INTELLIGENT SYSTEMS QUALIFIER

Spring 2014

Each IS student has two specialty areas. Answer all 3 questions in each of your specialty areas.

You will be assigned an identifying number and are required to hand in printed or written copies of your answers with each page identified only by that number. This process enables us to grade your answers anonymously. You should NOT identify yourself explicitly by name on your answer sheets or implicitly by referring in the first person to your work (my project on ABC).

Please answer each question starting on a new page, with your answer following the text of the question.

Place any relevant references that you cite in an answer at the end of that answer, NOT in a separate section at the end of the whole exam.

If you have any questions or feel it necessary to make any assumptions in your answers, do not seek clarification from faculty or staff. Simply record your assumptions as part of your answer.

If one of your areas is **Perception**, answer the three questions below:

Perception #1

For mapping and localization we can use Kalman Filters, Particle Filters and Factor Graphs.

a) What is the main advantage and disadvantage of each of these three techniques?

b) What is the computational characteristics of each technique?

c) When would you use each of them?

Perception #2

a) Discuss the notions of primary and secondary sparsity structure in the bundle adjustment problem.

b) Give examples of (realistic) best case and worst case camera network in the light of your definitions in a)

c) While these notions are not used in the MonoSLAM work by Davison et al., explain the effect of the camera motion on the secondary structure, and the consequences on computation.

Perception #3

A wealthy friend of yours is considering buying an antique home and wants your help in determining its value. The center piece of the home is a dining room which contains a rug, whose pattern resembles a Mondrian painting (for example, http://en.wikipedia.org/wiki/File:Mondrian_Composition_II_in_Red,_Blue,_and_Yellow.jpg), that is claimed to be antique and valuable. If you can reconstruct an image of the rug showing its true colors, your friend can easily verify its authenticity. The single floor-to-ceiling window of the dining room is made of large pieces of stained (colored) glass, and receives direct sunlight. You are free to take pictures from any location (but only during the day, and flash photography is forbidden).

a) Describe an algorithm for reconstructing the colors of the rug using photographs taken from inside the room. (You may make any reasonable simplifying assumptions about the daylight illumination).

b) The room also contains a valuable sculpture, which consists of a large smoothly-varying surface with constant albedo. Your friend would like to obtain a 3-D reconstruction of the shape of the sculpture in order to ascertain its value. Suppose you are allowed to capture time-lapse photography, can you think of a way to reconstruct the shape (e.g. in the form of a set of surface normals) from a series of photographs? You may make any reasonable simplifying assumption about the illuminant.

If one of your areas is **Planning & Search**, answer the three questions below:

Planning & Search #1

1) A* is a commonly used algorithm for pathfinding in fully-observable environments. D* and D*-lite can be used to perform pathfinding in partially-observable environments. A* can also be used in partially-observable environments by interleaving planning and execution.

a) Consider the performance metric that sums the number of state space expansions across all periods of planning between the time an agent starts to build its first plan and the time it completes execution to the goal. Explain how D*-lite is able to reduce this planning time metric over that of an agent that interleaves A* and execution.

b) Lifelong planning algorithms (LPA*, D*, and D*-lite) are regularly used in mobile robots. But despite the apparent benefits of reducing overall planning time, lifelong planning algorithms are considered inappropriate for computer games such as real-time strategy (RTS) games that involve many dozens or hundreds of units pathfinding simultaneously. This is the case even where there is dynamically deformable terrain in the game world (i.e., previously known paths can be blocked and new paths can be opened up). Explain why lifelong planning algorithms are not suitable for computer games with dozens or hundreds of agents. (Hint: think about the trade-offs necessary to achieve planning time reduction).

Planning & Search #2

The informal analysis of human planning by Hayes-Roth and Hayes-Roth suggests a “heterarchical” approach to planning, although they acknowledge other problems seem to be solved more hierarchically. Other studies conducted later suggest that humans also solve problems using means-ends analysis (working backward from a goal, satisfying the necessary conditions of each step, such as in UCPOP). Give arguments for and against “heterarchical”, hierarchical, and means-ends based approaches to planning. In your analysis, include the features of the problem that one might use to select one approach over another.

(Planning & Search continued next page)

Planning & Search #3

Consider a robot navigation problem. Assume: The robot must navigate across a 2D plane, that it has some form of accurate localization and heading detection, that it has an obstacle sensor with a limited range. The robot is provided the destination it should navigate to, but the robot does NOT have a map of the environment when it starts.

You are to design TWO systems for solving this problem: One that uses a planner (like A*) and the other using Reinforcement Learning. In both cases you should devise a solution that maximizes the value of information it gains from moving and sensing, and minimizes required interactions with the environment.

Explain any assumptions you make.

For each method, explain:

a) Which algorithm will you use?

b) How will you represent the task for the robot? What are the data structures?

Explain, for each method:

c) How are the data structures initialized?

d) What happens during each sensing, “thinking”, acting loop?

e) What happens when a previously unknown obstacle is encountered?

Overall, and this is the most important question:

f) Are planning and RL equivalent? Why or why not?

If one of your areas is **Machine Learning**, answer the three questions below:

Machine Learning #1

1) Gibbs Sampling is a widely-applicable technique for Bayesian inference in graphical models.

a) Briefly summarize Gibbs Sampling, explaining when it can be applied, what is needed to apply it, and what guarantees (if any) are available.

b) When would you apply an alternative such as slice sampling or Metropolis-Hastings instead of Gibbs sampling?

c) Consider the problem of inducing a discrete hidden Markov model, given data generated from a model with very strong probabilistic dependencies between adjacent hidden states. Why might Gibbs sampling perform poorly in this model? What alternative sampling algorithm might do better?

Machine Learning #2

(Fully Observable) Markov Decision Problems

Imagine that you're trying to build an autonomous car that will go around a race track. The actions are some version of going forward, backward, turning, applying power. The usual sort of thing. The problem here is that you want the car to start in the, um, starting position and end in the same position: In other words, something like:

```
A-----B
|     |
|     |
D-----C
```

where the task is to go A -> B -> C -> D -> A.

One can conceive, I hope, of a simple, straightforward MDP design where A is set as a goal with an appropriate reward function for entering A, but then the episode will end as soon as it starts, and the agent will learn nothing. In fact, it is even worse than that: even with a smarter reward function, if the state space is just the position of the agent then the problem becomes a POMDP and that is not easy to solve.

So, how would you design the state space and the reward functions such that the problem is solvable and also does not become a POMDP? Justify your decisions carefully and argue that one of the standard RL algorithms (please say which one) will be able to solve it in reasonable time.

(Machine Learning continued next page)

Machine Learning #3

Many recommender systems, such as for music, movies, etc., use feature vectors with dimensions for every item in their library. The user's past watching/listening history are entered into these vectors, resulting in a very sparse vector.

a) Describe techniques for dealing with this sparsity in trying to predict which item the user might be interesting in watching/listening to next.

b) The user's tastes might change over time. In fact, you suspect that taste changes are predictable and follow a pattern. How would you use a large database of user history from a recommender engine to test your hypothesis?

c) Suppose your hypothesis holds true. What type of model might you use to provide better recommendations for a user which takes into account taste changing over time?