HW 3

DREW MORRIS

Problem 1. *Compare the performance of the largest-coefficient and smallest-index pivoting rules on the following linear problem.*

$$\begin{array}{|ccccc|}\hline 3x_0 & + & 5x_1 & = & \xi \\ \hline x_0 & + & 2x_1 & \leq & 5 \\ x_0 & + & 0x_1 & \leq & 3 \\ 0x_0 & + & x_1 & \leq & 2 \\ x_0 & , & x_1 & \geq & 0 \\ \hline \end{array}$$

Proof. The smallest-index pivoting is more efficient for this problem. Observe.

(1) We perform simplex using largest-coefficient pivoting.

$$\begin{array}{|ccccccc|}\hline \xi & = & 0 & + & 3x_0 & + & 5x_1 \\ \hline y_0 & = & 5 & - & x_0 & - & 2x_1 \\ y_1 & = & 3 & - & x_0 & + & 0x_1 \\ y_2 & = & 2 & + & 0x_0 & - & x_1 \\ \hline \end{array} \quad \rightarrow \quad x_1 = 2 \quad \rightarrow$$

$$\begin{array}{|ccccccc|}\hline \xi & = & 10 & + & 3x_0 & - & 5y_2 \\ \hline y_0 & = & 1 & - & x_0 & + & 2y_2 \\ y_1 & = & 3 & - & x_0 & + & 0y_2 \\ x_1 & = & 2 & + & 0x_0 & - & y_2 \\ \hline \end{array} \quad \rightarrow \quad x_0 = 1 \quad \rightarrow$$

$$\begin{array}{|ccccccc|}\hline \xi & = & 13 & - & 3y_0 & + & y_2 \\ \hline x_0 & = & 1 & - & y_0 & + & 2y_2 \\ y_2 & = & 2 & + & y_0 & - & 2y_2 \\ x_1 & = & 2 & + & 0y_0 & - & y_2 \\ \hline \end{array} \quad \rightarrow \quad y_2 = 1 \quad \rightarrow$$

Date: September 25th 2023.

$$\begin{array}{|rcrcrcr|}\hline \xi & = & 14 & - & 2y_0 & - & \frac{1}{2}y_2 \\ \hline x_0 & = & 3 & + & y_0 & - & y_1 \\ y_2 & = & 1 & + & y_0 & - & \frac{1}{2}y_1 \\ x_1 & = & 1 & - & y_0 & + & \frac{1}{2}y_1 \\ \hline \end{array} \quad \rightarrow \quad \xi = 14$$

(2) We perform simplex using smallest-index pivoting.

$$\begin{array}{|rcrcrcr|}\hline \xi & = & 0 & + & 3x_0 & + & 5x_1 \\ \hline y_0 & = & 5 & - & x_0 & - & 2x_1 \\ y_1 & = & 3 & - & x_0 & + & 0x_1 \\ y_2 & = & 2 & + & 0x_0 & - & x_1 \\ \hline \end{array} \quad \rightarrow \quad x_0 = 3 \quad \rightarrow$$

$$\begin{array}{|rcrcrcr|}\hline \xi & = & 9 & - & 3y_1 & + & 5x_1 \\ \hline y_0 & = & 2 & + & y_1 & - & 2x_1 \\ x_0 & = & 3 & - & y_1 & + & 0x_1 \\ y_2 & = & 2 & + & 0y_1 & - & x_1 \\ \hline \end{array} \quad \rightarrow \quad x_1 = 1 \quad \rightarrow$$

$$\begin{array}{|rcrcrcr|}\hline \xi & = & 14 & - & \frac{1}{2}y_1 & - & 5y_0 \\ \hline x_1 & = & 1 & + & \frac{1}{2}y_1 & - & y_0 \\ x_0 & = & 3 & - & y_1 & + & 0y_0 \\ y_2 & = & 1 & - & \frac{1}{2}y_1 & + & y_0 \\ \hline \end{array} \quad \rightarrow \quad \xi = 14$$

We can see that the largest-coefficient pivoting takes 4 dictionaries to find the optimal answer whereas the smallest-index pivoting takes only 3. □

Problem 2. *Solve the Klee-Minty problem for $n = 3$.*

Proof. We wish the solve the following problem.

$$\begin{array}{|rcrcrcc|}\hline 4x_0 & + & 2x_1 & + & x_2 & = & \xi \\ \hline x_0 & + & 0x_1 & + & 0x_2 & \leq & 1 \\ 4x_0 & + & x_1 & + & 0x_2 & \leq & 100 \\ 8x_0 & + & 4x_1 & + & x_2 & \leq & 10000 \\ \hline \end{array}$$

Observe.

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 0 & + & 4x_0 & + & 2x_1 & + & x_2 \\ \hline y_0 & = & 1 & - & x_0 & + & 0x_1 & + & 0x_2 \\ y_1 & = & 100 & - & 4x_0 & - & x_1 & + & 0x_2 \\ y_2 & = & 10000 & - & 8x_0 & - & 4x_1 & - & x_2 \\ \hline \end{array} \quad \rightarrow \quad x_0 = 1 \quad \rightarrow$$

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 4 & - & 4y_0 & + & 2x_1 & + & x_2 \\ \hline x_0 & = & 1 & - & y_0 & + & 0x_1 & + & 0x_2 \\ y_1 & = & 96 & + & 4y_0 & - & x_1 & + & 0x_2 \\ y_2 & = & 9992 & + & 8y_0 & - & 4x_1 & - & x_2 \\ \hline \end{array} \quad \rightarrow \quad x_1 = 96 \quad \rightarrow$$

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 196 & + & 4y_0 & - & 2y_1 & + & x_2 \\ \hline x_0 & = & 1 & - & y_0 & + & 0y_1 & + & 0x_2 \\ x_1 & = & 96 & + & 4y_0 & - & y_1 & + & 0x_2 \\ y_2 & = & 9608 & - & 8y_0 & + & 4y_1 & - & x_2 \\ \hline \end{array} \quad \rightarrow \quad x_2 = 9608 \quad \rightarrow$$

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 9804 & - & 4y_0 & + & 2y_1 & - & y_2 \\ \hline x_0 & = & 1 & - & y_0 & + & 0y_1 & + & 0y_2 \\ x_1 & = & 96 & + & 4y_0 & - & y_1 & + & 0y_2 \\ x_2 & = & 9608 & - & 8y_0 & + & 4y_1 & - & y_2 \\ \hline \end{array} \quad \rightarrow \quad y_1 = 96 \quad \rightarrow$$

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 9996 & + & 4y_0 & - & 2x_1 & - & y_2 \\ \hline x_0 & = & 1 & - & y_0 & + & 0x_1 & + & 0y_2 \\ y_1 & = & 96 & + & 4y_0 & - & x_1 & + & 0y_2 \\ x_2 & = & 9992 & + & 8y_0 & - & 4x_1 & - & y_2 \\ \hline \end{array} \quad \rightarrow \quad y_0 = 1 \quad \rightarrow$$

$$\begin{array}{|rcrcrcrcr|} \hline \xi & = & 10000 & - & 4y_0 & - & 2x_1 & - & y_2 \\ \hline y_0 & = & 1 & - & x_0 & + & 0x_1 & + & 0y_2 \\ y_1 & = & 100 & - & 4x_0 & - & x_1 & + & 0y_2 \\ x_2 & = & 10000 & - & 8x_0 & - & 4x_1 & - & y_2 \\ \hline \end{array} \quad \rightarrow \quad \xi = 10000$$

□

Problem 3. *Consider a linear programming problem that has an optimal dictionary in which exactly k of the original slack variables are non-basic. Show that by ignoring feasibility preservation of intermediate dictionaries this dictionary can be arrived at in exactly k pivots. Do not forget to allow for the fact that some pivot elements may be* 0.

Proof. Assume the premise of the problem above and assume that our problem has m constraints and thus m basic variables in each dictionary where $m \geq k - 1$. Furthermore, since we know that optimal dictionary has k non-basic slack variables, we know that our objective function has at least k variables. Thus we know our problem looks like the following.

$$\xi = \max\left(\sum_{i=0}^{n} \delta_\xi(i)x_i\right)$$

subject to

$$\bigwedge_{i=0}^{m}\left(\sum_{j=0}^{n} \delta_i(j)x_j \le \varepsilon_i\right)$$

where $\delta_\xi : \mathbb{N}_0 \cap [0,n]$ is the coefficient mapping for the objective function, $\delta_i : \mathbb{N}_0 \cap [0,n]$ is the coefficient mapping for the ith constraint, and ε_i is the bounding constant for the ith constraint where $i \in \mathbb{N}_0 \cap [0,m]$, and $n \ge k-1$. Notice we can rewrite each constraint as a slack variable equation for our dictionary like the following.

$$y_i = \varepsilon_i - \sum_{j=0}^{n} \delta_i(j)x_j$$

Importantly, for k of the original slack variables (which all begin as basic) to become non-basic, k of the original non-slack variables (which all begin as non-basic) must become basic. Let $S_y \subseteq \{y_i : i \in \mathbb{N}_0 \cap [0,m]\}$ where $|S_y| = k$ be a subset of the original slack variables such that an optimal dictionary of our problem has all of these substituted into the original objective function. Then, there must exist a complimentary subset of the original non-slack variables $S_x \subseteq \{x_i : i \in \mathbb{N}_0 \cap [0,n]\}$ where $|S_x| = k$ such that each of them is chosen exactly once as an entering variable. Furthermore, we can consider every new dictionary as a new linear programming problem. As such, we proceed by induction.
Base Case ($k = 0$): Let $k = 0$. Then our current dictionary represents an optimal solution for our linear programming problem.
Base Case ($k = 1$): Let $k = 1$. Then there exists a non-basic variable that can be chosen as the entering variable for the next iteration such that the resultant dictionary is optimal.
Inductive Step: Suppose we have a linear programming problem for which there exists an optimal dictionary containing $k+1$ of the original slack variables as non-basic variables and suppose there exists a subset of k of these variables such that they can be chosen in non-repeating sequence as entering variables. Then the resulting dictionary of that sequence of entering variables will be one iteration away from optimal with the remaining unchosen variable from the afforementioned $k+1$ being an entering variable that yields it. □

Problem 4. *For an undirected graph, $G = (V,E)$, an independent set is a set of nodes, $J \subseteq V$ such that there exists no edge in E among any two nodes in J. More precisely, $i,j \in J \implies (i,j) \notin E$. The indepdendence number of a graph, $\alpha(G)$ of a graph, G, is the size of the largest independent set in G. Do the following.*

(1) *Formulate finding the independence number of G as a linear programming problem (with the restriction that the decision variables will be binary).*

(2) *Write down the dual problem of the above problem.*

(3) *State in words what the solution to the dual problem describes about G. What does weak duality imply about the independence number and this new property.*

(4) *Does there exist a graph for which these problems have a duality gap? If yes, provide an example. Otherwise, explain why.*

(1) Here is the described problem formulated as a linear programming problem.

Let $x_i = \begin{cases} 1 & \text{if node } i \text{ is in the maximum anti-clique} \\ 0 & \text{otherwise} \end{cases}$

$$\max_x \sum_i x_i$$

subject to

$$\forall I \subseteq V, I \text{ is a dependent set} \implies \sum_{i \in I} x_i \leq 1$$

(2) Here is the dual problem of the immediately previous problem.

Let $y_J = \begin{cases} 1 & \text{if the dependent set, } J \text{ is assigned a color} \\ 0 & \text{otherwise} \end{cases}$

$$\min_y \sum_J y_J$$

subject to

$$\forall j \in V, j \in J \implies \sum_J y_J \geq 1$$

(3) Here is a description of the above dual problem. Choose a family of dependent sets such that each node is in at least one these dependent sets then assign each of them a unique color. Weak duality implies that every possible coloring that fits the constraints of the dual problem has more colors than any set of dependent sets that fit the constraints of the original problem.

(4) There is no graph that has a duality gap for these problems. Here is why. Partition any graph, G, into a set of sub-graphs, P, such that for any two nodes in the parent graph, a given sub-graph contains both nodes if and only if there exists a series of edges in said sub-graph connecting them. The optimal solution to the original problem is the cardinality of this partition, $|P|$, i.e. selecting exactly one node from each subgraph. Additionally, each sub-graph in P is a dependent set (specifically each sub-graph in P is as large as it can be while not containing any disconnected nodes). Thus $|P|$ is also the optimal solution to the dual problem.