CSCI 2400: Artificial Intelligence

Midterm Study Guide
Instructor: David Byrd

This list is a best effort. It is not guaranteed to be exhaustive. The exam can cover anything we have discussed in class, or covered through problem sets or programming projects.

1 Agents and Environments

1.1 Conceptual Material

1. Types of AI problems: thinking/acting human/rationally
2. Strong AI vs Weak AI
3. What is Machine Learning?
4. What's an agent? What are the minimum requirements?
5. Terminology: percepts, percept sequence, agent function, agent program
6. What is a *rational* agent?
7. What's a performance metric?
8. Why can't agents usually achieve theoretically-optimal performance?
9. Agent hierarchy: what are reflex, model-based, goal-based, and utility agents?
10. Environmental classifications: observability, determinism, dynamicism, discretism, episodic, # agents, etc.

1.2 Potential In Depth Questions

Given a simple problem:

1. Draw the agent function table
2. Define the needed sensors, actions, goal conditions, transitions
3. Propose a reasonable performance metric
4. Draw the state space and transitions (visually)

2 Search

2.1 Conceptual Material

1. For what types of environments can we use planning algorithms?
2. Define the search problem: (S, S_0, S_F, A, T)
3. State the general problem that search solves (in English)
4. Understand *state space* versus a physical location or map
5. What is the result or output of a completed search function?
6. Successor function vs exhaustive T matrix
7. isGoal function vs exhaustive S_F set
8. Simple search evaluation criteria: completeness, optimality
9. Steps involved in a planning algorithm
10. Meaning of informed vs uninformed search
11. Admissibility of a heuristic
12. Dominance of a heuristic
13. Informedness of a heuristic
14. Consistency of a heuristic
15. Losing which of the above breaks answer-path optimality of A*?

2.2 Potential In Depth Questions

Given a simple problem:

1. Design the search problem representation (states, actions, etc)
2. Run one of the following algorithms "on paper": DFS, BFS, UCS, BestFS, greedyBestFS, A*
3. Modify the search space to help/hurt the above algorithms
4. Explain anything/everything about the above algorithms, plus Random Search
5. State (and defend) why the above algorithms would or wouldn't be a good fit for the problem
6. Contrast the benefits of different search algorithms

3 Constraint Satisfaction Problems

3.1 Conceptual Material

1. How are CSP different from the prior search problems?
2. How does that difference help solve them faster?
3. What is necessary to fully define a CSP?
4. What is the process of solving a CSP?
5. What is a consistent assignment?
6. What is a complete assignment?
7. What is a CSP "solution", precisely?
8. *When* is it appropriate to use CSP vs some other approach?
9. satisfiesConstraints function vs exhaustive C
10. Formulation, meaning, purpose of constraint graph
11. Constraint types

3.2 Potential In Depth Questions

Given a simple problem, puzzle, or game:

1. Formally formulate it as a CSP and solve it
2. Given the CSP version, explain it in simple real-world terms
3. Run one of the following algorithms (or pieces) "on paper":
   - backtracking DFS
   - variable ordering (and heuristics)
   - value ordering (and heuristics)
   - failing early via forward checking
   - failing early via MAC
4. Explain in detail any of the above algorithms or optimizations and understand all about how they work and *why* they help

4 Logical Agents

4.1 Conceptual Material

1. Why are logical agents special (vs previous agents)?
2. How do they work? What pieces do they need?
3. What is propositional logic? Atoms? Operations? Constants? Formulae?
4. What is the process by which a logical agent works?
5. What does "coercing the environment" mean?
6. What are the limitations of a propositional logic agent?

4.2 Potential In Depth Questions

Given a simple puzzle, problem, or game:

1. Describe, create, or translate between English meaning and propositional logic descriptions of the problem, world, knowledge, etc
2. Given some atom definitions, English rules, etc, express those rules as propositional logic statements
3. Trace the operation, actions, and deductions of a propositional logic agent

5 Probability, Inference

5.1 Conceptual Material

1. Basic terms: universe, random variable, event, probability
2. Basic rules: calculating $P(a \vee b)$ or $P(\neg a)$, given $P(a), P(b), P(a \wedge b)$, etc.
3. Full joint distribution - what is it? Use it to calculate things
4. Marginalization (or conditioning) - what is it? Use it to calculate things
5. Conditional probability - what is it? Use it to calculate things
6. Normalization constant - how to calculate it? What is it for? Why?
7. Product rule
8. Chain rule
9. Total Probability ("Conditioning") rule

10. What the above "rules" let you do in practical terms
11. Bayes Rule - why are the unconditional probabilities where they are?
12. Independence of R.V.
13. Conditional Independence of R.V.
14. Definition, significance, and uses of Indep. and Cond. Indep.

5.2 Potential In Depth Questions

1. Given some probabilities, calculate related probabilities (like our diagnostic test example)
2. Solve simple problems involving sequences of random events, independent or dependent
3. Apply probability knowledge to solve simple agent problems or puzzles (like the wumpus agent and the pits)

6 Bayes Nets

6.1 Conceptual Material

1. Requires all the basic probability stuff (see Exam 1 guide)
2. Generalized Bayes model: define it and understand it
3. Naive Bayes assumption: what is it and *why* do we make it?
4. Bayes Net: purpose, use, structure, meaning
5. Equation for joint probability in a Bayes Net
6. Definition and significance of Markov blanket

6.2 Potential In Depth Questions

1. Various probability calculations but Bayes-flavored
2. Bayes Nets: create one, use it to solve a problem, explain what it means/shows, reason about it, modify it
3. Understand, explain, use conditional probability tables

7 Dynamic Bayes Nets, Exact Filtering

7.1 Conceptual Material

1. What problem do DBNs solve that previous agent types did not?
2. When and why would we need to use a DBN?
3. How is time represented in a DBN?
4. What are the key assumptions required for a DBN? What do they mean?
5. What is the minimum representation needed to fully specify a DBN?
6. What is a transition model? A sensor model? What do they do?
7. What is a Bayesian Prior and why do you need it?
8. What are posterior probabilities?
9. What are the four main problems we can solve in the context of a DBN?
10. What is the filtering problem? Define it precisely.
11. Give the iterative steps involved in exact filtering.
12. What's the problem with exact filtering?
13. What is the difference between filtering and prediction?

7.2 Potential In Depth Questions

1. Given problem information, construct a complete, valid DBN to use in solving it.
2. Given a DBN specification, explain the underlying problem and what it represents.
3. Run some steps of the exact filtering algorithm on a specific problem.