《无人系统设计》课程作业 04

第23组 陶青筱 陆伊敏 汪逊杰 黄喆敏

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. 代码逻辑

我们均采用**TD3模型**进行训练。因为TD3对于DDPG进行了优化，用类似Double DQN的方式，解决了DDPG中Critic对Q值过估计的问题；并且延迟了Actor的更新，**使得Actor的训练更加稳定**。我们在经过尝试后，也发现TD3比DDPG稳定得多，更容易收敛。

在查找相关文档后，我们发现在MATLAB R2022a版本的Reinforcement Learning包中，并不支持使用GPU训练，因为相应的接口为老版本接口。因此，我们采用了demo里的方式，使用CPU并行的方法训练。

对于代码，我们进行了以下修改：

1. 向前直线行走

我们在demo的基础上，修改了训练参数。根据尝试，当Reward大于150时，效果较好，且最终Reward峰值能够达到200以上。因此设置为Reward到达200时，自动保存Agent；Episode数量最大设为4000。

我们对Reward进行了修改，修改为。即去掉了前一时间步的扭矩项。因为demo的目标是使用最小控制力，而我们并没有这一目标。

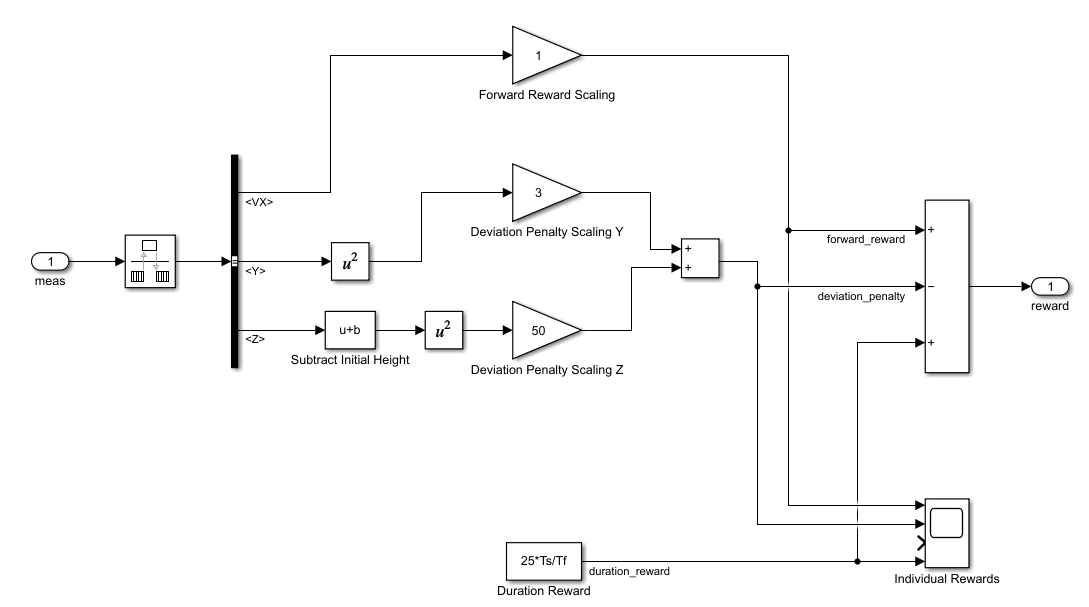


图1.1 向前行走Reward示意图

1. 向后直线行走

与向前行走类似，我们将训练参数设置为Reward到达200时，自动保存Agent；Episode数量最大设为4000。

在前一问的基础上，我们对Reward进行了修改，修改为。因为是向后行走，因此x方向速度系数为-1；我们增加了y方向速度惩罚项的系数，防止其走偏。

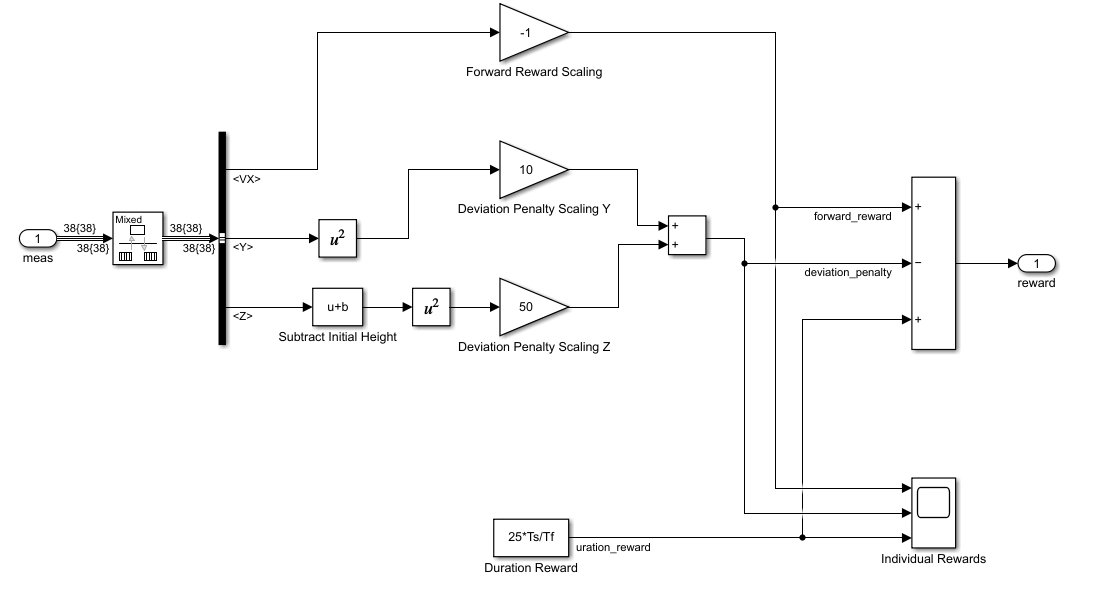


图1.2 向后行走Reward示意图

最后模拟时，需要调整轴的方向，使得机器人走在轴上。因此我们调整Walking Robot中的Spline参数，改为[0 0 0.025; -5 0 0.025]。

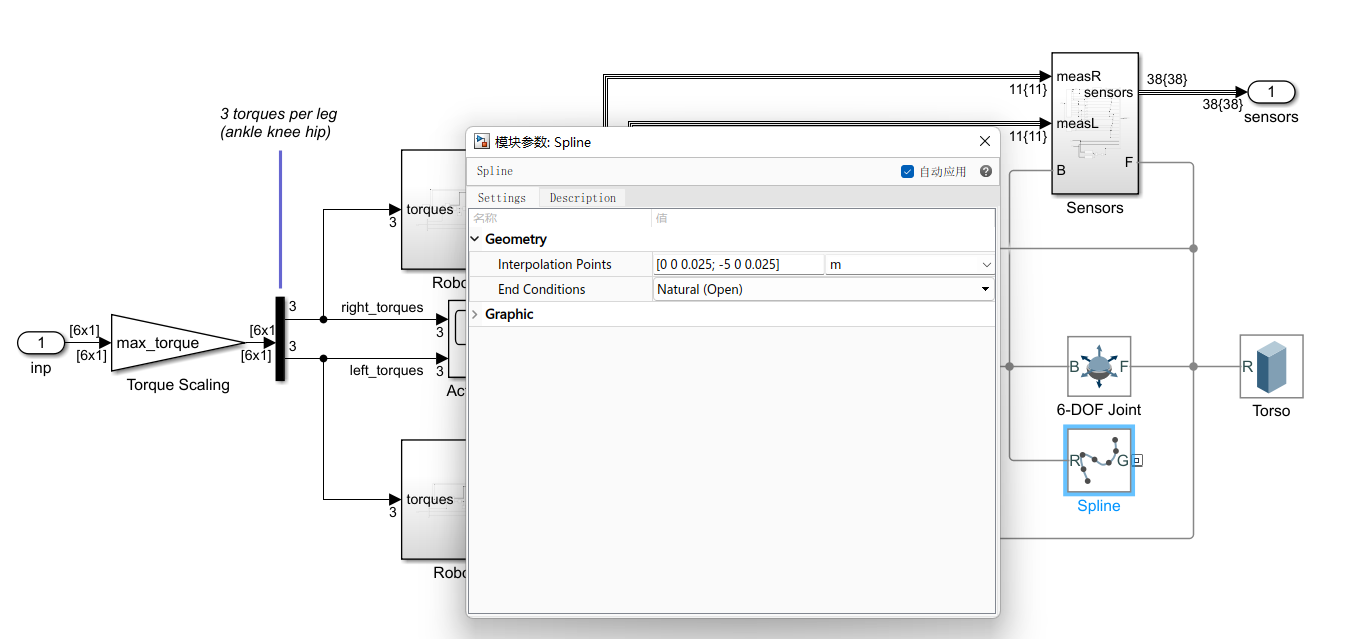


图1.3 向后行走调整轴方向示意图

1. 训练结果
2. 向前直线行走

训练的Reward以及结果如下。您可以在video/walkingForward.mp4中查看训练效果。

表一 向前直线行走训练结果

|  |  |
| --- | --- |
|  | RlTD3Agent |
| Status | Training finished |
| Episode number | 2000 |
| Episode reward | 179.8511 |
| Episode steps | 400 |
| Total agent steps | 129539 |
| Average reward | 76.7949 |
| Average steps | 191.612 |
| Episode Q0 | 16.0536 |
| Averaging window length | 250 |
| Training stopped by | EpisodeCount |
| Training stopped at | 2000 |

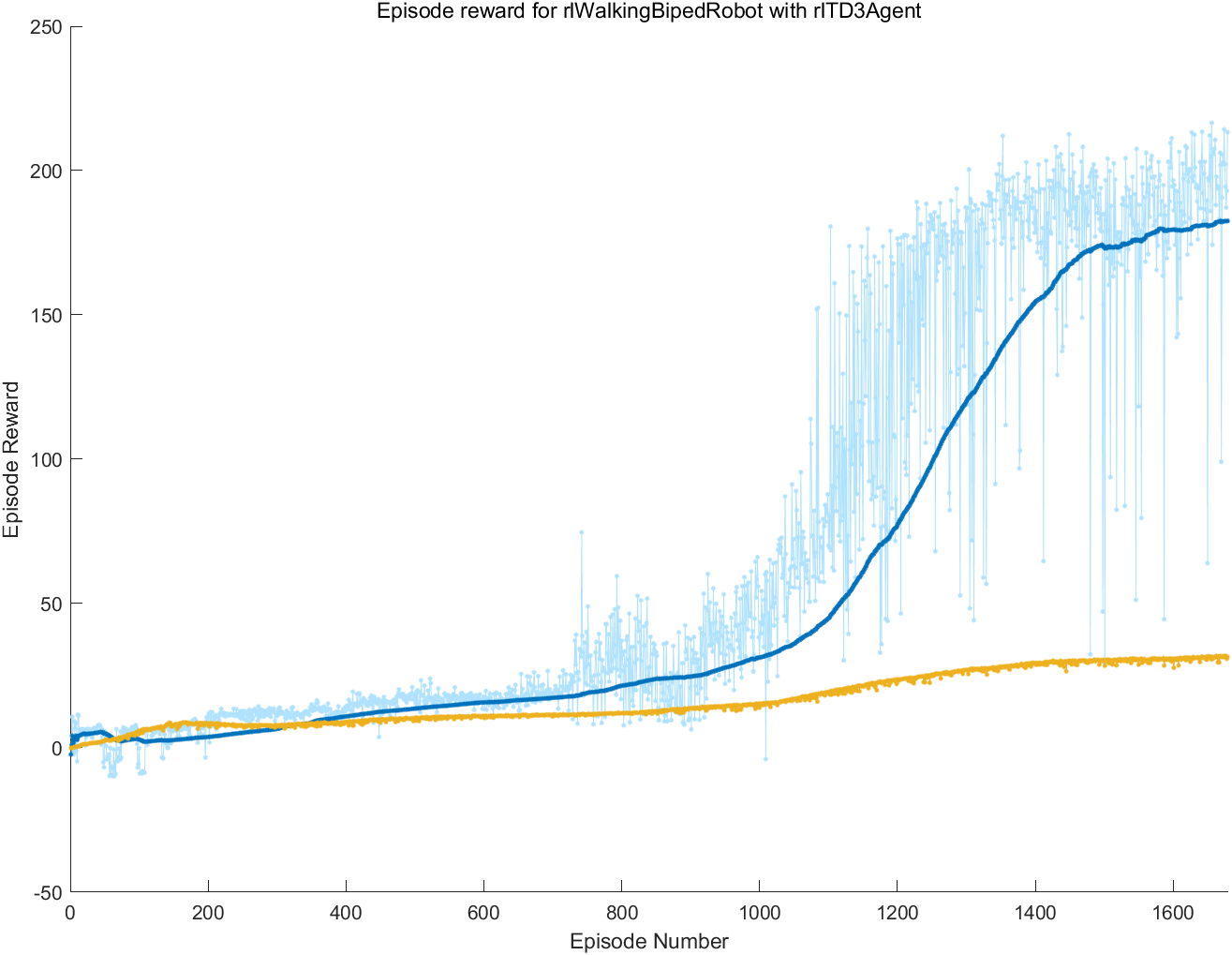


图2.1 向前行走Reward

1. 向后直线行走

训练的Reward以及结果如下。您可以在video/walkingBackward.mp4中查看训练效果。

表二 向后直线行走训练结果

|  |  |
| --- | --- |
|  | RlTD3Agent |
| Status | Training finished |
| Episode number | 4000 |
| Episode reward | 209.9923 |
| Episode steps | 400 |
| Total agent steps | 331937 |
| Average reward | 166.9581 |
| Average steps | 381.096 |
| Episode Q0 | 30.0144 |
| Averaging window length | 250 |
| Training stopped by | EpisodeCount |
| Training stopped at | 4000 |

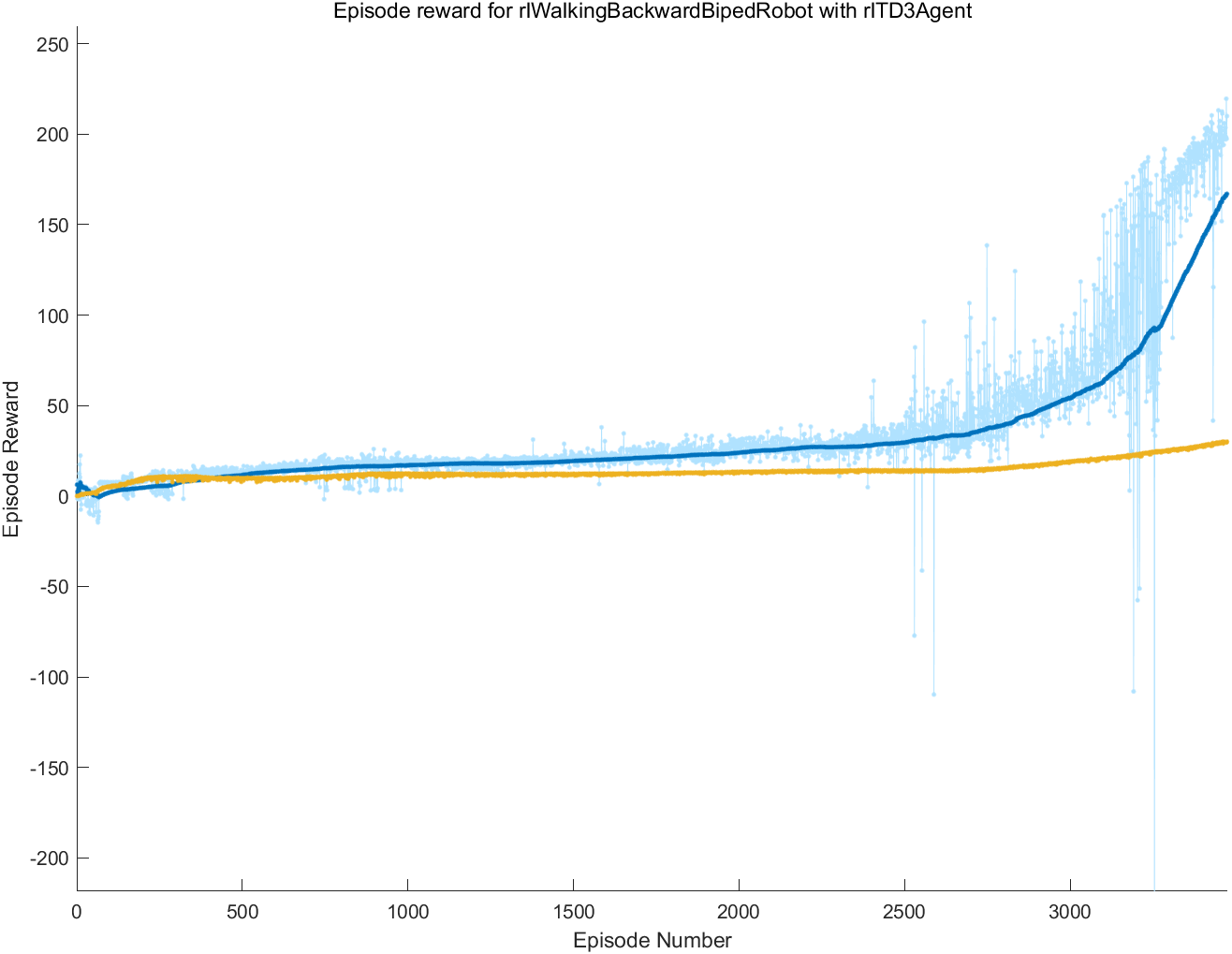


图2.2 向后直线行走Reward

1. 组员分工

Reference

[1] <https://ww2.mathworks.cn/help/reinforcement-learning/ug/train-agents-using-parallel-computing-and-gpu.html>