Intelligent Racecars:

Comparing the effectiveness of Q Learning and Value Iteration in a race track setting

Derek Reimanis

December 5, 2012

**Abstract**

This paper analyzes the use of two algorithms, Q Learning and Value Iteration, in a race track setting. A race car is initialized with one of these two algorithms and told to reach the finish line. Each movement has, in addition to a negative cost, only a chance of being executed successfully. The racer is tested across a multitude of tracks with different shapes, different learning constants, and various punishments for not staying on track. The time in which the racer took to finish is recorded, as well as number of movements, consistency of path, and how close to optimality the path chosen was.

**1 Introduction**

Artificial Intelligence, that is, an ‘intelligent’ problem solver, generally revolves around the idea of a model in which an agent and an environment exist. The agent can perceive the environment using its percepts. An agent can also take actions that alter the environment using its actuators. In combination, an agent perceives the environment and returns an action that alters the environment. In what is referred to as a reflex-based model, an agent will apply the current state of the environment to a set of internal rules, thus modeling a virtual environment. This is done without taking any actions upon the actual environment, allowing the agent to effectively ‘think’ about future environment states. An agent can now plan ahead, using iterations of a virtual environment state to eventually reach a goal state. This is referred to as a goal-based agent; that is, an agent that applies the rules of the environment to a virtual copy of the environment to eventually reach a desired state (Stuart Russell, 2010).

In the process of having an agent learn the best path a goal state, there are a few different methods to use. The first method typically presented is that of ‘supervised learning’. Supervised learning means that the correct goal state is known, and the agent merely checks to see if it has made progress towards that goal state after an action. The defining point in supervised learning is that the goal state is known a priori, and the agent tries to optimize the correct solution. A second method presented in agent learning is referred to as ‘un-supervised learning’. As the name would suggest, un-supervised learning is the process of moving through a problem, yet the agent knows of no solution to this problem. The learning agent therefore has no method of determining if it is in fact progressing, or even regressing, away from the solution. A final learning method an agent can utilize is that of reinforcement learning. Reinforcement learning acts as a middle ground between supervised and unsupervised learning; that is, reinforcement learning ranks the agent’s actions based on some utility, such that the agent can progress through a problem. Now, there may or may not be an answer to this problem, yet the agent doesn’t care. The reinforcement learning agent is only concerned with maximizing a utility, which will hopefully end up in a solution state.

Reinforcement learning is a practical field in Artificial Intelligence. Reinforcement learning entails the process of having an agent worl

These two algorithms employ different strategies to reach a nearly-optimal path. The model-based Value Iteration algorithm focuses on a greedy approach to rank the value of certain states over others, while needing a model of the environment to perform in. The model-free Q Learning algorithm doesn’t have knowledge of the environment’s model, so it must find a near-optimal path through the use of exploration and trial-and-error methods.