Questions

1. The input data does not include any cases where the starting situation was low traffic level in all three directions. Is this normal? If you had this kind of data, what would happen or what should we have done?

Such data was not included because the low traffic in all directions is the end goal. We want our AI to make decisions based on when we are not at our end goal. Had that data been added, we would program our AI to not do anything with the lights.

1. The statement does not say anything about the cost of the actions. What reasonable assumption could we make?

We should assume all actions cost the same. Whether it be 20 seconds for the wait time in between changing lights, or 1, the costs must be assumed to be uniform in this problem statement.

1. What are the expected values of the states? Provide the values to six decimal places. The precision must be greater than one thousandth.

[ 0. , 39.67884991, 36.91828749, 29.3448567 , 37.23555611,

37.07418881, 29.63256563, 30.73464376])

The expected values of the states are as following:

**‘LowLowLow’: 0**

**‘HighHighHigh’:** 39.67884991

**‘HighHighLow’:**  36.91828749

**‘HighLowLow’:** 29.3448567

**‘LowHighHigh’:** 37.23555611

**‘HighLowHigh’:** 37.07418881

**‘LowLowHigh’:** 29.63256563

**‘LowHighLow’:** 30.73464376

1. What is the optimal policy?

[None, 'E', 'E', 'N', 'E', 'W', 'W', 'E']

self.states = {

'LowLowLow': 0,

'HighHighHigh': 1,

'HighHighLow': 2,

'HighLowLow': 3,

'LowHighHigh': 4,

'HighLowHigh': 5,

'LowLowHigh': 6,

'LowHighLow': 7

}

The indices of the states in the object correspond to the output arrays. The optimal policies for each traffic state are the following:

**‘LowLowLow’: None**

**‘HighHighHigh’: ‘E’**

**‘HighHighLow’: ‘E’**

**‘HighLowLow’: ‘N’**

**‘LowHighHigh’: ‘E’**

**‘HighLowHigh’:’W’**

**‘LowLowHigh’: ‘W’**

**‘LowHighLow’: ‘E’**

LowLowLow has no optimal policy because it is the goal state.

1. If we had also had incoming traffic from the South and for each direction we had measured traffic at 5 levels instead of 2, what would have changed in the problem?

We would then have (5 \* 5 \* 5 \* 5) - 1= 625 – 1 = 624 possible starting combinations (excluding LLLL). Thus, the probability of, say, N being green would be less than before.

(5^4)\*4\*(5^4)= 1562500 possible data points to pick from. This would require a larger sample size for creating a more accurate optimal policy. This is a significant difference to the problem we had where there were 192 possible states (2^3\*3\*2^3), and we were given an adequate sample set.