IE-406 **2019 Fall**

Lecture 1 Farkas's Lemma, Strong Duality and Criss-Cross Algorithm

Instructor: Tamás Terlaky Scribe: Yutong Dai Last Modified: 2019-09-22

Disclaimer: *This note has not been subjected to the usual scrutiny reserved for formal publications.*

Consider the following primal-dual problem

$$\begin{cases} \min & c^T x \\ s.t & Ax = b \\ & x \geq 0, \end{cases} \quad (P) \qquad\qquad \begin{cases} \max & b^T y \\ s.t & A^T y \leq c \end{cases} \quad (D).$$

Correspondingly, we have following version of Farkas' Lemma

Lemma 1.1: Farkas' Lemma (Primal Form)

One and only one of the following two systems of linear inequalities has a feasible solution.

$$(I_p)\begin{cases} Ax = b \\ x \geq 0 \end{cases} \qquad\qquad (II_p)\begin{cases} A^T y \leq 0 \\ b^T y > 0 \end{cases}.$$

Theorem 1.2: Weak Duality

For any primal-dual feasible solution pair (x, y), we have $c^T x \geq b^T y$. Moreover, $=$ holds iff

$$x^T(A^T y - c) = 0,$$

which is known as the complementary slackness condition.

Theorem 1.3: Strong Duality

Exactly one of the following could happen for a $(P) - (D)$ problem pair.

1. If both the primal and dual problems have a feasible solution, then for any optimal solutions $\bar{x}$ and $\bar{y}$, we have $c^T \bar{x} = b^T \bar{y}$.
2. If the primal infeasible and the dual is feasible, then the dual is unbounded.
3. If the dual infeasible and the primal is feasible, then the primal is unbounded.
4. Both the primal and dual can be infeasible.

The equivalence of Farkas' Lemma and Strong Duality

First, let's consider when strong duality holds. Then consider the following case, (P) is infeasible with $c = 0$ and (D) is feasible. Then, we know that (D) is unbounded. Therefore, we know that $\exists\, y^*$ such that $A^T y^* \leq 0$ and $b^T y^* > 0$. This implies, when (II_p) is feasible, (I_p) is infeasible. When (D) is infeasible, then (II_p) is also infeasible, and (I_P) is feasible by strong duality.

Then we consider using Farkas' Lemma to prove Strong duality.

1. By weak duality, we have $c^T x \geq b^T y$, for any feasible solution pair (x, y). Now, we want to prove $b^T y \geq c^T x$. It's equivalent to prove the system (1.1) is feasible. So if (1.1) is feasible, then we have 1 in strong duality proved.

$$\left\{ \begin{array}{l} -b^T y + c^T x + \rho = 0, \rho \geq 0 \\ Ax = b, x \geq 0 \\ A^T y + s = b, s \geq 0, y \text{ is free.} \end{array} \right. \qquad \Longrightarrow \underbrace{\begin{bmatrix} 0 & 0 & A & 0 & 0 \\ -A^T & A^T & 0 & I & 0 \\ b^T & -b^T & c^T & 0 & 1 \end{bmatrix}}_{\bar{A}} \underbrace{\begin{bmatrix} y^- \\ y^+ \\ x \\ s \\ \rho \end{bmatrix}}_{\bar{x}} = \underbrace{\begin{bmatrix} b \\ c \\ 0 \end{bmatrix}}_{\bar{b}} \tag{1.1}$$

2. Assume the system (1.1) is infeasible, then by Fakars' lemma, we know that the system (1.2) must be feasible.

$$\left\{ \begin{array}{l} \bar{A}^T \bar{y} \leq 0 \\ \bar{b}^T \bar{y} > 0 \end{array} \right. \qquad \text{where } \bar{y} = \begin{bmatrix} u \in \mathbb{R}^{\gtrdot} \\ v \in \mathbb{R}^{\ltimes} \\ l \in \mathbb{R}^{\nVdash} \end{bmatrix} \qquad \Longrightarrow \left\{ \begin{array}{l} Av = bl \\ A^T u + cl \leq 0 \\ b^T u + c^T v > 0 \\ v \leq 0, l \leq 0 \end{array} \right. \tag{1.2}$$

Set $\bar{l} = -l \geq 0, \bar{v} = -v \geq 0$. If $\bar{l} > 0$. Then set $x = \frac{\bar{v}}{\bar{l}}, y = \frac{u}{\bar{l}}$. It's easy to verify that (x, y) is a feasible primal-dual solution pair. But $b^T y = \frac{1}{\bar{l}} b^T u > \frac{1}{\bar{l}} c^T \bar{v} = c^T x$, which contradicts with the weak duality. This implies $\bar{l} = 0$.

Provided $\bar{l} = 0$, then (1.3) always holds.

$$① \left\{ \begin{array}{l} A\bar{v} = 0 \\ \bar{v} \geq 0 \end{array} \right. \qquad ② \left\{ \begin{array}{l} A^T u \leq 0 \\ b^T u > c^T \bar{v} \end{array} \right. \tag{1.3}$$

As ① and ② both hold, by Fakars' lemma, we know $c^T \bar{v} \neq 0$.

- $c^T \bar{v} < 0$. Assume the primal is feasible and dual is infeasible. Consider a primal feasible solution x^*, then $x(\alpha) = x^* + \alpha \bar{v}$, where $\alpha > 0$, is also a solution. Then $c^T x(\alpha) = c^T x^* + \alpha c^T \bar{v} \to -\infty$ as $\alpha \to +\infty$. Therefore, the primal is unbounded.

- $c^T \bar{v} > 0$. This implies $b^T u > 0$. Assume the dual is feasible and primal is infeasible. Consider a dual feasible solution y^*, then $y(\alpha) = y^* + \alpha u$, where $\alpha > 0$, is also a solution. Then $b^T y(\alpha) = b^T y^* + \alpha b^T u \to +\infty$ as $\alpha \to +\infty$. Therefore, the dual is unbounded.

So (2) and (3) in the strong duality theorem are proved. It remains to prove (4). Consider $A = 0$, $c = -1$, $b = 1$. In this case both (P) and (D) are infeasible.

Criss-cross Algorithm and the Strong Duality

1. From Theorem 1.4, we know that if the Criss-cross algorithm terminates in finite steps and produce a primal-dual solution pair (x, y), then the complementary slack conditions hold. By the weak duality, we have $c^T x = b^T y$.

2. Suppose the primal is infeasible and the dual is feasible. Let $(\bar{y}, \bar{s})$ be a dual feasible solution such that $A^T\bar{y} + \bar{s} = c, \bar{s} \geq 0$. We need to find a $\hat{y}$such that $b^T\hat{y} > 0$. Then $b^T y(\alpha) = b^T\bar{y} + \alpha b^T\hat{y} \to +\infty$ as $\alpha \to +\infty$. Since the primal is infeasible, we must encounter

$$
\begin{array}{r|c|c|c|c|c|}
\cline{2-6}
 & & & & & \\
\cline{3-6}
\text{p-th row} & [A_B^{-1}]_p & - & \oplus & \cdots & \oplus \\
\cline{3-6}
 & & & & & \\
\cline{2-6}
\end{array}
$$

Let $\tilde{y}^T$ =p-th row of A_B^{-1}, then $\tilde{y}^T b < 0, A^T\tilde{y} \geq 0$. Set $\hat{y} = -\tilde{y}$, then $b^T\hat{y} > 0$. Hence, the dual is unbounded.

2. Suppose the dual is infeasible and the primal is feasible. Let $\bar{x}$ be a primal feasible solution such that $A^T\bar{x} = b, \bar{x} \geq 0$. We need to find a $\hat{x}$such that $c^T\hat{x} < 0$. Then $c^T x(\alpha) = c^T\bar{x} + \alpha b^T\hat{x} \to -\infty$ as $\alpha \to +\infty$. Since the dual is infeasible, we must encounter

$$
\begin{array}{|c|c|c|}
\multicolumn{3}{c}{\text{q-th row}} \\
\hline
 & + & \\
\hline
 & \ominus & \\
\hline
 & \vdots & \\
\hline
 & \ominus & \\
\hline
\multicolumn{1}{c|}{} & A_B^{-1}A[:, q] & \multicolumn{1}{c}{} \\
\cline{2-2}
\end{array}
$$

Define

$$
t_q = \begin{cases} \tau_{iq}, & \text{i in the basis} \\ -1, & \text{i=q} \\ 0, & \text{i not in the basis} \end{cases},
$$

we can see in the proof of Theorem 1.4 that $t_q \in \text{Null}(A)$, i.e., $At_q = 0$. From the table, we further know $t_{iq} = A_B^{-1}A[i, q] \leq 0$. Set $\hat{x} = -t_q$. Then $A\hat{x} = 0, \hat{x} \geq 0$.

$$
\begin{aligned}
c^T\hat{x} &= c_B^T\hat{x}_B + C_N^T\hat{x}_N \\
&\overset{(1)}{=} c_q - c_B^T(A_B^{-1}A[:, q]) \\
&\overset{(2)}{=} s_q < 0,
\end{aligned}
$$

where (1) holds as q not in the basis and (2) holds as $-s_q > 0$ (from the tableau). Hence, the primal is unbounded.

Appendix: Criss-Cross Algorithm

- Basic Tableau Setup

For a given coefficient matrix $A \in \mathbb{R}^{m \times n}, m < n$ and $\text{Rank}(A) = m$, we can partition A as $A = [A_B, A_N]$, where A_B is invertible Denote $I_B = \{i | i\text{th column of A in the} A_B\}, I_N = \{1, \cdots, n\} \backslash I_B$. We can rewrite the constraint in (D) as $A^T y + s = c$. Then, can partition c, s according to I_B, I_N, x, y respectively. Now, the solution pair (x, y) can be set to

$$x_B = A_B^{-1} b, \quad x_N = \mathbf{0}, \quad s_B = 0, \quad s_N = c_N^T - c_B^T A_B^{-1} A_N.$$

$c_B^T A_B^{-1} b$	$-s_B^T = 0$	$-s_N^T = -c_N^T + c_B^T A_B^{-1} A_N$
$x_B = A_B^{-1} b$	I	$A_B^{-1} A_N = (\tau_{ij})$

- Pivot

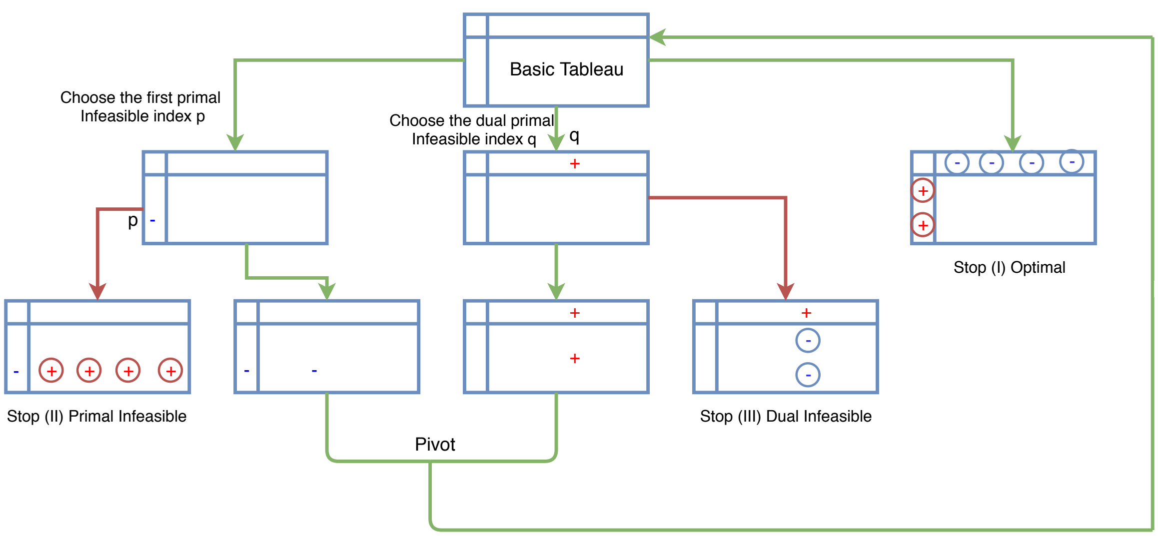
The pivot step can be described as

$$\begin{aligned}
I'_B &\leftarrow I_B \cup \{l\} \backslash \{k\} \\
\tau'_{ij} &= \tau_{ij} - \frac{\tau_{il}\tau_{kj}}{\tau_{kl}} && \forall i \in I'_B \backslash \{l\}; j \in I'_N \backslash \{k\} \\
\tau'_{ik} &= -\frac{\tau_{il}}{\tau_{kl}}, && \forall i \in I'_B \backslash \{l\} \\
\tau'_{lj} &= \frac{\tau_{kj}}{\tau_{kl}}, && \forall j \in I_N \backslash \{l\} \\
\tau'_{lk} &= \frac{1}{\tau_{kl}}
\end{aligned}$$

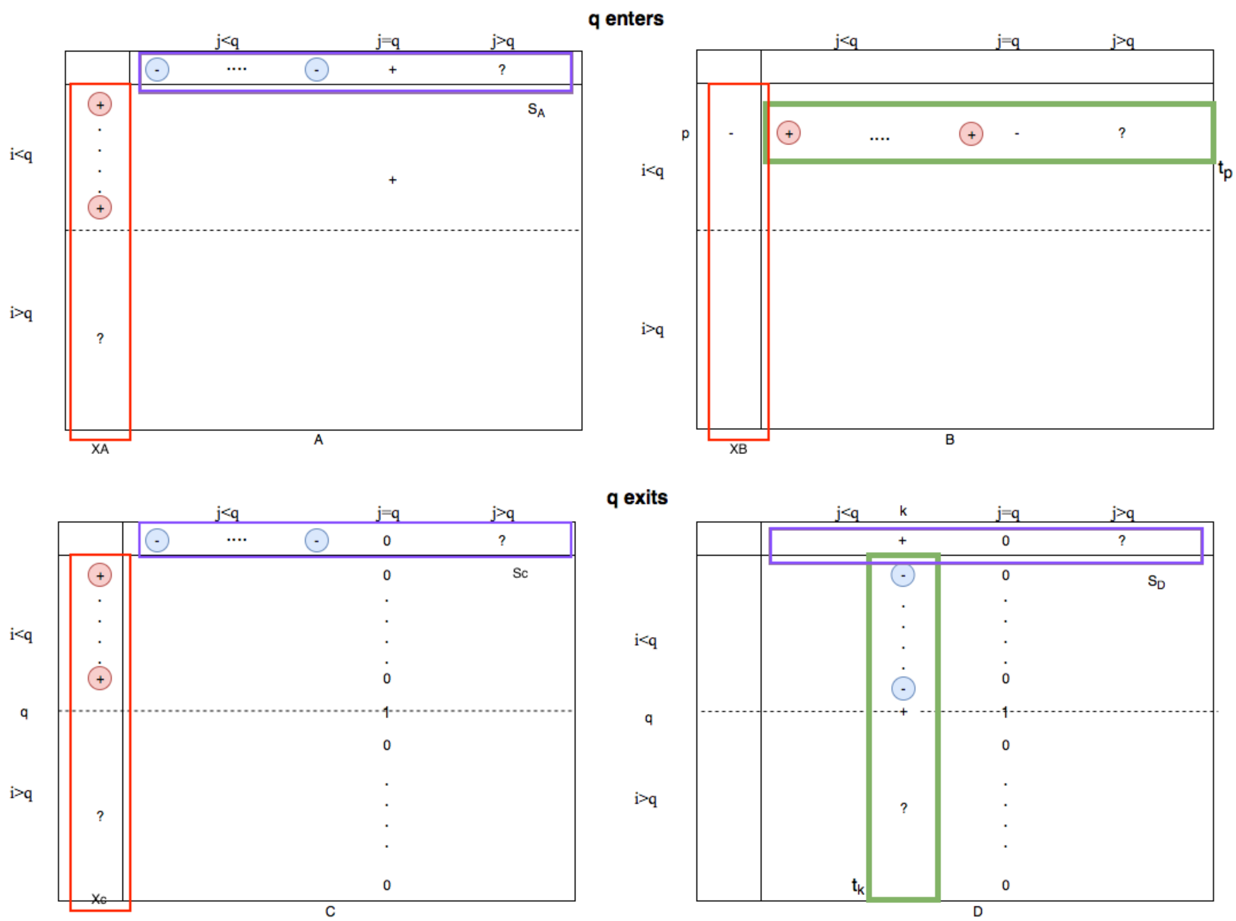
$$\begin{array}{c|c|cc|}
 & & \text{j} & \text{l} \\
\hline
 & & & \\
\hline
\text{i} & & \tau_{ij} & \tau_{il} \\
\text{k} & & \tau_{kj} & \tau_{kl} \\
\hline
\end{array}
\quad \xrightarrow{\text{Pivot on } \tau_{kl}} \quad
\begin{array}{c|c|cc|}
 & & \text{j} & \text{k} \\
\hline
 & & & \\
\hline
\text{i} & & \tau'_{ij} & \tau'_{ik} \\
\text{l} & & \tau'_{lj} & \tau'_{lk} \\
\hline
\end{array}$$

- Criss-cross Algorithm Procedure

Initialization
  let A_B be an arbitrary initial basis;
  I_B resp. I_N is the index set of
    the basis and nonbasis variables;
while true **do**
  if $x_B \geq 0$ and $s_N \geq 0$ **then**
(I)   **stop**: the current solution solves the LP problem;
  else
    $p := \min\{i \in I_B : x_i < 0\}$;
    $q := \min\{j \in I_N : s_j < 0\}$;
    $r := \min\{p, q\}$;
    if $r = p$ **then**
      if the p-row of the tableau is nonnegative **then**
(II)   **stop**: **(LP)** is inconsistent;
      else
        let $q := \min\{j \in I_N : \tau_{pj} < 0\}$;
      endif
    else (i.e. $r = q$)
      if the q-column of the tableau is nonpositive **then**
(III)   **stop**: **(LD)** is inconsistent;
      else
        let $p := \min\{i \in I_B : \tau_{iq} > 0\}$;
      endif
    endif
    perform a pivot: $I_B := I_B \cup \{q\} \setminus \{p\}$;
  endwhile
end.



Theorem 1.4
Criss-cross Algorithm terminates in finite steps.



Proof: Assume we will visit a basis twice, meaning we are trapped in a cycle. Denote $q = \max\{i \in \{1, 2, \cdots, n\} | i \text{ enters the basis during the cycle.}\} = \max\{i \in \{1, 2, \cdots, n\} | i \text{ leaves the basis during the cycle.}\}$. From the figure, we know that x_q can only enter the basis via either pattern A or B. Similarly, x_q can only leave the basis via either pattern C or D.

It remains to prove that one of the following four situations is impossible.

1. B$\Rightarrow$ D
2. B$\Rightarrow$ C
3. A$\Rightarrow$ D
4. A$\Rightarrow$ C

B$\Rightarrow$ D case:

The matrix A can be partitioned as $A = [A_B \mid A_n]$, then we can get the coefficient matrix in the tableau $[I \mid A_B^{-1} A_N]$. So we know that $t_p \in \mathbb{R}^{1\times n}$, where $p < q$, must be

$$[t_p]_j = \begin{cases} 0 & x_j \text{ in the basis and j} \neq p \\ 1 & j = p \\ \tau_{pj} & j \text{ not in the basis} \end{cases}$$

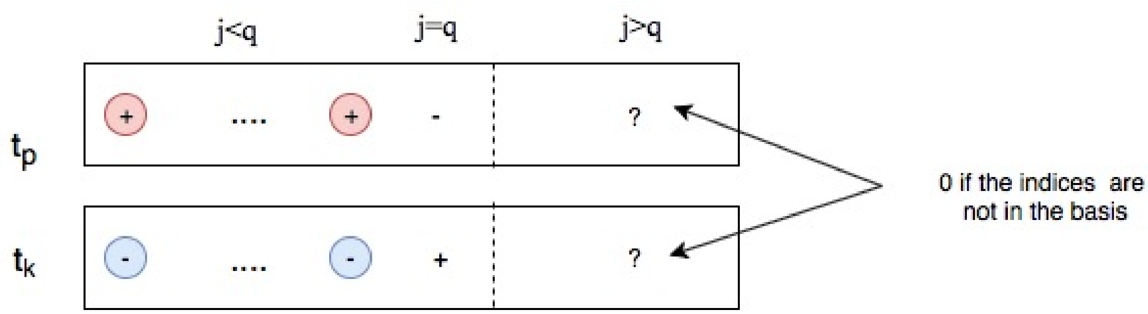
Similarly we can construct a $t_k \in \mathbb{R}^{n\times 1}$, where $k < q$, such that

$$[t_k]_i = \begin{cases} 0 & x_i \text{ not in the basis and i} \neq k \text{ (1)} \\ -1 & i = k \text{ (2)} \\ \tau_{ik} & i \text{ in the basis (3).} \end{cases}$$

(3) is directly from the tableau; (1) and (2) are constructed based the following fact

$$\begin{bmatrix} I & A_B^{-1}A_N \end{bmatrix} \begin{bmatrix} A_B^{-1}A_N \\ -I \end{bmatrix} = \mathbf{0}.$$

Therefore, we know that $\langle t_k, t_p \rangle = 0$. From the figure below, and the facts that



1) the way we define t_p and t_k;
2) If $j > p$, both sets $S_1 = \{j|x_j \text{ in the basis}\}, S_2 = \{j|x_j \text{ not in the basis}\}$ remain unchanged.

We know that $\langle t_k, t_p \rangle < 0$. Contradiction! So $B \Rightarrow D$ is impossible.

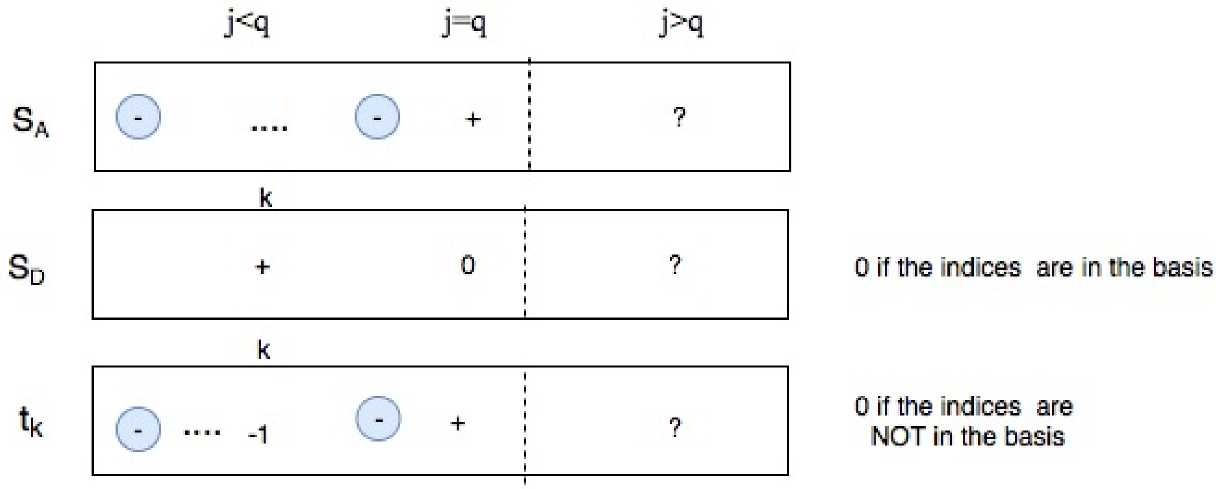
A⇒ D case:

From above analysis, we already construct t_k such that $t_k \in \text{Null}(A)$. So we want to find a vector in row(A).

Note that

$$\begin{cases} A^T y_A + (-S_A) = c \\ A^T y_D + (-S_D) = c, \end{cases}$$

we know $S \triangleq S_A - S_D = A^T(y_A - y_D)$. Therefore, $\langle t_k, t_p \rangle = 0$.

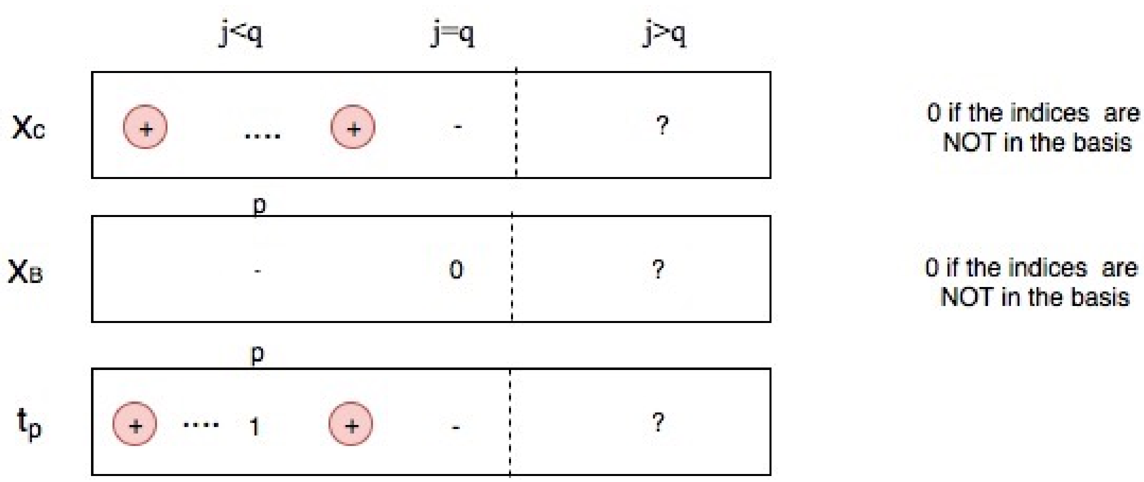


Similarly as the previous case, for $j > q$, x_j will be always [in/not in] basis when $A \Rightarrow D$.

- $\langle S_A, t_k \rangle = \underbrace{\text{non-negative}}_{j<q} + \underbrace{\text{positive}}_{j=q} + \underbrace{0}_{j>q} = \text{positive}$
- $\langle S_D, t_k \rangle = \underbrace{0}_{j<q, j\neq k} + \underbrace{\text{negative}}_{j=k} + \underbrace{0}_{j\geq q} = \text{negative}$

Therefore, $\langle S_A - S_D, t_k \rangle = \text{positive}$, Hence, contradiction!

B⇒ C case:



Consider the solution from tableau C and B and denote them as X_C and X_B. Then we know that $AX_C = b, AX_B = b$, (here we abuse the notation X_B,i.e., not the basic solution part.) So $X_C - X_B \in \text{Null}(A)$. As $t_p \in \text{Row}(A)$, we know that $\langle X_C - X_B, t_p \rangle = 0$.

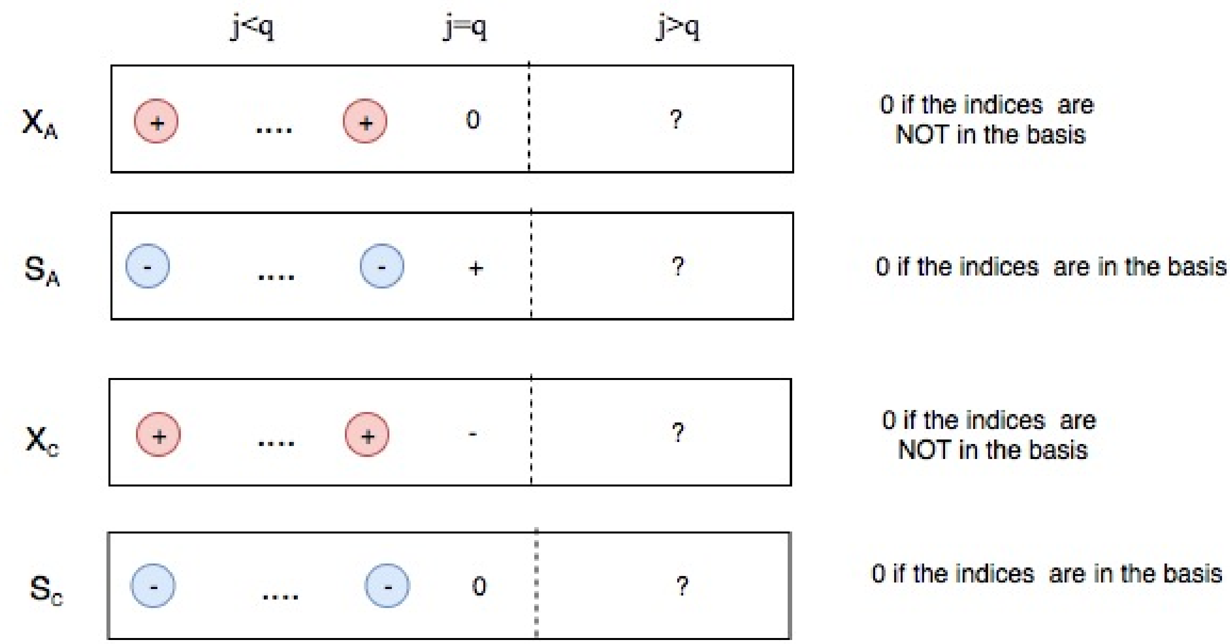
- $\langle X_C, t_p \rangle = \underbrace{\text{non-negative}}_{j<q} + \underbrace{\textit{positive}}_{j=q} + \underbrace{0}_{j>q} = \text{positive}$

- Based on the fact that X_B, t_p are from the same tableau,

$$\langle S_D, t_k \rangle = \underbrace{0}_{j<q, j\neq p} + \underbrace{\text{negative}}_{j=p} + \underbrace{0}_{j=q} + \underbrace{0}_{j\geq q} = \text{negative}$$

Therefore, $\langle X_C - X_B, t_p \rangle = \text{positive}$, Hence, contradiction!

A⇒ C



Based on the same reasons used before, we know that

$X_A - X_C \in \text{Null}(A), S_A - S_C \in \text{Row}(A)$, hence $\langle X_A - X_C, S_A - S_C \rangle = 0$.

- As S_A, X_C from the same tableau, we know that $\langle X_A, S_A \rangle = 0$. Similarly, $\langle X_C, S_C \rangle = 0$
- $\langle X_A, S_C \rangle = \text{non-positive}$, and $\langle X_C, S_A \rangle = \text{negative}$

So we know that $\langle X_A - X_C, S_A - S_C \rangle > 0$. Hence, contradiction!

To sum up, cycling is impossible during the criss-cross algorithm, hence terminating in finite steps.