Assignment 2

Joseph Wakim

January 29, 2021

Acknowledgements

I discussed problem 3 with Michael Bechinhausen, and we each independently implemented our solutions. LaTeX template for this submission comes from: `https://github.com/gijs-pennings/latex-homework`.

Snakes and Ladders

"Snakes and Ladders" is a classic board game in which players move along a tiled grid and race to a winning position at tile 100. Players take turns rolling a die (typically a fair, six-sided die), which indicates how many grid spaces to advance on the board. While traversing the board, players encounter ladders and snakes at fixed positions. By landing on a ladder, the player advances to a further position. Meanwhile, landing on a snake brings the player back to a previous tile.

The board game can be modeled as a Markov process by representing each board position as a state in $\mathcal{S}$, of which the winning tile represents the sole terminal state in $\mathcal{T}$ and the remaining tiles represent non-terminal states in $\mathcal{N}$:

$$\mathcal{S} = \mathbb{Z} : \mathcal{S} \in [0, 100]$$

$$\mathcal{T} = \{100\}$$

$$\mathcal{N} = \mathbb{Z} : \mathcal{N} \in [0, 99]$$

All players begin at the board's starting position before the first tile; the probability distribution of start states, μ, is give by:

$$\mu = \begin{cases} 1 & \mathcal{S}_0 = 0 \\ 0 & \mathcal{S}_0 \neq 0 \end{cases}$$

The transition probability function can be expressed in a curried format, in which non-terminal states transition to a set of new terminal states with finite probability distribution:

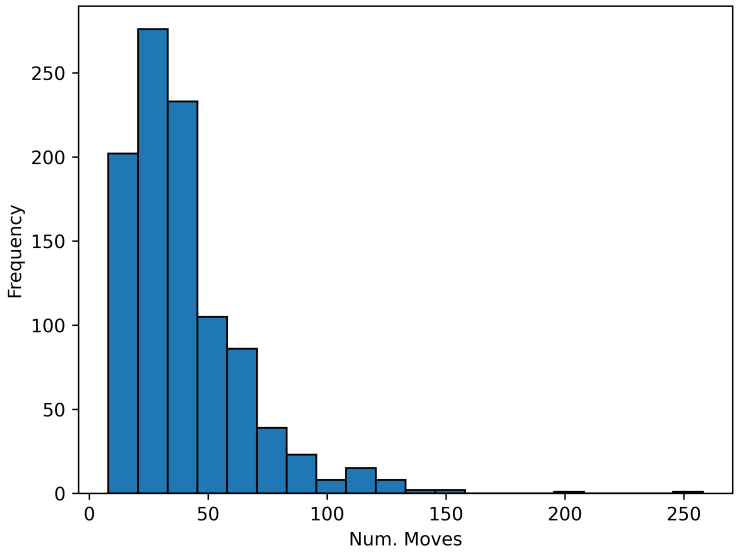

Figure 1: Distribution of "Snakes and Ladders" game lengths (from 1000 traces).

$$\mathcal{N} \to (\mathcal{S} \to [0, 1])$$

More specifically for a six-sided-die, and ignoring edge-cases, this transition probability function can be expressed as:

$$\mathcal{N} \to \left(\mathcal{S} \to \left\{ 0, \frac{1}{6} \right\} \right)$$

In this typical case, each position on the board transitions to six or fewer other positions with probability of 1/6, and any other transition occurs with probability of zero. Special cases occur at the five tiles preceding the winning position, since rolls exceeding tile 100 are rejected; these tiles have a probability of self-transition (corresponding to a rejected roll) given by:

$$P(S_{t+1} = S_t) = \max(0,\ S_t + 6 - 100)$$

One way to express the expected number of moves required for a player to complete the game is to simulate many traces of game-play and observe the distribution of trace lengths. This was implemented for **problem 2** of the assignment. The distribution of game lengths from 1000 traces is included in Figure 1. The average game length obtained by this approach was 39.5 moves.

Another approach is to model the game as a Markov reward process, in which each transition has a uniform reward of 1 and the discount factor is set equal to 1; in this case, the expected number of moves is given by the value function for the process at position 0 on the board. The Markov reward process implementation of "Snakes and Ladders" is completed for **problem 4** of this assignment. The average game length obtained by this approach was 37.7 moves.

Frog Problem

Consider a river containing a finite number of discrete lily pads between its river banks. The "Frog Problem" asks the following: if a frog has a uniform probability of hoping to a lily pad in front of it and never hops backwards, what is the expected number of hops required to cross the river?

For a given position in the river, since the probability of hopping to any lily pad is agnostic to the history of hops taken by the frog, the "Frog Problem" can be posed as a Markov process. To do so, define the states as the distance remaining to cross the river. In this case, the state space $\mathcal{S}$ is given by non-negative integers between 0 and N, where N is the number of lily pads in the river:

$$\mathcal{S} = \mathcal{Z} \ : \ \mathcal{S} \in [0, N]$$

Since the frog always starts on one bank of the river, where there are N lily pads ahead of the other river bank, the starting state distribution is given by:

$$\mu = \begin{cases} 1 & \mathcal{S}_0 = N \\ 0 & \mathcal{S}_0 \neq N \end{cases}$$

Since we care only about the number of hops required to fully cross the river, each simulation of the "Frog Problem" only terminates when the number of lily pads ahead of the frog is zero, and all other states constitute non-terminal states:

$$\mathcal{T} = \{0\}$$

$$\mathcal{N} = \mathbb{Z} \ : \ \mathcal{N} \in [1, N]$$

There are two ways to obtain the expected number of hops required to cross the river. One approach is to frame the problem as the Markov process described above, simulate many traces, and take the average trace length. Another approach is to frame the problem as a Markov reward process with a reward of one assigned to any transition; the expected number of hops is given by the value function for the starting state, where there are N lily pads ahead of the frog.

For **problem 3** of the assignment, I chose to take the first approach, representing the "Frog Problem" as a Markov process, since it offers the benefit of approximating the distribution of hops required to cross the river. Figure 2 plots the distribution of required hop counts from 1000 traces for a river with 1000 lily pads.

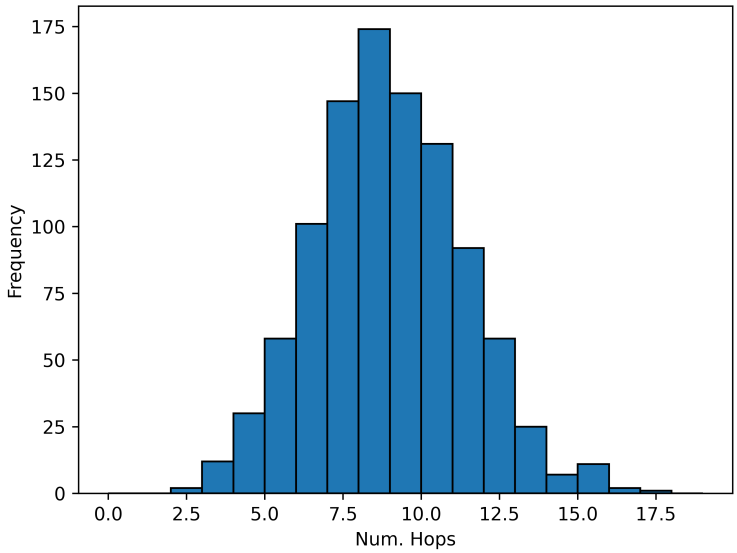

Figure 2: Distribution of hops required to cross a river containing 1000 lily pads (from 1000 traces).

Extension of Stock Price Example

For **problem 5**, I chose to represent the third stock price process described in Chapter 1 of **Foundations of Reinforcement Learning with Applications in Finance** as a Markov reward process. The process defines rewards of each transition as an arbitrary function of the new stock price obtained from the transition. To test my implementation of this Markov reward process, I defined this reward function by 98% of the new stock price. I specified a discount factor γ of 0.95 and a reverse-pull strength α of 0.75. Figures 3 and 4 plot traces for this Markov reward process obtained from 100 time steps.

Figure 3: 100 time-step trace of price obtained from MRP extension for stock price process three.

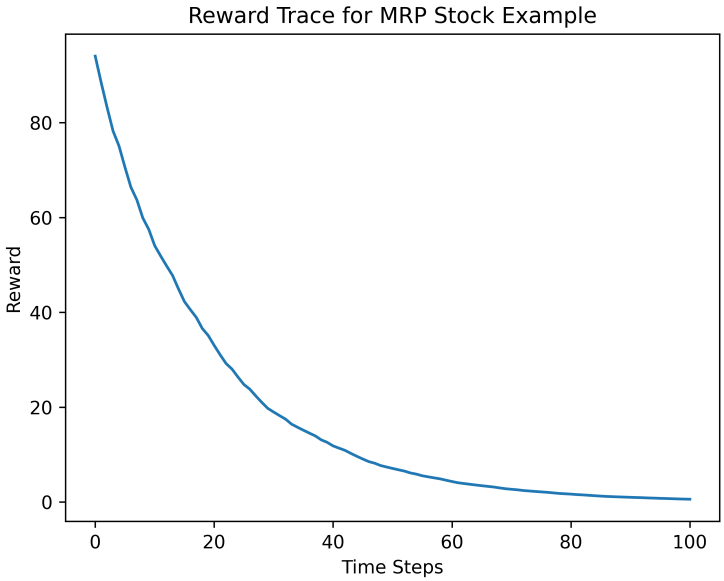

Figure 4: 100 time-step trace of reward obtained from MRP extension for stock price process three.