The Simplex Method

Eklavya Sharma

This document describes the *simplex method* for solving linear programs. The following result (proof omitted for brevity) will help us focus on a special case of linear programming.

1 Preliminaries

Theorem 1. *Any linear programming problem can be reduced to the following problem (called a* standard form linear program*):*

$$\min_{x \in \mathbb{R}^n} c^T x \text{ where } Ax = b \text{ and } x \geq 0.$$

Here $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ *and* $c \in \mathbb{R}^n$.

We will also assume without loss of generality that $\operatorname{rank}(A) = m$.

Read the following concepts at TheoremDep (`https://sharmaeklavya2.github.io/theoremdep/`):

- Basic feasible solution (BFS)
- Extreme point of a convex set
- Extreme point iff BFS
- LP in orthant is optimized at BFS

Due to the last point above, we will focus on finding an optimal solution that is also a BFS.

Lemma 2. *Let* $B = [u_1, u_2, \ldots, u_n]$ *be a basis of a vector space* V. *Let* $w = \sum_{i=1}^n \lambda_i u_i$. *Then* $B' = B - \{u_r\} \cup \{w\}$ *is a basis of* V *iff* $\lambda_r \neq 0$.

Proof. (See `https://sharmaeklavya2.github.io/theoremdep/nodes/linear-algebra/vector-spaces/basis/replace-vector.html`.) □

Lemma 3. *For any matrix* A, *we have* $\operatorname{rank}(A) = \operatorname{rank}(A^T)$.

Proof. (TODO: find proof) □

1.1 Notation

For any non-negative integer n, let $[n] := \{1, 2, \ldots, n\}$ (or $[n] := [1, 2, \ldots, n]$, depending on the context).

Let $v \in \mathbb{R}^n$ and $A \in \mathbb{R}^{m \times n}$. Let $i \in [m]$ and $j \in [n]$. Then the j^{th} element of v is denoted as v_j or $v[j]$. The element of A in the i^{th} row and j^{th} column of A is denoted as $A_{i,j}$ or $A[i,j]$. $A[*,j]$ denotes the j^{th} column of A and $A[i,*]$ denotes the i^{th} row of A.

Let $J = [j_1, j_2, \ldots, j_r]$ be a sequence of r integers in $[n]$. $v[J]$ is defined as the vector $[v[j_1], v[j_2], \ldots, v[j_n]]$. $A[*,J]$ is defined as the matrix whose k^{th} column is $A[*,j_k]$. Let $K = [k_1, k_2, \ldots, k_q]$ be a sequence of q integers in $[m]$. Then $A[K,*]$ is defined as the matrix whose i^{th} column is $A[k_i,*]$.

For matrices $A \in \mathbb{R}^{m \times n_1}$ and $B \in \mathbb{R}^{m \times n_2}$, let $C = [A, B]$ denote the matrix in $\mathbb{R}^{m \times (n_1+n_2)}$ where the first n_1 columns in C are the columns of A and the last n_2 columns in C are the columns of B. We call C the *horizontal concatenation* of A and B. We can similarly define horizontal concatenation of more than two matrices. We can similarly define vertical concatenation of A and B, which we denote as $\begin{bmatrix} A \\ B \end{bmatrix}$.

2 Bases

Consider this linear program:

$$\min_{x \in \mathbb{R}^n} c^T x \text{ where } Ax = b \text{ and } x \geq 0.$$

Here $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$.

Definition 1 (Basis)**.** *Let J be a sequence of m distinct numbers from $[n]$. Let $B := A[*,J]$. Then J is called a* basis *of the LP iff* $\operatorname{rank}(B) = m$. *$J$ is called a feasible basis iff it is a basis and $B^{-1}b \geq 0$.*

Let $\overline{J}$ be the increasing sequence of values of $[n]$ that are not in J. Define $\texttt{solve}(J)$ *as a vector $\widehat{x} \in \mathbb{R}^n$, where $\widehat{x}[J] = B^{-1}b$ and $\widehat{x}[\overline{J}] = 0$.*

The following two results show that to find an optimal BFS of the LP, we can find a feasible basis J that minimizes $c^T \texttt{solve}(J)$, and then return $\texttt{solve}(J)$.

Lemma 4. *Let J be a feasible basis and $\widehat{x} = \texttt{solve}(J)$. Then $\widehat{x}$ is a BFS of the LP.*

Proof. It's trivial to see that $\widehat{x} \geq 0$. Let $B = A[*,J]$ and $N = A[*,\overline{J}]$. Then

$$A\widehat{x} = B\widehat{x}[J] + N\widehat{x}[\overline{J}] = B(B^{-1}b) = b.$$

Hence, $\widehat{x}$ is feasible for the LP.

Because we can rearrange variables and constraints, we can assume without loss of generality that $J = [m]$. The equality constraints are tight, and their coefficient matrix is $A = [B, N]$. The non-negativity constraints $\{x_j \geq 0 : j \in \overline{J}\}$ are tight, and their coefficient matrix is $I_n[\overline{J},*] = [0, I_{n-m}]$, where I_k denotes the k-by-k identity matrix. Hence, the rank of the coefficient matrix of tight constraints at $\widehat{x}$ is

$$\operatorname{rank}\left(\begin{bmatrix} B & N \\ 0 & I_{n-m} \end{bmatrix}\right) = \operatorname{rank}\left(\begin{bmatrix} B & 0 \\ 0 & I_{n-m} \end{bmatrix}\right) = \operatorname{rank}(B) + (n-m) = n.$$

The first equation follows from the fact that rank is unaffected by row operations. The third equation follows from the fact that J is a basis. Since the coefficient matrix of tight constraints of $\widehat{x}$ has rank n, $\widehat{x}$ is a BFS of the LP. $\square$

Lemma 5. *Let $\widehat{x}$ be a BFS of the LP. Then there exists a feasible basis J such that $\widehat{x} = \texttt{solve}(J)$.*

Proof. Since $\widehat{x}$ is a BFS, there exist n linearly independent constraints that are tight at $\widehat{x}$. m of these are the equality constraints, whose coefficient matrix is A. The rest are inequality constraints. Let $\overline{J}$ be the indices of these $n-m$ inequality constraints. This implies $\widehat{x}[\overline{J}] = 0$. Since we can rearrange variables, assume without loss of generality that $\overline{J} = [m{+}1, m{+}2, \ldots, n]$. The coefficient matrix of these constraints is $I_n[\overline{J}, *] = [0, I_{n-m}]$.

Let $J = [m]$. Let $B = A[*, J]$ and $N = A[*, J]$. Then $A = [B, N]$. Since $\widehat{x}$ is a BFS, we get

$$n = \operatorname{rank}\left(\begin{bmatrix} B & N \\ 0 & I_{n-m} \end{bmatrix}\right) = \operatorname{rank}\left(\begin{bmatrix} B & 0 \\ 0 & I_{n-m} \end{bmatrix}\right) = \operatorname{rank}(B) + (n-m).$$

This implies that $\operatorname{rank}(B) = m$, which shows that J is a basis of the LP.

Furthermore, since $\widehat{x}$ is feasible for the LP, we get that $b = A\widehat{x} = B\widehat{x}[J] + N\widehat{x}[\overline{J}] = B\widehat{x}[J]$. Hence, $\widehat{x}[J] = B^{-1}b$. Since $\widehat{x}$ is feasible for the LP, we get $\widehat{x} \ge 0 \implies \widehat{x}[J] \ge 0 \implies B^{-1}b \ge 0$. Hence, J is a feasible basis and $\texttt{solve}(J) = \widehat{x}$. $\square$

3 The Simplex Algorithm

Algorithm 1 $\texttt{naiveSimplex}(A, b, c, J)$: Here $A \in \mathbb{R}^{m \times n}$ and J is a feasible basis for the standard LP given by A, b, c.

while true **do**
    $B = A[*, J]$
    $Y = B^{-1}A$ *// If B isn't invertible, throw an exception*
    $\overline{b} = B^{-1}b$
    $\texttt{assert}(\overline{b} \ge 0)$
    $z = Y^T c[J]$
    if $c - z \ge 0$ **then**
        return $(\text{optimal}, \texttt{solve}(J))$
    end if
    Find $k \in [n]$ such that $c_k - z_k < 0$.
    Define $y \in \mathbb{R}^n$ as $y_t = \begin{cases} -Y[i,k] & \text{if } t = j_i \\ 1 & \text{if } t = k \\ 0 & \text{otherwise} \end{cases}$.
    if $Y[*, k] \le 0$ **then**
        return $(\text{unbounded}, \texttt{solve}(J), y)$, where
    end if
    $r = \operatorname*{argmin}_{i \in [m] : Y[i,k] > 0} \dfrac{\overline{b}_i}{Y[i,k]}$
    $\delta = \overline{b}_r / Y[r, k]$
    $J[r] = k$ *// change the r^{th} entry in J to k.*
end while

Theorem 6. *If* `naiveSimplex` *outputs* (*optimal*, $\widehat{x}$)*, then* $\widehat{x}$ *is a BFS of the LP and an optimal solution to the LP.*

Proof. If `naiveSimplex` outputs (optimal, $\widehat{x}$) in an iteration, then the algorithm didn't fail at Lines 3 and 5, so $\operatorname{rank}(B) = m$ and $\overline{b} \geq 0$ in that iteration. This implies that J is a feasible basis in that iteration. Hence, by Lemma 4, $\widehat{x} = \texttt{solve}(J)$ is a BFS of the LP. Note that $c^T\widehat{x} = c[J]^T\widehat{x}[J] = c[J]^T\overline{b}$.

Let $\overline{J} = [n] - J$. Let $N = A[*, \overline{J}]$. Let $x_B = x[J]$ and $x_N = x[\overline{J}]$. Then

$$Ax = b \iff Bx_B + Nx_N = b \iff x_B = \overline{b} - B^{-1}Nx_N.$$

Note that since the constraint $x_B = \overline{b} - B^{-1}Nx_N$ is equivalent to $Ax = b$, we can replace $Ax = b$ by $x_B = \overline{b} - B^{-1}Nx_N$ in the LP without affecting the set of feasible solutions.

We can use these new constraints to express the objective value as a function of x_N.

$$\begin{aligned} c^Tx &= c[J]^Tx_B + c[\overline{J}]^Tx_N \\ &= c[J]^T\left(\overline{b} - B^{-1}Nx_N\right) + c[\overline{J}]^Tx_N \\ &= c[J]^T\overline{b} + (c[\overline{J}]^T - c[J]^TB^{-1}N)x_N \end{aligned}$$

$$\begin{aligned} &z[\overline{J}]^T = (c[J]^TY)[\overline{J}] = c[J]^TB^{-1}A[*, \overline{J}] = c[J]^TB^{-1}N. \\ &\implies c^Tx = c[J]^T\overline{b} + (c - z)[\overline{J}]^Tx_N. \end{aligned}$$

From the non-negativity constraints, we know that $x_N \geq 0$. We also know that $c - z \geq 0$, since `naiveSimplex` returned (optimal, $\widehat{x}$). Hence, for every feasible x, we have $c^Tx = c[J]^T\overline{b} + (c - z)[\overline{J}]^Tx_N \geq c[J]^T\overline{b} = c^T\widehat{x}$. Hence, $\widehat{x}$ is an optimal solution to the LP. □

Lemma 7. $z[J] = c[J]$ *(before* J *changes at Line 17).*

Proof.

$$z[J]^T = c[J]^T(B^{-1}A)[*, J] = c[J]^TB^{-1}A[*, J] = c[J]^T.$$ □

Lemma 7 implies that $k \notin J$, since $c_k - z_k < 0$.

Lemma 8. $Y[*, J] = I$*. Let* $J = [j_1, j_2, \ldots, j_m]$*. Then* $Y[i, j_p] = \begin{cases} 1 & \text{if } p = i \\ 0 & \text{if } p \neq i \end{cases}$.

Proof.

$$Y[*, J] = (B^{-1}A)[*, J] = B^{-1}A[*, J] = B^{-1}B = I.$$

$$Y[i, j_p] = Y[*, J][i, p] = I[i, p] = \begin{cases} 1 & \text{if } p = i \\ 0 & \text{if } p \neq i \end{cases}.$$ □

We will now show that the simplex algorithm moves in the direction y in each iteration, and y is a direction in the nullspace of A in which the cost c^Ty reduces.

Lemma 9. $Yy = Ay = 0$.

Proof.

$$\begin{aligned}(Yy)_i &= \sum_{j=1}^{n} Y[i,j]y_j = \sum_{p=1}^{m} Y[i,j_p]y_{j_p} + Y[i,k]y_k \\ &= y_{j_i} + Y[i,k]y_k = -Y[i,k] + Y[i,k] = 0.\end{aligned}$$

$$Ay = B^{-1}Yy = B^{-1}0 = 0.$$

□

Lemma 10. $c^T y = c_k - z_k < 0$.

Proof.

$$\begin{aligned}c^T y &= \sum_{j=1}^{n} c_j y_j = c_k y_k + \sum_{p=1}^{m} c_{j_p} y_{j_p} = c_k - \sum_{p=1}^{m} c_{j_p} Y[p,k] \\ &= c_k - \sum_{p=1}^{m} Y^T[k,p]c[J]_p = c_k - (Y^T c[J])_k = c_k - z_k < 0.\end{aligned}$$

□

Theorem 11. *If* `naiveSimplex` *outputs* $(\text{unbounded}, \widehat{x}, y)$*, then the LP's cost reduces along the ray* $\{\widehat{x} + \lambda y : \lambda \ge 0\}$ *and the ray is feasible, which implies that the LP is unbounded.*

Proof. Since `naiveSimplex` didn't fail at Lines 3 and 5, we know that $\operatorname{rank}(B) = m$ and $\overline{b} \ge 0$. Hence, J is a feasible basis. So, by Lemma 4, we know that $\widehat{x} = \texttt{solve}(J)$ is a BFS of the LP.

By Lemma 9, we know that $Ay = 0$. Hence, $A(\widehat{x}+\lambda y) = A\widehat{x} = b$. Since `naiveSimplex` returned $(\text{unbounded}, \widehat{x}, y)$, we get that $Y[*,k] \le 0$ (by Line 12). Hence, $y \ge 0$ and so $\widehat{x} + \lambda y \ge \widehat{x} \ge 0$. Hence, $\widehat{x} + \lambda y$ is feasible for the LP for all $\lambda \ge 0$.

By Lemma 10, we know that $c^T y < 0$, Hence, moving along the ray will reduce cost indefinitely. This implies that the LP is unbounded. □

Suppose `naiveSimplex` doesn't return an output in an iteration. Then it will change J to, say, $\widetilde{J}$ in that iteration (at Line 17). We will show that $\widetilde{J}$ is also a feasible basis of the LP, and hence, `naiveSimplex` will not fail at Lines 3 and 5.

Lemma 12. *Suppose* `naiveSimplex` *changes* J *to* $\widetilde{J}$ *in an iteration. Then* $\widetilde{J}$ *is a basis of the LP.*

Proof. Let $J = [j_1, j_2, \ldots, j_m]$. The set of values in $\widetilde{J}$ is $J - \{j_r\} \cup \{k\}$. Since $k \notin J$, $\widetilde{J}$ has distinct values.

Let a_j be the j^{th} column of A. Let $B = A[*,J]$. Let $\widetilde{B} = A[*,\widetilde{J}]$. Let $S = \{a_{j_1}, a_{j_2}, \ldots, a_{j_m}\}$ be the set of columns of B and let $\widetilde{S} = S - \{a_{j_r}\} \cup \{a_k\}$ be the set of columns of $\widetilde{B}$. Since J is a basis, $\operatorname{rank}(B) = m$, so S is a set of linearly independent vectors. Since $|S| = m$, we get that S is a basis of $\mathbb{R}^m$. Hence, $a_k \in \operatorname{span}(S)$.

Let $a_k = \sum_{i=1}^{m} \lambda_i a_{j_i}$. Let $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_m]$. Then $B\lambda = \sum_{i=1}^{m} \lambda_i a_{j_i} = a_k$. Hence, $\lambda = B^{-1}a_k = Y[*,k]$. Since $Y[r,k] > 0$, we get that $\lambda_r > 0$. Hence, by Lemma 2, we get that $\widetilde{S}$ is also a basis of $\mathbb{R}^m$. Hence, $\operatorname{rank}(\widetilde{B}) = m$, so $\widetilde{J}$ is a basis. □

Lemma 13. *Suppose* `naiveSimplex` *changes* J *to* $\widetilde{J}$ *in an iteration. Let* $\widehat{x} = \texttt{solve}(J)$ *and* $\widetilde{x} = \widehat{x} + \delta y$*. Then* $\widetilde{x} = \texttt{solve}(\widetilde{J})$ *and* $\widetilde{J}$ *is a feasible basis.*

Proof. By Lemma 9, we get that $Ay = 0$. Hence, $A\widetilde{x} = A\widehat{x} + \delta(Ay) = A\widehat{x} = b$.

If $i \notin J$ or $Y[i,k] \le 0$, then $y_i \ge 0$, and hence $\widetilde{x}_i = \widehat{x}_i + \delta y_i \ge \widehat{x}_i \ge 0$. Now let $i \in J$ and $Y[i,k] > 0$. Let $J = [j_1, j_2, \ldots, j_m]$. Then

$$\delta = \frac{\overline{b}_r}{Y[r,k]} \le \frac{\overline{b}_i}{Y[i,k]}.$$

$$\implies \widetilde{x}_{j_i} = \widehat{x}_{j_i} + \delta y_{j_i} = \overline{b}_i - \delta Y[i,k] \ge 0.$$

Hence, $\widetilde{x} \ge 0$. Therefore, $\widetilde{x}$ is feasible for the LP.

Let $i \in [n] - \widetilde{J}$. If $i = j_r$, then

$$\widetilde{x}_i = \widehat{x}_{j_r} + \delta y_{j_r} = \overline{b}_r - \delta Y[r,k] = 0.$$

If $i \in [n] - J - \{k\}$, then $\widetilde{x}_i = \widehat{x}_i + \delta y_i = 0 + \delta 0 = 0$. Hence, $\widetilde{x}_i = 0$ when $i \notin \widetilde{J}$. Let $\widetilde{B} := A[*, \widetilde{J}]$. Then

$$b = A\widetilde{x} = A[*, \widetilde{J}]\widetilde{x}[\widetilde{J}] = \widetilde{B}\widetilde{x}[\widetilde{J}].$$

By Lemma 12, $\widetilde{J}$ is a basis, so $\widetilde{B}$ is invertible. Hence, $\widetilde{x}[\widetilde{J}] = \widetilde{B}^{-1}b$. Furthermore, $\widetilde{x}[[n] - \widetilde{J}] = 0$, so $\widetilde{x} = \texttt{solve}(\widetilde{J})$. Since $\widetilde{x} \ge 0$, we get that $\widetilde{B}^{-1}b \ge 0$. Hence, $\widetilde{J}$ is a feasible basis. □

4 The Tableau Method and the Revised Simplex method

The naive simplex method has a large running time of $O(m^2(m+n))$, since we compute B^{-1}, Y, $\overline{b}$ and z afresh in each iteration. The tableau method and the revised simplex method are two ways to get around this problem.

Let J be a feasible basis of the LP. Let $B = A[*, J]$, $Y = B^{-1}A$, $\overline{b} = B^{-1}b$ and $z = Y^T c[J]$. Suppose $c_k - z_k < 0$ for some $k \in [n] - \widetilde{J}$ and $Y[r,k] > 0$ for some $r \in [m]$. Assume without loss of generality that $r = \operatorname{argmin}_{i \in [m]: Y[i,k] > 0} \frac{\overline{b}_i}{Y[i,k]}$. Let $\widetilde{J}$ be a sequence obtained by changing the r^{th} element of J to k. By Lemma 13, $\widetilde{J}$ is a feasible basis. Let $\widetilde{B} = A[*, \widetilde{J}]$, $\widetilde{Y} = \widetilde{B}^{-1}A$, $\overline{\overline{b}} = \widetilde{B}^{-1}b$ and $\widetilde{z} = \widetilde{Y}^T c[\widetilde{J}]$. We will now see how to compute $\widetilde{Y}$, $\widetilde{z}$ and $\overline{\overline{b}}$ from $\widetilde{Y}$, z and $\overline{b}$.

Define the matrix $\widehat{Y}$ as

$$\widehat{Y}[i,j] = \begin{cases} \dfrac{Y[r,j]}{Y[r,k]} & \text{if } i = r \\[2ex] Y[i,j] - \dfrac{Y[i,k]}{Y[r,k]} Y[r,j] & \text{if } i \ne r \end{cases}.$$

Note that $\widehat{Y}$ is obtained from Y by elementary row operations. This is called *pivoting*, and $Y[r,*]$ is a column vector where the r^{th} entry is 1 and the others are 0. Let R be the matrix of these row operations. Then $\widehat{Y} = RY$. We can find R by applying these row operations to the m-by-m identity matrix.

$$R[i,j] = \begin{cases} \dfrac{I[r,j]}{Y[r,k]} & \text{if } i = r \\ I[i,j] - \dfrac{Y[i,k]}{Y[r,k]} I[r,j] & \text{if } i \neq r \end{cases}$$
$$= \begin{cases} \dfrac{1}{Y[r,k]} & \text{if } i = r = j \\ -\dfrac{Y[i,k]}{Y[r,k]} & \text{if } i \neq r \wedge j = r \\ 1 & \text{if } i \neq r \wedge j = i \\ 0 & \text{if } j \notin \{i, r\} \end{cases}.$$

Lemma 14. *$\widetilde{B}^{-1} = RB^{-1}$ and $\widetilde{Y} = RY$ and $\bar{\bar{b}} = R\bar{b}$.*

Proof. Let $J = [j_1, j_2, \ldots, j_m]$. $\widetilde{J} = J - \{j_r\} \cup \{k\}$. By Lemma 8, we get that $Y[*,J] = \widetilde{Y}[*,\widetilde{J}] = I$. We will try to show that $\widehat{Y}[*,\widetilde{J}] = I$.

Let $p, q \in [m] - \{r\}$.

$$\widehat{Y}[*,\widetilde{J}][r,r] = \widehat{Y}[r,\widetilde{J}[r]] = \widehat{Y}[r,k] = 1.$$

$$\widehat{Y}[*,\widetilde{J}][r,q] = \widehat{Y}[r,\widetilde{J}[q]] = \widehat{Y}[r,j_q] = \frac{Y[r,j_q]}{Y[r,k]} = 0. \qquad \text{(by Lemma 8)}$$

$$\widehat{Y}[*,\widetilde{J}][p,r] = \widehat{Y}[p,\widetilde{J}[r]] = \widehat{Y}[p,k] = Y[p,k] - \frac{Y[p,k]}{Y[r,k]} Y[r,k] = 0.$$

$$\widehat{Y}[*,\widetilde{J}][p,q] = \widehat{Y}[p,\widetilde{J}[q]] = \widehat{Y}[p,j_q] = Y[p,j_q] - \frac{Y[p,k]}{Y[r,k]} Y[r,j_q] = Y[p,j_q]$$
$$= \begin{cases} 1 & \text{if } p = q \\ 0 & \text{otherwise} \end{cases}. \qquad \text{(by Lemma 8)}$$

Hence, $\widehat{Y}[*,\widetilde{J}] = I$.

$$I = \widehat{Y}[*,\widetilde{J}] = (RB^{-1}A)[*,\widetilde{J}] = RB^{-1}A[*,\widetilde{J}] = RB^{-1}\widetilde{B}.$$

Hence, $\widetilde{B}^{-1} = RB^{-1}$.

$$\widetilde{Y} = \widetilde{B}^{-1}A = RB^{-1}A = RY.$$

$$\bar{\bar{b}} = \widetilde{B}^{-1}b = RB^{-1}b = R\bar{b}. \qquad \square$$

Define $\widehat{z} \in \mathbb{R}^n$ and η as

$$\widehat{z}_j = z_j + \frac{c_k - z_k}{Y[r,k]} Y[r,j] \qquad\qquad \eta = c[J]^T\bar{b} + \frac{c_k - z_k}{Y[r,k]} \bar{b}_r.$$

Lemma 15. *$\widehat{z} = \widetilde{z}$ and $\eta = c[\widetilde{J}]^T \overline{\overline{b}}$.*

Proof. Let $J = [j_1, j_2, \ldots, j_m]$. Then $\widetilde{J} = J - \{j_r\} \cup \{k\}$. Let $i \in [m] - \{r\}$. Then

$$\widehat{z}[\widetilde{J}]_i = \widehat{z}_{j_i} = z_{j_i} + \frac{c_k - z_k}{Y[r,k]} Y[r, j_i] = z_{j_i}.$$

By Lemma 8, we get $Y[r, j_i] = 0$. By Lemma 7, we get $z_{j_i} = c_{j_i}$. Hence, $\widehat{z}[\widetilde{J}]_i = c_{j_i} = c[\widetilde{J}]_i$.

$$\widehat{z}[\widetilde{J}]_r = \widehat{z}_k = z_k + \frac{c_k - z_k}{Y[r,k]} Y[r,k] = c_k = c[\widetilde{J}]_r.$$

Hence, $\widehat{z}[\widetilde{J}] = c[\widetilde{J}]$.

$$Y[r, *] = (B^{-1}A)[r, *] = B^{-1}[r, *]A.$$

$$\overline{b}_r = (B^{-1}b)_r = B^{-1}[r, *]b.$$

Let $\alpha = (c_k - z_k)/Y[r, k]$. Then

$$\widehat{z}^T = z^T + \alpha Y[r, *] = c[J]^T B^{-1} A + \alpha B^{-1}[r, *]A.$$

$$\eta = c[J]^T \overline{b} + \alpha \overline{b}_r = c[J]^T B^{-1} b + \alpha B^{-1}[r, *]b.$$

Let $u^T = c[J]^T B^{-1} + \alpha B^{-1}[r, *]$. Then $\widehat{z}^T = u^T A$ and $\eta = u^T b$.

$$c[\widetilde{J}]^T = \widehat{z}[\widetilde{J}]^T = (u^T A)[\widetilde{J}] = u^T A[*, \widetilde{J}] = u^T \widetilde{B}.$$

Hence, $u^T = c[\widetilde{J}]^T \widetilde{B}^{-1}$. So, $\widehat{z} = c[\widetilde{J}]^T \widetilde{B}^{-1} A = c[\widetilde{J}]^T \widetilde{Y} = \widetilde{z}$ and $\eta = c[\widetilde{J}]^T \widetilde{B}^{-1} b = c[\widetilde{J}]^T \overline{\overline{b}}$. □

4.1 Tableau method

Note that $c[J]^T \overline{b} = c^T \texttt{solve}(J)$. In the tableau method, we compute B^{-1}, Y, $\overline{b}$, z and $c[J]^T \overline{b}$ in the first iteration. In subsequent iterations, we update Y and $\overline{b}$ by applying elementary row operations given by R (see Lemma 14), and update z and $c[J]^T \overline{b}$ as per Lemma 15. All of these operations can be done together as a pivoting operation on the following matrix, called the tableau:

$$\begin{bmatrix} c - z & c[J]^T \overline{b} \\ Y & \overline{b} \end{bmatrix}.$$

The time per iteration is, therefore, $O(mn)$. We can reduce this to $O(m(n-m))$ if we observe that when we pivot on row r and column k, then $m - 1$ entries in $Y[r, *]$ are 0 (by Lemma 8). The space complexity is $O(mn)$.

4.2 Revised simplex method

In the revised simplex method, we compute B^{-1} in the first iteration. Henceforth, we will assume that B^{-1} is available in the beginning of each iteration and we need to update it at the end of each iteration.

Compute $c[J]^T B^{-1}$ by multiplying $c[J]^T$ and B^{-1}. Then we iterate over $j \in [n] - \widetilde{J}$, and compute $z_j = (c[J]^T B^{-1})A[*, j]$. We stop iterating when we find a suitable $k \in [n] - \widetilde{J}$ such that $c_k - z_k < 0$, or if $c_j - z_j \geq 0$ for all $j \in [n] - \widetilde{J}$.

Next, we compute $u = B^{-1}A[*, k]$ and $\bar{b} = B^{-1}b$. Note that $u = Y[*, k]$. Then we continue from Line 11 onwards (i.e., compute y, δ and r, and update $\widetilde{J}$). At the end of the iteration, we can update B^{-1} using row operations as per Lemma 14. This is possible since R is defined by u.

The time taken is $O(m(t + m))$, where t is the number of variables that need to be considered till we find k. Note that $t \leq n - m$. The space complexity (in addition to storing the input) is $O(m^2)$.