TJHSST Computer Systems Lab Senior Research Project
Reinforcement Learning in Connect 4
2007-2008

Michael Yura

November 2, 2007

Abstract

Although an AI is often thought of as being only as intelligent as its programmer, this is not exactly the case; this project will attempt to create an dynamically learning Machine Learner for Connect 4 by using reinforcement learning, as well as to determine what degree of reinforcement allows for the Machine Learner to learn to play the best in the shortest amount of time.

Keywords: Multiagent, dynamic simulation, group navigation, herds, swarms

1 Introduction

I will create a dynamically learning ML (Machine Learner) for Connect 4. These ML's will learn through reinforcement learning. I they are successful (and win the game), they will do what they did more often; conversely, if they are not successful (and lose the game), they will do what they did less often.

I expect to have an ML that throughly and hopefully quickly learns to play Connect 4 to an advanced level. Through this project, I hope to learn how fast and to what quality reinforcement learning allows for the learning

of a simple game; these methods can hopefully be extended to other, more complex tasks for machines to learn.

To illustrate the use of dynamically simulated characters, we created a group of simulated human bicyclists and a group of alien bicyclists that ride on a bicycle race course (figure 2). Our earlier results indicate 1,2 that we can generate algorithms that support characters of different types and groups of varying size, however, manual tuning was required to obtain good performance. In this paper we describe automatic tuning methods and algorithms that generate improved group performance.

2 Background

Connect 4 has already been solved by James D. Allen and Victor Allis; I will attempt to compare the way the ML plays to the strategies outlined in Allis's A Knowledge-based Approach of Connect-Four

3 Procedures and Methodology

I have currently programmed the connect 4 game itself, as well as created an ML (Machine Learner) abstract class that other ML's will be based upon, with methods to load save its board data. I currently have an ML that does not change the way it places pieces, playing completely randomly. I will create ML's that will change the way they play to different degrees some radically changing their strategies after each game, and others doing so to a more moderate degree.

ML's currently store their board data in a single text file. I may attempt to find a more efficient method of storing this data, as the reading and writing of this data may take a long time as the board data of many games begins to accumulate.

I will keep track of when different ML's win or lose games against one another and save this data. I will analyze this data to detect any trends. I may try to play against a few of the ML's myself.

ML's will train against one another, so that they can rapidly change their piece placement with little human effort. I will test to see if an ML can learn by playing against itself, learning by both winning and losing at the same time.

4 Expected Results

Through this project, I hope to find a degree of reinforcement learning that allows the computer to learn to play connect 4 quickly and throughly. I hope that this project may add to the creation process of AI's.