Dual Simplex Method

$$\begin{array}{ll} \text{maximize} & -x_1 - x_2 \\ \text{subject to} & -2x_1 - x_2 \le 4 \\ & -2x_1 + 4x_2 \le -8 \\ & -x_1 + 3x_2 \le -7 \\ & x_1,\ x_2 \ge 0\,. \end{array}$$

$$\begin{array}{ll} \text{minimize} & 4y_1 - 8y_2 - 7y_3 \\ \text{subject to} & -2y_1 - 2y_2 - y_3 \ge -1 \\ & -y_1 + 4y_2 + 3y_3 \ge -1 \\ & y_1,\ y_2,\ y_3 \ge 0\,. \end{array}$$

Dual Simplex Method

$x_j z_j = 0$

$y_i w_i = 0$

(P)

$$
\begin{array}{rcl}
\zeta & = & \quad - x_1 - x_2 \\
\hline
w_1 & = & 4 + 2x_1 + x_2 \\
w_2 & = & -8 + 2x_1 - 4x_2 \\
w_3 & = & -7 + x_1 - 3x_2
\end{array}
$$

(D)

$$
\begin{array}{rcl}
-\xi & = & \quad - 4y_1 + 8y_2 + 7y_3 \\
\hline
z_1 & = & 1 - 2y_1 - 2y_2 - y_3 \\
z_2 & = & 1 - y_1 + 4y_2 + 3y_3 \, .
\end{array}
$$

$y_2 \sim w_2$

$z_1 \sim x_1$

(P)

$$
\begin{array}{rcl}
\zeta & = & -4 - 0.5w_2 - 3x_2 \\
\hline
w_1 & = & 12 + w_2 + 5x_2 \\
x_1 & = & 4 + 0.5w_2 + 2x_2 \\
w_3 & = & -3 + 0.5w_2 - x_2
\end{array}
$$

-4	-0.5	-3
12	1	5
4	0.5	2
-3	0.5	-1

(D)

$$
\begin{array}{rcl}
-\xi & = & 4 - 12y_1 - 4z_1 + 3y_3 \\
\hline
y_2 & = & 0.5 - y_1 - 0.5z_1 - 0.5y_3 \\
z_2 & = & 3 - 5y_1 - 2z_1 + y_3 \, .
\end{array}
$$

$y_3 \sim w_3$

$y_2 \sim w_2$

4	-12	-4	3
0.5	-1	-0.5	-0.5
3	-5	-2	1

(P)

$$
\begin{array}{rcl}
\zeta & = & -7 - w_3 - 4x_2 \\
\hline
w_1 & = & 18 + 2w_3 + 7x_2 \\
x_1 & = & 7 + w_3 + 3x_2 \\
w_2 & = & 6 + 2w_3 + 2x_2
\end{array}
$$

(D)

$$
\begin{array}{rcl}
-\xi & = & 7 - 18y_1 - 7z_1 - 6y_2 \\
\hline
y_3 & = & 1 - 2y_1 - z_1 - 2y_2 \\
z_2 & = & 4 - 7y_1 - 3z_1 - 2y_2 \, .
\end{array}
$$

Dual Simplex Method - Phase 1

$$\begin{array}{lrcr} \text{maximize} & -x_1 + 4x_2 & & \\ \text{subject to} & -2x_1 - x_2 & \le & 4 \\ & -2x_1 + 4x_2 & \le & -8 \\ & -x_1 + 3x_2 & \le & -7 \\ & x_1,\ x_2 & \ge & 0\,. \end{array}$$

$\eta = -x_1 - x_2$

$$\text{(P)} \qquad \begin{array}{rcl} \zeta & = & - x_1 + 4x_2 \\ \hline w_1 & = & 4 + 2x_1 + x_2 \\ w_2 & = & -8 + 2x_1 - 4x_2 \\ w_3 & = & -7 + x_1 - 3x_2\,, \end{array}$$

$$\text{(D)} \qquad \begin{array}{rcl} -\xi & = & - 4y_1 + 8y_2 + 7y_3 \\ \hline z_1 & = & 1 - 2y_1 - 2y_2 - y_3 \\ z_2 & = & -4 - y_1 + 4y_2 + 3y_3\,. \end{array}$$

$$\text{(D')} \qquad \begin{array}{rcl} -\xi & = & -4y_1 + 8y_2 + 7y_3 \\ z_1 & = & 1 - 2y_1 - 2y_2 - y_3 \\ z_2 & = & 1 - y_1 + 4y_2 + 3y_3 \end{array}$$

$$\begin{array}{rcl} \eta & = & -7 - \ \ w_3 - 4x_2 \\ \hline w_1 & = & 18 + 2w_3 + 7x_2 \\ x_1 & = & \ \ 7 + \ \ w_3 + 3x_2 \\ w_2 & = & \ \ 6 + 2w_3 + 2x_2 \ . \end{array}$$

$\zeta = -x_1 + 4x_2 = -(7 + w_3 + 3x_2) + 4x_2$

$$\begin{array}{rcl} \zeta & = & -7 - \ \ w_3 + \ \ x_2 \\ \hline w_1 & = & 18 + 2w_3 + 7x_2 \\ x_1 & = & \ \ 7 + \ \ w_3 + 3x_2 \\ w_2 & = & \ \ 6 + 2w_3 + 2x_2 \ . \end{array}$$

The dual simplex algorithm: example

$$
\begin{array}{rcrcrcrcr}
\eta & = & 12 & - & 4x_1 & - & x_2 & - & x_3 \\
\hline
x_4 & = & -4 & + & 3x_1 & - & 11x_2 & + & x_3 \\
x_5 & = & 3 & - & x_1 & + & 3x_2 & - & 2x_3
\end{array}
$$

1. dual pivot: x_4 leaves and x_3 enters

$$
\begin{array}{rcrcrcrcr}
\eta & = & 8 & - & x_1 & - & 12x_2 & - & x_4 \\
\hline
x_3 & = & 4 & - & 3x_1 & + & 11x_2 & + & x_4 \\
x_5 & = & -5 & + & 5x_1 & - & 19x_2 & - & 2x_4
\end{array}
$$

2. dual pivot: x_5 leaves and x_1 enters

$$
\begin{array}{rcrcrcrcr}
\eta & = & 7 & - & 0.2x_5 & - & 15.8x_2 & - & 1.4x_4 \\
\hline
x_3 & = & 1 & - & 0.6x_5 & - & 0.4x_2 & - & 0.2x_4 \\
x_1 & = & 1 & + & 0.2x_5 & + & 3.8x_2 & + & 0.4x_4
\end{array}
$$

Dual feasible, so go on with primal pivots: done right away!

Duality, other forms

Our usual form of (P) and (D) is:

$$
\begin{array}{lllll}
\text{(P)} & \max & \sum_{j=1}^{n} c_j x_j & & \\
& \text{subject to} & & & \\
& & \sum_{j=1}^{n} a_{ij} x_j & \leq b_i & \text{for } i = 1, \ldots, m \\
& & x_j & \geq 0 & \text{for } j = 1, \ldots, n.
\end{array}
$$

$$
\begin{array}{lllll}
\text{(D)} & \min & \sum_{i=1}^{m} b_i y_i & & \\
& \text{subject to} & & & \\
& & \sum_{i=1}^{m} y_i a_{ij} & \geq c_j & \text{for } j = 1, \ldots, n \\
& & y_i & \geq 0 & \text{for } i = 1, \ldots, m.
\end{array}
$$

In matrix form:

$$
\begin{array}{ll}
\text{(P)} & \max\{c^T x : Ax \leq b,\ x \geq O\} \\
\text{(D)} & \min\{b^T y : A^T y \geq c,\ y \geq O\}.
\end{array}
$$

One may meet LP problems in other forms. But: every LP problem may be rewritten in the form (P). To do so, certain techniques are needed:

$[E = b] \Leftrightarrow [E \leq b \,,\, -E \leq -b]$

- each equation is written as two inequalities
- $\min f = -\max(-f)$

  $x = x^{+} - x^{-}$
- a free variable x is replaced by $x^{+} - x^{-}$ where $x^{+}, x^{-} \geq 0$

One may then (if desirable) find the dual problem (since the primal now has the "right" form) and write this in the simplest form possible.

It is important to practise the techniques to

- write any LP problem on the form (P), and
- find the dual of any LP problem.

It is recommended to use the matrix form in this rewriting of the problem.

We then need to work on partitioned matrices, see section on this in the linear algebra book (MAT1120). In particular, we need a rule for matrix multiplication:

$$\begin{array}{c} \begin{array}{cc} \overbrace{\quad}^{m_1} & \overbrace{\quad}^{m_2} \end{array} \\ \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \end{array} \begin{bmatrix} x^1 \\ x^2 \end{bmatrix} \begin{matrix} \}\, n_1 \\ \}\, n_2 \end{matrix} = \begin{bmatrix} A_{11}x^1 + A_{12}x^2 \\ A_{21}x^1 + A_{22}x^2 \end{bmatrix}$$

Another useful rule is for the transpose of a partitioned matrix:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^T = \begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix}$$

Example:

$$\max\{c^T x^1 + d^T x^2 : A^1 x^1 \geq b^1,\ A^2 x^1 + A^3 x^2 \leq b^2,\ x^1, x^2 \geq O\}$$

Here the variables are x^1 og x^2 (suitable vectors). We may write this in the form (P):

$$\Leftrightarrow \quad -A^1 x^1 \leq -b^1$$
$$A^2 x^1 + A^3 x^2 \leq b^2$$

$$\begin{bmatrix} x^1 \\ x^2 \end{bmatrix} \geq 0$$

$$\max\{\begin{bmatrix} c \\ d \end{bmatrix}^T \begin{bmatrix} x^1 \\ x^2 \end{bmatrix} : \begin{bmatrix} -A^1 & O \\ A^2 & A^3 \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \end{bmatrix} \leq \begin{bmatrix} -b^1 \\ b^2 \end{bmatrix},\ x^1, x^2 \geq O\}$$

Then the dual may be determined and, finally, one sees if the dual may be simplified.

This, and related, examples are given on the blackboard. (For instance, where a variable vector x is free (that is, no sign constraint) and is replaced by $x' - x''$ where $x', x'' \geq O$.)

Last comment on this, a connection between the primal and the dual:

- an equation in one of the problems corresponds to a free variable in the other problem,
- an inequality in one of the problems corresponds to a nonnegative variable in the other problem.

$$E = b \iff \overset{(P)}{\begin{array}{l} E \le b \\ -E \le -b \end{array}} \quad \longrightarrow \quad \overset{(D)}{\begin{array}{l} y_1 \ge 0 \\ y_2 \ge 0 \end{array}}$$

always $\underset{\displaystyle y}{\underset{\parallel}{y_1 - y_2}}$ or $\underset{\displaystyle -y}{\underset{\parallel}{y_2 - y_1}}$

y not constr.

Resource Allocation

Recall the resource allocation problem ($m = 2$, $n = 3$):

$$\begin{array}{lrcrcrcl} \text{maximize} & c_1x_1 & + & c_2x_2 & + & c_3x_3 & & \\ \text{subject to} & a_{11}x_1 & + & a_{12}x_2 & + & a_{13}x_3 & \leq & b_1 \\ & a_{21}x_1 & + & a_{22}x_2 & + & a_{23}x_3 & \leq & b_2 \\ & & & & x_1, & x_2, \; x_3 & \geq & 0, \end{array}$$

where

$$\begin{aligned} c_j &= \text{profit per unit of product } j \text{ produced} \\ b_i &= \text{units of raw material } i \text{ on hand} \\ a_{ij} &= \text{units raw material } i \text{ required to produce } 1 \text{ unit of prod } j. \end{aligned}$$

Closing Up Shop

If we produce one unit less of product j, then we free up:

- a_{1j} units of raw material 1 and
- a_{2j} units of raw material 2.

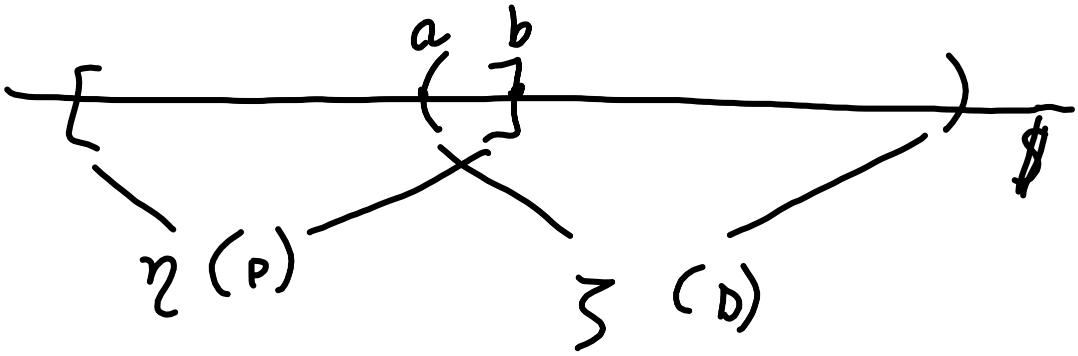
Selling these unused raw materials for y_1 and y_2 dollars/unit yields $a_{1j}y_1 + a_{2j}y_2$ dollars.
Only interested if this exceeds lost profit on each product j:

$$a_{1j}y_1 + a_{2j}y_2 \geq c_j, \qquad j = 1, 2, 3.$$

Consider a buyer offering to purchase our entire inventory.
Subject to above constraints, buyer wants to minimize cost:

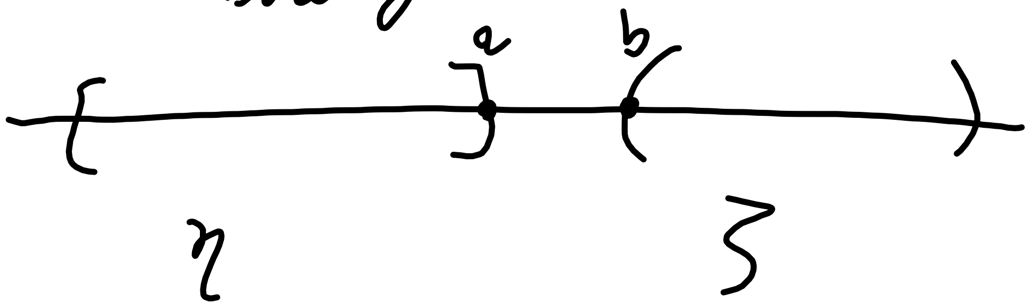
$$\begin{array}{lrcrcl} \text{minimize} & b_1y_1 & + & b_2y_2 & & \\ \text{subject to} & a_{11}y_1 & + & a_{21}y_2 & \geq & c_1 \\ & a_{12}y_1 & + & a_{22}y_2 & \geq & c_2 \\ & a_{13}y_1 & + & a_{23}y_2 & \geq & c_3 \\ & & & y_1, \ y_2 & \geq & 0 \ . \end{array}$$

⊙ no weak d.



-a +b -a +b ...

⊙ no strong d.



-b +a

LP. Lecture 6: Ch. 6: the simplex method in matrix form, and Section 7.1: sensitivity analysis

- matrix notation
- simplex algorithm in matrix notation
- example
- negative transpose property: proof
- sensitivity analysis (section 7.1)

Matrix notation

Dictionary contra matrix form:

- dictionary best for understanding simplex algorithm and calculation by hand of minor examples
- in larger calculation the simplex algorithm in matrix form is used.
- matrix form is more efficient. Uses numerical linear algebra.
- important questions: (i) pricing, (ii) quick updating of basis, (iii) LU-factorization, (iv) exploiting sparsity

We will just explain the algorithm in matrix form, without discussing the numerical questions.

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & & \\ b_{n1} & \cdots & b_{nm} \end{bmatrix}$$

$$A \cdot B = C \qquad c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

⊙ (def) $\mathcal{O}(n^3)$

⊙ 1969 Strassen $\mathcal{O}(n^{2.803\ldots})$

(tensor factorization)

Consider LP problem in standard form

$$\begin{array}{ll} \max & \sum_{j=1}^{n} c_j x_j \\ \text{s.t.} & \\ & \begin{array}{rll} \sum_{j=1}^{n} a_{i,j} x_j & \leq b_i & \text{for } i = 1, \ldots, m \\ x_j & \geq 0 & \text{for } j = 1, \ldots, n. \end{array} \end{array}$$

Converting to equations by using slack variables.

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{i,j} x_j \quad \text{for } i = 1, \ldots, m.$$

Matrix form:

$$\begin{array}{ll} \max & c^T x \\ \text{s.t.} & \\ & \begin{array}{rcl} Ax & = & b, \\ x & \geq & O \end{array} \end{array}$$

where

$$(Ax)_i = \sum_{k=1}^{n+m} A_{ik} x_k = \sum_{k=1}^{n} a_{ik} x_k + x_{n+i} = b_i$$

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n} & 1 & 0 & \dots & 0 \\ a_{21} & \dots & a_{2n} & 0 & 1 & \dots & 0 \\ \vdots & & \vdots & & & \ddots & \\ a_{m1} & \dots & a_{mn} & 0 & 0 & & 1 \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix},$$

Note: A has full rowrank, which means that the rows in A are linearly independent.

$$c = \begin{bmatrix} c_1 \\ \vdots \\ c_n \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ x_{n+1} \\ \vdots \\ x_{n+m} \end{bmatrix}$$

step 0 $B = \{n+1, \dots, n+m\}$

$N = \{1, \dots, n\}$

$\Rightarrow A_B = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix}$

The objective function is $c^T x = \sum_{j=1}^{n} c_j x_j$.

The simplex algorithm will in each iteration have split the variables into two groups: the basic and the nonbasic variables. Will as usual let B and N be the index sets of, respectively, the basic variables and the nonbasic variables.

We let A_B and A_N be the submatrices of A which corresponds to the columns with indices B and N, respectively. So, we have

$$P \cdot A = \begin{bmatrix} A_B & A_N \end{bmatrix} \quad , P \text{ perm. matrix}$$

Note: here $B = \{1, \ldots, m\}$, but this is just to simplify the notation. In general the basis indices are spread out. Mathematically we can imagine permuting columns in A and elements in x (correspondingly) so that we have the form above.

Primal simplex algorithm

$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$ $x_B = \begin{bmatrix} x_1 \\ x_3 \end{bmatrix}$ $x_N = \begin{bmatrix} x_2 \\ x_4 \end{bmatrix}$

Split x and c similarly as

$$x = \begin{bmatrix} x_B \\ x_N \end{bmatrix}, \quad c = \begin{bmatrix} c_B \\ c_N \end{bmatrix}$$

With that

$$\begin{aligned} Ax &= \begin{bmatrix} A_B & A_N \end{bmatrix} \begin{bmatrix} x_B \\ x_N \end{bmatrix} = A_B\, x_B + A_N\, x_N, \\ c^T x &= \begin{bmatrix} c_B^T & c_N^T \end{bmatrix} \begin{bmatrix} x_B \\ x_N \end{bmatrix} = c_B^T x_B + c_N^T x_N. \end{aligned}$$

The set of equations $Ax = b$ is now

$$A_B^{-1} \cdot \quad A_B x_B + A_N x_N = b \quad A_B^{-1} \cdot$$

We now assume that A_B is nonsingular; A_B is then called a basis in A. The columns in A_B are then a basis for $\mathbb{R}^m$ (i.e., m linearly independent vectors in $\mathbb{R}^m$). Solve the set of equations:

$$x_B = A_B^{-1}b - A_B^{-1}A_N x_N \qquad (1)$$

which expresses the basis variables x_B through the nonbasic variables x_N.

Note: *Any* solution of $Ax = b$ can be written in this form $x = \begin{bmatrix} x_B \\ x_N \end{bmatrix}$ where x_N is chosen appropriately and x_B is determined uniquely based on (1).

We now eliminate x_B from the objective function:

$$\begin{aligned} \eta &= c_B^T x_B + c_N^T x_N = \\ &= c_B^T(A_B^{-1}b - A_B^{-1}A_N x_N) + c_N^T x_N = \\ &= c_B^T A_B^{-1}b - ((A_B^{-1}A_N)^T c_B - c_N)^T x_N. \end{aligned}$$

We now have

$$
\begin{array}{rcl}
\eta & = & \overset{\bar{\eta}}{\overset{\shortparallel}{c_B^T A_B^{-1} b}} \;+\; (c_N - (A_B^{-1} A_N)^T c_B)^T x_N \\
\hline
x_B & = & A_B^{-1} b \;-\; A_B^{-1} A_N x_N
\end{array}
$$

which is the dictionary we have used so far. Here

$$
\begin{aligned}
c_B^T A_B^{-1} b &= \bar{\eta} \\
(c_N - (A_B^{-1} A_N)^T c_B)^T &= [\bar{c}_j] \\
A_B^{-1} b &= [\bar{b}_i] \\
A_B^{-1} A_N &= [\bar{a}_{ij}].
\end{aligned}
$$

where the vectors on the right-hand side have components indexed with $i \in B$ and $j \in N$.

The basic solution Associated to the dictionary (the choice of basis B) is

$$x_N^* = O, \;\; x_B^* = A_B^{-1} b.$$

Will now look at the dual. Recall the correspondence

- the primal variable x_j corresponds to the dual slack variable z_j
- the primal slack variable w_i corresponds to the dual slack variable y_i

We say that x_j and z_j are complementary, and that w_i and y_i are complementary. Complementary variables have opposite roles in the equations: they are on the opposite sides.

This means that: a variable is in basis if and only if the complementary variable is out of basis.

$w_i \Rightarrow x_{n+i}$, $y_i = z_{n+i}$

Example:

$$
\text{(P)}\quad
\begin{array}{rcrcrcrcr}
\eta & = & 0 & + & 4x_1 & + & x_2 & + & 3x_3 \\
\hline
w_1 & = & 1 & - & x_1 & - & 4x_2 & & \\
w_2 & = & 3 & - & 3x_1 & + & x_2 & - & x_3
\end{array}
$$

$$
\text{(D)}\quad
\begin{array}{rcrcrcr}
-\xi & = & 0 & - & y_1 & - & 3y_2 \\
\hline
z_1 & = & -4 & + & y_1 & + & 3y_2 \\
z_2 & = & -1 & + & 4y_1 & - & y_2 \\
z_3 & = & -3 & & & + & y_2
\end{array}
$$

Initially x_1 is not in the basis in (P), while the complementary variable z_1 is in basis in (D) and so on. Pivot now in (P) by taking x_3 into basis and w_2 out of basis. By corresponding pivot in (D): z_3 (complementary of x_3) goes out of basis and y_2 (complementary of w_2) goes into basis.

In each pivot in (P) one basic variable and one nonbasic variable switch roles. By a corresponding pivot in (D) the complementary variables in (D) will also switch roles, but the opposite way.

This means that complementary variables still have opposite roles when it comes to being in basis. Because of this complementary property we choose to arrange the variables in the two problems like this:

$$
\begin{aligned}
(x_1, \dots, x_n, w_1, \dots, w_m) &\quad\rightarrow\quad (x_1, \dots, x_n, x_{n+1}, \dots, x_{n+m}) \\
(z_1, \dots, z_n, y_1, \dots, y_m) &\quad\rightarrow\quad (z_1, \dots, z_n, z_{n+1}, \dots, z_{n+m})
\end{aligned}
$$

So then x_j and z_j are complementary for $j = 1, \dots, n+m$. In particular, the basic variables in (D) be z_N (not z_B!).

compl. slack : $x_j z_j = 0$

	(P)	(D)
BASIC	x_B	z_N
NON-BASIC	x_N	z_B

Because of the negative transpose property the dual dictionary (with basis B) is given by:

$$\begin{array}{rcrcr} -\xi & = & -c_B^T A_B^{-1} b & - & (A_B^{-1} b)^T z_B \\ \hline z_N & = & (A_B^{-1} A_N)^T c_B - c_N & + & (A_B^{-1} A_N)^T z_B. \end{array}$$

The corresponding dual basic solution for this dictionary is

$$z_B^* = O, \ \ z_N^* = (A_B^{-1} A_N)^T c_B - c_N.$$

We now introduce

$$\eta^* = c_B^T A_B^{-1} b$$

which is the value of the objective function η in (P) for the basis solution associated to B.

Conclusion: primal and dual dictionary for basis B now becomes

(P)
$$\begin{array}{rcrcl} \eta & = & \eta^* & - & (z_N^*)^T x_N \\ \hline x_B & = & x_B^* & - & A_B^{-1} A_N x_N \end{array} \qquad (2)$$

(D)
$$\begin{array}{rcrcl} -\xi & = & -\eta^* & - & (x_B^*)^T z_B \\ \hline z_N & = & z_N^* & + & (A_B^{-1} A_N)^T z_B \end{array} \qquad (3)$$

Note the negative transpose property.

$$\begin{bmatrix} \eta^* & -(z_N^*)^T \\ x_B^* & -A_B^{-1} A_N \end{bmatrix}$$

$$\begin{bmatrix} -\eta^* & -(x_B^*)^T \\ z_N^* & (A_B^{-1} A_N)^T \end{bmatrix}$$

simplex algorithm (primal) in short version:

- starts with a basis B so that x_B^* is feasible in (P)
- makes thereafter a sequence of pivots.
- each pivot is to find a neighboring basis (that equals the previous basis except for one index) so that η increases and also determine the corresponding primal- and dual basic solutions.

The simplex algorithm in matrix form will find the same solutions (in each iteration) as the dictionary approach. The difference is just that we are now going to operate with matrices and vectors. We use the notation from the dictionary form above.

An iteration in the simplex algorithm:

Step 1. Test optimality. If $z_N^* \geq 0$, stop. The present basic solution is optimal.

Step 2. Choose entering basic variable. Choose an index $j \in N$ where $z_j^* < 0$. Call x_j entering basis variable.

Step 3. Calculate the primal search direction. Will now let $x_N = te_j$ where e_j is the j'th unit vector; this determines the change of (primal) nonbasic variables. The primal basic variables will then be given by (see (2))

$$x_B = x_B^* - A_B^{-1} A_N t e_j = x_B^* - t \cdot \Delta x_B \tag{4}$$

where the search direction is given by

$$\Delta x_B = A_B^{-1} A_N e_j.$$

(Δx_B contains the coordinates of the jth column in A_N expressed in basis A_B.)

Step 4. Calculate primal step length. We choose t as large as possible so that x_B still is nonnegative. From equation (4) we get that the new value of the basic variable x_i is

$$x_i = x_i^* \underbrace{- t \cdot \Delta_i}.$$

$x_i \geq 0$

$x_i^* \geq t\Delta_i$

$t \leq \frac{x_i^*}{\Delta_i}$

So if $\Delta_i \leq 0$ for all i, the problem (P) is unbounded. Otherwise the maximal t is given by

$$t = \min\{x_i^*/\Delta_i : \Delta_i > 0\}. \tag{5}$$

Based on steps 3 and 4 we can determine the new primal solution (see Step 8).

Step 5. Choose the leaving basic variable. Choose an index i where the minimum occurred in (5), and let x_i be the leaving basic variable.

Step 6. Calculate the dual search direction. It still remains to find the change in the dual variables (we need these to find the new coefficients in the objective function in (P)). This is determined by the choice of i and j over. Since x_i leaves basis in (P), the complementary variable z_i will go into basis in (D), so it has to be increased from zero to a certain value s. The dual basic variables are then given by (see (3))

$z_i = s$

$$z_N = z_N^* + (A_B^{-1}A_N)^T s e_i = z_N^* - s \cdot \Delta z_N \tag{6}$$

where the search direction is given by

$z_B = s e_i$

$$\Delta z_N = -(A_B^{-1}A_N)^T e_i.$$

Step 7. Calculate the dual step length. We can determine the dual step length s based on that z_j leaves basis (which happens because the complementary variable x_j goes into primal basis). Since z_j becomes zero we get from (6) that

$$s = z_j^*/\Delta_j.$$

Step 8. Update primal and dual solution. Primal solution is updated by

$$x_j^* := t, \quad x_B^* := x_B^* - t \cdot \Delta x_B$$

and dual solution is updated by

$$z_i^* := s, \quad z_N^* := z_N^* - s \cdot \Delta z_N.$$

Step 9. Update basis. Finally the basis is updated by

$$B := (B \setminus \{i\}) \cup \{j\}.$$

$$N := (N \setminus \{j\}) \cup \{i\}$$

Final comments:

- example: see section 6.3 in Vanderbei's book
- dual simplex in matrix form: see section 6.4
- summary: see slide number 2.