

Introduction to Multi-Agent-Programming

B. Nebel, A. Kleiner
C. Dornhege, D. Zhang
Winter Semester 2009/2010

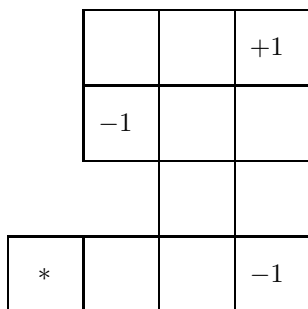
University of Freiburg
Department of Computer Science

Exercise Sheet 12

Due: February 15th, 2009

Exercise 12.1 (Q-Learning (1pt, written))

Consider the following grid world, where the numbers are rewards associated with the cells. An agent starts at the left-bottom corner. It can perform four possible actions: North, South, East and West. With probability 0.6 the agent reaches the intended state, with probability 0.2 it moves to the right of the intended direction, and 0.2 to left. Compute the optimal policy using Q-Learning.



Exercise 12.2 (Reinforcement Learning)

(a) Problem Formulation (1pt, written)

Consider a game of writing a string with nine chars, each char can have three possible values "X", "O", or "-". Each time, an agent can choose a place to put "X" in the string. Suppose the putting action is stochastic, and there is an oracle, which tells the agent the resulted string after each round, and three possible awards +1, -1, or 0 after some rounds (when the game is terminated). Formulate the description to a Q-Learning problem in a formal form.

(b) Implementation (2pt, programming)

Implement the formulated learning algorithm in C++. The implementation should be based on the framework available at `grpX@augusta:/home/dapeng/public/stringgame.tar.gz`.

(c) The learning curve (2pt, programming)

Generate a data file which contains the learning curve of last exercise. Write a script to output the data to a jpg file. We suggest you to use

gnuplot.

A tutorial: <http://www.duke.edu/~hpgavin/gnuplot.html>

**Please put the programming part in your group directory in augusta
The written part should be submitted via e-mails or at the consulta-
tion time on Feb. 15th)**