

SAI Assignment 4
19-12-2021

Group 3

Burak Ozdemir s2656205
Tolunay Ahmedov s2689596
Cem Ayerdem s2690713

1.

c. Indeed, the agent finds the best action in the current state and executes this on his own.

2.

c. -

3.

a. Implemented in prison_3.txt.

b. We used the Value iteration to understand the path the agent takes. The agent looks for the shortest path to a goal including the path of reaching the key which leads to the goal door.

If we take two goals which are both exactly far away from the starting point but the key for goal b for example is closer to the agent, the agent will go for goal b instead of the other goal where the key is a little bit further away from the agent.

4.

a. 10.000 free agent locations, 30 keys and doors and a world of 100x100

$10.000 \times 2^{30} = 10.737.418.240.000$ unique states

Amount of locations x key or no key (50/50 so we need to check all possible states)³⁰(all possible keys).

b. $10.737.418.240.000 \times 4 \text{ bytes} = 42.949.672.960.000$ bytes which is roughly 39 terabytes.

c. prison.txt is 64 bytes which gets compiled in 0.21ms so 39 terabytes will be compiled in 4.500 years approximately.

d. R. Bellman in his book "Dynamic Programming" in 1957: "*The curse of dimensionality is the problem caused by the exponential increase in volume associated with adding extra dimensions to Euclidean space.*"

Curse of dimensionality means that the chance of an error increases with the increase of the number of features. Algorithms are harder to design in high dimensions and often have a running time exponential in the dimensions. Theoretically using higher dimensions allows us to store more information, but in reality it does not help because of the higher chance of noise and redundancy in the real-world data. The amount of keys is the aspect of

exponential growth in this problem.¹

5. Comparison to search:

- 1.
- 2.
- 3.

1

<https://analyticsindiamag.com/curse-of-dimensionality-and-what-beginners-should-do-to-overcome-it/>