Midterm Exam SOLUTIONS

16-281: General Robotics

2026

This exam is being administered in person. You should write all of your work and solutions by hand. We will collect your answers when the test is finished.

- You will have 1 hour and 10 minutes to complete this exam
- There are 5 sections on 20 pages. Make sure you have all of the pages. There may be blank pages throughout the sections.
- When making drawings, please be precise. Rounded edges should look rounded, sharp edges should look sharp, sizes should be close to scale. Neatness counts.
- Show your work. Partial credit may apply. Where applicable, algebraically justify your work and show intermediate steps to ensure full credit.
- It should be very clear what your final answer is. Circling your final answer is recommended.
- You may need to make certain assumptions to answer a problem. State them (e.g. what is optimal?).
- You are allowed one handwritten two-sided reference sheet for the exam. No cell phones, laptops, neighbors, etc. allowed.
- Good luck and you can do it.

Contents

1 Vision [20 points]

Alan and Rahul have taken a trip to the nation of Suriname to look at some blue poison dart frogs. Since these frogs are poisonous, Alan and Rahul use their 16-281 knowledge to make a robot to get up close and take pictures of the frogs.

Figure 1: A blue poison dart frog

In this question, you will help our explorers use computer vision to take some pictures of these amazing amphibians.

1. We want to create a program that thresholds the frog from the forest background. What color representation would be the best for this? How would you tune this threshold for varying light conditions? [5 points]

   We would use HSV (hue-saturation-value) as color representation. While one might be tempted to use RGB for this (maybe thinking that the frog would have low red and green and high blue), variations in the conditions between images (such would lighting) could greatly affect all three values. Hue would allow us to separate based on the color itself, saturation on its intensity, and value on its brightness. **Along those lines, we would tune the threshold based on the value.**

2. Due to poor lighting conditions, after thresholding you are tasked with highlighting the edges of the frog. Explain how you would do this using convolution. Write out the matrix of the kernel/filter that you would use. [5 points]

Possible Answers:

1. Use two kernels to compute the gradients in the x and y directions and use the L2 norm of their outputs (or some other method of combining the outputs) then choose some threshold to identify a line.

$$\text{x\_grad: } \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ y\_grad: } \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

2. Detect horizontal and vertical edges simultaneously using the following kernel and define edges as those above some specific threshold:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

3. Oh no! It started to rain! After using your proposed method above, you realize that there are many false edge detections due to image noise. What filtering method can be used to resolve this? Why does this reduce noise/improve edge detection? [5 points]

   Use a gaussian filter on the image before edge detection to smooth out the noise. This helps with edge detection because it reduces the spike in the gradient on the noisy parts of the image by averaging the pixels with its surroundings.

4. Our explorers have found a frog! Using a LIDAR, the explorers find that the frog is 3 meters away. They decide to take pictures of the frog using a camera with these specifications:

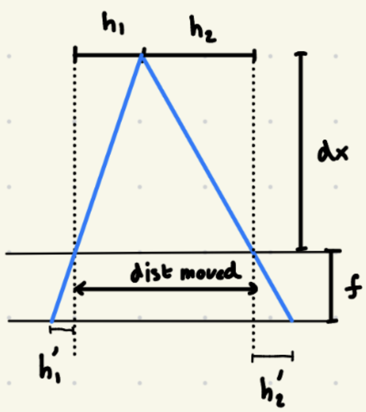
   - Focal length: 30 mm
   - Camera resolution: 5000 x 3500 pixels
   - Pixel size: 8 μm

   In the first picture, the center of the frog is 55 pixels to the left of the center of the image. The robot moves to the right and the second picture. The center of the frog is now 35 pixels to the right of center. The slippery rainforest terrain means that odometry can't be trusted. Use the information provided to calculate how far the robot moved between the first and second pictures. The robot moves while maintaining the same camera orientation. [5 points]

NOTE: The robot should move to the **left** not right. Saying the robot moves to the right creates an inconsistency in the information given in the camera. The reference solution solves the question with the robot moving to the left as intended by the authors of the question. Our sincere apologies for the oversight here, but we graded the questions very generously to compensate for this.

Δpixels = 90

$\Delta x = 90\ px \cdot 0.008\ mm = 0.72\ mm$



$h_1' = 55\ px \cdot 8 \cdot 10^{-6}\ m = 440 \cdot 10^{-6}$

$h_2' = 35\ px \cdot 8 \cdot 10^{-6} = 280 \cdot 10^{-6}$

$\frac{h_1}{dx} = \frac{h_1'}{f} \qquad \frac{h_2}{dx} = \frac{h_2'}{f}$

$h_1 = dx\, \frac{h_1'}{f} \qquad h_2 = dx\, \frac{h_2'}{f}$

$h_1 = 3 \cdot \frac{440 \cdot 10^{-6}}{0.03} = 0.044\ m$

$h_2 = 3 \cdot \frac{280 \cdot 10^{-6}}{0.03} = 0.028\ m$

$h_1 + h_2 = \boxed{7.2\ cm}$

PAGE INTENTIONALLY LEFT BLANK
You can work out problem 1 here

2 Control [20 points]

A telescope must precisely track planets as they move across the sky. The telescope rotates about a single axis with angular position θ. Due to disturbances, the telescope's pointing angle drifts over time, there will be steady-state error. To maintain accurate tracking, the system uses a PID controller to regulate the telescope's angular position.

The control system measures the error

$$e(t) = \theta_{desired}(t) - \theta(t)$$

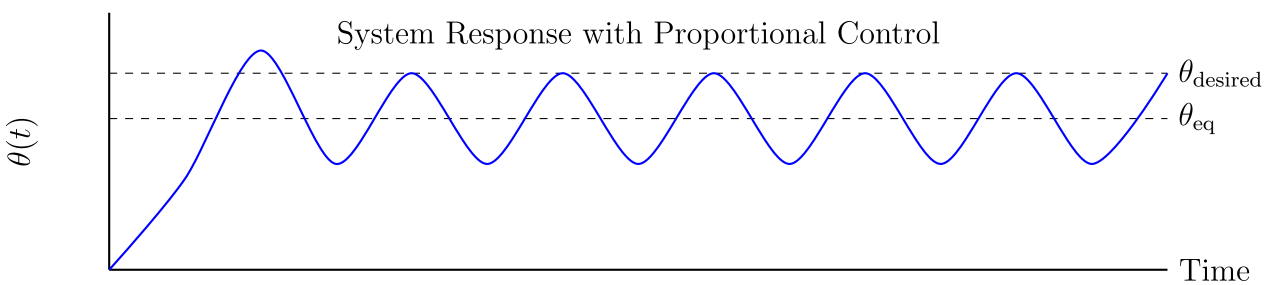
and computes the control input (a torque)

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt}$$

where K_p, K_i, and K_d are the proportional, integral, and derivative gains respectively.

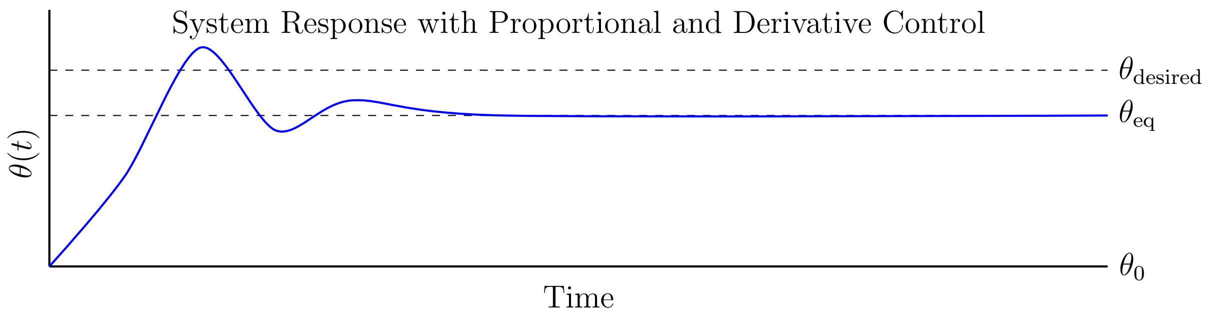
1. Suppose the telescope is commanded to rotate to a new angle using a step input. At first, the controller only uses proportional control ($K_p > 0$, $K_i = 0$, $K_d = 0$).

   Sketch the expected response of the telescope angle $\theta(t)$ over time. Note, there will be some steady-state error. [4 points]



2. Engineers notice that the telescope overshoots the target angle significantly when the proportional gains are large and decide to add a derivative term ($K_d > 0$).

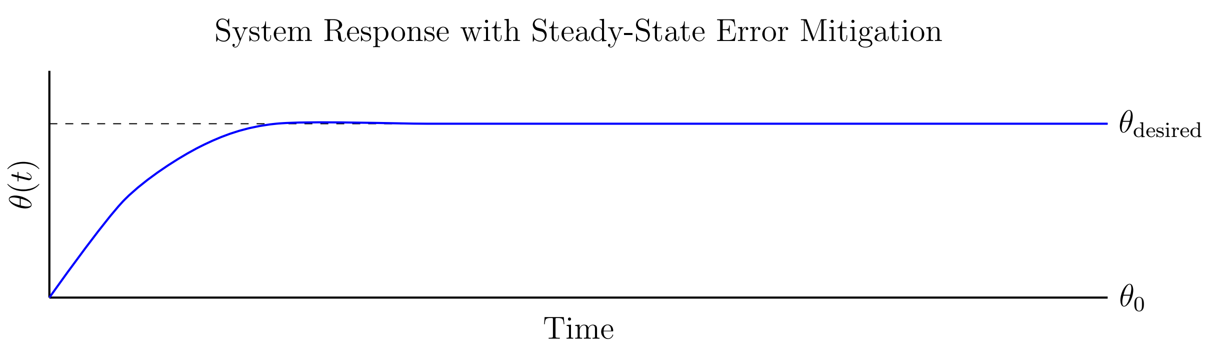
   Explain what effect the derivative term has on the response and briefly sketch the new response. [4 points]



   The derivative term adds damping because it reacts to how quickly the angle is changing. As the telescope approaches the target, the derivative component resists rapid motion, which reduces overshoot and oscillations. This makes the response smoother and helps the telescope settle more quickly near the equilibrium angle. In general, adding derivative control improves stability, but it does not remove steady-state error by itself.

3. The steady-state error is due to a constant external disturbance torque.

   Which term(s) in the PID controller help reject this disturbance and why? Briefly sketch the new response after adding your selected term(s). [4 points]



   The integral term (K_i) is responsible for rejecting constant disturbances. It accumulates the error over time, and as long as a steady-state error exists, the integral term continues to grow. This builds up additional control effort until the error is driven to zero. In contrast, the proportional term only reacts to the current error and cannot eliminate steady-state error on its own, while the derivative term responds to changes in error and has no effect once the system reaches steady state.

4. Suppose the derivative gain K_d is increased too much.

   Describe one drawback of very high derivative gain in a real robotic system with noisy sensors. [4 points]

   A very high derivative gain amplifies sensor noise because the derivative term depends on the rate of change of the measured signal. Noise causes rapid fluctuations in the measurement, which appear as large derivatives, leading to erratic and high-frequency control inputs. This can cause jittery actuator behavior, reduced stability, and potential wear or damage to the system. It also leads to under-damped control, where the system responds sluggishly.

5. The telescope must follow a predictable trajectory when tracking a moving planet. Engineers add a feedforward control term based on the desired motion.

   Explain the purpose of feedforward control and how it differs from feedback control. [4 points]

   Feedforward control uses a model of the system and the desired trajectory to predict the required control input in advance, allowing the system to follow the motion more accurately without waiting for error to occur. It improves tracking performance, especially for known, predictable motions.

   In contrast, feedback control reacts to the error between the desired and actual output and corrects deviations after they happen. Feedforward acts proactively based on the reference signal, while feedback acts reactively based on measured error. In practice, they are often combined to achieve both accurate tracking and robustness to disturbances.

3 Path Planning [25 points]

The following map below has a robot (green circle) and a goal marker:

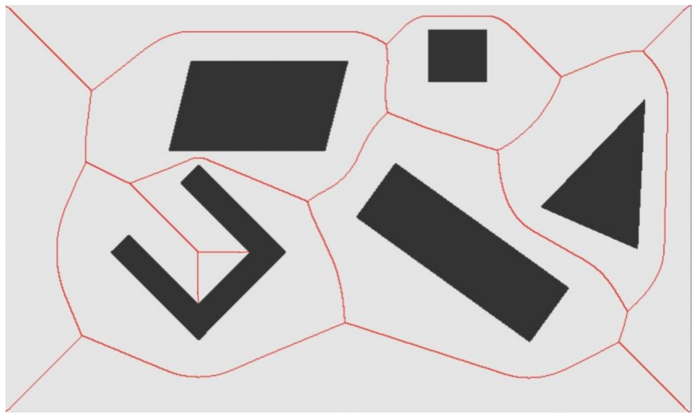


Figure 2: Map with start and goal positions

1. Your friend wants the robot to traverse this map to the goal using a potential function. Is this a good idea? Why or why not? [3 points]

   No, you using a potential function on the current map puts you at risk at being trapped at the local minima from the hollow open square block

2. Draw a Voronoi diagram using the map above [4 points]

   See solution image

3. Using path on your Voronoi diagram that connects the start and goal, comment on whether or not this path would be considered optimal. State your assumptions and explain your answer. [4 points]

   Full points earned as long as you gave a related metric to test optimality and gave a correct answer with proper explanation to go along with the metric (ex. optimal with respect to safety due to nature of voronoi keeping robot as far away from closest obstacles as possible or not optimal with respect to (L2) distance because turns around obstacles can be taken tighter for less distance

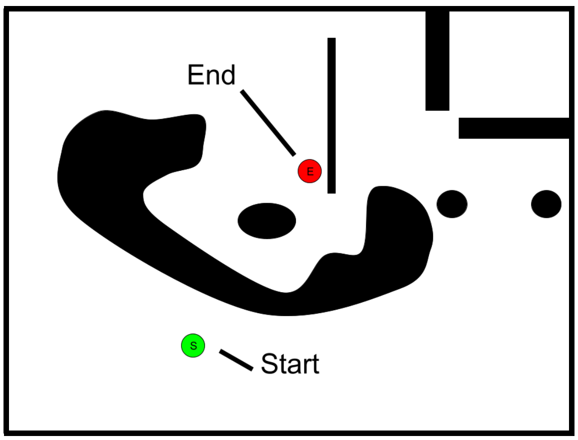
4. What is an alternative method to Voronoi that could perform better by some metric of your choosing? Explain your answer. [4 points]

   Full points earned as long as the alternative method is reasonably applicable to the scenario and a correct explanation to said method as to why it could be better (ex. using either wavefront or visibility graph to get better distance.)

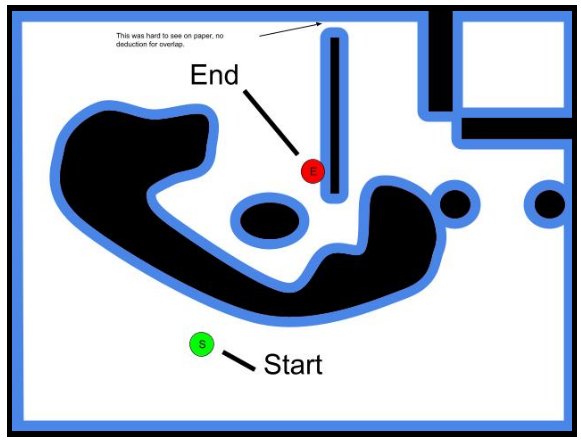
For this portion of the exam, we will be using the following map. Assume the "position" of the circular robot is defined by its center on the map and the size of the robot is proportional to the rest of the map (e.g. it cannot travel through to the upper right corner from its start position). The robot is 1 meter in diameter.

5. Indicate (draw) the configuration space and the free space of the robot for this map. Consider the entire map. Make it clear what is and is not included in each space. [4 points]

Free Space



Configuration Space

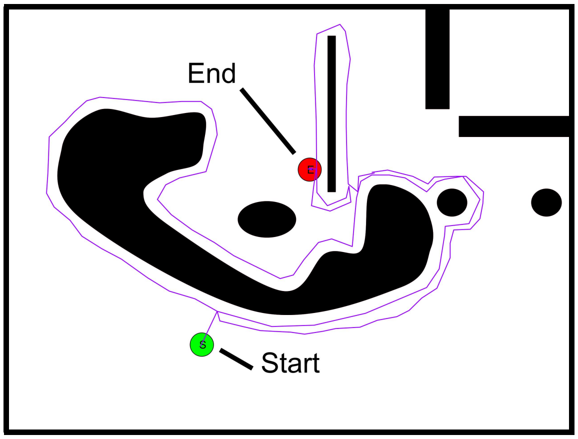


6. How does the configuration space compare to the free space of this map (i.e. is one a subset of the other)? [3 points]

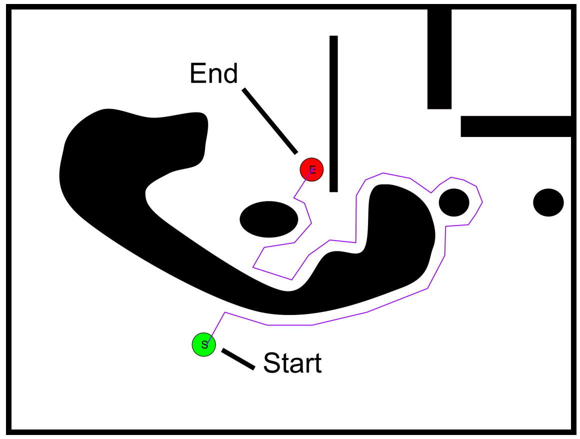
Full points earned as long as you didn't state anything clearly incorrect and mentioned that configuration space is a subset of the free space.

7. Draw the paths that would be found by the Bug 1 and Bug 2 algorithms **for this robot.** Assume both algorithms prefer to travel right. You should be drawing the path of the center of the robot according to your configuration space. [3 points]

Bug 1

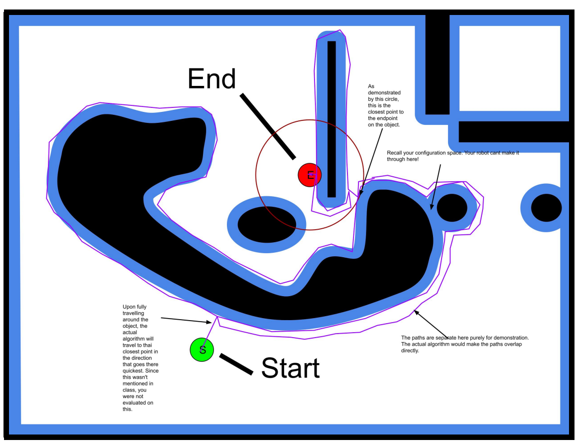


Bug 2

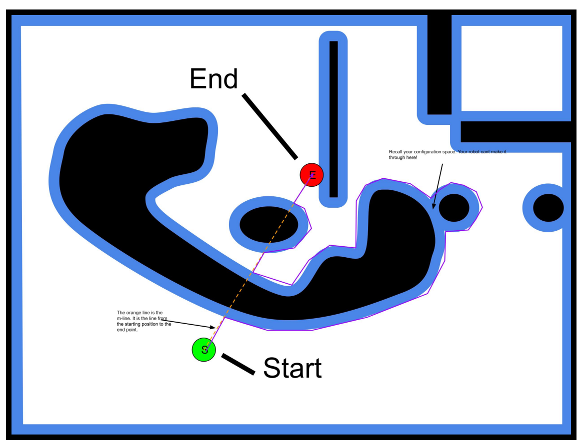


Explanation:

Bug 1



Bug 2



4 Graph Search [15 points]

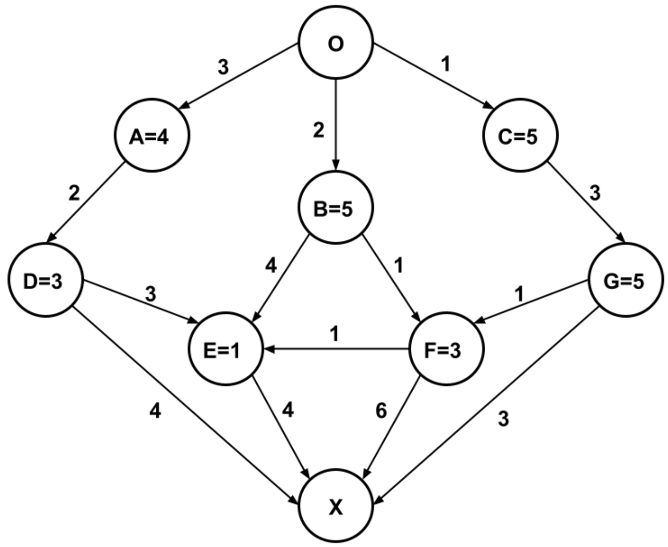


Figure 3: Graph for the A^* search

1. Perform an A^* search on the graph shown in Figure 4. O is the starting node, and X is the goal node. Break ties alphabetically. List the nodes that are EXPANDED and added to the closed list in the A^* algorithm above in the order in which they are expanded: [10 points]

   OCABFEDX

2. What is the path found by the A^* algorithm and how long is it? [5 points]

   OBFEX

3. Is the heuristic admissible? How is that related to finding the shortest path? [BONUS 5 points]

   No, the heuristic is not admissible. This is because **G is an overestimate of the cost from O to X.** Admissible heuristics are necessary to guarantee we have found the shortest path.

5 Localization [20 points]

You're on a spaceship many years into the future that has traveled to another universe in which the planets are stationary around its star. You begin your journey orbiting the star when suddenly you travel through a meteor field and damage your GPS (Galactic Positioning System) system. Now you don't know where you are! You must rely on odometry and a sensor that can detect the planets around you, and apply probabilistic localization methods (Bayesian) to determine your most likely position in the orbit.

At time $t-1$, the spaceship's position x is represented as a probability distribution $p(x_{t-1})$ using samples. When the spaceship moves at time t based on a control signal u_t, its new position follows an updated probability distribution $p(x_t)$.

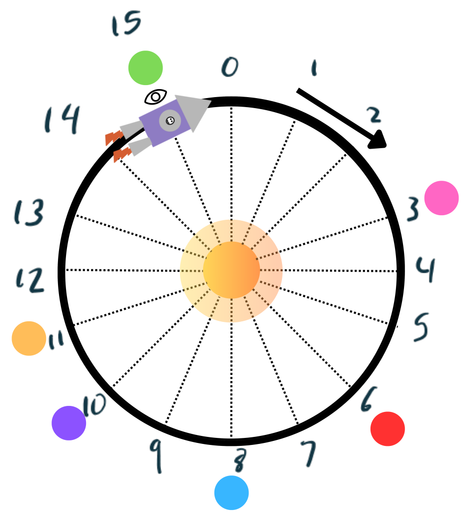
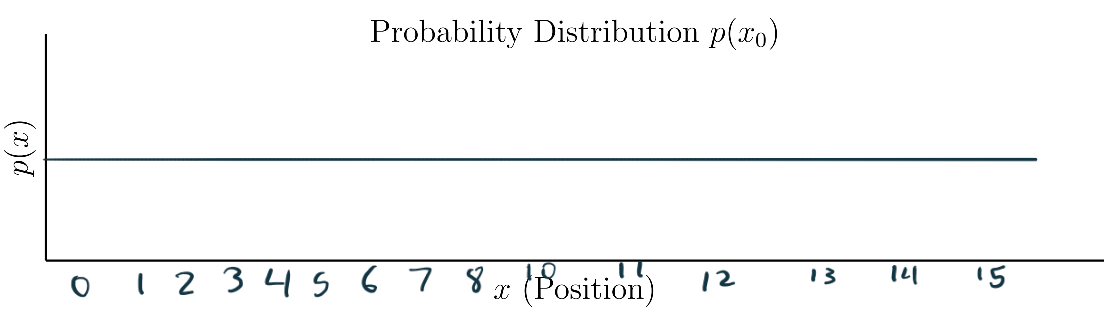
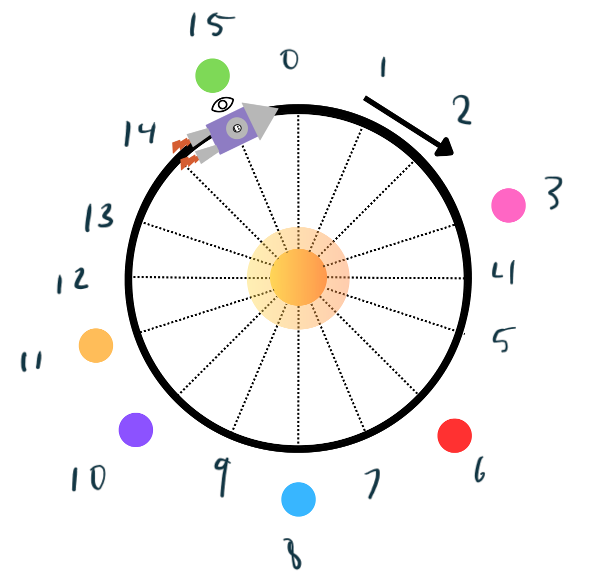


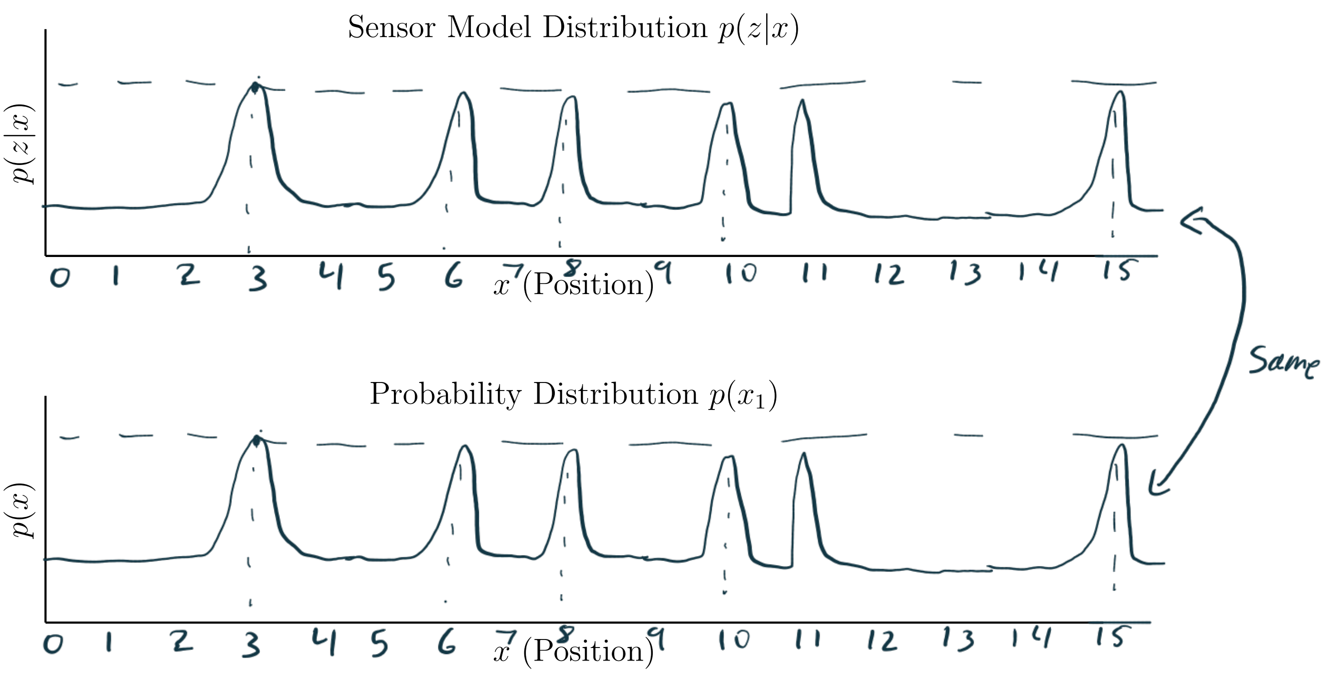
Figure 4: Orbit of our spaceship, *The Howie*

1. The spaceship is on a circular path around the star. Assuming that the spaceship can travel perfectly around this circular path and that we know the relative positioning of the planets, but not where the spaceship is on the circular path. **Sketch a continuous initial probability distribution** $p(x_0)$ **before any sensor readings.** [2 points]

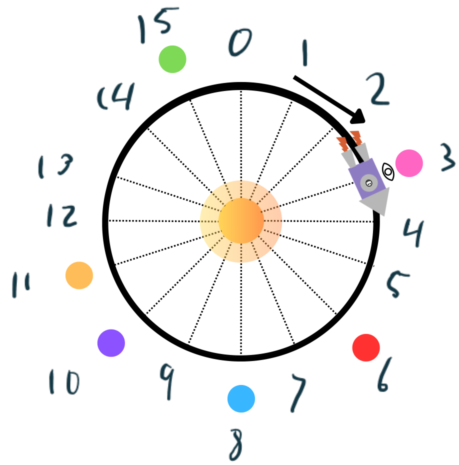


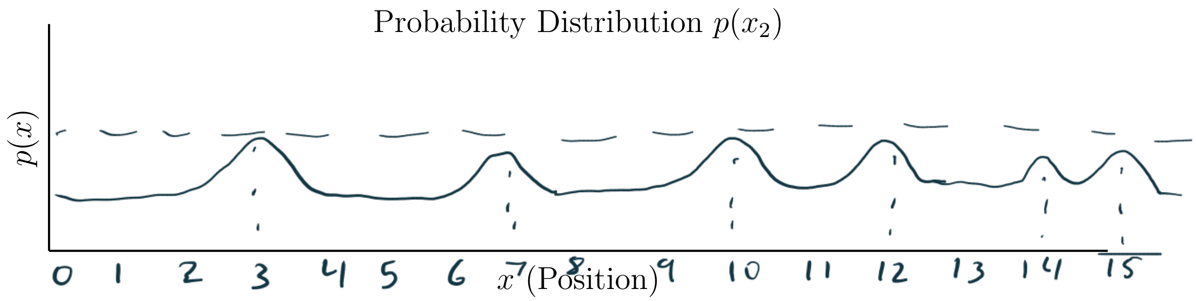
2. The spaceship takes an initial measurement and detects a planet in its surroundings! **Sketch both $p(z|x)$ (sensor model) and $p(x_1)$ (updated belief) after the measurement.** [6 points]





3. The spaceship moves forward around the orbit 4 sectors, as pictured. **Sketch the new continuous probability distribution $p(x_2)$ after the spaceship moves.** [4 points]



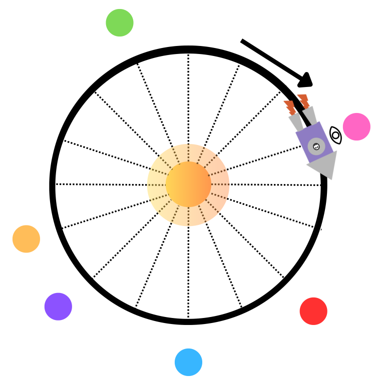


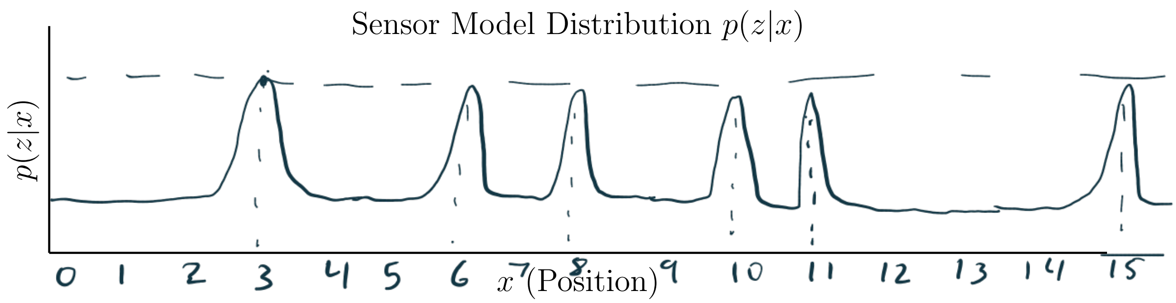
4. As you wait for the sensor to detect the next planet, you begin to think about other ways you could've localized. You remember from taking 16-281 that you can also represent probabilities discretely! If you used a particle filter, what would be one advantage and one drawback of increasing the amount of particles the algorithm uses? [4 points]

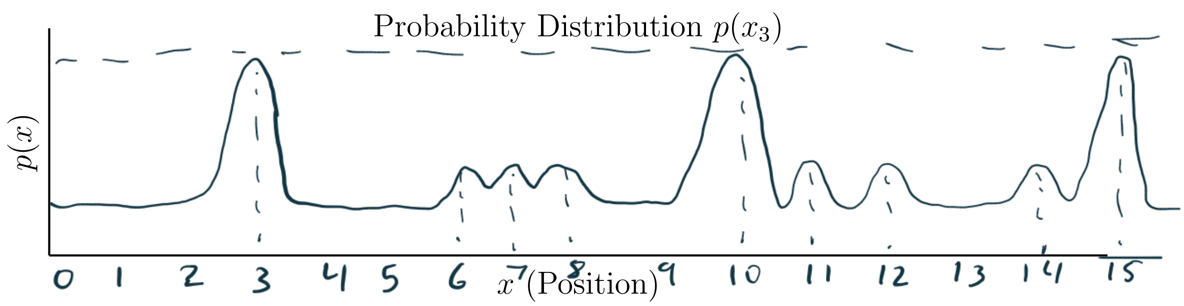
Adv: Localization will be more accurate.

Dis: More time spent computing

5. The spaceship takes another sensor measurement and detects another planet, helping refine its belief. **Sketch both $p(z|x)$ (sensor model) and $p(x_3)$ after incorporating the second sensor reading.** Did the spaceship localize to a single point? [4 points]







No.

PAGE INTENTIONALLY LEFT BLANK
This is the end of the test.