CIS 620 — Advanced Topics in AI
Profs. M. Kearns and L. Saul
Problem Set 1
Distributed: Wednesday, January 9, 2002
Due: Wednesday, January 23, 2002 (start of class)

1. *Effective horizon for discounted return.* Let $0 \leq \gamma < 1$, and let $\sum_{i=0}^{\infty} \gamma^i r_i$ be an infinite sum (which we may regard as the discounted return in an MDP) with all $r_i \in [0,1]$. Let $0 < \epsilon < 1$. Prove that, for some constant $c_0 > 0$,

$$t \geq \frac{c_0}{\log(1/\gamma)} \log \frac{1}{(1-\gamma)\epsilon}$$

implies $\sum_{i=t}^{\infty} \gamma^i r_i \leq \epsilon$. Thus, for any chosen amount of tolerated approximation (ϵ), we can view infinite-horizon discounted return as similar to finite horizon return, where the length of this finite horizon grows as $\gamma \to 1$. Note that as $\gamma \to 1$, $\log(1/\gamma)$ behaves like $1/(1-\gamma)$.

2. *Approximation to optimal value function yields near-optimal policy.* Let V^* be the value function for the optimal policy π^* in an MDP, and let $\hat{V}$ be an approximation to V^* (as might be computed, for instance, via the value iteration algorithm). Let $\hat{\pi} = \mathit{greedy}(\hat{V})$. Recall that this means

$$\hat{\pi}(s) = \mathrm{argmax}_a \left\{ R(s,a) + \gamma \sum_{s'} P(s'|s,a)\hat{V}(s') \right\}$$

for every state s, where γ is the discount factor. (Note that $V^{\hat{\pi}} \neq \hat{V}$ in general.) Define the *regret* $\hat{L}(s)$ of $\hat{\pi}$ from s as

$$\hat{L}(s) = V^*(s) - V^{\hat{\pi}}(s).$$

Show that if $|V^*(s) - \hat{V}(s)| \leq \epsilon$ for every s, then $\max_s\{\hat{L}(s)\} \leq 2\gamma\epsilon/(1-\gamma)$. Thus, following the greedy policy determined by a good approximation to the optimal value function is, in fact, a near-optimal policy. You may find it helpful to break the proof into the following two steps (though you are free to use any proof you like):

- Let $a = \pi^*(s)$ and $b = \hat{\pi}(s)$. First use the assumed approximation bound on $\hat{V}$ and the greediness of $\hat{\pi}$ to give a bound on $R(s,a) - R(s,b)$.
- Substitute your bound on $R(s,a) - R(s,b)$ into a one-step expansion of $\hat{L}(s)$.

3. *Computation of optimal policy via linear programming.* A *linear program* is a maximization (or minimization) problem with the following special form: maximize the linear function $\vec{w} \cdot \vec{x}$, subject to the linear inequalities $A\vec{x} \geq \vec{b}$. Here $\vec{w}, \vec{b} \in \Re^n$ are given vectors, A is a given n by n matrix of reals, $\cdot$ denotes inner product, and the problem is to compute $\vec{x} \in \Re^n$ accomplishing the stated maximization. Show that the problem of computing the optimal policy in a given MDP can be formulated as a linear program. Thus, standard linear programming algorithms (such as the simplex algorithm, whose worst-case running time may be exponential in n, or Karmarkar's algorithm, whose running time is polynomial) can be used to compute (exactly) optimal policies.

4. *Policy iteration improves policies.* Recall that policy iteration maintains a policy $\hat{\pi}_t$, and for each state s, sets

$$\hat{\pi}_{t+1}(s) \leftarrow \mathrm{argmax}_a\{Q^{\hat{\pi}_t}(s,a)\} = \mathrm{argmax}_a \left\{ R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\hat{\pi}_t}(s) \right\}$$

where the computation of $V^{\hat{\pi}}$ can be accomplished via the solution of a system of linear equations, and no change is made to $\hat{\pi}_t(s)$ if the argmax_a is already achieved. Prove that if the policy $\hat{\pi}_{t+1}$ is different than $\hat{\pi}_t$, it is strictly better than $\hat{\pi}_t$ — that is, $V^{\hat{\pi}_{t+1}}(s) \geq V^{\hat{\pi}_t}(s)$ for all s, with strict inequality for at least one state. (Hint: consider only the change at a single state, and look at the time-dependent policy that makes the suggested change on the first i steps of a random walk under $\hat{\pi}_t$, but not afterwards. Show that $i+1$ is better than i.)

5. *Relating value iteration and policy iteration.* For any natural number $k \geq 1$, define the algorithm *rollout*(k) as follows. Like value iteration, *rollout*(k) will proceed in rounds, and maintain a current policy $\hat{\pi}_t$ and value function $\hat{V}_t$ at round t. The update equations are

$$\hat{V}_{t+1}(s) \leftarrow \left(\sum_{i=0}^{k-1} \gamma^i \sum_{s'} P(s'|s,\hat{\pi}_t,i) R(s',\hat{\pi}_t(s')) \right) + \gamma^k \sum_{s'} P(s'|s,\hat{\pi}_t,k) \hat{V}_t(s')$$

and $\hat{\pi}_{t+1} = \mathit{greedy}(\hat{V}_{t+1})$. Here $P(\cdot|s,\hat{\pi}_t,i)$ is the distribution induced over states by taking an i-step walk under $\hat{\pi}_t$ starting from s. Prove that value iteration is equivalent to *rollout*(1) and that policy iteration is equivalent to *rollout*(∞). Based on this observation, conjecture which algorithm is better, and give your reasons. (Extra credit: prove your conjecture.)