CMPSCI 683 **Fall 1998**
FINAL EXAM **Prof. Lesser**

---

1) Agent Architectures/Real-Time (15 points)

a) Sketch out the extensions to PRS that would be needed to introduce reasoning about hard real-time deadlines. If you had available anytime algorithms or a design-to-time criterion scheduler how could these capabilities be exploited in your extended PRS? Hint: if you are going to do real-time you need to have declarative constructs that define the performance characteristics of methods.

b) Explain how Guardian exploits a separate preprocessor of sensory data to focus computational resources on the key problems that need to be solved, while still permitting important conditions to be responded to quickly.

c) Do you see any differences in the subsumption architecture of Brooks and the layered hierarchy of modules described in the Question 3 on learning in the robotic soccer application?

---

2) Multi-Agent Systems (15 points)

Remember the crossword puzzle problem in the homework. Suppose we cut the puzzle right down the middle on letter boundaries and assigned each half of the puzzle to a different agent. What are the issues involved in the agents working together to generate an overall solution that meets all the constraints?

a) What type of subproblem interactions will there be between agents?

b) What type of coordination between agents would be worthwhile to solve the interacting subproblems?

c) What type of changes (if any) would you need to make to your local agent problem-solving strategy in order to coordinate effectively with the other agent?

---

3) General Learning (18 points)

I just read a Ph.D. thesis on the use of learning in the application of robotic soccer. This is an interesting application because it involves teams of opposing players, the rewards of players' actions are not immediate, the state of all the players is not known to an individual player, the sensing of the environment is noisy and the result of a players action such as kicking are not deterministic.

In order to develop robotic soccer players, the thesis used a learning approach to the development of a player. Instead of trying to perform learning on the entire activities of a player, learned was layered. The learned behavior at one level was used as the basis for learning more complicated behaviors. Learning was used at three levels:

1) ball-interception (how to capture a moving ball) -- one player
2) pass evaluation (deciding on the likely success) -- one-one-player
3) pass selection (deciding who to pass to) – one-to-many player

a) Explain the potential benefits of this layered learning approach against learning a direct mapping from sensory data to action.

b) On the third level, a reinforcement learning algorithm was used. Instead of having a reward only when a goal occurred, intermediate rewards were introduced such as when corner kicks occurred, kicks on goals, sustained movement towards the goal, etc. Explain the need to introduce such intermediate rewards. Do you see any drawbacks to these intermediate rewards?

c) On the first level, a neural network was used for learning the local decision of how best to accomplish ball interception. In contrast, a decision tree algorithm was used for learning at level 2 where the number of variables involved in the decision was much larger. What reasons would you give for each of these choices about which learning algorithm to use?

d) For learning at level 1 and 2, is instance-based learning appropriate? Explain your answer.

---

4) Decision Trees (18 points)

A feature of a decision tree algorithm that we have not discussed in class is its ability to associate a confidence factor with its classification decision: the probability of a decision being correct. This probability is computed based on the training data and the specific leaf node that is used to make the decision.

a) Explain how this probability is computed;

b) Can a similar approach be used with a neural network?

One way of handling continuous attributes in a decision tree is to automatically generate a characteristic function such as X less than or equal to 4.5 as part of the decision tree learning algorithm

c) How could you use information entropy measures to decide on an appropriate characterization?
d) How can this approach be extended to multiple splits at a node (X less than or equal to 4.5, X>4.5 and X<7.1, X greater than or equal to 7.1)
e) What do you see as the potential advantages/disadvantages of introducing multiple splits?

5) Genetic Algorithms (10 points)

a) How can you use a genetic algorithm to learn a decision tree? What would be the fitness measure you would use? How would the mutation and crossover operators be applied to the decision tree? (Please show a simple example.)

b) Would this be a good approach for learning a decision tree versus a more conventional approach? Explain your answer.

---

6) Neural Networks (10 points)

a) How and why are neural networks used as part of a reinforcement learning system?

b) How does the number of hidden nodes in a neural network affect its expressability, robustness and learning time?

---

7) Reinforcement Learning (14 points) [See attached]