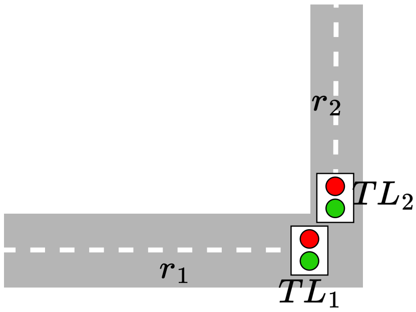
IoT final project - smart traffic light

Consider the following road intersection:



The two traffic lights have two possible states: green and red. When one traffic light is green, the other one is red, and vice versa. Traffic light TF_1 is located at the end of road r_1. Traffic light TF_2 is located at the end of road r_2. At each time step, n_1 cars occupy road r_1, and n_2 cars occupy road r_2. When TL_1 is green, a number of cars equal to $\min\{n_1, 3\}$ pass the intersection and free road 1. When TL_2 is green, a number of cars equal to $\min\{n_1, 2\}$ pass the intersection and free road 2. Independently from the state of the traffic lights, at each time step, a number of new cars in $\{0, 1, \ldots, 5\}$ can enter road 1, and a number of new cars in $\{0, 1, \ldots, 3\}$ can enter road 2. Assume that these events are independent and equally likely (meaning: with probability $\frac{1}{6}$ i new cars enter road 1, $i = 0, 1, \ldots, 5$, and with probability $\frac{1}{4}$ i new cars enter road 2, $i = 0, 1, \ldots, 3$). Nevertheless, if a road is highly congested, no more cars enter. In particular, the number of cars in r_1 cannot be more than 40, and the number of cars in r_2 cannot be more than 25. The traffic on the two roads, r_1 and r_2 can be modeled as a Markov Decision Process, where each state s is defined as $s = (n_1, n_2, TL_1, TL_2, N)$, where $n_1 \in \{0, 1, \ldots, 40\}$, $n_2 \in \{0, 1, \ldots, 25\}$, TL_i is either green or red, $\forall i = 1, 2$, and N is the total traffic, i.e., $N = n_1 + n_2$.
There are two possible actions: $\mathcal{A} = \{TL_1 = \text{green and } TL_2 = \text{red}, TL_1 = \text{red and } TL_2 = \text{green}\}$. From a state s, if we take action $(TL_1 = g, TL_2 = r)$, the set of states reachable from s is:

$$S := \{s' : s' = (\min\{n_1 - \min\{n_1, 3\} + i, 40\}, \min\{n_2 + j, 25\}, TL_1 = g, T_2 = r, N = min\{n_1 - \min\{n_1, 3\} + i, 40\} + \min\{n_2 + j, 25\})\}$$

for $i = 0, 1, \ldots, 5$ and $j = 0, 1, 2, 3$, representing the new cars on the two roads. Therefore, in total there are 6×4 possible states by taking action $(TL_1 = g, TL_2 = r)$, each reachable with equal probability $\frac{1}{24}$. In a similar way, we can define the set of states reachable from s by taking action $(TL_1 = r, TL_2 = g)$ as:

$$S = \{s' : s' = (\min\{n_1 + i, 40\}, \min\{n_2 - \min\{n_1, 2\} + j, 25\}, TL_1 = r, T_2 = g, N = \min\{n_1 + i, 40\} + \min\{n_2 - \min\{n_1, 2\} + j, 25\}\}$$

.
The total number of cars in each state, N, represents a metric of the traffic congestion. If $N < 15$, then the traffic condition is "low". If $15 \leq N < 30$, then the traffic condition is "medium". If $30 \leq N \geq 65$, the traffic condition is "high". Each action gets a +1 reward if it brings the system into a "low" traffic condition. Each action gets a 0 reward if it brings the system into a "medium" traffic condition. Each action gets a -1 reward if it brings the system into a "high" traffic condition.
Find the optimal policy implementing any algorithm you like. Run tests and see how changing the parameters of the system changes the cumulated reward. (The space state is huge and it is taking forever to solve? Consider using lower numbers - e.g., maximum number of cars in the two roads, and adjust the ranges for the three traffic conditions accordingly).