工作总结：

细读论文《nips-2017-maddpg》，maddpg算法以Actor-critic为框架，结合的PG算法（处理连续动作，变异性小收敛稳点的特点）和DQN算法（使用经验回放提高sample efficiency， 使用target network提高估计准确性）的优点。Critic 收集所有agent的观测和动作，评估当前agent的策略好坏，actor分布执行每个agent的动作。使maddpg可以在，competition， cooperation，mix的任意环境下使用。

但算法critic集中式训练收集所有agent的观测，算法复杂度随agent的数量线性增加。而在环境中并不是每个agent都具有强相关性的。所以如何区分agent的相关程度并提高算法的性能，是下一步的改进方向。

查询文献

[《AAMAS-2018, VDN: Value-Decomposition Networks For Cooperative Multi-Agent Learning》](https://arxiv.org/pdf/1706.05296)

[《ICML-2018, QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning》](https://arxiv.org/pdf/1803.11485)

[《ICML-2019, QTRAN: Learning to Factorize with Transformation for Cooperative Multi-Agent Reinforcement learning》](https://arxiv.org/pdf/1905.05408)

这三篇文章对此有改进，我将泛读并按重要程度依次阅读分析。

《nips-2017-maddpg》基于openai-gym，开源了multi-agent环境MPE，在MPE实现论文中一个listener-speaker的实例，下面会借用这个框架，改写我们的多车辆协同控制的应用场景。