ORIE 6300 Mathematical Programming I October 10, 2008

Problem Set 6

Due Date: October 17, 2008

1. When I was a researcher at IBM, I once needed to solve several linear programs that had a very large number of constraints relative to the number of variables. I was trying to solve $\min c^T x : Ax \leq b, x \geq 0$, for $A \in \Re^{m \times n}$, where m was exponential in n. To get this into standard form, add slack variables s for each constraint, so that the problem becomes $\min c^T x : Ax + s = b, x \geq 0, s \geq 0$. The running time for solving each LP was several hours. I wandered down the hall and asked John Forrest, the author of IBM's linear programming code (OSL), if there was anything I could do to speed up the solution time. He suggested a simple reformulation of my problem that dropped the running time to under 20 minutes. Try to explain a change such that the simplex method as we have described it (running on the primal in standard form) might run much more quickly in such a situation. (Hint: think about the complexity of a pivot operation).

2. (Chvatal 7.12) Prove that every $m \times m$ non-singular matrix B has an eta factorization consisting of at most m matrices, $B = E_1 \cdots E_k$, with $k \leq m$, where each E_i is an eta matrix with the columns potentially permuted. Give an algorithm to find such a factorization. What is the order of the worst-case running time of your algorithm (as a function of m)? Hint: show by induction on j that one can find j eta matrices whose product agrees with B in the first j columns.