Localization Problems and Hints

1. \( p(s|d,\phi) \) table
   a. densities vs probabilities: use either one; probabilities give higher numbers
   b. \( s=m \) and \( s<m \) values don’t mix, even for multiple-sonar update
   c. table scaling for \( s<m \) is small compared to \( s=m \)

2. Gain
   a. Gain is implemented as a mixture of a uniform distribution and the on-axis and off-axis mixture
   b. A mixture is just a weighted average of the two distributions
   c. Might want to use a slightly smaller value of uniform distribution, so that there is some probability left over for \( s=m \)

3. Update step
   a. The post-probabilities will be very small, especially if you use a lot of sonars. Expect a factor of 1/100 for every sonar.
   b. Make sure you get the right sample when searching the CDF with binary search!!!!
   c. Smearing is a good idea to keep samples diversified. Use uniform smearing (random x,y,\( \theta \)).

4. Parameters
   a. There are a lot of parameters, and it helps to understand what they do and how to adjust them
   b. #define’s are a bad way to set up parameters, because you have to compile to reset them. Make the parameters available through Colbert, so you can change them on-the-fly.
   c. Things to change: robot error (relax it), smearing (increase it), number of points, update distance/angle

5. Moving the robot
a. Biggest problem here is that the predict/update cycle runs asynchronously. When it gets finished, the robot has moved from its location at the start of the cycle. How do you update the old position / new position?

b. Check out the SfROBOT->moveTo(ArPose p, ArPose q) function

6. Debugging tips
   a. Update step
      • Isolate the update step, so it can be performed on command and with a single sonar.
      • Set up a situation with a single wall, samples mostly in the same pose, and see what happens with various single sonars. Vary the gain and smearing.
   b. Predict/Update cycle
      • Set the gain to 0, and check that the update/resampling step moves the cloud as with prediction alone.
      • Turn off robot jumping and watch how the cloud evolves
      • Make sure to compensate for robot jumping is setting up the old robot pose for the predict/update cycle