INFO8006 : Project - Part 3

Roumache Grégoire - 20162533 and Stoquart Colin - 20161140

December 22, 2018

1 Convergence of the Belief State

When we launch the program with the belief state agent on the observer layout, if we use the parameters $w = 1$ and $p = 1$, for the *rightrandy* ghost, then the convergence takes only 4 steps.
Initially, the *beliefStates* (related to the ghost) all have the value 0.00178253. But once we have updated it for the first time, we get a matrix full of zeros, well... not full of zeros actually. There are also non-zero probabilities in the matrix. Those probabilities are the probabilities of finding the ghost at this position, their values are comprised between 0.09 and 0.23. This value is not surprising seeing that, we could assume the ghost to be in a square with length equals to $2w + 1$ which means that each square should have a probabilities equals to roughly $P \approx \frac{1}{(2w+1)^2} = 0.111$.

But how do we know that the convergence only takes 4 steps and not 2 or 20 ? What we did is that we checked the *beliefStates* matrix at every step until we had what we wanted, i.e. a matrix full of zeros except for only one 1. This is how we knew the *beliefStates* matrix had converged. This method is rather arbitrary seeing that the result may change over different attempts. Furthermore, once the distribution has converged (in the sense we have defined it) it could once again diverge and the distribution could be different from zero for more than one square.

	w = 0	w = 1	w = 2	w = 3
p = 0	1	15	16	22
p = 0.25	1	9	16	15
p = 0.5	1	6	13	18
p = 0.75	1	5	7	16
p = 1	1	4	3	15

Table 1: Convergence of the *beliefStates* Matrix

Note that these results can change from one time to the other. The location of the ghost influences the amount of steps needed to converge.
We noticed two general rules:

1. The number of steps needed for the convergence of the matrix increases as the parameter w increase. This makes sense. Indeed, w characterizes the noise of the sonar. So if the noise increase, it is more difficult to accurately locate the ghost. As we could have expected it, the convergence occurs in only one step for $w = 0$. In this case, pacman knows directly and without hesitation where's the ghost.

2. The number of steps needed for the convergence of the matrix is reduced when the value of p is high. Once again, this is coherent. If we increase p, we increase the probability of taking "EAST" for the next action. So, if we increase p, we reduce uncertainty about the next state. For example, if $p = 1$ and "EAST" is a legal action, we are sure that the action chosen by the ghost will be "EAST". On the other hand, if we decrease p, ghost could take several different actions and we can't perfectly predict which ones. The reason why we can't converge directly when $p = 1$ is that we "EAST" could be an illegal action.

2 Agent Improvement

We could have greatly improved the amount of computation the agent takes if we didn't had to re-compute the transition matrix every time we have to compute a new distribution.

At first, we had a problem because the agent took into account measurements that are not physically possible. Since the walls are not taken into account in the evidence matrix, we get non-zero probabilities on walls. Yet, the transition matrix doesn't ignore the walls like the evidence matrix does. It means we can't go to the location of a wall but if we're on a wall's position already, we can only go to empty positions. In the end, we improved

the agent by saying that we can't go anywhere since we're on a wall's position, i.e. since the position we're on is not possible, it should not influence the probabilities further.
In other words, we have (arbitrarily) defined :

$$P(x_{t+1}|\text{"}x_t = \texttt{walls}\text{"}) = 0 \qquad \forall x_{t+1}$$

We could have done a more clever choice but this seems to work well.
Finally, note that only the first belief states has non-zeros probabilities on walls. Then, we use the transition matrix to stop it. The rest of the belief states matrix will have a zero probability on a wall's position.

An other way to improve our prediction is to try to find better models. For example, we use for the sonar model a uniform law on a square $W \times W$. However, we could expect that the center of the square should have an higher probability than its border. One way to implement it is to use a Gaussian distribution center on the center of the square with a standard deviation of W.