоритм машинного обучения Q-learning**.

Определение среды и состояний включает:

- **Состояния:** положение робота, расположение стен.
- **Действия:** движение робота вперед, назад, поворот налево или направо.
- **Вознаграждение:** положительное за приближение к цели, отрицательное за столкновение с препятствием.

Алгоритм Q-learning:

- Робот обучается на основе взаимодействия с окружающей средой.
- Q-функция обновляется на основе действий, которые выбирает робот, и получаемого вознаграждения.

Инструменты: модель

Алгоритм Q-learning:

- Алгоритм обучения с подкреплением
- Цель: оптимизация стратегии действий
- **Принцип работы:** Агент выбирает действия на основе данных из Q-таблицы.

  Q-таблица

  - отражает ожидаемую **суммарную награду** за выполнение действия
  - Строки таблицы — возможные состояния среды, столбцы — возможные действия агента в этих состояниях
  - В начале обучения Q-таблица **инициализируется нулями** или случайными значениями. После каждого шага агента **Q-значения** в таблице **обновляются** с использованием **уравнения Беллмана**.
  - Действие выбирается с помощью жадного алгоритма на основе значений из Q-таблицы

Инструменты: модель

Уравнение Беллмана

$$Q(s,a) = r + \gamma \max_{a'} Q(s',a')$$

Где:

- *Q(s, a)* — ожидаемое Q-значение состояния и действия,
- *r* — немедленная награда, полученная после выполнения действия *a* из состояния *s*,
- *γ* — коэффициент дисконтирования, который представляет собой важность будущих наград (обычно значение между 0 и 1),
- $\max_{a'} Q(s',a')$ — максимальное Q-значение для всех возможных действий *a'* из следующего состояния *s'*.

Инструменты: реализация

Модель была реализована с использованием средств языка Python.

Для моделирования **среды** использована библиотека **PyBullet** — симулятор физики с поддержкой 3D-сред. Она позволяет симулировать движения робота с учетом столкновений и динамических препятствий.

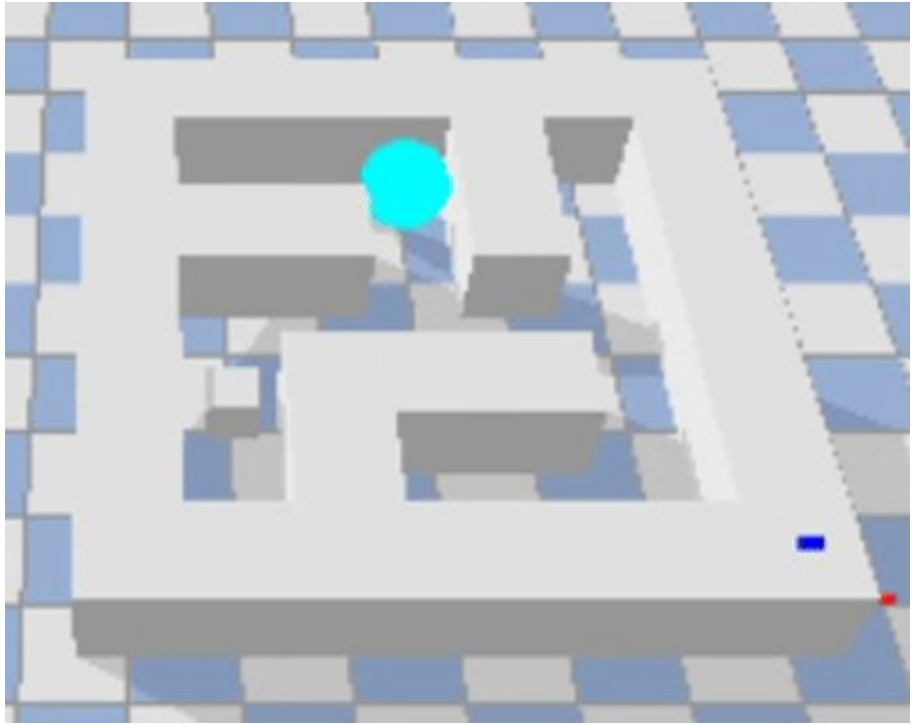
Также был реализован алгоритм **Q-learning** в соответствии с описанным ранее алгоритмом.

Результаты

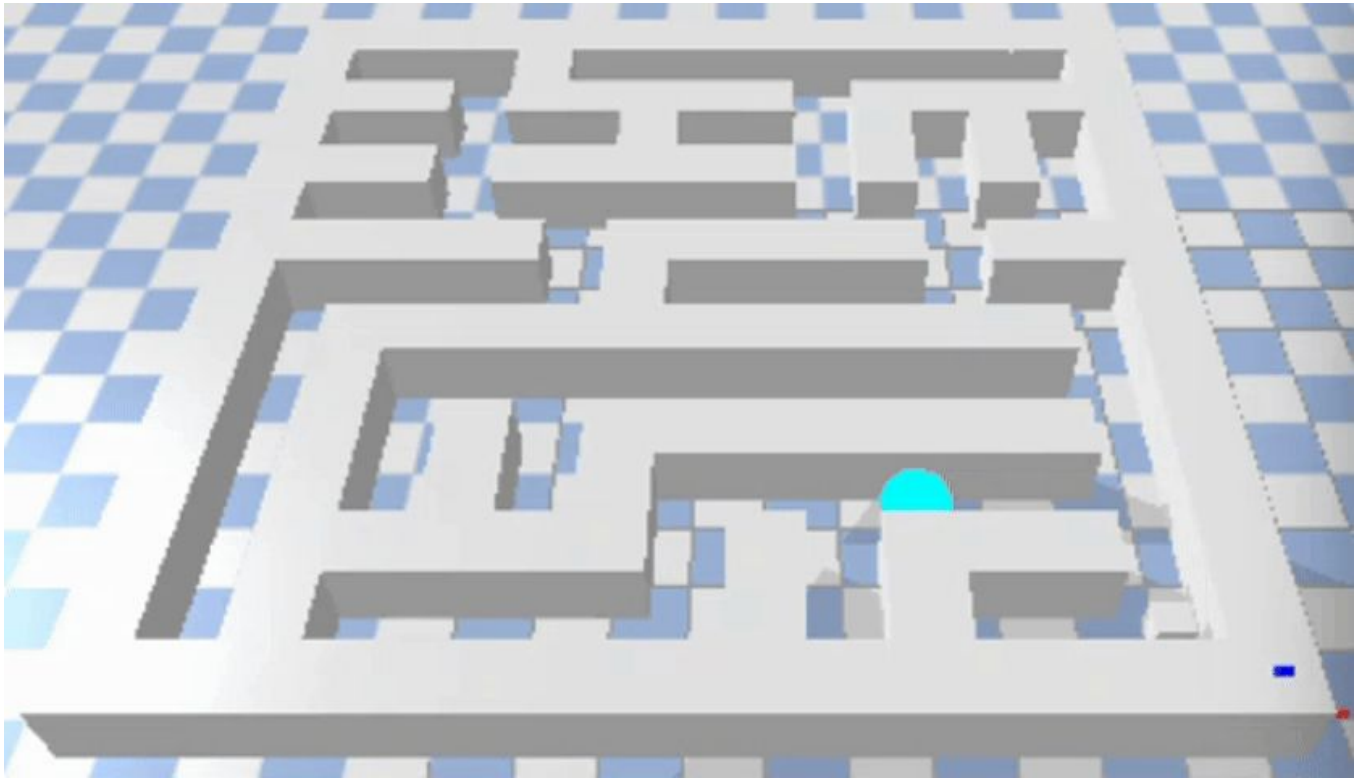
Целевая точка: [-5, 2, 0.5]

Траектории: [[[-3, 5, 0.5], [-3, 4, 0.5], [-3, 3, 0.5], [-4, 3, 0.5], [-5, 3, 0.5], [-5, 2, 0.5]], [[-3, 5, 0.5], [-3, 4, 0.5], [-3, 3, 0.5], [-4, 3, 0.5], [-5, 3, 0.5], [-5, 2, 0.5]], [[-3, 5, 0.5], [-4, 5, 0.5], [-3, 5, 0.5], [-3, 4, 0.5], [-3, 3, 0.5], [-4, 3, 0.5], [-5, 3, 0.5], [-5, 2, 0.5]]...

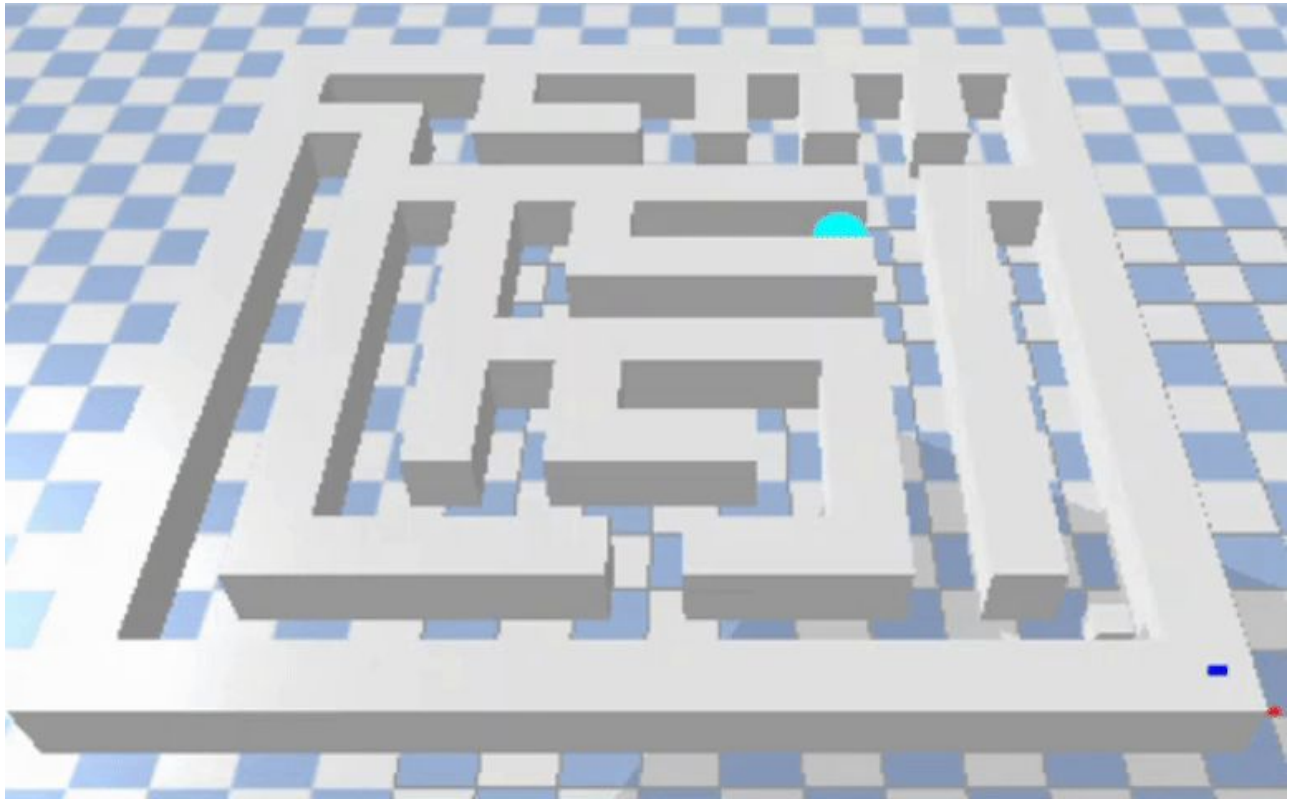
Выбрана траектория: [-3, 5, 0.5], [-3, 4, 0.5], [-3, 3, 0.5], [-4, 3, 0.5], [-5, 3, 0.5], [-5, 2, 0.5]



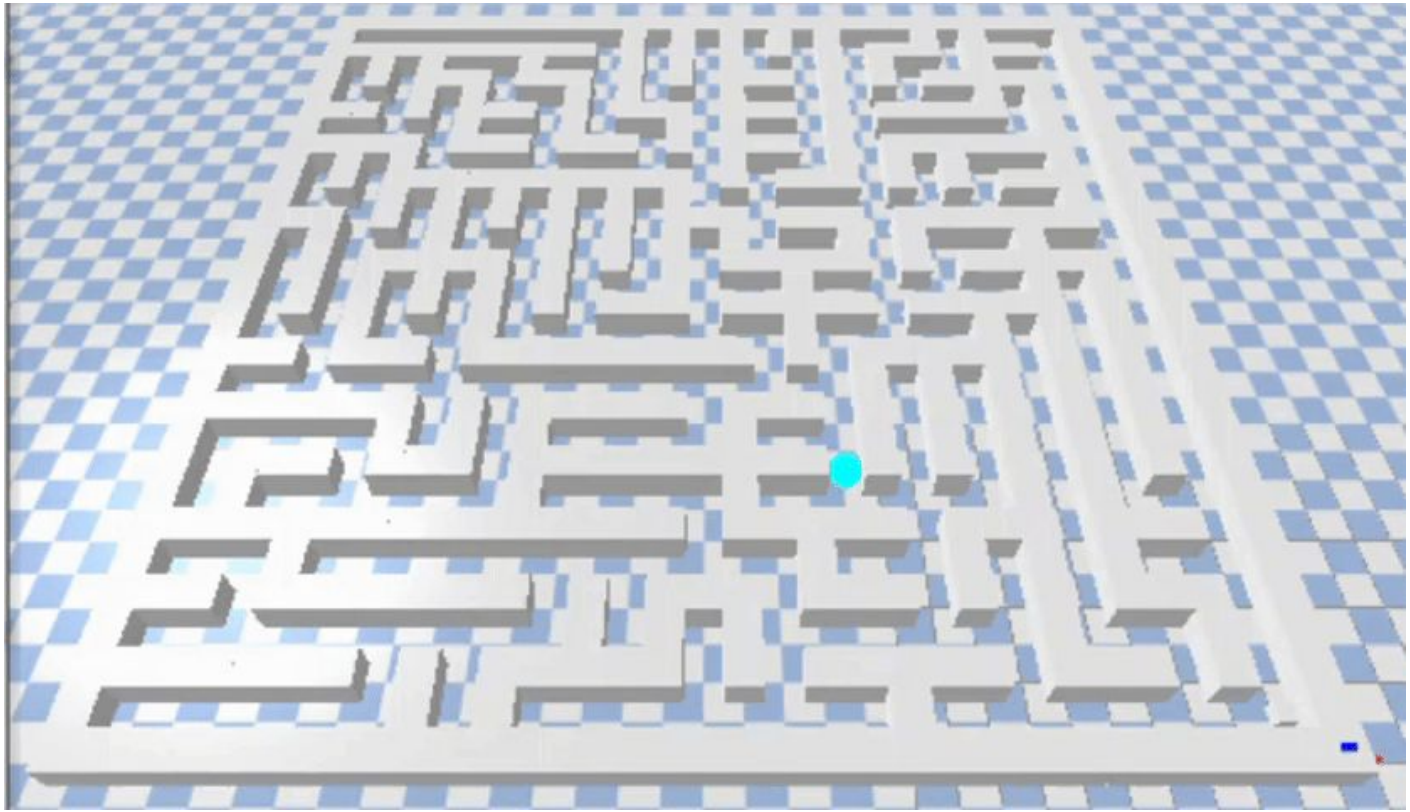
Результаты



Результаты



Результаты



Выводы

При решении задачи

- Проведено моделировании среды — склада, представленного лабиринтом
- Реализован алгоритм Q–learning
- Реализован выбор агентом (роботом) траектории на основании результатов работы алгоритма Q–learning

Спасибо за внимание!