Homework 2 - Probabilistic aspects of computer science

1 The maximal expected reward

Let $X_i$ denote the random state at time $i$ and $Y_i$ denote the random action at time $i$ of an MDP. Given a policy $\pi$, the maximal expected reward at time horizon $t$ of $\pi$ is defined by:

$$M^\pi_t \overset{\text{def}}{=} \mathbb{E}^{\pi}(\max(r(X_i, Y_i) \mid 0 \leq i < t))$$

The corresponding vectorial reward (which depends on the initial state) is denoted $M^\pi_t$. As usual, the optimal vectorial reward $M^*_t$ is defined by: for all $s \in S$, $M^*_t[s] \overset{\text{def}}{=} \sup_{\pi}(M^\pi_t[s])$.

**Question 1.** Show an example of MDP such that no Markovian policy is optimal for the (vectorial) maximal expected reward at time horizon 3.

**Question 2.** Let $M$ be an MDP and $t$ be an horizon. Propose an algorithm that finds the optimal reward and an optimal policy for the maximal expected reward problem in polynomial time w.r.t. the size of $M$ and in pseudo-polynomial time w.r.t. $t$.

*Hint: The algorithm builds an MDP $M'$ such that from the optimal reward and an optimal policy for the pure total expected reward in $M'$, one can recover the optimal reward and an optimal policy for the maximal expected reward in $M$.*

2 Terminal components of a MDP

Let $M$ be an MDP, we introduce the notion of a subMDP. A subMDP $M'$ of $M$ is a non empty set of pairs state-action such that $(s, a) \in M'$ implies that $s \in S$ and $a \in A_s$. The underlying graph of $M'$, $G_{M'} = (S', E')$ is defined by:

1. $S' \overset{\text{def}}{=} \{s \in S \mid \exists(s, a) \in M'\}$;
2. $E' \overset{\text{def}}{=} \{(s, s') \in (S')^2 \mid \exists(s, a) \in M' \text{ with } p(s'|s, a) > 0\}$.

A subMDP $M'$ is a terminal component of $M$ if:

1. For all $s, s' \in S$, $a \in A_s$, $(s, a) \in M'$ and $p(s'|s, a) > 0$ implies $s' \in S'$;
2. $G_{M'}$ is strongly connected.

$M'$, a terminal component of $M$, is maximal if there is no terminal component $M''$ with $S' \subseteq S''$, $E' \subseteq E''$ and $S' \cup E' \subseteq S'' \cup E''$.
We have drawn above $G_M$ the underlying graph of a MDP $M$ where an action $a$ labels an edge $(s, s')$ if $p(s' | s, a) > 0$.

**Question 3.** Let $M$ be the MDP whose graph is drawn above. Find a maximal terminal component of $M$ and a non maximal terminal component of $M$.

Let $\rho = s_0, a_0, s_1, a_1, \ldots$ be an infinite path. Define $\omega(\rho) \overset{\text{def}}{=} \{(s, a) \mid \forall i \in \mathbb{N} \exists j \geq i (s_j, a_j) = (s, a)\}$, the set of pairs state-action infinitely occurring in $\rho$.

**Question 4.** Let $\pi$ be a policy and $\rho = X_0, Y_0, X_1, Y_1, \ldots$ the random path of an MDP. Prove that:

$$\Pr^\pi(\omega(\rho) \text{ is a terminal component}) = 1$$

**Algorithm 1:** Computing the maximal terminal components

```plaintext
MaxTerminalComponents(M)

Input: M, an MDP
Output: SM, the set of maximal terminal components of M

Data: $i$ integer, $s, s'$ states, $a$ action, sub, sub' subMDP, stack, a stack of subMDP

sub $\leftarrow \{(s, a) \mid s \in S, a \in A\}$; Push(stack, sub); SM $\leftarrow \emptyset$

while not Empty(stack) do
  sub $\leftarrow$ Pop(stack); $S' \leftarrow \{s \mid \exists (s, a) \in \text{sub}\}$
  for $(s, a) \in \text{sub}$ do
    for $s' \in S$ do
      if $p(s' | s, a) > 0$ and $s' \notin S'$ then sub $\leftarrow$ sub \ {$(s, a)$}
    end
  end
  if sub $\neq \emptyset$ then
    Compute the strongly connected components of $G_{\text{sub}}, S_1, \ldots, S_K$
    if $K > 1$ then
      for $i$ from 1 to $K$ do
        sub' $\leftarrow \{(s, a) \in \text{sub} \mid s \in S_i\}$; Push(stack, sub')
      end
    else SM $\leftarrow$ SM $\cup$ sub
  end
end
return SM
```

**Question 5.** Prove that algorithm $\square$ returns the set of maximal terminal components.

**Question 6.** Analyse the (worst-case) complexity of algorithm $\square$ w.r.t. $|S|$ and $|A|$.

3 Minimising the reachability cost

Let $M$ be an MDP with non negative rewards and an absorbing state $s_e$: $A_{s_e}$ is a singleton whose Dirac distribution leads to $s_e$ and whose reward is null. We assume that there exist policies that ensure to reach $s_e$ with probability 1 and such policies are called winning policies. In this case, there exists a stationary deterministic winning policy.

The reachability cost of a policy $\pi$ (which may be infinite) is defined by:

$$R^\pi \overset{\text{def}}{=} \sum_{i \in \mathbb{N}} \mathbb{E}^\pi(r(X_i, Y_i))$$

The corresponding vectorial cost (which depends on the initial state) is denoted $R^\pi$. The optimal vectorial cost $R^*$ is defined by: for all $s \in S, R^*[s] \overset{\text{def}}{=} \inf_\pi \{R^\pi[s] \mid \pi \text{ is winning}\}$. The reachability cost problem consists to find the minimal reachability cost $R^*$ and an optimal winning policy.
Question 7. Using the MDP figured below (with only Dirac distributions) show that a non
winning strategy can have a smaller reachability cost than any winning strategy.

In the sequel, we assume that for all non winning policy \( \pi \) there exists \( s \in S \) such that: \( R^\pi[s] = \infty \).

Let the operator \( L \) on \( \text{Rew} \) defined by:
\[
\forall s \in S \quad \min_{a \in A_s} \left( r(s, a) + \sum_{s' \in S} p(s'|s, a)v[s'] \right)
\]

Question 8. Let \( v \in \text{Rew} \) be a fixpoint of \( L \). Prove that \( v \leq R^* \).

Question 9. Let \( \mathcal{d}^\infty \) be a stationary policy. Show that \( R^\mathcal{d}^\infty = \sum_{i \in \mathbb{N}} (P_d)^i r_d \) (using the notations of the lecture notes).

Given \( d \) a decision rule, the operator \( L_d \) on \( \text{Rew} \) is defined by:
\[
L_d(v) = r_d + P_d v
\]

Question 11. Let \( \mathcal{d}^\infty \) be a winning policy. Show that \( R^\mathcal{d}^\infty \) is a fixpoint of \( L_d \).

Question 12. Let \( \mathcal{d} \) be a decision rule such that there exists \( v \in \text{Rew} \) with \( L_d(v) \leq v \).

Show that \( \mathcal{d}^\infty \) is a winning policy. Hint: use the assumption about non winning policies.

Question 13. Let \( \mathcal{d}^\infty \) be a deterministic stationary winning policy such that \( L(R^\mathcal{d}^\infty) \leq R^\mathcal{d}^\infty \).

Let \( d' \) be a deterministic decision rule such that \( L(R^\mathcal{d}^\infty) = L_d'(R^\mathcal{d}^\infty) \).

Show that \( R^{d'}^\infty \leq R^{d^\infty} \).

Question 14. Deduce from the previous questions that there exists a winning deterministic station-
ary policy \( \mathcal{d}^\infty \) such that \( L(R^\mathcal{d}^\infty) = R^\mathcal{d}^\infty \) and that \( \mathcal{d}^\infty \) is an optimal policy for the reachability
cost problem.

Question 15. Design a linear programming problem such that its solution is \( R^* \).