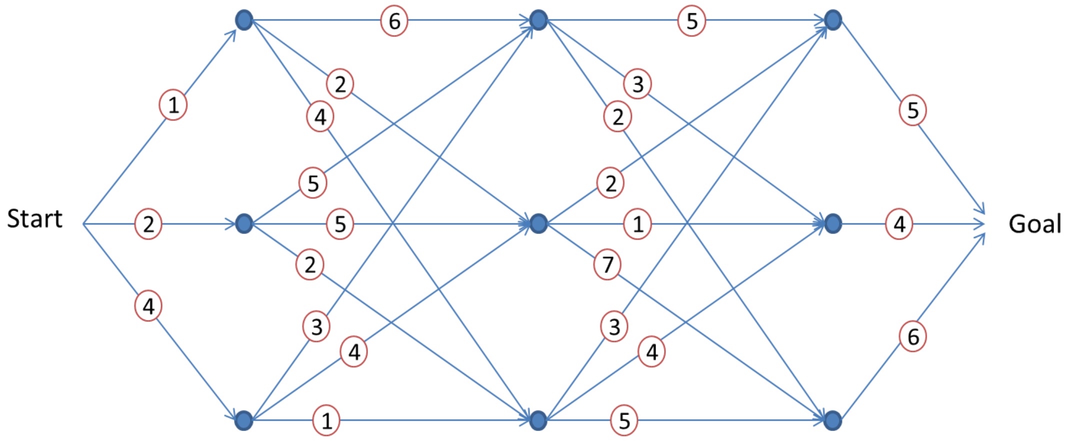
Class Tutorial 2

1. Shortest path on a decision graph

Consider the following decision graph:



1. What are the states, actions, reward, and transition function?

2. Write down the performance objective for the shortest path.

3. How many different trajectories exist for this graph? What is the complexity of a direct search?

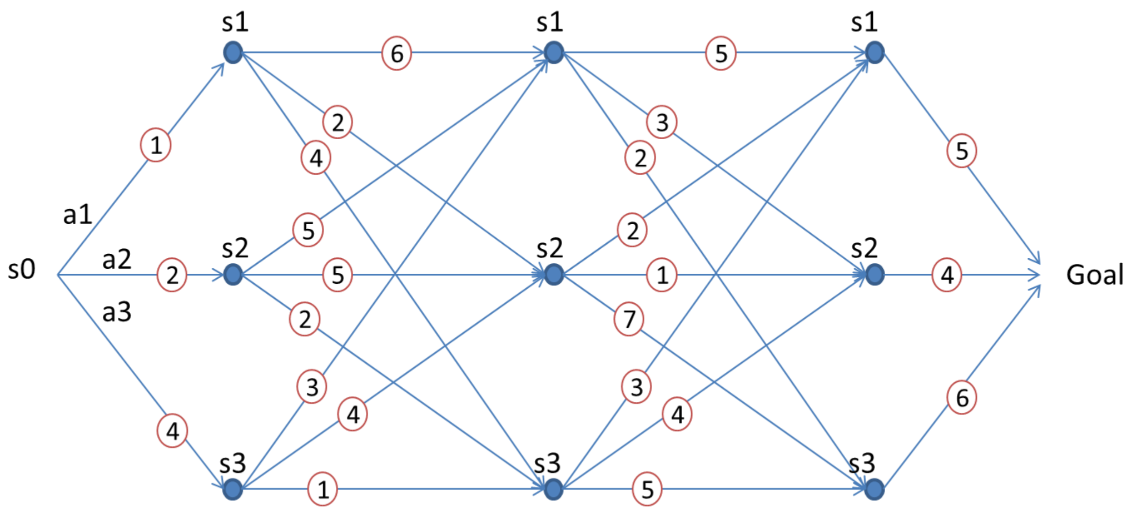
4. Calculate the value function $V_k(s)$ for all states. What is the complexity of this calculation?

5. What is the optimal policy?

6. What would be the complexities if we added k layers to the graph?

Solution

1. For example, label the states and actions as follows:



Then $f_0(s_0,a_1)=s_1$, $f_1(s_1,a_2)=s_2$, etc.

2. Minimize $\sum_{t=0}^{3} c_t(s,a)$.

3. We have $3\times 3\times 3=3^3=27$ trajectories. The complexity is 3^3 additions of 4 terms.

4. We follow the finite horizon dynamic programming algorithm:

(i) Initialize the value function: $V_N(s)=c_N(s)$, $s\in S_N$.

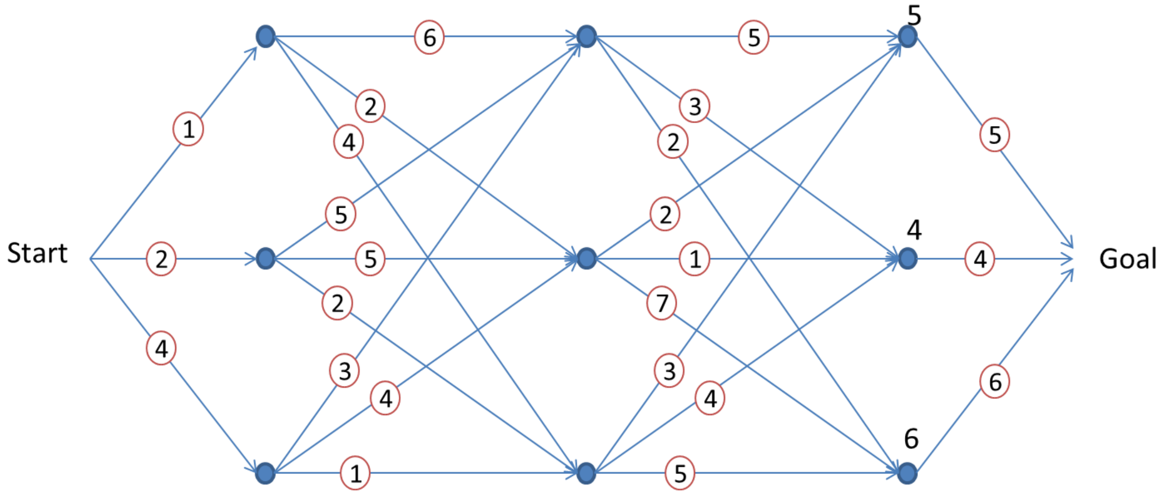
(ii) Backward recursion: For $k=N-1,\text{K} ,0$, compute

$$V_k(s)=\min_{a\in A_k}\left\{c_k(s,a)+V_{k+1}(f_k(s,a))\right\},\quad s\in S_k$$

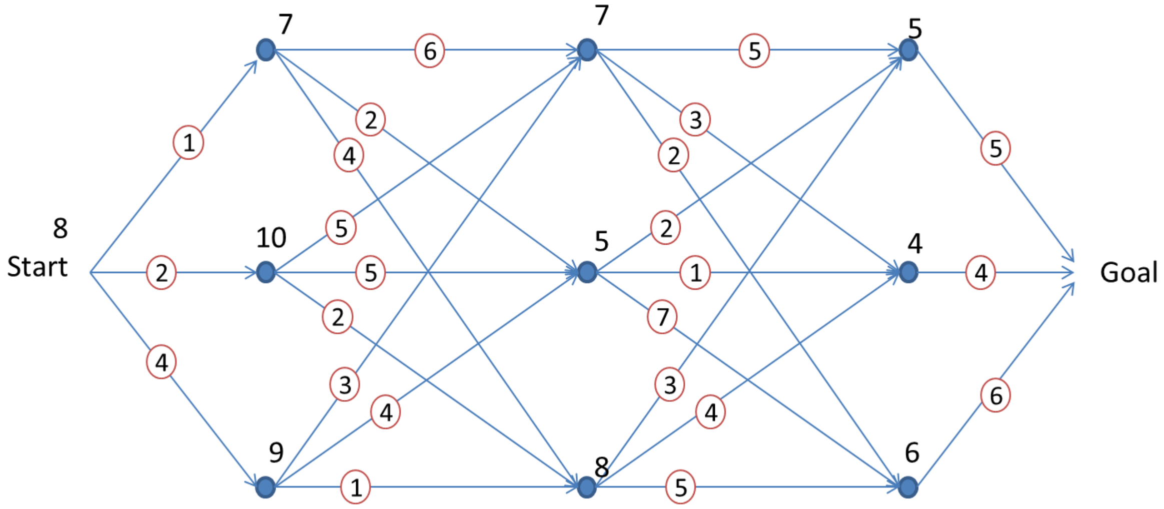
(iii) Optimal policy: Choose any control policy $\pi=\{\pi_k\}$ that satisfies:

$$\pi_k(s)\in\arg\min_{a\in A_k}\left\{c_k(s,a)+V_{k+1}(f_k(s,a))\right\},\quad k=0,\text{K} ,N-1$$

We start with $V_3(s)$:

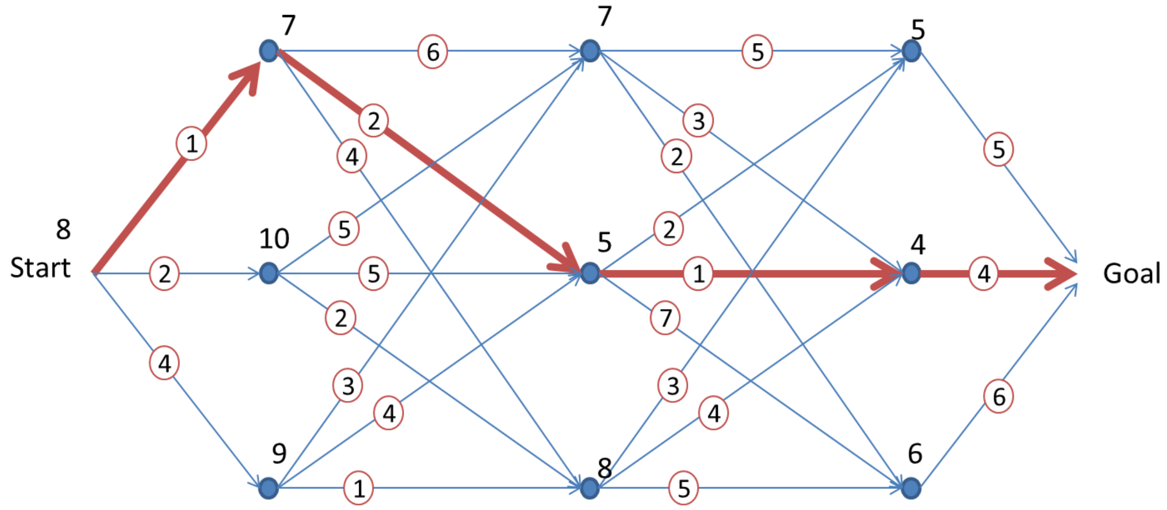


And do the backward recursion:



A total of $3+3\times3+3\times3+3=24$ additions of 2 terms and 7 max operations.

5. The optimal policy:



6. In this case the direct search would require 3^{3+k} computations while dynamic programming would require $6+9(2+k)$ computations. In order notation, the DP solution yields $O(|S||A|H)$ complexity, and assuming that $|S|\approx|A|$, the exhaustive solution yields $O(|S|^{H})$.

2. Traveling Salesman Problem

We are given N cities and distances $d_{i,j}$ between the cities. We wish to find the shortest route that starts at city 1, visits all cities $2,\ldots,N$ and comes back to city 1. We assume that all cities are connected.

1. What is the complexity of a naïve algorithm that checks every possible route?

2. Our state is defined by $s_k \, \mathrm{B}\{U_k, j\}$, where U_k is some subset of k cities containing city 1, and $j \neq 1$ is the final city in U_k. The action set A_k for each state is $A_k(s_k) = \bar{U}_k$, and the cost is $c(s_k, a_k) = d_{j,a_k}$. Explain the meaning of the states and actions, and write down the transition function.

3. Write down a DP algorithm for this problem.

4. What is the complexity of the DP approach? Compare to the naïve algorithm for $N = 20$.

Solution

1. $O\big((N-1)!\big)$.

2. The system transition function $f(s_k, a_k) = (\{U_k, a_k\}, a_k)$

3. Initialize: $V_N(U_N, j) = d_{j,1}$

For $k = N-1,\ldots,1$ compute

$$V_k(s_k) = \min_{a \in A_k(s_k)} \{d_{j,a} + V_{k+1}(f(s_k, a))\}$$

4. For each k we have $\binom{N-1}{k-1}(k-1)$ states (possible subsets times the last city), and for each state we require less than N computations. Thus we require

$$N \sum_{k=1}^{N} \binom{N-1}{k-1}(k-1) \leq N^2 \sum_{k=1}^{N} \binom{N-1}{k-1}$$

$$= N^2 \sum_{k=0}^{N-1} \binom{N-1}{k} = N^2 (1+1)^{N-1} = N^2 2^{N-1}$$

computations. For $N = 20$ we have $(N-1)!\ ;\ 1.2 \times 10^{17}$ and $N^2 2^{N-1}\ ;\ 2 \times 10^{8}$.