Ramon Pena

Assignment 2

1. Computing:

$$P\left(Target\ in\ Cell_i\ \middle|Observations_t\ \wedge\ Failure\ in\ Cell_j\right) = B_{t+1}$$
$$= \frac{P\left(Target\ in\ Cell_i\ \middle|Observations_t\right)P\left(Target\ in\ Cell_i \middle|\ Failure\ in\ Cell_j\right)}{P(Observations_t\ \wedge failed\ in\ j)}$$
$$= B_t(i) * \frac{P(Failure\ in\ j|Target\ in\ i)}{P(Observatons_t \wedge failed\ in\ j)}$$

$Since\ \ \frac{1}{P(Observations_t\ \wedge failed\ in\ j)}\ is\ \ just\ the\ normalization\ factor, it\ can\ be\ showed\ as$ $\propto. Therefore, we\ can\ use\ \boldsymbol{B_{t+1}(i) = \propto B_i(t) P(failure\ in\ j\ |target\ is\ in\ i)}.$
$Using\ this\ we\ can\ efficiently\ update\ the\ belief\ state\ since, we're\ using\ our\ prior\ belief\ and\ our\ current\ result\ for\ the\ update.\ Then, normalizing\ the\ belief.$

2. $The\ probability\ that\ the\ target\ would\ be\ found\ in\ Cell\ i\ if\ it\ is\ searched\ is\ the\ prior\ probabil$ $multiplied\ by\ the\ false\ negative\ rate$ $\rightarrow P(Target\ not\ found\ in\ Cell_i\ |\ Target\ in\ Cell_i)\ of\ the\ current\ cell. Then\ dividing\ by\ the\ sum\ of\ all\ the\ evidence\ collected\ so\ far.$
$In\ the\ program, this\ can\ be\ done\ by$
$multiplying\ the\ current\ belief\ at\ the\ current\ cell\ by\ the\ probability\ of\ its\ terrain.$
$Then\ normalizing\ by\ multiplying\ by \frac{1}{\sum evidence\ collected}$

3. $Rule\ 2\ on\ average\ requires\ less\ searches. I\ think\ this\ is\ because\ rule\ 2\ skips\ cells\ that\ have\ a\ low\ probability\ of\ \ finding\ the\ target\ within\ it.$
$That\ is\ P(Target\ not\ in\ found\ in\ cell\ I\ |\ Target\ in\ cell\ i) <\ 0.2.$
$This\ means\ that\ cells\ that\ the\ only\ \ cells\ that\ are\ searched\ are\ those\ that\ have\ a\ probability\ of\ finding\ the\ target\ in\ the\ cell\ given\ that\ it\ is\ in\ there\ =\ 0.8.$
$This\ holds\ across\ multiple\ maps. It\ consistently\ does\ less\ searches\ than\ rule\ 1.$

4. $You\ can\ use\ your\ current\ belief\ state\ and\ your\ current\ location\ to\ determine\ where\ to\ search\ next\ is\ by\ checking\ the\ probability\ of\ the\ surrounding\ cells\ containing\ the\ target\ and\ checking\ the\ probability\ of\ finding\ the\ target\ within\ that\ cell. This\ is\ a\ combination\ of\ rules\ 1\ and\ 2. This\ means\ that\ for\ any\ Cell\ (\ i\ )at\ most\ it\ will\ conduct\ four\ searches\ plus\ one\ action\ to\ move. The\ performance\ compared\ to\ the\ first\ two\ rules\ worse\ due\ to\ there\ being\ more\ constraints\ on\ where\ to\ move\ and\ where\ to\ search.\ This\ means\ that\ it\ fails\ to\ find\ the\ target\ more\ often\ than\ the\ other\ two\ rules, but\ the\ times\ that\ it\ does\ find\ the\ target, the\ amount\ of\ actions\ are\ less\ than\ the\ first\ rule.$

5. This project is like the old joke by that the rules we're searching each cell by is like the light. Searching blinding in any given cell because we think that the target might be there. The light in this case tells us that even if the search is easier, the false positive rate will always be 0. We won't ever find the target, were it is not located. The "park" in this project is the terrain and the probability of finding the target within the given terrain. Some of the rules used to search was like the man, ignoring the park because it was more difficult to search there. In this case, some of the rules ignored certain terrains because the probability of finding the target in those terrains was very low, compared to others. So, even if the target was in one of the terrains that was ignored, it would never be found since we weren't even trying to search in there.