Problem Definition and Formulation

Suppose there is a set, $S$, of suppliers, each with supply $s_i$ and a set, $D$, of customers each with demand $d_j$ that must be met. Further, suppose that each unit shipped from supply node $i$ to demand node $j$ incurs a cost $c_{ij}$. Find a minimum-cost shipping scheme that satisfies all the demand and supply restrictions.

Clearly, this problem is only feasible if $\sum_i s_i \geq \sum_j d_j$. Without loss of generality, suppose $\sum_i s_i = \sum_j d_j$ since otherwise we can set a dummy demand node $k$ with $d_k = \sum_i s_i - \sum_j d_j$. Then we can formulate this as a linear program by:

$$\text{min} \quad \sum_{i \in S, j \in D} c_{ij}x_{ij}$$

s.t.

$$\sum_{j \in D} x_{ij} = s_i \quad \forall i \in S$$

$$\sum_{i \in S} x_{ij} = d_j \quad \forall j \in D$$

$$x_{ij} \geq 0.$$

In some cases, we might wish to restrict the value of $x_{ij}$ to be integral.

First, note that the rows of the constraint matrix are linearly dependent because:

$$\sum_{i \in S} s_i = \sum_{i \in S} \sum_{j \in D} x_{ij} = \sum_{j \in D} \sum_{i \in S} x_{ij} = \sum_{j \in D} d_j$$

i.e., adding all the rows in the first constraint set gives the same result as adding all the rows of the second constraint set. Hence, without loss of generality, we can remove one constraint arbitrarily.

Let $|S| = m, |D| = n$. Consider a graphical representation of this problem where $G$ is a bipartite graph with vertex sets corresponding to $S$ and $D$ and an edge set $E = S \times D$, i.e., $\{i, j\} \in E$ for every $i \in S, j \in D$. Note that each decision variable corresponds to an edge in the graph. Hence, a basic solution to the LP corresponds to some subgraph.

Claim 1 The graph corresponding to a basic solution to the LP doesn’t contain cycles.

Proof: Let $C$ be a cycle in the corresponding subgraph induced by the basic solution. Then $C = i_1 \to j_1 \to \ldots \to j_k \to i_1$ where $i_1 \in S, j_1 \in D$. Consider the vector $x$ which equals 1 if $x_i$ corresponds to a cycle edge from $S$ to $D$, $-1$ if $x_i$ corresponds to a cycle edge from $D$ to $S$, and 0 otherwise.

Note that in any cycle in our graph, every vertex $k$ in the cycle has exactly one edge $x_{ik}$ going from $S$ to $D$ and one edge $x_{kj}$ going from $D$ to $S$ (or vice versa). Hence, it must be that $Ax = 0$, where $A$ is our constraint matrix.

So, consider any basis, $B$ and suppose $C$ is a cycle using only our basis variables. Let $A_B$ be the corresponding basis matrix and $x_B$ be the matching components of $x$. Since $x_i = 0$ for $i \notin B$, $0 = Ax = [A_B|A_N][x_B|x_N] = A_Bx_B + A_N0 = A_Bx_B$. Hence, $x_B$ is a non-zero element in the null space of $A_B$, so $B$ is not a basis. 

\[ \square \]

\[1\text{Based on previous notes of Rajithkumar Rajagopalan and Dennis Leventhal} \]
Noting that our graph has \( n + m \) vertices, and each basis has \( n + m - 1 \) edges (since there are that many constraints), and each basis cannot contain a cycle, it follows that each basic solution corresponds to a spanning tree of our graph.

**A Simplex-like Algorithm**

First, note that linear dependence of the primal constraints is equivalent to linear dependence of the dual variables. Since there is no non-negativity in the dual problem, we can choose one dual variable and set it arbitrarily (for example, set it to 0 for convenience).

Let’s see how a simplex-like algorithm can be used to solve the Transportation Problem. Using the characterization of basic feasible solution, notice that if we add an edge corresponding to a non-basic variable to the subgraph spanned by the edges corresponding to the basic variables, this creates a unique cycle. The algorithm is as follows:

**Initialization:** A basic feasible solution, \( x \), to the primal LP

**Step 1:** Choose a dual variable and set it arbitrarily.

**Step 2:** Solve for the other dual variables to maintain the complementary slackness condition, i.e., if \( x_{ij} \) is basic, then \( u_i + v_j = c_{ij} \).

**Step 3:** For every non-basic edge \( \{i, j\} \), compute the reduced costs \( \bar{c}_{ij} = c_{ij} - u_i - v_j \). If all reduced costs are non-negative, STOP: Dual solution is feasible. Else, choose \( x_{ij} \) such that \( \bar{c}_{ij} < 0 \). This will be the entering basic variable (this creates a cycle in the graph).

**Step 4:** Alternatingly increase (from \( i \to j \)) and decrease (from \( j \to k \)) flow around edges in the cycle to preserve 0 net flow. Let \( \delta = \min_{(i,j)} \{ x_{ij} : x_{ij} \text{ basic variable}, (i,j) \text{ a decreasing edge} \} \). Increase all “forward” edges by \( \delta \), decrease all “backwards” edges by \( \delta \). Remove from the basis an edge which achieves the minimum. Goto Step 1.