Massively Multiagent Inverse Reinforcement Learning in Open-world Environments

William H. Guss*
wguss@cs.cmu.edu

Ruslan Salakhutdinov*
rsalakhu@cs.cmu.edu

January 17, 2018

1   Introduction

Over recent years inverse reinforcement learning (IRL) has revolutionized the state of the art in apprenticeship learning, cooperative and adversarial modeling, and the modeling of intent in human and animal behaviour**TODO: cite**. At its core, IRL solves the problem of learning expert policies indirectly: in direct juxtaposition to behavioral cloning IRL learns the reward function of an expert agent and then produces a policy which maximizes that reward. In this regime, the resultant policy is often far more interpratable, robust, and sample efficient than that of behavioural cloning.

Mechanically, inverse reinforcement learning is optimal not only for cloning expert policies in applications where thousands if not millions of demonstrations are not possible, but also as a substitute for traditional reinforcement learning when exploration is extremely expensive. For example, in robotic manipulation tasks, where typical ϵ-greedy exploration polices would result in potential damage to the robot, apprenticeship learning via IRL is a powerful alternative. Furthermore, by learning the reward function directly, IRL is an effective, interpritable forecasting mechanism in tasks such as epidimiological modeling, traffic prediciton, and first person activity forecasting.**TODO: cite**
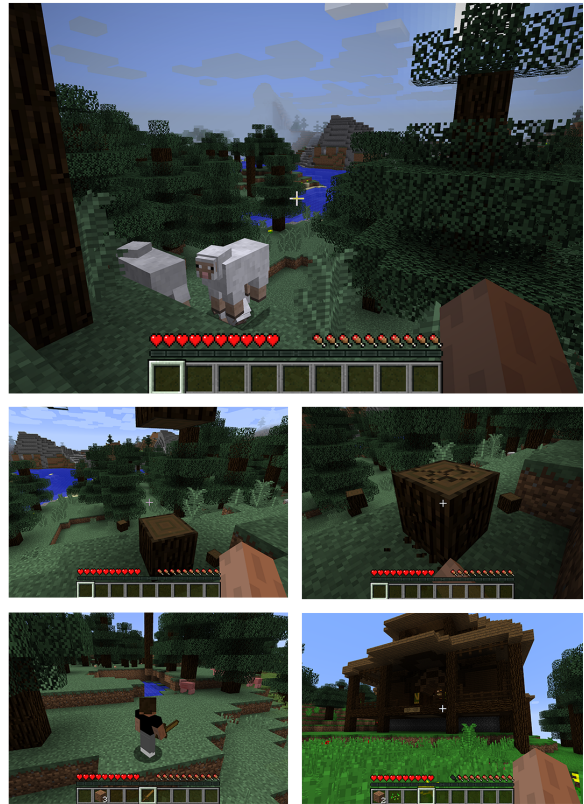

Figure 1: An example of different states in Minecraft, such as resource gathering, environment discruction, combat, building, and mining.

---

*Carnegie Mellon University, Machine Learning Department.

Despite its numerous applications and avantages, IRL is an underconstrained optimzation problem; in particular, there are potentially inifinitely many reward functions which explain an expert policies behaviour. To see this formally, let (S, A, T, γ, D) be a rewardless Markov deciion process (MDP) with state space S, action space A, state-to-state transition function T, γ some marginal utility discount, and D an initial state distribution. Given some expert policy $\pi^*(a|s)$, where $a \in A, s \in S$, inverse reinforcement learning aims to find a reward function $R : S \times A \to \mathbb{R}$ such that

$$\pi^* = \arg\max_{\pi} \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t)\right] \tag{1}$$

where $s_t \sim T(s_{t-1}, a_{t-1}), a_t \sim \pi(\cdot|s_t)$, and $s_0 \sim D$. In this setup, it is clear that degenerate reward functions such as $R = 0$ suffice in explainig π^*. Constraining the space of candidate reward functions has therefore become central problem in inverse reinforcement learning.

TODO: Previous research on reward constraints, plus hook.

2 Our Approach

In this project we propose a novel approach the problem of reward space regularization by leveraging the statistical power of large scale massively multiexpert policy trajectories in open-world environments. In particular, we will develop new algorithms which allow inverse reinforcement learning to utilize policy rollouts from multiple distinct experts whose reward functions are assumed to be samples from some continuously parameterized manifold of reward functions. Thereafter, we will develop a cloud-based massively multiplayer environment to record a sufficiently rich dataset of expert trajectories in Minecraft, an open-world game with inductively ambiguous reward structures. Finally, we will apply the foregoing methods to train deep reinforcement learning agents using different player reward functions sampled from the learned reward space. In addition to being introduced to the envrionment, these agents will serve as a benchmark for the efficacy of the inverse reinforcement learning procedure evaluated by several metrics.

Multi-reward Inverse Reinforcement Learning. In contrast to existing methods, our proposed approach, multi-reward inverse reinforcement learning (MIRL), exploits the following fact: although in multiagent open-world environments, each individual agent is governed by its own reward function, between any two agents their respective rewards are similar; that is, there is a semi-continuous parameterization of the space of reward functions over some reasonable population of agents in an open-world environment. This assumption leads to a powerful regularization scheme wherein degeneracy of a particular agents reward function is penalized as that agent's reward must be ϵ-sufficient for explaining the actions of another similar but distinct agent.

Open-world environment. To collect sufficiently realistic and open-world samples for MIRL we will develop a custom extension to the multiplayer version of Minecraft. Minecraft, see Figure 1, is a survival-based sandbox game wherein players are given full agency to build, destroy, and explore their environment. In addition to the huge state space, in an online setting players can fight or collaborate in constructing various buildings, cities, and communities. Generally speaking there is no particular goal, score, or objective[1] which governs players actions; that is, each player defines

---

[1]With the exception of experience, which will be used as a metric for the evaluation of MIRL.

their own reward structure and the game is as seemingly open-world as the real world.

Over the duration of the project, we will host a minecraft server which records every players screen and actions and collates this data into a coherent data-store. At scale we anticipate the server will host more than 1000 unique players generating 30 frames per second each. The resulting dataset of player trajectories will be the largest of its kind. As aforementioned, we will also develop tooling for the introduction of learned agents into the environment with which players will continually interact.

2.1 Experiments

The proposed project will consist of multiple experiments for testing the efficacy of the developed algorithms. In particular we will

2.2 Metrics