Value Iteration Algorithm by Alapan Sau Apr 8, 2021

Part 2 Report | MDL

Task 1:

We have used the **Value Iteration Algorithm** with $\delta = 0.001, \gamma = 0.999$ to determine a Policy for the given Markov Decision Process. The details regarding each and every iteration of the algorithm is provided in the `part_2_trace.txt` file.

We have analysed the Policy and some of the key inferences are discussed below.

Important Observations regarding the Policy:

Position: East

1. IJ `never` tries to change his position in East.
2. If IJ is in the East, he tries to attack, either by `shooting` or by `hitting` with blade, depending on the number of arrows and health of MM. This is irrespective of the state of MM.

Position: West

3. Provided MM is Dormant and there are no arrows, IJ `always` prefers to move `right` to Center from West.
   In case of presence of an arrow, it only takes a `shoot` action in West when health of MM is `25` for the greed of a high reward.
4. Incase of MM is Ready, IJ `always` prefers to `stay` in West if there are no arrows available.
   However, with a presence of arrows, IJ `mostly Shoots` .

Position: North

5. Provided MM is Dormant and with absence of materials, IJ `mostly` prefers to move `down` to Center from North. However with a presence of materials, IJ generally `crafts` .
6. When MM is Ready, IJ tries to remain in the North position, by either `crafting` or `staying` (when he does not have materials to craft).

Position: South

7. When MM is Dormant, IJ `almost always` (one exception) prefers to move `up` to the centre from South.
8. When MM is Ready, IJ prefers to `stay` in South mostly, and `gathers` depending on the number of materials.

Position: Center

9. Provided MM is Dormant, IJ `always` prefers to move `right` towards East from Center.
10. If MM is Ready, he `mostly` tries to avoid the East block, either by `shooting` or moving `up` to North or `down` to South, depending on the number of Arrows, number of Materials as well as Health of MM.

Rate of Convergence:

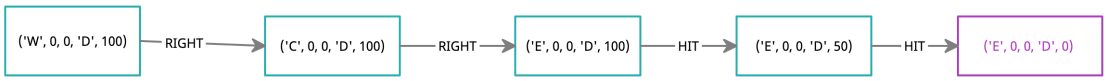
The number of iterations taken for the Value Iteration to converge is 115.

Some Observed Patterns

- IJ is more risk seeking when it is farther away from the goal, i.e, MM health is 100.
- As the goal approaches closer, the risk seeking nature decreases.
- As MM state becomes Ready, IJ avoids the East and Centre.
- When MM is dormant, IJ tries to attack and moves into the positions which enable it. For instance, when it has no arrows in the Centre, it goes to the East and tries to hit with blade, even though the hit probability is low.

Simulation

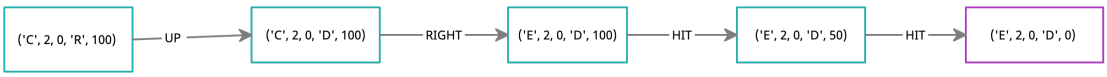
Start State (W, 0, 0, D, 100)



In the most optimal scenario, IJ tries to move towards East and attack MM by hitting.

It chooses to hit with blade rather than shoot arrows inspite of the large difference in because crafting arrows would require additional steps and also, blade deals a heavier damage.

Start State (C, 2, 0, R, 100)



Here the starting state, MM is in Ready state. MM attacks when IJ takes the `UP` action which fails due to the same. However considering the optimal scenario, IJ eventually moves towards the East and attack MM by `HIT` using blades

Task 2:

Case 1 :

Left action at East Square leads to West Square

Trace file: `part_2_task_2.1_trace.txt`

- The number of iterations required for convergence is 129.
- In this case, when MM is in Ready State, IJ has a way of getting away from a potentially high negative reward function and thus, the policy changes such that most of the cases when MM is in ready state and IJ is in the East square now prefer action Left, except those in which the end state is one step away.

Case 2

$Step\ Cost = 0$

Trace file: `part_2_task_2.2_trace.txt`

- The number of iterations required for convergence is 104.
- The number of `Stay` actions in the policy `increase drastically` .
- When MM is in Ready state, IJ prefers to `stay` in `West, North, or South` positions. This is understandable as IJ does not move to a high risk situation, and also does not incur a step cost.

- Even when MM is in dormant state, sometimes IJ tends to prefer the `Stay` action due to no step cost and the comparable utility values of neighbouring states. Stay action is not always successful in North and South and thus, IJ might eventually get out and move to East, which has a high utility value.

  However, as the Stay action is always successful in the West, IJ `loops infinitely in the West block`.

Case 3

$\gamma = 0.25$

Trace file: `part_2_task_2.3_trace.txt`

- The number of iterations required to converge is `8` which is `much lesser` than the other cases. This is because the algorithm does not look too much into the future.
- IJ becomes more risk-seeking, and it moves to the states which might give it a negative reward. This is due to the fact that it `prefers short term gains and ignores long term risks`, due to the reduction of gamma.
- This preference of short term gains also lead IJ to pick `Stay` action in the `West and North`. As Stay is always successful, IJ might `loop infintely` in case he ends up in that state.