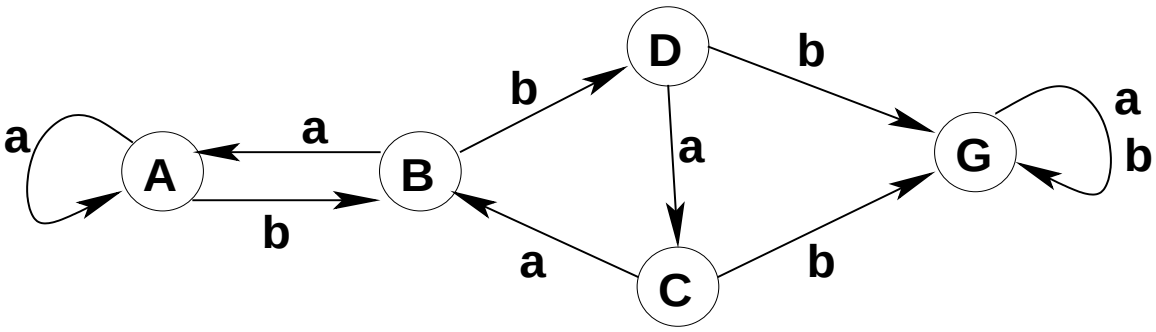
COS 402: Artificial Intelligence

Homework #6 — Fall 2004
Cat and mouse — Due: Wednesday, December 8

Part I: Written Exercises

See instructions on the assignments page on how and when to turn these in. Approximate point values are given in parentheses. Be sure to show your work and justify all of your answers.

1. (15) Consider the following MDP:



There are five states: A, B, C, D and G. The reward at every state is -1, except at G where the reward is 0. There are two actions, a and b, and the effect of each action is deterministic as indicated in the figure. For instance, executing a in state B leads to state A. Assume $\gamma = 1$ in this problem.

a. Show the sequence of utility estimates U_i that would result from executing value iteration on this MDP. Also show the optimal policy that is computed using the final utility estimate.

b. Show the sequence of policies π_i and utility estimates U^{π_i} that would result from executing policy iteration on this MDP. Assume that you start with a policy that assigns action a to every state. Note that U^{π_i} will be infinite for some states. Also, assume that all ties between the actions a and b in the policy improvement step are always broken in favor of a.

c. Generalizing this example, suppose we are given a graph with a distinguished node (i.e., state) G, and k edges emanating from every node corresponding to k (deterministic) actions. As in this example, all of the edges emanating from G are self-loops, the node G is assigned reward 0, and all other nodes are assigned reward -1. In terms of properties of the graph, what is the optimal utility function U^*, and what is the optimal policy π^*? If value iteration is applied to this graph (viewed as an MDP), exactly how many iterations will be needed until the algorithm converges? How about for policy iteration?

2. (10) Sometimes MDP's are formulated with a reward function $R(s, a)$ that depends on the action taken, or a reward function $R(s, a, s')$ that also depends on the outcome state. For each of these formulations, show how to appropriately modify each of the following:

- the Bellman equation (Eq. (17.5) in R&N);
- the formula for converting the optimal utility U^* (denoted simply U in R&N) into an optimal policy π^* (Eq. (17.4) in R&N);
- the value iteration algorithm;
- the policy iteration algorithm.

3. (15) Let $B(U)$ and $\|\cdot\|_\infty$ be as defined in class. (This is the same as BU and $\|\cdot\|$ defined in Section 17.2 of R&N.) The purpose of this exercise is to prove that B is a *contraction*, i.e., that $\|B(U) - B(U')\|_\infty \leq \gamma \|U - U'\|_\infty$. We will begin by proving some basic facts. Be sure to give genuine mathematical proofs for each part of this problem. Also, your proofs should use elementary facts — in other words, do not give proofs that rely on mathematical sledge-hammers like the Cauchy-Schwartz inequality.

a. Let $u_1, \ldots, u_n$ and $v_1, \ldots, v_n$ be any sequences of real numbers. Prove that if $u_i \leq v_i$ for all i then

$$\max_i u_i \leq \max_i v_i.$$

b. Let $x_1, \ldots, x_n$ and $y_1, \ldots, y_n$ be any sequences of real numbers. Prove that

$$\left(\max_i x_i\right) - \left(\max_i y_i\right) \leq \max_i (x_i - y_i),$$

and also that

$$\max_i (x_i - y_i) \leq \max_i |x_i - y_i|.$$

(Hint: both of these inequalities can be proved using part (a) for an appropriate choice of u_i and v_i.)

Finally, use these facts to prove that

$$\left|\left(\max_i x_i\right) - \left(\max_i y_i\right)\right| \leq \max_i |x_i - y_i|.$$

c. Let $x_1, \ldots, x_n$ be any real numbers, and suppose that $p_1, \ldots, p_n$ are nonnegative real numbers such that $\sum_i p_i = 1$. Use the fact that $|a + b| \leq |a| + |b|$ for any real numbers a and b to prove that

$$\left|\sum_i p_i x_i\right| \leq \max_i |x_i|.$$

d. Now let s be any state, and let $(B(U))(s)$ denote the value of $B(U)$ at state s. By plugging in the definition of B, and using the properties proved above, prove that

$$|(B(U))(s) - (B(U'))(s)| \leq \gamma \|U - U'\|_\infty.$$

Conclude that

$$\|B(U) - B(U')\|_\infty \leq \gamma \|U - U'\|_\infty.$$

4. (15) Suppose we start flipping a fair coin. What is the expected number of coin flips until we get three heads in a row? In this problem, we will use a Markov process formulation to solve this problem. The Markov process has six states: Four of the states are HH, HT, TH and TT corresponding to the last two times that the coin was tossed. Thus, if we last flipped tails, and the time before that we flipped heads, then we must be in state HT. There is also a "dead" state called D which we reach upon flipping three heads in a row for the first time. Once we reach D, we never exit from it. Finally, the starting state is called S. Thus, we begin in S. After flipping the coin twice, we traverse to one of the four states HH, HT, TH or TT, corresponding to the outcome of the two flips. Each time the coin is flipped, we traverse to the appropriate state. For instance, if we are in state HT, and the coin comes up tails, then we traverse to TT. Finally, when the coin comes up heads three times in a row, we immediately traverse to the dead state D.

a. Formulate this problem as a Markov process (i.e., an MDP in which there is no choice of action at each time step) in such a way that the "utility" of every state s is equal to the expected number of coin flips until three heads come up for a random sequence of coin flips beginning in state s. What is the probability of transitioning from every state to every other state? What is the "reward function" at each state?

b. Write down Bellman-style equations for the utility at each state s in terms of the utilities of the states that can be reached from s in one step.

c. Solve the system of equations in part b.

d. Now answer the original question: What is the expected number of coin flips until we get three heads in a row?

Part II: Programming

The programming part of this assignment is described on the course website at:
`http://www.cs.princeton.edu/courses/archive/fall04/cos402/assignments/mdp`