Computer Science 576: HW2

Handed out: Sept. 24, 2002. Due: Oct. 8, 2002.

1. Show that Sen's axioms α continues to hold of $\succ$ is a partial order, but show by example that Sen's β may fail to hold.

2. GRADUATE: Find a variant of Sen's β that, together with α, characterizes those $C(B)$ which are a $C(B, \succ)$ for some partial order $\succ$.

3. Let $\succ$ be a partial order on a denumerable set X. Define $\succeq$ and $\sim$ in the usual way. Define $x \approx y$ if for all z, $x \sim z$ iff $y \sim z$. Show

   (a) $\approx$ is an equivalence relation.

   (b) If $w \approx x$, $x \succ y$, and $y \approx z$, then $w \succ y$ and $x \succ z$.

   (c) There is a function $U : X \to I\!R$ such that if $x \succ y$, then $U(x) > U(y)$ and $x \approx y$ iff $U(x) = U(y)$.

   Does this still hold true if $\succ$ is only acyclic rather than transitive?

4. Show that the lexicographic order is in fact a preference order.

5. Consider the utility function $U(z_1, z_2) = \min(z_1, z_2)$, which in the uncertainty context gives rise to the *maximin criterion.* Which of the assumptions in Theorem 7 in Prof. Blume's notes does its ordering violate?

6. Prove the following proposition from the class notes:

   Proposition: Let f be an increasing function. Then $\text{maximin}(u) = \text{maximin}(f(u))$ and $\text{maximax}(u) = \text{maximax}(f(u))$. Show by example that $opt^\alpha(u)$ may not be the same as $opt^\alpha(f(u))$, and that $regret(u)$ may not be the same as $regret(f(u))$.

7. Prove the following proposition from the class notes:

   Proposition: Let f be a positive affine transformation. Then

   - $\text{maximin}(u) = \text{maximin}(f(u))$

- $\text{maximax}(u) = \text{maximax}(f(u))$
- $opt^\alpha(u) = opt^\alpha(f(u))$
- $regret(u) = regret(f(u))$.

8. Recall the algorithm for the location problem: The robot goes to $+1$, then -2, then $+4$, then -8, $\ldots$, until it finds the object.

    (a) Show that this algorithm has a competitive ratio of 9. That is, show that compared that, no matter where the object is located, it will take the robot at most 9 times as many steps to find the object using this algorithm as it would to go directly to the object.

    (b) Show that if $k < 9$, then for all c, there is a location N_c for the object such that the robot will take more than $kN_c + c$ steps to find the object.

9. GRADUATE: Show that no algorithm for the robot location problem has competitive ratio better than 3. (Hint: first show that, without loss of generality, there exist two sequences $N_1, N_2, \ldots$ and $N_1', N_2', \ldots$ such that $0 < N_1 < N_2 < \ldots$ and $0 < N_1' < N_2' < \ldots$ such that the robot goes to $+N_1, -N_1', +N_2, -N_2', \ldots$, until it finds the object.)

10. Consider the ski rental problem from the notes. Recall that you can rent skis for $\$r$/day, or purchase them for $\$p$, and you'll be using the skis for at most N days. Assume that whether or not you buy skis does not affect how many days you ski. (Note: this is almost certainly false for many people. Once they buy skis, they view it as a commitment and are less likely to quit skiing.) With this assumption, model the ski rental problem as a decision problem.

    (a) Which act(s) are optimal according to the maximin rule?

    (b) Which act(s) are optimal acording to the minimax regret rule?

    (c) Which act(s) give the optimal competitive ratio?

    (d) Explain clearly why this assumption is necessary in your analysis.

    Note: the act(s) that are optimal depend on the values of the parameters p, r, and N. (Actually, all that matters is p/r and N.)