Autonomous Systems - Perception, Markov Localisation

Rupp Matthias

04/2021

FH Vorarlberg
University of Applied Sciences

Contents

NOTE: Octave was used to solve this exercise. Since Octave-indices start with 1, not 0, all indices were incremented by one. So, the cells go from 1 to 50, and the cells where pillars are located where also shifted by one. This does not change the end result, but should be kept in mind when reading this documentation.

1 Description of models used

Two models need to be determined to properly calculate the robot's movements: the movement model and the measurement model.

1.1 Movement model

The movement model for this exercise is rather easy. The robot moves a certain distance with a certain probability, with the distances being $3, 4, 5, 6, 7$ and the probability being $0.1, 0.2, 0.4, 0.2, 0.1$, respectively. These values are saved in a vector with two rows, the first row containing the distance values and the second row containing the probabilities for said distances. This way, by taking a column of the vector, both the distance and the probability for the robot moving said distance can be extracted. This can be used, together with the "old" posteriors of the preceding steps, to calculate the priors.

1.2 Measurement model

The measurement model is more complex. First off, there is a measurement error, with a uncertainty of ± 1 cell. The actual distance is measured with a probability of 0.5, with one cell more or less being measured with a probability of 0.25. Secondly, the robot can only measure the next pillar it can see. Pillars are placed on cells $5, 14, 21, 32, 46$, so if a robot stands on cell 3, it can see the pillar on cell 5, but not the one on cell 14. This, in turn, means that a distance measurement of, say, 11 cannot have happened on cell 3, even though there is a pillar on cell 14, since cell 5 blocks the measurement. The robot will also not measure any pillars that are on the same cell as him, which also means that there will be no distance with value 0 measured.

All this leads to a measurement model that is built as follows.

1. Create a NxN matrix, where N is the number of cells (50 in this exercise). Note that the matrix could be chosen smaller in this exercise, but a NxN matrix could capture measurements from each cell to every other cell, so it will be used.

2. Let the index of the matrix row be r and the index of the matrix column be c

3. For each matrix cell r, c, enter the probability that the robot measured the distance c from the cell r with 0.5

4. For the cells $r, c - 1$ and $r, c + 1$, enter the probability 0.25, which is the probability of the measurement error (meaning the robot measured on cell less/more than the cell the actual obstacle is in)

The result of this process is a NxN matrix from which, using the cell index as r and the measured distance as c, the probability that this distance was measured from said cell can be extracted from cell r, c of the matrix. This matrix, together with a vector containing the measured distances for each step, $5, 10, 7, 2, 10$, will be used in the posterior calculation.

2 Documentation of procedure

For this exercise, we start with a uniform distribution for the robot's placement. Since we have 50 cells, this means a probability of $\frac{1}{50}$ for every cell. The robot now takes the first step. This means the calculation of the new position based on the movement model. Equation 1 shows the formula for this process, taken from page 19 of Merz (2021).

$$\overline{\text{bel}}(x_t) = \sum_{x_{t-1}} p(x_t \mid u_t, x_{t-1}) \, \text{bel}(x_{t-1}) \tag{1}$$

We want to calculate the probability (or belief), that we are at the position x_t after moving. To this end, we take the probability that we have moved to position x_t given the movement information u_t, x_{t-1} and multiply this with the probability that the robot was located at x_{t-1} in the first place. We then sum up these calculations for each cell and all movement information to get the probabilities that the robot moved to each cell.

In this concrete example, where the robot moves $3, 4, 5, 6, 7$ steps with the probabilities $0.1, 0.2, 0.4, 0.2, 0.1$, we go through each cell, calculate into which cells it can move from the cell we are currently looking at, multiply the movement possibilities with the possibility that the robot is located in the current cell in the first place, and remember said possibilities for the cells the robot may have moved to. To give a concrete example: the robot is in cell 5 with a probability of 0.2. It moves 3 cells with a probability of 0.1, so the probability that the robot moved to cell $5 + 3 = 8$ from cell 5 is $0.1 * 0.2 = 0.02$. For moving 4 cells, the calculation is $0.2 * 0.2 = 0.04$ for moving to cell $5 + 4 = 9$ and so on. By doing this for every cell and summing up the resulting probabilities (so we sum up all the probabilities that the robot could have moved to a certain cell from any other cell, and we do this for each cell), we get a new distribution of probabilities that shows where the robot might be after having moved. These are our priors. The next step is for the robot to measure distances.

How to calculate the robot's position after sensor measurements, also called the posteriors, can be seen in Equation 2, also from page 19 of Merz (2021).

$$\text{bel}(x_t) = \eta p(z_t \mid x_t, M) \overline{bel}(x_t) \tag{2}$$

What we want to know is the probability that the robot is on certain cell given the measured distance. However, we do not know this. Instead, we calculate our posteriors by calculating the probability that the robot measured a certain distance z_t given his current position x_t and the map with our obstacles M, which we multiply with the probability that the robot is located at said current position in the first place (which is the prior we calculated in the step before, $\overline{bel}(x_t)$). A concrete example would be: we are at cell 12 with a probability (the prior) of 0.35. The measured distances, as specified in the task sheet, are $5, 10, 7, 2, 10$. Assuming we are currently measuring after step 4, this means we measure a distance of 2. Since the next (and only) pillar the robot can see from cell 12 is the pillar at cell 14, the probability that we measured the distance 2 given the position 12 and the map with the obstacles is 0.5, due to our sensor having uncertainties. So, the calculation is $0.5 * 0.35 = 0.175$ for cell 12.

The principle is the same for all other cells. Then, we need to multiply this result with a factor η, to ensure a sum of 1 over all posteriors. Before calculating this η, it is a good idea to replace all posteriors which are smaller than a certain value min_p with said value min_p, to get rid of any zeros or very small values. η can then be calculated with the simple formula $\eta = \frac{1}{sum(posteriors)}$. Now, we multiply our posteriors with η, and have finished the calculation of the posteriors for this step.

In the next step the robot makes, these posteriors will be used to calculate the priors, and the process repeats.

3 Results for 5 steps

In this section, the prior and posterior plots for 5 steps will be shown.

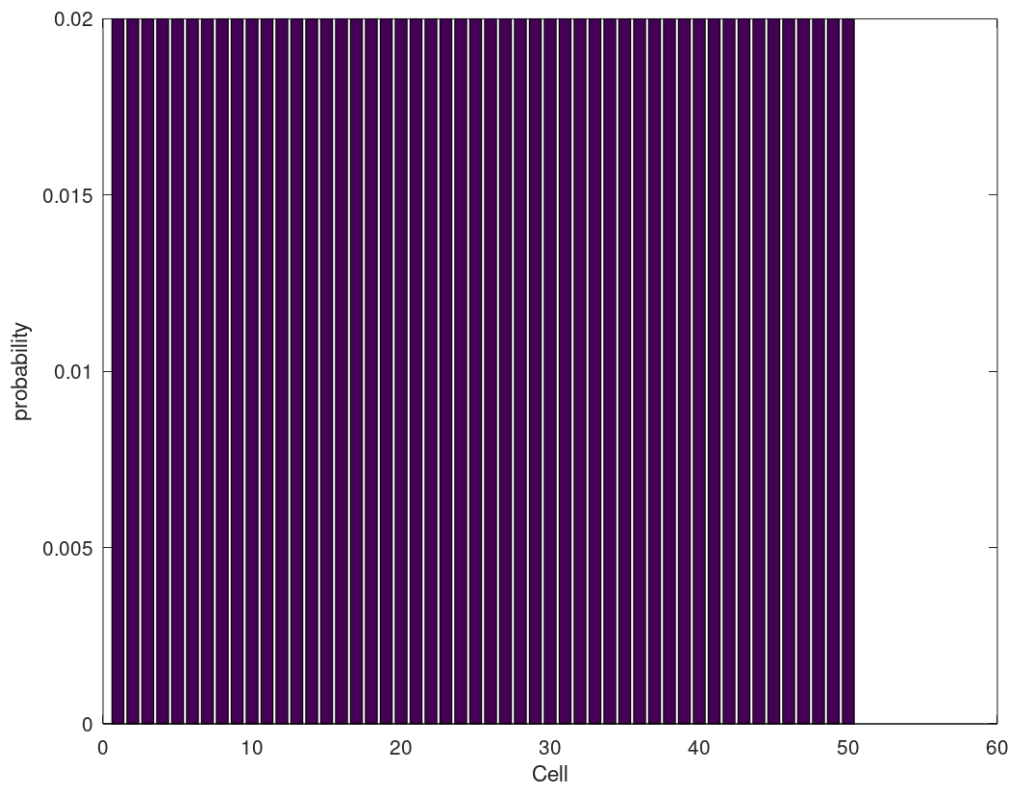Figure 1 and Figure 2 show the priors and posteriors for step 1.
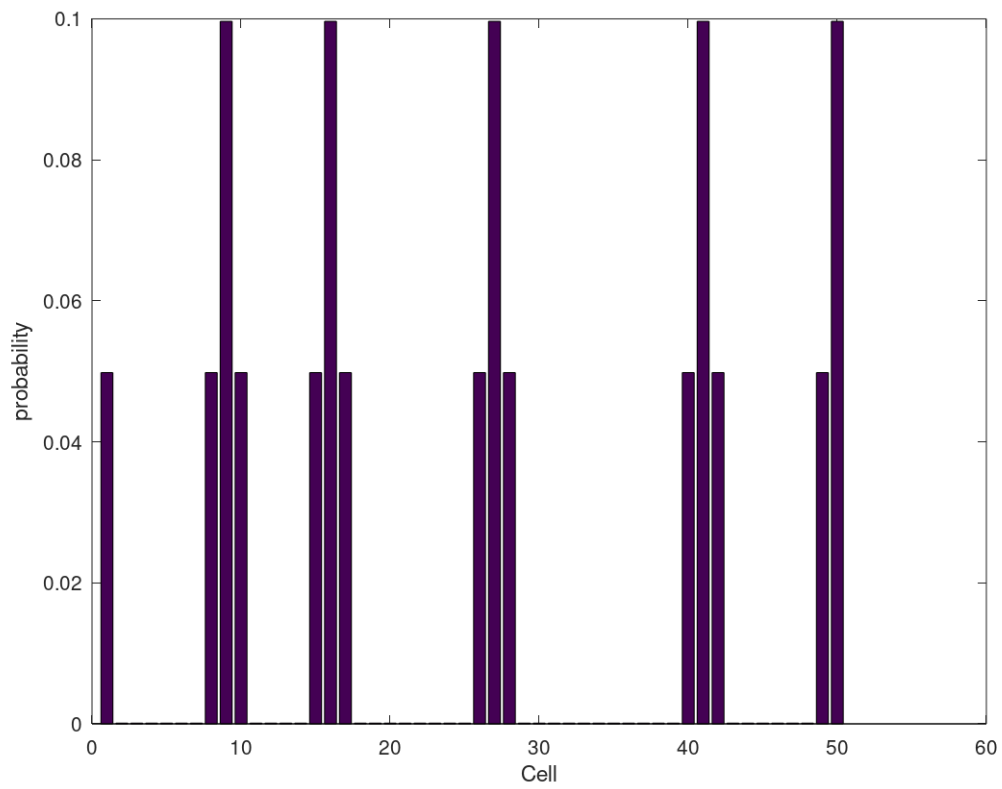
Figure 1: Prior, step 1

Figure 2: Posterior, step 1

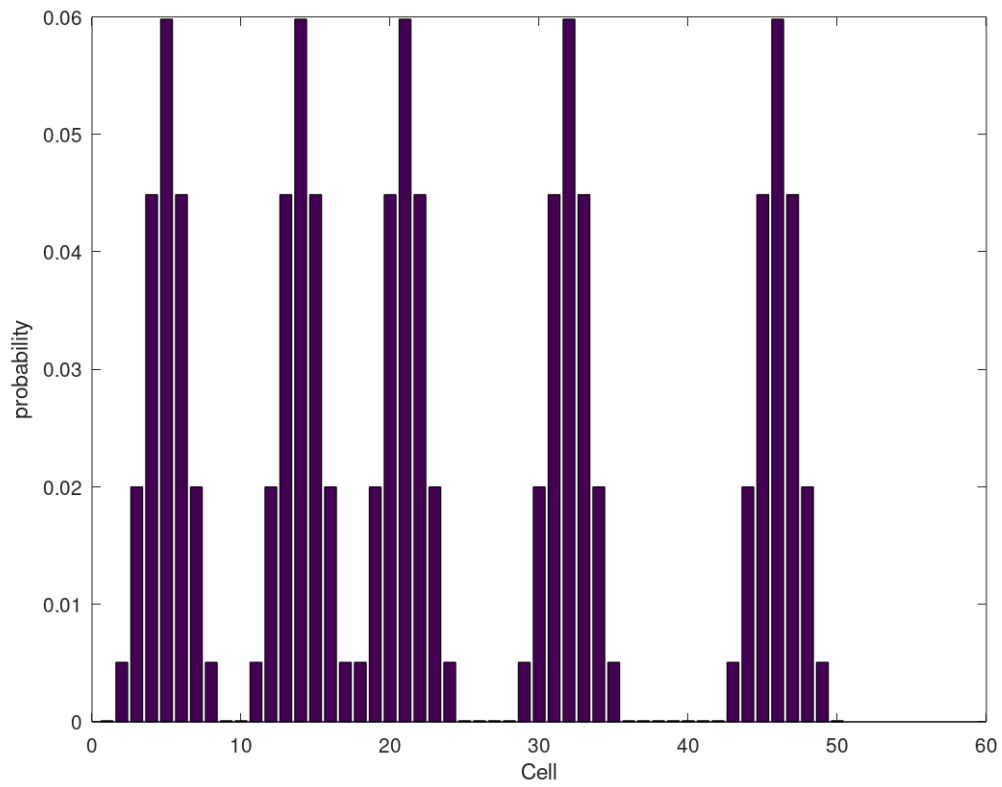Figure 3 and Figure 4 show the priors and posteriors for step 2.
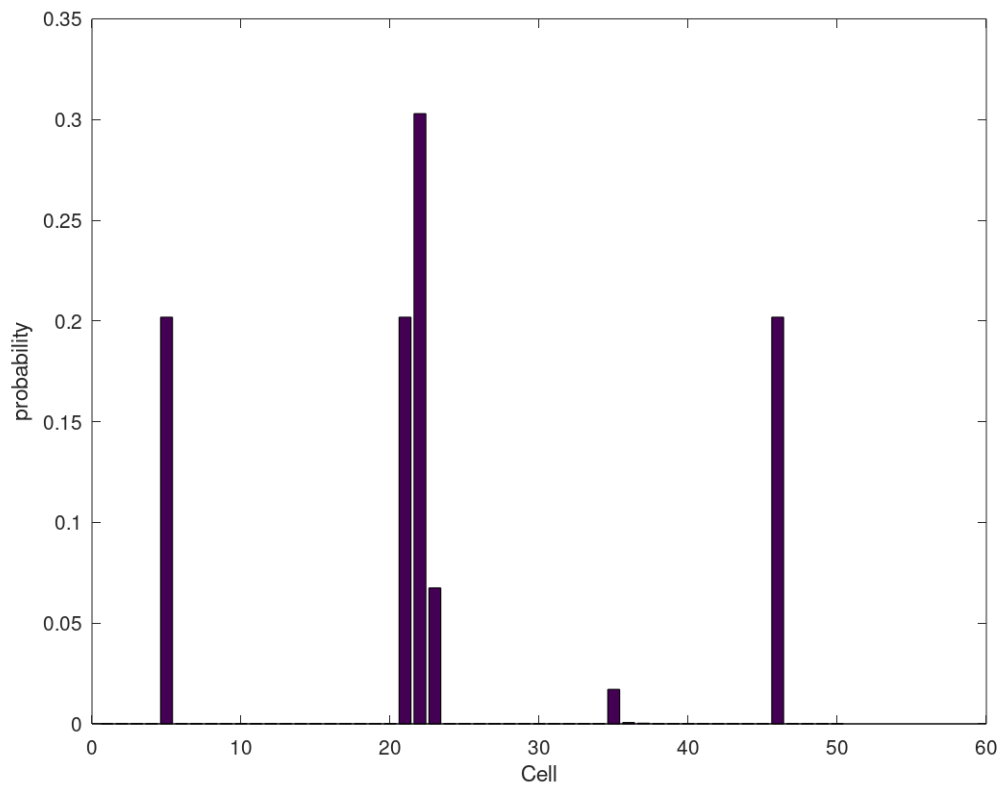
Figure 3: Prior, step 2

Figure 4: Posterior, step 2

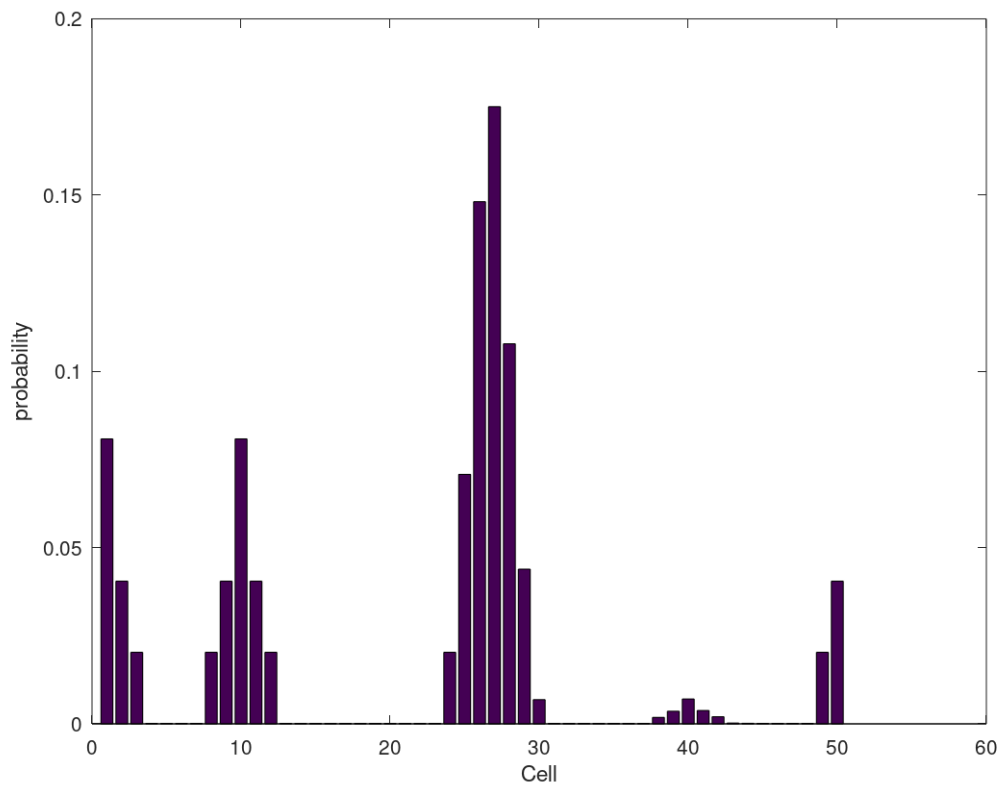Figure 5 and Figure 6 show the priors and posteriors for step 3.
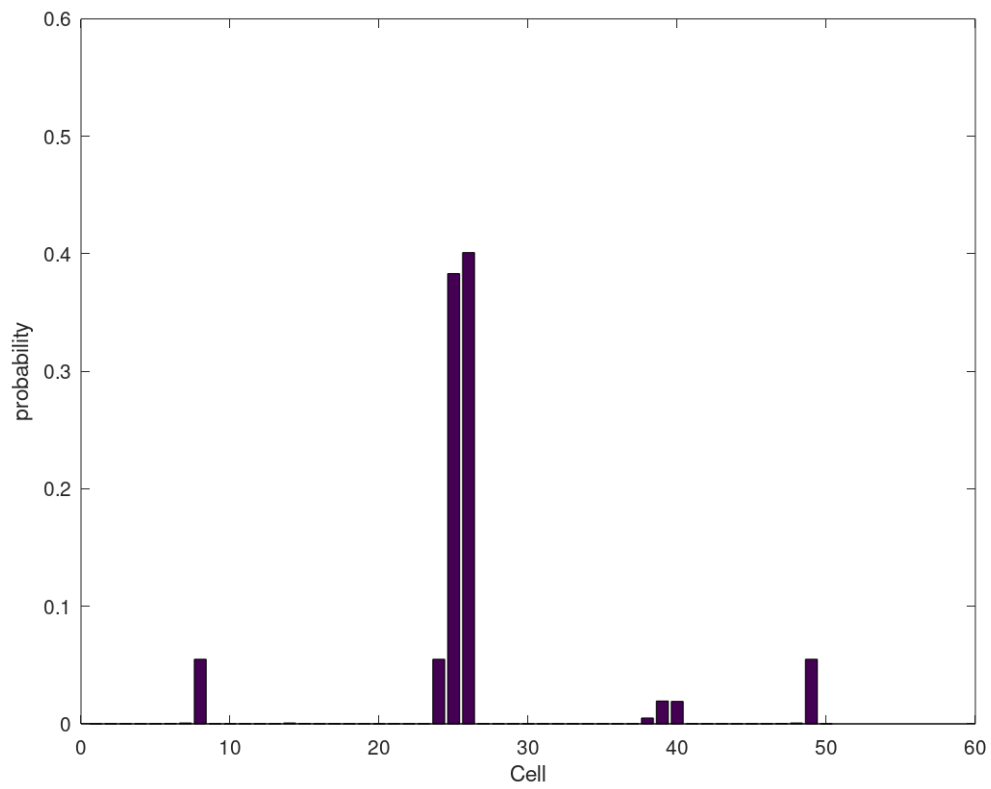
Figure 5: Prior, step 3

Figure 6: Posterior, step 3

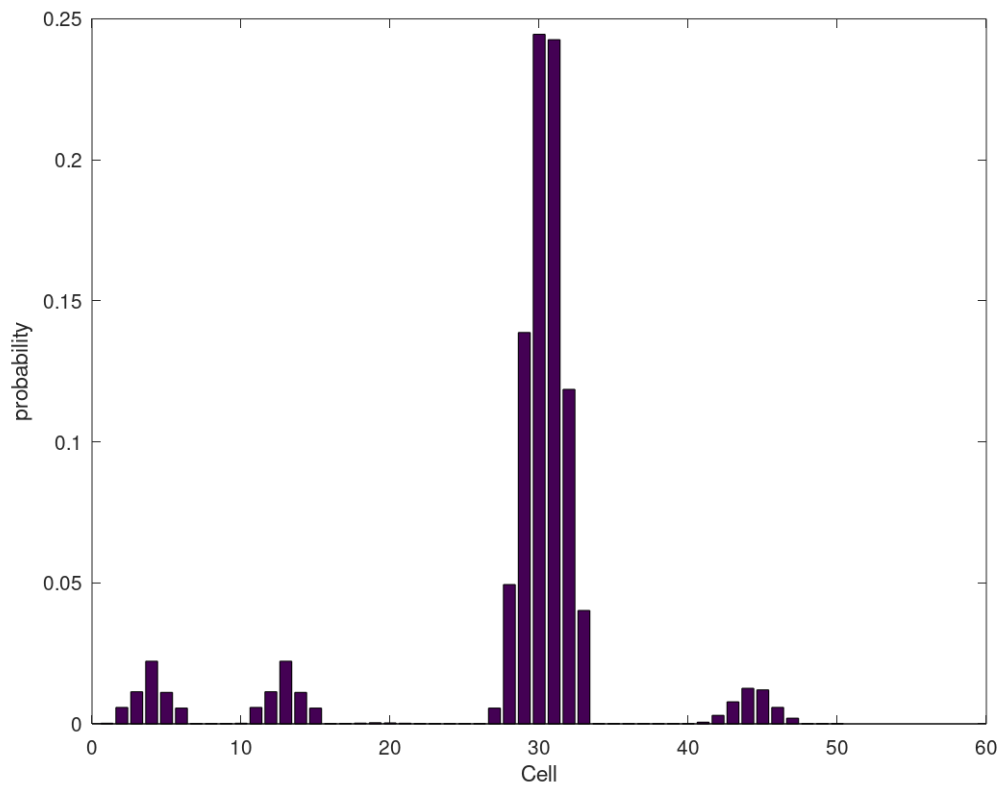Figure 7 and Figure 8 show the priors and posteriors for step 4.

Figure 7: Prior, step 4

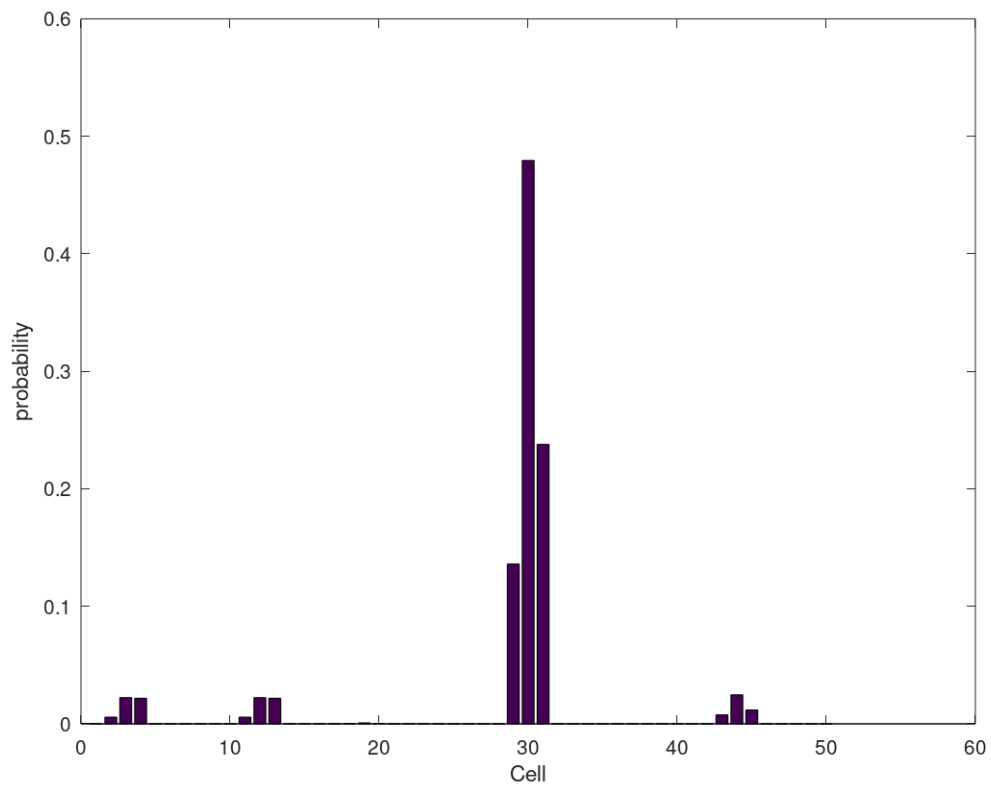Figure 8: Posterior, step 4

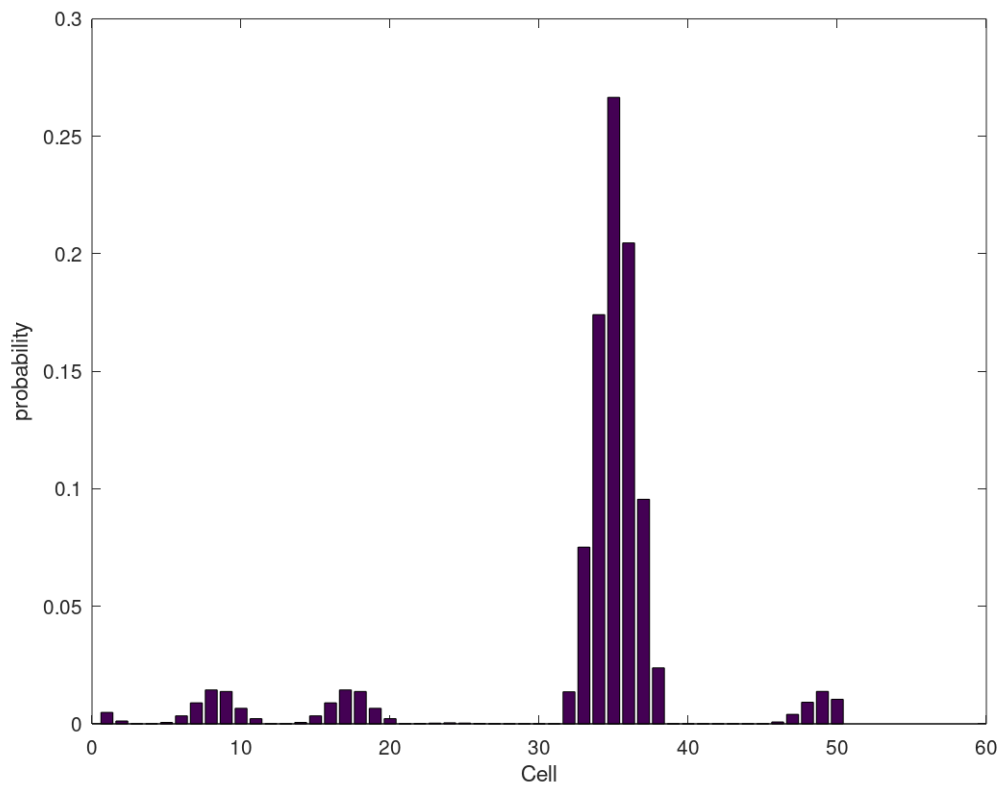Finally, Figure 9 and Figure 10 show the priors and posteriors for step 5.
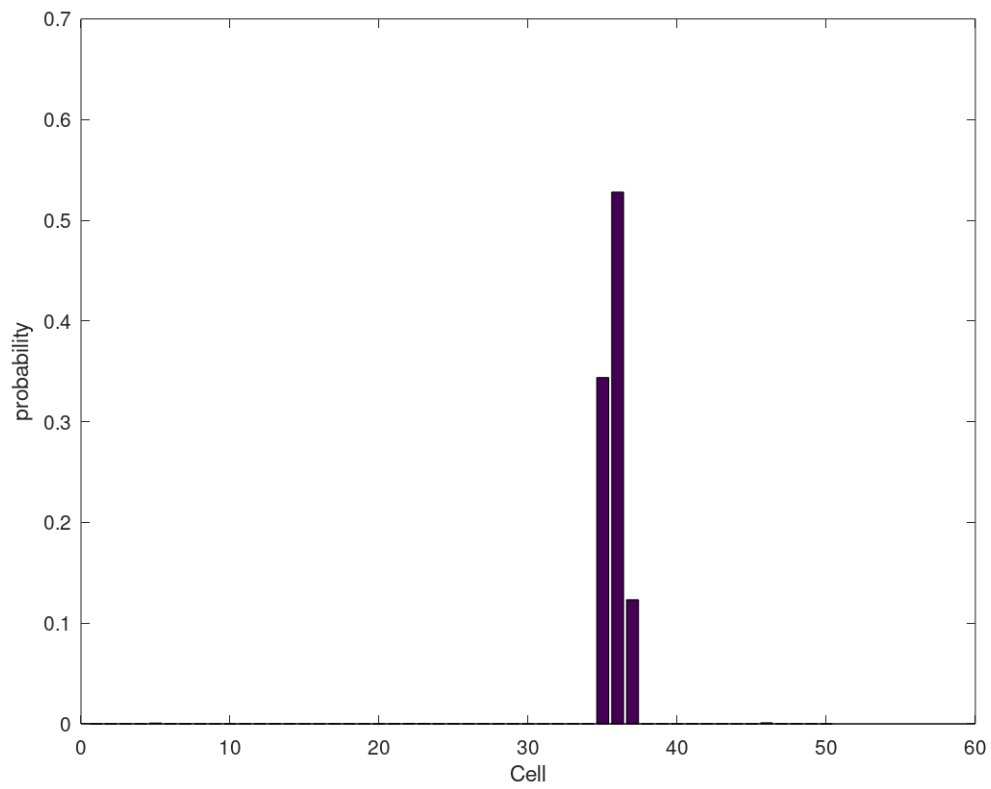
Figure 9: Prior, step 5

Figure 10: Posterior, step 5

References

Merz, Robert (2021). "Autonome Systeme 2 - Lokalisierung, Kartierung, Pfadplanung".

List of Figures