# 中国人民大学附属中学

# 2019-2020 学年度高二研究性学习课题开题报告

课题编号：

课题名称：强化学习实现简易自动驾驶

课题成员：

指导教师：

# 课题的目的及意义

目的：实现赛车游戏的自动化。

意义：这个时代是一个自动化的时代，更多的体力劳动被自动的机器代替。在新时代中，自动化解放的人力可以更多的投入到创新和文化艺术中去。由此可见，自动化的意义无疑非常重大。

自动化的中心在于算法。一个好算法对于自动化有着重大的影响。一个暴力的算法让机器变得蠢笨，而一个优秀的算法让机器变得高效。近年来，人工智能的成就有目共睹。使用人工智能来实现自动化理所当然成为了研究的课题。

实践是检验真理的唯一标准，虽然我没有条件在真是环境中驾驶汽车，但是我可以研究在赛车游戏中的自动化。游戏是一种对真实的虚拟，游戏中的自动化可以推动真实世界里的自动化。

实现赛车游戏的自动化有着重大的意义！赛车游戏本质上是经过了高度简化的虚拟环境，对于算法的优化和验证有着指导性的意义。在研究这个课题的过程中，我可以深入的了解强化学习算法，对算法中可以改进的地方进行改进。有可能发现某些参数对学习速度的影响，积累调参的经验。

# 国内外研究动态

## 2.1 OpenAI Dota2 Rerun

Dota2是一个电子游戏，在人工智能OpenAi Five面前，人类的Dota2冠军队Team OG不堪一击，现在，OpenAI的新人工智能Rerun全面碾压OpenAI Five，胜率达到98%。Rerun中就应用了强化学习技术。

Dota2比围棋更复杂，围棋一局大约要走150步，而Dota2一局比赛要dui 电脑做出超过20000次操作。而且由于战争迷雾的存在，Dota2中，只能看到局部的信息，其他的信息都处于隐藏的状态。

训练使用的是扩展版的近端策略优化（PPO）方法，这也是OpenAI现在默认的强化学习训练方法。这些智能体的目标是最大化未来奖励的指数衰减和。在训练策略的过程中，OpenAI Five没有用到人类游戏数据，而是通过自我博弈。在围棋、象棋等问题上，也应用了类似的方式训练。其中，80%的战斗中对手是使用了最新参数的分身，而20%的对手是老参数的分身。每经过10次迭代之后，新训练出的分身就被标为老前辈。如果当前正训练的AI击败了新秀或者老前辈，系统就会根据学习率更新参数。按照OpenAI CTO此前的说法，击败OG前OpenAI Five已经练习了相当于45000年Dota。AI每天的训练量相当于人类打180年游戏。

从这个应用例子中可以看出，用强化学习玩对战类的游戏，可以让电脑自我对弈，其中对手使用上一次训练的分身。此外，还可以人工写一些脚本和电脑对战，让电脑学习人类游戏的经验。

## 2.2雷神之锤III竞技场

继OpenAI之后，DeepMind也在多智能体强化学习方面秀肌肉：首次在第一人称射击游戏的多人模式中完胜人类，而且没有使用大量训练局数，轻松超过了人类水平。

从实验结果看，他们的AI已经超越了人类：在与由40个人类玩家组成的队伍对战时，纯AI的队伍完胜纯人类的队伍（平均多抢到16面旗），并且有95%的几率战胜AI与人混合组成的队伍。这个AI名叫“为了赢”（For the Win，FTW），只玩了将近45万场游戏，理解了如何有效地与人和其他的机器合作与竞争。

智能体必须从头开始学习如何在不可见（unseen）的环境中观察、行动、合作和竞争，所有这些都来自每场比赛的一个强化信号：他们的团队是否获胜。这是一个具有挑战性的学习问题，它的解决方法基于强化学习的三个一般思路:

* 我们不是训练一个智能体，而是训练一群智能体，它们通过组队玩游戏来学习，提供了多样化的队友和敌手。
* 群体中的每个智能体都学习自己的内部奖励信号，这些信号使得智能体能够产生自己的内部目标，例如夺取一面旗子。双重优化过程可以直接为了获胜优化智能体的内部奖励，并使用内部奖励的强化学习来学习智能体的策略。
* 智能体在两个时间尺度上运行，快速和慢速，这提高了它们使用内存和产生一致动作序列的能力。

从以上材料看出，强化学习的功能十分强大，用其训练出的智能体甚至可以有团队合作的能力。这也提示我们，训练时可以使用多个智能体一起训练，效果可能比只训练一个庞大的智能体效果更好。

**2.3 总结**

这些神级的人工智能是我不可能做出来的，我只是想了解一下最前沿的科技。

# 研究内容与目标

## 学习强化学习基础算法

### 3.1.1 Q Learning

Q Learning需要一张Q表，用来记录每一个状态Si和行为Ai对应的奖励值，记为Q(s, a)。每一种状态时要进行什么动作只需要查Q表就能知道了。那么如何更行Q表呢？

假设现在已知Q(s1, a1) Q(s1, a2)。在s1状态后进行了a1的操作，变成了s2状态。然后在s2状态下尝试a1,a2两种动作，得到奖励值，于是就有了Q(s2, a1), Q(s2, a2)。下面用Q(s2)来更行Q(s1, a1)。设到达s2时获得的奖励值为r，s2状态下进行的所有a得到的奖励值中最大的为maxQ(s2)。把Q(s1, a1)更新为Q(s1, a1) + α \* ( r + y \* maxQ(s2) – Q(s1, a1) )。现实意义为把原价值加上新价值和原价值的差值，α是学习效率，y是衰减值（未来的奖励有多重要）。

### 3.1.2 Sarsa

Sarsa和Q Learning非常相似，都使用了Q表，只有状态转移不一样。Sarsa并不尝试动作，而是直接采取行动，所以Sarsa的状态转移方程为Q(s1, a1) = Q(s1, a1) + α \* ( r + y \* Q(s2, a1) – Q(s1, a1) )。

Q learning倾向于选择最有可能成功的路线，而Sarsa倾向于选择危险最少的路线。Q learning是离线学习，Sarsa是在线学习。

### 3.1.3 Sarsa(λ)

在Sarsa的基础上，Sarsa(λ)记录每回合走过的路线，λ是脚步衰减值，离奖励越近的动作受到更新的力度越大。

Sarsa(λ)是回合更新，Sarsa是单步更新。

## 用以上算法实现简单的游戏

游戏来自Open AI的开源游戏环境gym

### Acrobot-v1

机器人系统包括两个关节和两个链接，其中两个链接之间的关节被致动。 最初，连杆是向下悬挂的，目标是将下部连杆的末端摆动到给定的高度。

### CartPole-v1

杆通过不可操作的接头连接到手推车，手推车沿着无摩擦的轨道移动。 通过向推车施加+1或-1的力来控制系统。 钟摆开始直立，目的是防止其跌落。 杆保持直立的每个时间步长都提供+1的奖励。 当极点与垂直方向的夹角超过15度时，或者推车从中心移出2.4个单位以上时，情节结束。

### Pendulum-v0

倒立摆摆动问题是控制文献中的经典问题。 在此问题的版本中，摆锤开始于随机位置，目标是将其摆动以使其保持直立。

## 用神经网络替代Q表

学习tensorflow，搭建一个人工神经网络，并用其替代基本强化学习算法中的Q表，这么做是因为传统的Q表法不能适应现在的机器学习任务，因为围棋等游戏的状态数太多，Q表存不下来。

## 挑战自动驾驶/登月

### CarRacing-v0

从赛车环境中学习像素。状态由96x96像素组成。 奖励是每帧-0.1，所访问的每个轨道图块的+ 1000 / N，其中N是轨道中图块的总数。 访问所有图块时，情节结束。 窗口底部和状态RGB缓冲区显示一些指示器。 从左到右：真实速度，四个ABS传感器，方向盘位置，陀螺仪。

### LunarLander-v2

如果着陆器离开着陆垫，它将失去奖励。 如果着陆器坠毁或静止下来，则情节结束，获得了额外的-100或+100点。 每条腿的接地为+10。 射击主机每帧为-0.3分。 解决的是200分。 可以在着陆垫外面着陆。 燃料是无限的，因此特工可以学会飞行然后首次尝试着陆。 共有四个离散操作：不执行任何操作，左向发动机点火，主发动机点火，向右发动机点火。

# 研究方案

## 学习基础强化学习算法

预计用时两周

## 学习tensorflow

预计用时两周

## 实现二维自动驾驶

预计用时两周

# 可行性分析

## 编程硬件设施

现有一台联想笔记本。一个显示器。一个键盘。一个鼠标。

缺少训练用的显卡，准备用cpu凑合。

## 5.2 软件环境

已安装Linux系统，python，gym。

## 学习资料

可以上网查找，已知有莫烦的python教程、csdn博客。

## 我坚定的意志和高尚的节操，伟大的爱国精神再加上纯洁的心灵

## 预期成果

在模拟环境中，程序根据图像识别出汽车的位置等状态，并作出对汽车的控制。让汽车行驶在路上，不走出路线区域外，并且尽可能快的走完赛道。

# 参考文献

百度百科、csdn、知网、必应学术、谷歌翻译、搜狐新闻