Report - Assignment 3 : Probabilistic Search (and Destroy)

Diksha Prakash

October 2019

Abstract

The purpose of this assignment is to model your knowledge/belief about a system probabilistically, and use this belief state to efficiently direct future action.

1 Introduction

1. Default dimension is 50. Hence the grid size is 50 X 50
2. The terrain specification, for every map, is as under:
   (a) 20% of the cells are 'Flat' cells and have an associated 'False Negative' rate of 0.1
   (b) 30% of the cells are 'Hilly' cells and have an associated 'False Negative' rate of 0.3
   (c) 30% of the cells are 'Forest' cells and have an associated 'False Negative' rate of 0.7
   (d) 20% of the cells are 'Cave' cells and have an associated 'False Negative' rate of 0.9
3. The target is placed at a random location and is equally likely to be anywhere in the map.

2 A Stationary Target

2.1 Given observations upto time 't' i.e. Observations $Observations_t$, and a faliure searching Cell j $Observations_{t+1} = Observations_t \wedge Faliure\ inCell_j$ how can Bayes' theorem be used to effectively update the belief state i.e. compute: $P(Target\ in\ Cell_i \mid Observations_t \wedge Failure\ in\ Cell_j)$

The 'Belief Matrix' (dimension: 50 X 50) with belief of target in each cell is given as: $Belief\ [Cell_i]\ =\ P\ (Target\ in\ Cell_i\ |\ Observations\ through\ time\ 't')$
At time $t = 0$,

$$Belief[Cell_i] = P(Target\ in\ Cell_i) = \frac{1}{Number\ of\ cells} = \frac{1}{2500}$$

When the target cell isn't found in $Cell_j$, the belief matrix is updated as follows: (NOTE: Detailed steps for the derivation of next couple of equations can be found in Figure 1.)

$$Belief\ [Cell_i]\ =\ P\ (Target\ in\ Cell_i\ |\ Observations\ through\ time\ 't' \wedge Failure\ in\ Cell_j)$$

The above equation can also be written as: $Belief_{it}\ =\ P(T_i\ |\ X_t)$
Defining Δ as :

$$\Delta = P(\ S_j\ =\ Success\ |\ T_j,\ X_t)\ P(\ T_j\ |\ X_t)$$

$$=> \Delta = 1 - P(\ S_j\ =\ Failure\ |\ T_j,\ X_t)\ P(\ T_j\ |\ X_t)$$

Defining α as :

$$\alpha = \frac{1}{P(S_j = Failure|T_j, X_t)P(T_j|X_t) + \sum_{k \neq i} P(S_j = Failure|T_k, X_t)P(T_k|X_t)}$$

Intuitively, $P(S_j = Failure|T_k, X_t) = 1$ as the target can only be found at its location and duplicate targets don't exist, we can conclude the following:

$$P(T_i|S_j = Failure, X_t) = \alpha(Belief_{it} - \Delta) \quad given, i = j$$

$$P(T_i|S_j = Failure, X_t) = \alpha.Belief_{it} \quad given, i \neq j$$

where, $Belief_{it}$: $(Target\ is\ in\ Cell_i\ |\ Observations\ through\ time\ 't'$
S_j : $Result\ of\ search\ in\ Cell_j$
T_i : $Target\ in\ Cell_i$
X_t : $Observations\ till\ time\ 't'$

Let T_i be Target in cell 'i'

S_i be the searching in cell 'i' i.e. Search [$Cell_i$]

$$\therefore P(T_i \mid X_{t+1}) = P(T_i \mid S_i = Faliure, X_t)$$

$$= \frac{P(S_i = Faliure \mid T_i, X_t) \cdot P(T_i, X_t)}{P(S_i = Faliure, X_t)}$$

$$= \frac{P(S_i = Faliure \mid T_i, X_t)\, P(T_i \mid X_t)\, P(X_t)}{P(S_i = Faliure \mid X_t)\, P(X_t)}$$

$$= \frac{P(S_i = Faliure \mid T_i, X_t)\, P(T_i \mid X_t)}{P(S_i = Faliure \mid X_t)}$$

Now, $P(S_i = Faliure \mid X_t)$ can be further bifurcated into 'True Negative and 'False Negative' cases.

$$\therefore P(S_i = Faliure \mid X_t) = P(S_i = Faliure \mid T_i, X_t) \cdot P(T_i \mid X_t) + \sum_{i \neq K} P(S_i = Faliure \mid T_K, X_t) \cdot P(T_K \mid X_t)$$

CASE : For any '$Cell_j$' different from the searched '$Cell_i$'

$$P(T_j \mid S_i = Faliure, X_t) = \frac{P(S_i = Faliure \mid T_j, X_t) \cdot P(T_j \mid X_t)}{P(S_i = Faliure \mid T_i, X_t)\, P(T_i \mid X_t) + \sum_{K \neq i} P(S_i = Faliure \mid T_K, X_t)\, P(T_K \mid X_t)}$$

now, $P(S_i = Faliure \mid T_j, X_t) = 1$ <As duplicate targets don't exist>

Let $Belief_{tj} = P(T_j \mid X_t)$

Let $\Delta = P(S_i = Success \mid T_j, X_t)\, P(T_j \mid X_t)$

$= (1 - P(S_i = faliure \mid T_j, X_t))\, P(T_j \mid X_t)$

Let $\alpha = \left[P(S_i = Faliure \mid T_i, X_t)\, P(T_i \mid X_t) + \sum_{K \neq i} P(S_i = Faliure \mid T_K, X_t)\, P(T_K \mid X_t)\right]^{-1}$

$$\therefore P(T_j \mid S_i = Faliure, X_t) = \begin{cases} \alpha(Belief_{tj} - \Delta) & ; j = i \\ \alpha\, Belief_{tj} & ; j \neq i \end{cases}$$

Figure 1: Derivation of equations

2.2 Given the observations up to time t, the belief state captures the current probability the target is in a given cell. What is the probability that the target will be found in Cell i if it is searched:

$$P(Target\ found\ in\ Cell_i | Observations_t)?$$

$$P(Target\ found\ in\ Cell_i|Observations_t) = P(Target\ found\ in\ Cell_i|X_t)?$$
$$= P(\ S_i\ =\ Success,\ T_i\ |\ X_t)$$
$$= P(\ S_i\ =\ Success\ |\ T_i\ ,\ X_t).P(\ T_i\ |\ X_t\ )$$

where, the first half is the success rate of the Cell and the second half is the 'Prior Belief'.
The success rate depends on the terrain type and doesn't change with time. Thus:

$$= P(\ S_i\ =\ Success\ |\ T_i\ ,\ X_t)\ =\ (1\ -\ P(\ S_i\ =\ Failure)\ |\ T_i).\ Belief_{it}$$

2.3 Consider comparing the following two decision rules:

1. **Rule 1: At any time, search the cell with the highest probability of containing the target.**
2. **Rule 2: At any time, search the cell with the highest probability of finding the target. For either rule, in the case of ties between cells, consider breaking ties arbitrarily.**

How can these rules be interpreted / implemented in terms of the known probabilities and belief states?
For a fixed map, consider repeatedly using each rule to locate the target (replacing the target at a new, uniformly chosen location each time it is discovered). On average, which performs better (i.e., requires less searches), Rule 1 or Rule 2? Why do you think that is? Does that hold across multiple maps?

In the program, the solver recursively chooses the cell having the best value on the basis of Rule 1 and Rule 2. The idea is as under:

1. RULE 1 : $Belief_{it} = P(T_i|X_t)$
2. RULE 2 : $(1 - P(S_i = Failure|T_i))Belief_{it})$

The solver has knowledge of the prior belief of the cells i.e. $P(T_k|X_t)$ and the false negative rates i.e. $P(S_i = Failure|T_i, X_t)$ associated with all the terrain types. The latter is a constant quantity i.e. it doesn't change with time however, the former is updated and stored in a belief matrix whenever a search fails. (This is done via 'update_belief' function in the code). The updating happens as under:

$$P(T_k|X_{t+1}) = \alpha(Belief_{it} - \Delta) \quad given, i = j$$

$$P(T_k|X_{t+1}) = \alpha.Belief_{it} \quad given, i \neq j$$

In each round, the solver finds a best cell to search based on either of the rules. After getting a response from the Terrain Map, updating of the belief matrix for the decision in the next round.
To compare the performance of the two rules, the target was randomly placed in all the four types of terrain and the iterations were set to a max of 100 for a total of 8 cases (4 types of terrain X 2 Rules). The average number of steps needed to find the target is tabulated as under:

Target in Terrain Type	Rule 1	Rule 2
Flat	3127.78	1179.19
Hilly	4545.92	2678.17
Forest	9548.81	5227.85
Cave	10717.09	13832.46
Average	6984.9	5729.42

Result: Rule 2 performs better in cases where the target is in Flat and Hilly regions, and somewhat better in Forested regions. However, when target is in cave, Rule 1 performs better than Rule 2. The reason for the same is that in Rule 2, we consider the probability of finding the target in a given terrain along with the belief. Whereas, Rule 1 gives equal importance to all the types of cells. Rule 2, by its nature, first searches the Flat terrain, followed by Hilly and Forested terrains and finally Caves. This is the reason why the performance is poor in the case of caves and good in the case of flat terrain.
In other words, changing the map does not change the fraction of each terrain. Furthermore, the target is set randomly among the cells with the same terrain, and the agent also searches randomly among the cells with the best probability, this confirms that the order where the Target is placed in the same terrain cells is irrelevant to this problem.
The graphs depicting the number of steps taken in each terrain by both Rule 1 and Rule 2 are shown in Figure 2 and Figure 3.

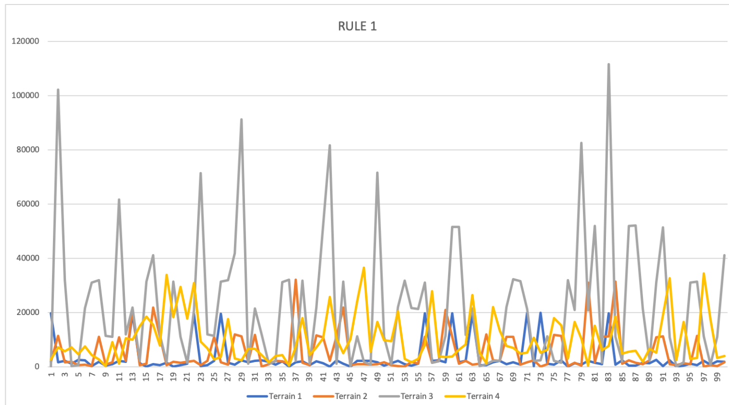

Figure 2: Number of Steps taken in each terrain for Rule 1

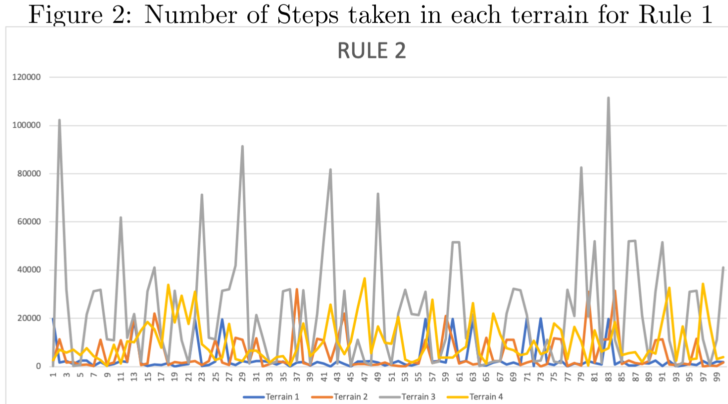

Figure 3: Number of Steps taken in each terrain for Rule 2

2.4 Consider modifying the problem in the following way: at any time, you may only search the cell at your current location, or move to a neighboring cell (up/down, left/right). Search or motion each constitute a single 'action'. In this case, the 'best' cell to search by the previous rules may be out of reach, and require travel. One possibility is to simply move to the cell indicated by the previous rules and search it, but this may incur a large cost in terms of required travel. How can you use the belief state and your current location to determine whether to search or move (and where to move), and minimize the total number of actions required? Derive a decision rule based on the current belief state and current location, and compare its performance to the rule of simply always traveling to the next cell indicated by Rule 1 or Rule 2. How does Utility apply here? Discuss.

Here, two parameters are considered while selecting the next cell. Along with the current belief, the distance of the current cell from the target cell is also taken into consideration. The cell with the maximum value of $\frac{Belief\ of\ current\ state}{Distance\ from\ the\ target}$ is selected. The distance of the current cell from the target is computed using the 'Manhattan Distance' i.e. $abs(x_{target} - x_{cell}) + abs(y_{target} - y_{cell})$.

The following table concludes the average number of steps taken over 100 iterations for each terrain type for both the rules.

Target in Terrain Type	Rule 1	Rule 2
Flat	6211.36	913.16
Hilly	8870.34	2311.76
Forest	23169.29	19012.63
Cave	10054.3	11509.77
Average	12076.32	8436.83

It can be seen that Rule 2 is performing better than Rule 1, although the average steps for both the rules are greater as compared to before. The reason for the same is that we are mostly moving to the neighbouring cells as the motion of agent is restricted to limited distances as the cost of movement is taken into account.

The graphs depicting the number of steps taken in each terrain by both Rule 1 and Rule 2 are shown in Figure 4 and Figure 5.

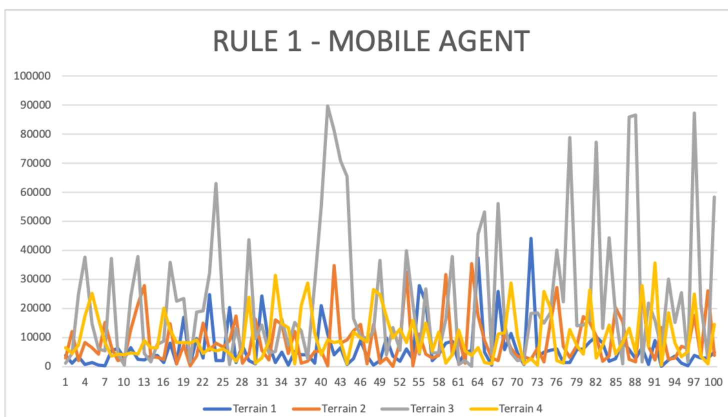

Figure 4: Number of Steps taken in each terrain for Rule 1

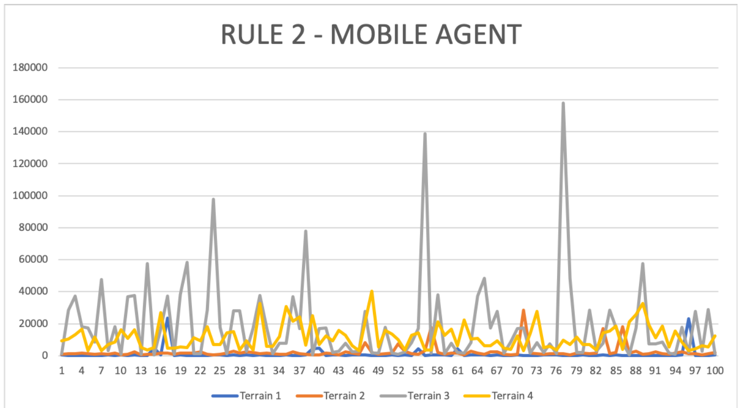

Figure 5: Number of Steps taken in each terrain for Rule 2

2.5 An old joke goes something like the following: A policeman sees a drunk man searching for something under a streetlight and asks what the drunk has lost. He says he lost his keys and they both look under the streetlight together. After a few minutes the policeman asks if he is sure he lost them here, and the drunk replies, no, and that he lost them in the park. The policeman asks why he is searching here, and the drunk replies, "the light is better here". In light of the results of this project, discuss.

In the light of the project, the following analogies can be made:

1. The Drunk man and the Police officer are the people 'Searching" here
2. The "Target" are the lost keys
3. The different terrain types are as under:
   (a) The area under the street light is "Flat" terrain as it is least difficult to search
   (b) The area around the street light is "Hilly" as it is relatively difficult to search there but not very difficult
   (c) The area near the park can be described as "Forested"
   (d) Finally the park can be considered as "Cave" terrain as it will be most difficult to look for the keys there

Probability of losing the keys in the park is high from the belief state of the drunk man. This belief keeps on decreasing as we recede from the park to the street light.

Probability of finding the keys if they're lost under the street light is quite high as the terrain is "Flat". This decreases rapidly as we move from the street light to the park.
I believe that the drunk man is using Rule 2 to find the keys. Provided the man had lost the keys under the street light, the approach taken by him is right. However, the probability of losing the keys in the park is quite high, the drunk man will waste a lot of time under the street light and then should consider moving towards and in to the park to continue the search.