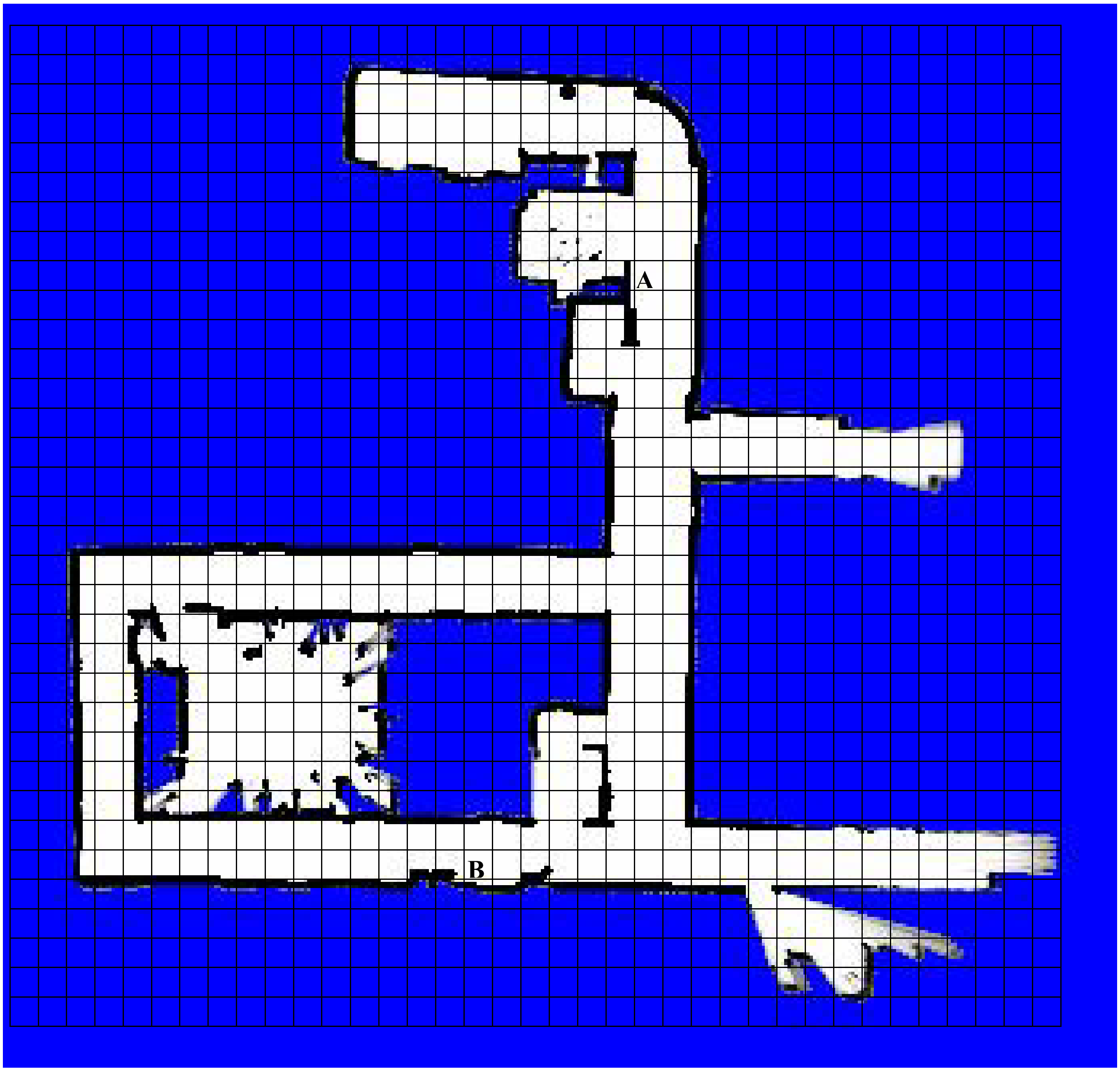
CS594: Artificial Intelligence, Fall 2004
Due: 12:00 noon, Friday, October 22, via email to parker@cs.utk.edu or delivered to Lynne Parker's office (Claxton 220) (slide under door if no one there).

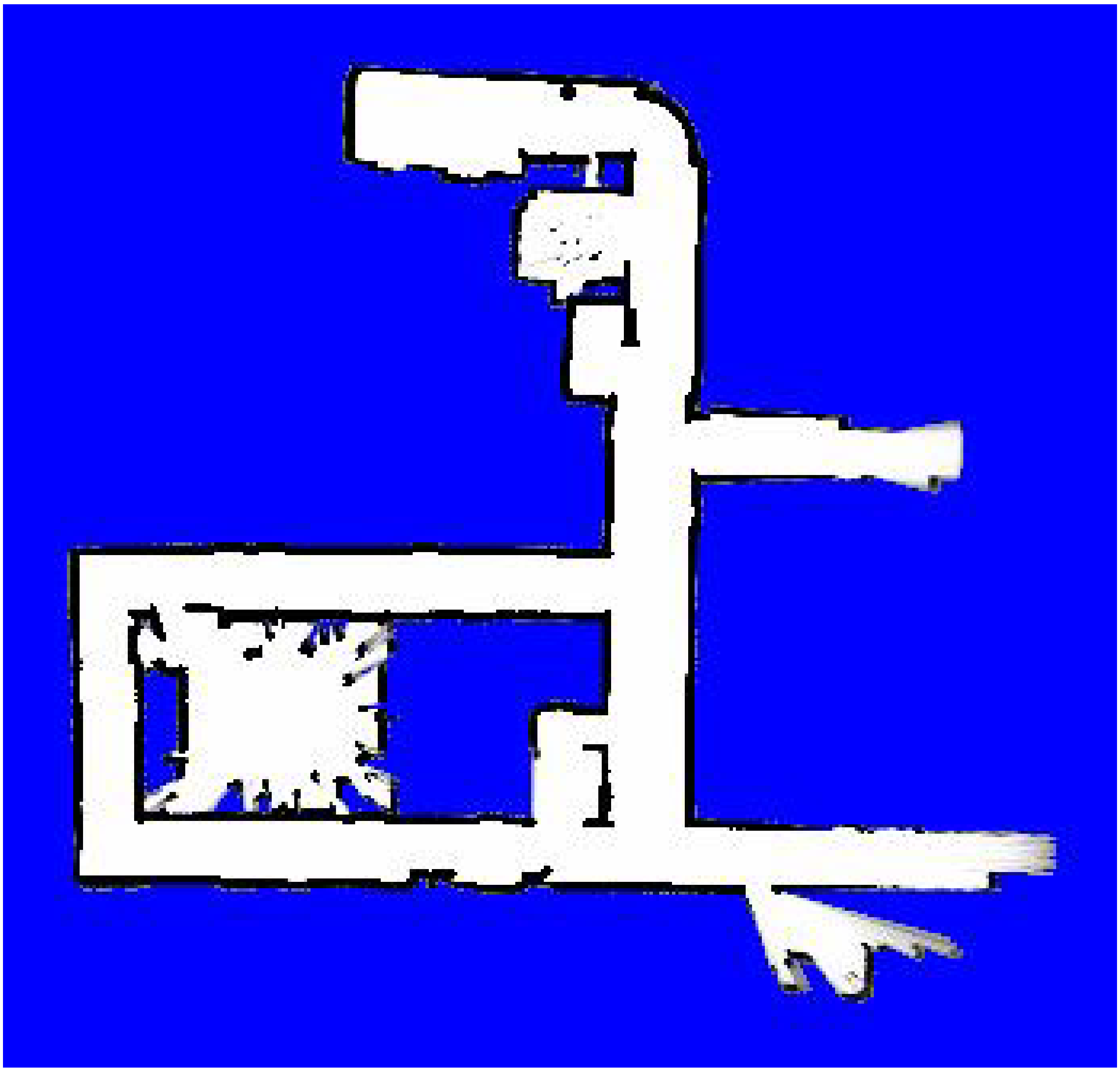
Homework 4

---

1. (Robotics) Below is a robot-generated metric map, divided into grids. Show the planning process of the dual wavefront path planner in planning a path from A to B by shading in the cells that are visited during the planning process. Clearly show the resulting path that would be generated from this planning process. Assume obstacles have already been grown.



2. (Perception and Robotics) Below is our robot-generated metric map again. Convert this metric map into a topological map, using the techniques discussed in class. (To simplify the problem, treat the two rooms that are cluttered as if the clutter was not present (i.e., as open rooms).) Show your intermediate Voronoi diagram, as well as the critical points and lines generated in the process of creating the topological map. Also show clearly the resulting topological map.



3. (Robotics) In this exercise, you are to specify the desired robot control action given a perceived input state. Here, your robot has a camera similar to the CMUCam, with an image of size r x c pixels (assume the origin, (0,0) is in the lower left corner of the image, with $+r$ to the right and $+c$ upwards). This sensor has a supplied function, called *Perceive*(Green), which returns a vector <*x, y, p*>, where *x, y* is the centroid of a Green color blob in the image, and p is the number of pixels of the Green color in the image. (If no Green is found in the image, the vector returned by Perceive is <-1, -1, 0>). Assume you have a Green target rectangle that appears to be of size m pixels when placed 1 meter in front of the CMUCam, and to have a centroid at <*r/2, c/2*>. Assume that there is no other Green color in the image other than your target. You want to control your robot so that it follows your moving Green target around the environment at a 1 meter distance. Your robot only moves forward, so if the target gets closer than 1 meter, your robot just has to stop (and perhaps turn in place). The control actions available to your robot are:

- *Move*(*steer,velocity*), where *steer* = {HardLeft, Left, Straight, Right, HardRight}, and *velocity* = {Stop, SlowForward, ModerateForward, FastForward}, which moves the robot according to the defined steering and velocity commands. The robot can turn in place, so a command of *Move*(*HardLeft,*0) will rotate the robot quickly to the left about its axis.
- *CallForHelp*, which stops the robot and communicates a message that it is lost.

Write the robot control specification in the form of a mapping from perceived states to robot actions. Also include a brief explanation of each mapping, so that the meanings of your defined states and actions are clear. For example, one such mapping is:

Perceived State	*Action*	*Explanation*
$x < 0, y < 0, p = 0$	*CallForHelp*	Robot has lost target; stop and ask for help.

To keep a manageable table size, you should define "equivalence classes" of states that share the same robot action, based upon ranges of values of *x, y,* and *p.*

4. (Uncertainty, Ch. 13) Work problem 13.6 on page 489 ("Given the full joint…").

5. (Uncertainty, Ch. 13) Work problem 13.8 on page 489 ("After your yearly checkup…").

6. (Uncertainty, Ch. 13) Work problem 13.15 on page 491 ("Suppose you are a witness…").

7. (Uncertainty, Ch. 13) Using the techniques of section 13.7, show the calculation of $\mathbf{P}(P_{2,2} \mid known, b)$. Be sure it is clear how you have obtained your answer.

8. (Uncertainty, Ch. 13) Work problem 13.19 on page 491 ("In our analysis…").