LINEAR PROGRAMMING

[V. CH6]: THE SIMPLEX METHOD IN MATRIX NOTATION

Phillip Keldenich   Ahmad Moradi

Department of Computer Science
Algorithms Department
TU Braunschweig

December 19, 2023

LP in Matrix Notation

Standard Form LP

As usual, we begin our discussion with the standard-form linear programming problem:

$$\begin{aligned} \max_x \quad & \sum_{j=1}^{n} c_j x_j \\ \text{subject to} \quad & \sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, 2, \cdots, m \\ & x_j \geq 0, \quad j = 1, 2, \cdots, n \end{aligned}$$

Standard Form LP

As usual, we begin our discussion with the standard-form linear programming problem:

$$\begin{aligned} \max_x \quad & \sum_{j=1}^{n} c_j x_j \\ \text{subject to} \quad & \sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, 2, \cdots, m \\ & x_j \geq 0, \quad j = 1, 2, \cdots, n \end{aligned}$$

It is convenient to introduce slack variables as follows:

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j, \qquad i = 1, \ldots, m$$

w_i renamed as x_{n+i}

Matrix Form

With these slack variables, we now write our problem in matrix form:

$$\begin{aligned} \max_x \quad & c^T x \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

where

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} & 1 & & & \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} & & 1 & & \\ \vdots & \vdots & \ddots & \vdots & & & \ddots & \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} & & & & 1 \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}, c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ 0 \\ \vdots \\ 0 \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ x_{n+1} \\ \vdots \\ x_{n+m} \end{pmatrix}$$

Separating Basic and Non-Basic Parts

Consider an iteration of the Simplex algorithm where $\mathcal{B}$ and $\mathcal{N}$ are the set of basic and non-basic indices.

The ith component of Ax can be broken up into a *basic* and a *nonbasic* part:

$$\sum_{j=1}^{n+m} a_{ij}x_j = \sum_{j\in\mathcal{B}} a_{ij}x_j + \sum_{j\in\mathcal{N}} a_{ij}x_j.$$

SEPARATING BASIC AND NON-BASIC PARTS

Consider an iteration of the Simplex algorithm where $\mathcal{B}$ and $\mathcal{N}$ are the set of basic and non-basic indices.

The ith component of Ax can be broken up into a *basic* and a *nonbasic* part:

$$\sum_{j=1}^{n+m} a_{ij}x_j = \sum_{j\in\mathcal{B}} a_{ij}x_j + \sum_{j\in\mathcal{N}} a_{ij}x_j.$$

To break up the matrix product Ax analogously:

- Let B denote an $m \times m$ matrix whose columns are indexed by $\mathcal{B}$.
- Similarly, let N denote an $m \times n$ matrix whose columns are indexed by $\mathcal{N}$.

Separating Basic and Non-Basic Parts

Consider an iteration of the Simplex algorithm where $\mathcal{B}$ and $\mathcal{N}$ are the set of basic and non-basic indices.

The ith component of Ax can be broken up into a *basic* and a *nonbasic* part:

$$\sum_{j=1}^{n+m} a_{ij}x_j = \sum_{j\in\mathcal{B}} a_{ij}x_j + \sum_{j\in\mathcal{N}} a_{ij}x_j.$$

To break up the matrix product Ax analogously:

- Let B denote an $m \times m$ matrix whose columns are indexed by $\mathcal{B}$.
- Similarly, let N denote an $m \times n$ matrix whose columns are indexed by $\mathcal{N}$.

Now, one could write A and x in a partitioned-matrix form as:

$$A = \begin{bmatrix} B & N \end{bmatrix}, x = \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix}$$

Separating Basic and Non-Basic Parts

Consider an iteration of the Simplex algorithm where $\mathcal{B}$ and $\mathcal{N}$ are the set of basic and non-basic indices.

The ith component of Ax can be broken up into a *basic* and a *nonbasic* part:

$$\sum_{j=1}^{n+m} a_{ij}x_j = \sum_{j\in\mathcal{B}} a_{ij}x_j + \sum_{j\in\mathcal{N}} a_{ij}x_j.$$

To break up the matrix product Ax analogously:

- Let B denote an $m \times m$ matrix whose columns are indexed by $\mathcal{B}$.
- Similarly, let N denote an $m \times n$ matrix whose columns are indexed by $\mathcal{N}$.

Now, one could write A and x in a partitioned-matrix form as:

$$A = [B \;\; N]\,, x = \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix}$$

This is a rearrangement (relabeling of the variables) so that basic columns/variables are listed first, followed by the nonbasic columns/variables. So technically, the equality is not correct.

SIMPLIFYING

Now, we can write:

$$Ax = [B \ \ N] \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix} = Bx_{\mathcal{B}} + Nx_{\mathcal{N}}.$$

Check that this equality holds (and matrices have the right format).

Simplifying

Now, we can write:

$$Ax = [B \;\; N] \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix} = Bx_{\mathcal{B}} + Nx_{\mathcal{N}}.$$

Check that this equality holds (and matrices have the right format).
Similar partitioning on c gives:

$$c^T x = \begin{bmatrix} c_{\mathcal{B}} \\ c_{\mathcal{N}} \end{bmatrix}^T \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix} = c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}}.$$

Simplifying

Now, we can write:

$$Ax = [B \;\; N] \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix} = Bx_{\mathcal{B}} + Nx_{\mathcal{N}}.$$

Check that this equality holds (and matrices have the right format).
Similar partitioning on c gives:

$$c^T x = \begin{bmatrix} c_{\mathcal{B}} \\ c_{\mathcal{N}} \end{bmatrix}^T \begin{bmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{bmatrix} = c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}}.$$

Example: Write the following LP in matrix form, then calculate the above values for $\mathcal{B} = \{1, 2\}$

$$\begin{array}{rrrrl} \max\limits_{x} & 3x_1 + & 4x_2 - & 2x_3 & \\ \text{subject to} & x_1 + & 0.5x_2 - & 5x_3 & \leq 2 \\ & 2x_1 - & x_2 + & 3x_3 & \leq 3 \\ & x_1, & x_2, & x_3 & \geq 0 \end{array}$$

LP IN MATRIX NOTATION

SIMPLEX METHOD IN MATRIX NOTATION
- Primal Simplex Algorithm
- Dual Simplex Algorithm
- Two-Phase Methods

Dictionaries in Matrix Form: Basic Variables

A dictionary has the property that the

basic variables are written as functions of the nonbasic variables.

In matrix notation, we see that the constraint equations $Ax = b$ can be written as

$$Bx_{\mathcal{B}} + Nx_{\mathcal{N}} = b.$$

Dictionaries in Matrix Form: Basic Variables

A dictionary has the property that the

basic variables are written as functions of the nonbasic variables.

In matrix notation, we see that the constraint equations $Ax = b$ can be written as

$$Bx_{\mathcal{B}} + Nx_{\mathcal{N}} = b.$$

$x_{\mathcal{B}}$ can be written as a function of the nonbasic variables $x_{\mathcal{N}}$ iff the matrix B is *invertible*,

$$x_{\mathcal{B}} = B^{-1}b - B^{-1}Nx_{\mathcal{N}}.$$

Dictionaries in Matrix Form: Basic Variables

A dictionary has the property that the

basic variables are written as functions of the nonbasic variables.

In matrix notation, we see that the constraint equations $Ax = b$ can be written as

$$Bx_{\mathcal{B}} + Nx_{\mathcal{N}} = b.$$

$x_{\mathcal{B}}$ can be written as a function of the nonbasic variables $x_{\mathcal{N}}$ iff the matrix B is *invertible*,

$$x_{\mathcal{B}} = B^{-1}b - B^{-1}Nx_{\mathcal{N}}.$$

The fact that B must be invertible means that its m column vectors are linearly independent and therefore form a basis for $\mathbb{R}^m$. This is why the basic variables are called *basic*.

Dictionaries in Matrix Form: Objective

Similarly, the objective function can be written as

$$\begin{aligned}\zeta &= c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T \left(B^{-1}b - B^{-1}N x_{\mathcal{N}}\right) + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}}\right)^T x_{\mathcal{N}}\end{aligned}$$

Dictionaries in Matrix Form: Objective

Similarly, the objective function can be written as

$$\begin{aligned}\zeta &= c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T \left(B^{-1}b - B^{-1}Nx_{\mathcal{N}}\right) + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T B^{-1}b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}}\right)^T x_{\mathcal{N}}\end{aligned}$$

Putting all together, we can write the dictionary associated with basis $\mathcal{B}$ as

$$\begin{aligned}\zeta &= c_{\mathcal{B}}^T B^{-1}b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}}\right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1}b \quad - B^{-1}Nx_{\mathcal{N}}\end{aligned}$$

Dictionaries in Matrix Form

$$
\begin{aligned}
\zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\
x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}}
\end{aligned}
$$

Comparing against the component-form notation, we make the following identifications:

$$
\begin{aligned}
c_{\mathcal{B}}^T B^{-1} b &= \bar{\zeta} \\
c_{\mathcal{N}} - \left(B^{-1}N\right)^T c_{\mathcal{B}} &= [\bar{c}_j] \\
B^{-1} b &= [\bar{b}_i] \\
B^{-1} N &= [\bar{a}_{ij}]
\end{aligned}
$$

bracketed expressions on the right denote vectors and matrices with the index i running over $\mathcal{B}$ and the index j running over $\mathcal{N}$.

Matrix Form: Basic Solutions

Recall: We rewrote dictionaries in matrix form as

$$\begin{aligned}\zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}}\end{aligned}$$

The basic solution associated with this dictionary is obtained by setting $x_{\mathcal{N}}$ equal to zero.

$$x_{\mathcal{N}}^* = 0, \quad x_{\mathcal{B}}^* = B^{-1} b$$

Matrix Form: Basic Solutions

Recall: We rewrote dictionaries in matrix form as

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1} N \right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

The basic solution associated with this dictionary is obtained by setting $x_{\mathcal{N}}$ equal to zero.

$$x_{\mathcal{N}}^* = 0, \quad x_{\mathcal{B}}^* = B^{-1} b$$

Insight:
The basic solution $x_{\mathcal{B}}^*$ for a given $\mathcal{B}$ is simply obtained by solving the linear system $Bx_{\mathcal{B}} = b$!
All other variables are simply set to 0.

Matrix Form: Basic Solutions

Recall: We rewrote dictionaries in matrix form as

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1} N \right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

The basic solution associated with this dictionary is obtained by setting $x_{\mathcal{N}}$ equal to zero.

$$x_{\mathcal{N}}^* = 0, \quad x_{\mathcal{B}}^* = B^{-1} b$$

Insight:
The basic solution $x_{\mathcal{B}}^*$ for a given $\mathcal{B}$ is simply obtained by solving the linear system $Bx_{\mathcal{B}} = b$!
All other variables are simply set to 0.
Solving $Bx_{\mathcal{B}} = b$ for some b may give us a feasible (or infeasible) solution: $x_{\mathcal{B}}^*$ might be negative!

Example Time

As an example, consider the same LP

$$\begin{aligned} \max_{x} \quad & 3x_1 + 4x_2 - 2x_3 \\ \text{subject to} \quad & x_1 + 0.5x_2 - 5x_3 \leq 2 \\ & 2x_1 - x_2 + 3x_3 \leq 3 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

Write the initial dictionary. Do first pivot. You get a new basis $\mathcal{B} = \{2, 5\}$. Compute different part of this dictionary in matrix form and compare.

Towards Dual Simplex

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

To write the associated dual dictionary using the negative transpose property, it is important to correctly associate *complementary pairs of variables*.

Recall that, we have appended the primal slack variables to the end of the original variables:

$$(x_1, \dots, x_n, w_1, \dots, w_m) \rightarrow (x_1, \dots, x_n , x_{n+1}, \dots, x_{n+m})$$

Recall that,

- dual slacks are complementary to the primal originals, and
- dual originals are complementary to the primal slacks

using similar index for complementary variables,

$$(z_1, \dots, z_n, y_1, \dots, y_m) \rightarrow (z_1, \dots, z_n , z_{n+1}, \dots, z_{n+m})$$

Dual Dictionary: Negative Transpose Property

Primal dictionary in matrix form:

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1} N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

Dual Dictionary: Negative Transpose Property

Primal dictionary in matrix form:

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1} N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

and the corresponding dual dictionary:

$$\begin{aligned} -\xi &= -c_{\mathcal{B}}^T B^{-1} b \qquad\qquad - \left(B^{-1} b\right)^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= \left(\left(B^{-1} N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right) + \left(B^{-1} N\right)^T z_{\mathcal{B}} \end{aligned}$$

Dual Dictionary: Negative Transpose Property

Primal dictionary in matrix form:

$$\begin{aligned} \zeta &= c_{\mathcal{B}}^T B^{-1} b - \left(\left(B^{-1} N \right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= B^{-1} b \quad - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

and the corresponding dual dictionary:

$$\begin{aligned} -\xi &= -c_{\mathcal{B}}^T B^{-1} b \qquad\qquad - \left(B^{-1} b \right)^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= \left(\left(B^{-1} N \right)^T c_{\mathcal{B}} - c_{\mathcal{N}} \right) + \left(B^{-1} N \right)^T z_{\mathcal{B}} \end{aligned}$$

The dual solution associated with this is obtained by setting $z_{\mathcal{B}}$ equal to zero:

$$z_{\mathcal{B}}^* = 0, \quad z_{\mathcal{N}}^* = \left(B^{-1} N \right)^T c_{\mathcal{B}} - c_{\mathcal{N}}$$

Succinct Primal and Dual Dictionary

Using the shorthands

$$z_{\mathcal{N}}^* = \left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}},$$
$$x_{\mathcal{B}}^* = B^{-1}b \text{ and}$$
$$\zeta^* = c_{\mathcal{B}}^T B^{-1} b,$$

we write the primal dictionary as

$$\begin{aligned} \zeta &= \zeta^* - \left(z_{\mathcal{N}}^*\right)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x_{\mathcal{B}}^* - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

and dual dictionary as

$$\begin{aligned} -\xi &= -\zeta^* - \left(x_{\mathcal{B}}^*\right)^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= z_{\mathcal{N}}^* + \left(B^{-1}N\right)^T z_{\mathcal{B}} \end{aligned}$$

PRIMAL SIMPLEX IN MATRIX NOTATION

Primal Dictionary

$$\begin{aligned} \zeta &= \zeta^* - (z^*_{\mathcal{N}})^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x^*_{\mathcal{B}} - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

Dual Dictionary

$$\begin{aligned} -\xi &= -\zeta^* - (x^*_{\mathcal{B}})^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= z^*_{\mathcal{N}} + \left(B^{-1} N\right)^T z_{\mathcal{B}} \end{aligned}$$

Assumptions: Given initial $\mathcal{B}$ with m elements, B invertible, $x^*_{\mathcal{B}} = B^{-1} b \geq 0$.
Let us discuss the details of a single Simplex iteration (move to adjacent basis) in matrix notation.

STEP 1. *Check for optimality*

if $z^*_{\mathcal{N}} \geq 0$ stop
//Primal feasibility and complementary is already maintained, just check dual feasibility.

STEP 2. *Select entering variable*

Else: pick $j \in \mathcal{N}$ with $z^*_j < 0$.
//variable x_j is the entering variable.

Primal Simplex in Matrix Notation

Primal Dictionary

$$\begin{aligned} \zeta &= \zeta^* - (z_{\mathcal{N}}^*)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x_{\mathcal{B}}^* - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

Dual Dictionary

$$\begin{aligned} -\xi &= -\zeta^* - (x_{\mathcal{B}}^*)^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= z_{\mathcal{N}}^* + \left(B^{-1} N\right)^T z_{\mathcal{B}} \end{aligned}$$

Step 3. *Compute primal step direction* $\Delta x_{\mathcal{B}}$

Having selected the entering variable, we let

$$x_{\mathcal{N}} = t e_j$$

where e_j denote the unit vector that is zero in every component except for a one in the position associated with index j,

$$x_{\mathcal{B}} = x_{\mathcal{B}}^* - B^{-1} N t e_j$$

Hence, the step direction $\Delta x_{\mathcal{B}}$ for the primal basic variables is given by

$$\Delta x_{\mathcal{B}} = B^{-1} N e_j$$

PRIMAL SIMPLEX IN MATRIX NOTATION

STEP 4. *Compute primal step length t*

We wish to pick the largest $t \geq 0$ for which

$$x^*_{\mathcal{B}} \geq t\Delta x_{\mathcal{B}}.$$

(every component of $x_{\mathcal{B}}$ remains nonnegative)

Since, for each $i \in \mathcal{B}^*, x^*_i \geq 0$ and $t \geq 0$

$$\frac{1}{t} \geq \frac{\Delta x_i}{x^*_i}, \quad \forall i \in \mathcal{B}$$

Hence, the largest t for which all of the inequalities hold is given by

$$t = \left(\max_{i \in \mathcal{B}} \frac{\Delta x_i}{x^*_i}\right)^{-1} \text{ // convention for } \frac{0}{0} \text{ is to set such ratios to 0}$$

if $t \leq 0$ stop //*primal is unbounded*

STEP 5. *Select leaving variable* The leaving variable is chosen as any variable $x_i, i \in \mathcal{B}$, for which the maximum in the calculation of t is obtained.

Primal Simplex in Matrix Notation

Primal Dictionary

$$\begin{aligned} \zeta &= \zeta^* - (z^*_{\mathcal{N}})^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x^*_{\mathcal{B}} - B^{-1} N x_{\mathcal{N}} \end{aligned}$$

Dual Dictionary

$$\begin{aligned} -\xi &= -\zeta^* - (x^*_{\mathcal{B}})^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= z^*_{\mathcal{N}} + \left(B^{-1} N\right)^T z_{\mathcal{B}} \end{aligned}$$

Essentially all that remains is to explain changes to the objective function.

Step 6. *Compute dual step direction $\Delta z_{\mathcal{N}}$*
Since in dual dictionary z_i is the entering variable, we see that

$$\Delta z_{\mathcal{N}} = -\left(B^{-1} N\right)^T e_i$$

Step 7. *Compute dual step length s*
Since we know that j is the leaving variable in the dual dictionary,

$$s = \frac{z^*_j}{\Delta z_j}$$

Primal Dictionary	Dual Dictionary
$$\begin{aligned} \zeta &= \zeta^* - (z_{\mathcal{N}}^*)^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x_{\mathcal{B}}^* - B^{-1} N x_{\mathcal{N}} \end{aligned}$$	$$\begin{aligned} -\xi &= -\zeta^* - (x_{\mathcal{B}}^*)^T z_{\mathcal{B}} \\ z_{\mathcal{N}} &= z_{\mathcal{N}}^* + \left(B^{-1} N\right)^T z_{\mathcal{B}} \end{aligned}$$

We now have everything we need to update the data in the dictionary:

Step 8. *Update current primal and dual solutions*

$$\begin{aligned} x_j^* &\leftarrow t \\ x_{\mathcal{B}}^* &\leftarrow x_{\mathcal{B}}^* - t\Delta x_{\mathcal{B}} \end{aligned}$$

$$\begin{aligned} z_i^* &\leftarrow s \\ z_{\mathcal{N}}^* &\leftarrow z_{\mathcal{N}}^* - s\Delta z_{\mathcal{N}} \end{aligned}$$

Step 9. *Update basis*

$$\mathcal{B} \leftarrow (\mathcal{B} \setminus \{i\}) \cup \{j\}$$

As an example, let us solve the following LP:

$$\begin{aligned} \max_{x} \quad & 4x_1 + 3x_2 \\ \text{subject to} \quad & x_1 - x_2 \leq 1 \\ & 2x_1 - x_2 \leq 3 \\ & + x_2 \leq 5 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Lets now derive *Dual simplex* in matrix notation.

RECALL:
Dual Simplex is to apply simplex to the dual LP but (indirectly) on a sequence of primal pivots

Here, instead of assuming that the primal dictionary is feasible ($x_{\mathcal{B}} \geq 0$), we now assume that the dual dictionary is feasible ($z_{\mathcal{N}} \geq 0$) and perform the analogous steps

Primal Dictionary

$$\begin{aligned} \zeta &= \zeta^* - (z^*_\mathcal{N})^T x_\mathcal{N} \\ x_\mathcal{B} &= x^*_\mathcal{B} - B^{-1} N x_\mathcal{N} \end{aligned}$$

Dual Dictionary

$$\begin{aligned} -\xi &= -\zeta^* - (x^*_\mathcal{B})^T z_\mathcal{B} \\ z_\mathcal{N} &= z^*_\mathcal{N} + \left(B^{-1}N\right)^T z_\mathcal{B} \end{aligned}$$

STEP 1. *Check for optimality*

if $x^*_\mathcal{B} \geq 0$ stop

STEP 2. *Select Entering Variable*

Else: pick $i \in \mathcal{B}$ with $x^*_i < 0$.
//variable z_i is the entering variable.

STEP 3. *Compute Dual Step Direction $\Delta z_\mathcal{N}$*

$$\Delta z_\mathcal{B} = -\left(B^{-1}N\right)^T e_i$$

STEP 4. *Compute Dual Step Length, s*

$$s = \left(\max_{j \in \mathcal{N}} \frac{\Delta z_j}{z^*_j} \right)^{-1}$$

if $s \leq 0$ stop
//dual is unbounded, primal infeasible

STEP 5. *Select Leaving Variable*

Pick z_j, where the maximum obtained.

STEP 6. *Compute Primal Step Direction $\Delta x_\mathcal{B}$*

$$\Delta x_\mathcal{B} = B^{-1} N e_j$$

STEP 7. *Compute Primal Step Length, t*

$$t = \frac{x^*_i}{\Delta x_i}$$

STEP 8. *Update Current Primal and Dual Solutions*

$$z^*_i \leftarrow s \qquad x^*_j \leftarrow t$$
$$z^*_\mathcal{N} \leftarrow z^*_\mathcal{N} - s\Delta z_\mathcal{N} \qquad x^*_\mathcal{B} \leftarrow x^*_\mathcal{B} - t\Delta x_\mathcal{B}$$

STEP 9. *Update Basis*

$$\mathcal{B} \leftarrow (\mathcal{B} \setminus \{i\}) \cup \{j\}$$

Primal Simplex	Dual Simplex
Suppose $x^*_{\mathcal{B}} \geq 0$	Suppose $z^*_{\mathcal{N}} \geq 0$
while $(z^*_{\mathcal{N}} \ngeq 0)$ {	while $(x^*_{\mathcal{B}} \ngeq 0)$ {
pick $j \in \{j \in \mathcal{N} : z^*_j < 0\}$	pick $i \in \{i \in \mathcal{B} : x^*_i < 0\}$
$\Delta x_{\mathcal{B}} = B^{-1} N e_j$	$\Delta z_{\mathcal{N}} = -(B^{-1}N)^T e_i$
$t = \left(\max_{i \in \mathcal{B}} \frac{\Delta x_i}{x^*_i} \right)^{-1}$	$s = \left(\max_{j \in \mathcal{N}} \frac{\Delta z_j}{z^*_j} \right)^{-1}$
pick $i \in \operatorname{argmax}_{i \in \mathcal{B}} \frac{\Delta x_i}{x^*_i}$	pick $j \in \operatorname{argmax}_{j \in \mathcal{N}} \frac{\Delta z_j}{z^*_j}$
$\Delta z_{\mathcal{N}} = -(B^{-1}N)^T e_i$	$\Delta x_{\mathcal{B}} = B^{-1} N e_j$
$s = \frac{z^*_j}{\Delta z_j}$	$t = \frac{x^*_i}{\Delta x_i}$
$x^*_j \leftarrow t$	$x^*_j \leftarrow t$
$x^*_{\mathcal{B}} \leftarrow x^*_{\mathcal{B}} - t \Delta x_{\mathcal{B}}$	$x^*_{\mathcal{B}} \leftarrow x^*_{\mathcal{B}} - t \Delta x_{\mathcal{B}}$
$z^*_i \leftarrow s$	$z^*_i \leftarrow s$
$z^*_{\mathcal{N}} \leftarrow z^*_{\mathcal{N}} - s \Delta z_{\mathcal{N}}$	$z^*_{\mathcal{N}} \leftarrow z^*_{\mathcal{N}} - s \Delta z_{\mathcal{N}}$
$\mathcal{B} \leftarrow \mathcal{B} \setminus \{i\} \cup \{j\}$	$\mathcal{B} \leftarrow \mathcal{B} \setminus \{i\} \cup \{j\}$
}	}

Initially we set

$$\mathcal{N} = \{1, 2, \ldots, n\} \quad , \quad \mathcal{B} = \{n+1, n+2, \ldots, n+m\}$$

So we have

$$N = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \quad , \quad B = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}$$

and

$$c_{\mathcal{N}} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} \quad , \quad c_{\mathcal{B}} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Substituting these expressions into the definitions of $x^*_{\mathcal{B}}$, $z^*_{\mathcal{N}}$ and ζ^*, we find that

$$x^*_{\mathcal{B}} = B^{-1}b = b \quad , \quad z^*_{\mathcal{N}} = \left(B^{-1}N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} = -c_{\mathcal{N}} \quad , \quad \zeta^* = 0$$

$$\boxed{\begin{aligned} \zeta &= \zeta^* - (z^*_{\mathcal{N}})^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= x^*_{\mathcal{B}} - B^{-1} N x_{\mathcal{N}} \end{aligned}}$$

$$x^*_{\mathcal{B}} = B^{-1} b = b \quad , \quad z^*_{\mathcal{N}} = \left(B^{-1} N\right)^T c_{\mathcal{B}} - c_{\mathcal{N}} = -c_{\mathcal{N}} \quad , \quad \zeta^* = 0$$

Hence, the initial *primal* dictionary reads

$$\boxed{\begin{aligned} \zeta &= c^T_{\mathcal{N}} x_{\mathcal{N}} \\ x_{\mathcal{B}} &= b - N x_{\mathcal{N}} \end{aligned}}$$

$$\boxed{\begin{aligned} \zeta &= \quad c_{\mathcal{N}}^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= b - N x_{\mathcal{N}} \end{aligned}}$$

If $b \geq 0$ and $c \leq 0$
 // *primal feasibility + optimality observed* $\Rightarrow$ *stop*

If $b \geq 0$ and $c \not\leq 0$
 // *primal feasibility observed* $\Rightarrow$ *start with primal simplex*

If $b \not\geq 0$ and $c \leq 0$
 // *dual feasibility observed* $\Rightarrow$ *start with dual simplex*

If $b \not\geq 0$ and $c \not\leq 0$
 // *we need to do a Phase I*

$$\boxed{\begin{aligned} \zeta &= \quad c_{\mathcal{N}}^T x_{\mathcal{N}} \\ x_{\mathcal{B}} &= b - N x_{\mathcal{N}} \end{aligned}}$$

If $b \ngeq 0$ and $c \nleq 0$

a Phase I could be :

⇝ Define the *auxiliary problem*, as in Ch. 2, and apply primal simplex.
  // In Phase II, proceed with primal simplex.

⇝ Replace $c_{\mathcal{N}}$ with a *non-positive* one and apply dual simplex.
  // In Phase II, proceed with primal simplex.

⇝ Replace b with a *non-negative* one and apply primal simplex.
  // In Phase II, proceed with dual simplex.