CS234 lecture 02) Mars Rover Problem

Jiin Sang

2021-01-13

Contents

Exercise 1) States Classification

Determine states of the Mars rover problem whether they're stationary, transient or recurrent, and give reasons why.

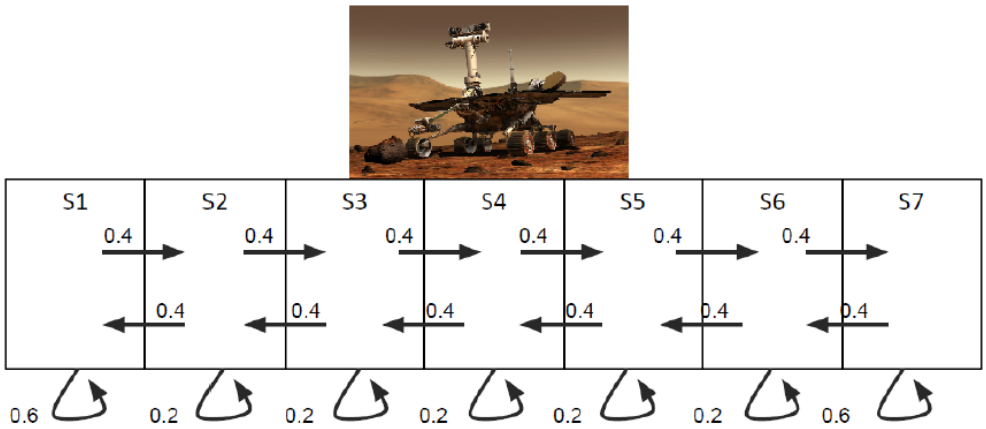

Figure 1: Mars Rover Markov Process

Transition Probability Matrix of Mars rover problem:

$$P = \begin{pmatrix} 0.6 & 0.4 & 0 & 0 & 0 & 0 & 0 \\ 0.4 & 0.2 & 0.4 & 0 & 0 & 0 & 0 \\ 0 & 0.4 & 0.2 & 0.4 & 0 & 0 & 0 \\ 0 & 0 & 0.4 & 0.2 & 0.4 & 0 & 0 \\ 0 & 0 & 0 & 0.4 & 0.2 & 0.4 & 0 \\ 0 & 0 & 0 & 0 & 0.4 & 0.2 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.6 \end{pmatrix}$$

Let X be a DTMC on state space S with transition matrix P. For each state $i \in S$, let τ_i denote the first $n \geq 1$ such that $X_n = i$.

1. State i is said to be recurrent, if $Pr(\tau_i < \infty | X_0 = i) = 1$.
2. State i is said to be transient, if it is not recurrent.
3. State i is said to be absorbing, as a special case of recurrent state, if $Pr_{ii} = 1$.

$\therefore$ All states are recurrent.

Exercise 2) Reward computation using iterative method

Compute reward for above Mars rover problem assuming time horizon = 10, starting from S4

```
import numpy as np

# Transition Probability Matrix
P = np.matrix ([[0.6,0.4,0,0,0,0,0],
                [0.4,0.2,0.4,0,0,0,0],
                [0,0.4,0.2,0.4,0,0,0],
                [0,0,0.4,0.2,0.4,0,0],
                [0,0,0,0.4,0.2,0.4,0],
                [0,0,0,0,0.4,0.2,0.4],
                [0,0,0,0,0,0.4,0.6]])

# Reward Matrix
R = np.matrix([1,0,0,0,0,0,10]).reshape(7,1)
# Time Horizon
H = 10

# V_(t+1)(s')
V_t1 = np.matrix([0,0,0,0,0,0,0]).reshape(7,1)
t = H-1

while(t >= 0):
    V_t = R + (P * V_t1)
    t = t-1
    V_t1 = V_t

print(V_t[3])

## [[8.11882086]]
```

$\therefore$ Reward is 8.119.