Linear Programming

Linear Optimization Problem

$$\begin{array}{rll} \text{minimize} & f(x) & \\ \text{by varying} & x \in \mathbb{R}^n & \\ \text{subject to} & g_i(x) \leq 0 & i = 1, \ldots, n_g \\ & h_j(x) = 0 & j = 1, \ldots, n_h \end{array}$$

We assume that

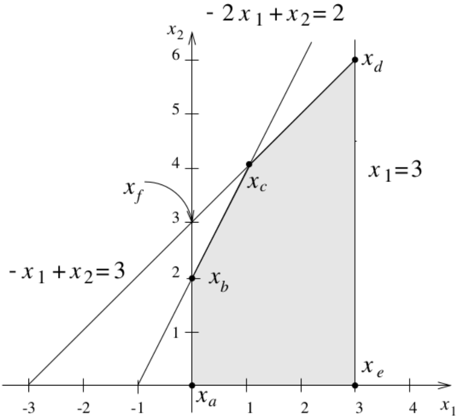
- f is linear, i.e.,

$$f(x) = c^\top x \qquad \text{here } c \in \mathbb{R}^n$$

- each g_i is linear,
- each h_j is linear.

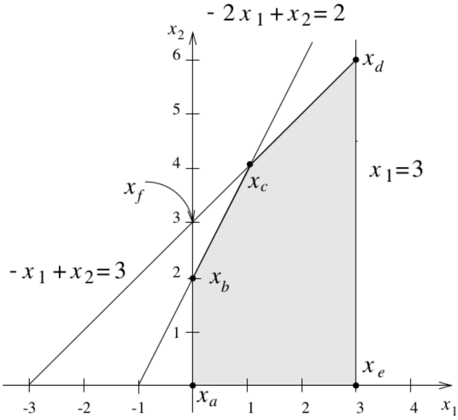
For convenience, in what follows, we also allow constraints of the form $g_i(x) \geq 0$.

Example



$$
\begin{aligned}
\text{minimize} \quad & z = -x_1 - 2x_2 \\
\text{subject to} \quad & -2x_1 + x_2 - 2 \leq 0 \\
& -x_1 + x_2 - 3 \leq 0 \\
& x_1 - 3 \leq 0 \\
& x_1, x_2 \geq 0.
\end{aligned}
$$

Example



The lines define the boundaries of the feasible region

$$-2x_1 + x_2 = 2$$
$$-x_1 + x_2 = 3$$
$$x_1 = 3$$

$$x_1 = 0$$
$$x_2 = 0$$

Standard Form

The *standard form linear program*

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

Here

- $x = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n$
- $c = (x_1, \ldots, c_n)^\top \in \mathbb{R}^n$
- A is an $m \times n$ matrix of elements a_{ij} where $m < n$ and rank$(A) = m$

  That is, all rows of A are linearly independent.
- $b = (b_1, \ldots, b_n)^\top \geq 0$

  $b \geq 0$ means $b_i \geq 0$ for all i.

Standard Form

The *standard form linear program*

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

Here

- $x = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n$
- $c = (x_1, \ldots, c_n)^\top \in \mathbb{R}^n$
- A is an $m \times n$ matrix of elements a_{ij} where $m < n$ and $\text{rank}(A) = m$

  That is, all rows of A are linearly independent.
- $b = (b_1, \ldots, b_n)^\top \geq 0$

  $b \geq 0$ means $b_i \geq 0$ for all i.

Every linear optimization problem can be transformed into a standard linear program such that there is a one-to-one correspondence between solutions of the constraints preserving values of the objective.

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

2. Transform every $g_i(x) \leq 0$ to $g_i(x) + s_i = 0, s_i \geq 0$. Here s_i are new variables (*slack variables*).

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

2. Transform every $g_i(x) \leq 0$ to $g_i(x) + s_i = 0, s_i \geq 0$. Here s_i are new variables (*slack variables*).

3. Move all constant terms to the right side of the constraints.

   Now we have constraints of the form $Ax = b, x \geq 0$.

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

2. Transform every $g_i(x) \leq 0$ to $g_i(x) + s_i = 0, s_i \geq 0$. Here s_i are new variables (*slack variables*).

3. Move all constant terms to the right side of the constraints.

   Now we have constraints of the form $Ax = b, x \geq 0$.

4. Remove linearly dependent equations from $Ax = b$.

   This step does not alter the set of solutions.

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

2. Transform every $g_i(x) \leq 0$ to $g_i(x) + s_i = 0, s_i \geq 0$. Here s_i are new variables (*slack variables*).

3. Move all constant terms to the right side of the constraints.

   Now we have constraints of the form $Ax = b, x \geq 0$.

4. Remove linearly dependent equations from $Ax = b$.

   This step does not alter the set of solutions.

5. If $m \geq n$, the constraints either have a unique or no solution. Neither of the cases is interesting for optimization. Hence, $m < n$.

Transformation to Standard Form

1. For every variable x_i introduce new variables x_i', x_i'', replace every occurrence of x_i with $x_i' - x_i''$, and introduce constraints $x_i', x_i'' \geq 0$.

   Note that if a constraint is in the form $x_i + \zeta \geq 0$ we may simply replace x_i with $x_i' - \zeta$ and introduce $x_i' \geq 0$.

2. Transform every $g_i(x) \leq 0$ to $g_i(x) + s_i = 0, s_i \geq 0$. Here s_i are new variables (*slack variables*).

3. Move all constant terms to the right side of the constraints.

   Now we have constraints of the form $Ax = b, x \geq 0$.

4. Remove linearly dependent equations from $Ax = b$.

   This step does not alter the set of solutions.

5. If $m \geq n$, the constraints either have a unique or no solution. Neither of the cases is interesting for optimization. Hence, $m < n$.

6. Multiplying equations with $b_i < 0$ by -1 gives $b \geq 0$

Transformation Example

$$\begin{aligned} \text{maximize} \quad & z = -5x_1 - 3x_2 \\ \text{subject to} \quad & 3x_1 - 5x_2 - 5 \leq 0 \\ & -4x_1 - 9x_2 + 4 \leq 0 \end{aligned}$$

Transformation Example

$$
\begin{array}{ll}
\text{maximize} & z = -5x_1 - 3x_2 \\
\text{subject to} & 3x_1 - 5x_2 - 5 \leq 0 \\
 & -4x_1 - 9x_2 + 4 \leq 0
\end{array}
$$

Introduce the bounded variables:

$$
\begin{array}{ll}
\text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\
\text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' - 5 \leq 0 \\
 & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + 4 \leq 0 \\
 & x_1', x_1'', x_2', x_2'' \geq 0
\end{array}
$$

Transformation Example

$$\begin{array}{ll} \text{maximize} & z = -5x_1 - 3x_2 \\ \text{subject to} & 3x_1 - 5x_2 - 5 \leq 0 \\ & -4x_1 - 9x_2 + 4 \leq 0 \end{array}$$

Introduce the bounded variables:

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' - 5 \leq 0 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + 4 \leq 0 \\ & x_1', x_1'', x_2', x_2'' \geq 0 \end{array}$$

Introduce the slack variables:

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 - 5 = 0 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 + 4 = 0 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Transformation Example

$$\begin{aligned}
\text{maximize} \quad & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\
\text{subject to} \quad & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 - 5 = 0 \\
& -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 + 4 = 0 \\
& x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0
\end{aligned}$$

Transformation Example

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 - 5 = 0 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 + 4 = 0 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Move constants to the right:

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 = 5 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 = -4 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Transformation Example

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 - 5 = 0 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 + 4 = 0 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Move constants to the right:

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 = 5 \\ & -4x_1' + 4x_1'' - 9x_2' + 9x_2'' + s_2 = -4 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Check if all equations are linearly independent.

Multiply the last one with -1:

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 = 5 \\ & 4x_1' - 4x_1'' + 9x_2' - 9x_2'' - s_2 = 4 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

Transformation Example

$$\begin{array}{ll} \text{maximize} & z = -5x_1' + 5x_1'' - 3x_2' + 3x_2'' \\ \text{subject to} & 3x_1' - 3x_1'' - 5x_2 + 5x_2'' + s_1 = 5 \\ & 4x_1' - 4x_1'' + 9x_2' - 9x_2'' - s_2 = 4 \\ & x_1', x_1'', x_2', x_2'', s_1, s_2 \geq 0 \end{array}$$

In matrix form:

$$A = \begin{pmatrix} 3 & -3 & -5 & 5 & 1 & 0 \\ 4 & -4 & 9 & -9 & 0 & -1 \end{pmatrix}$$

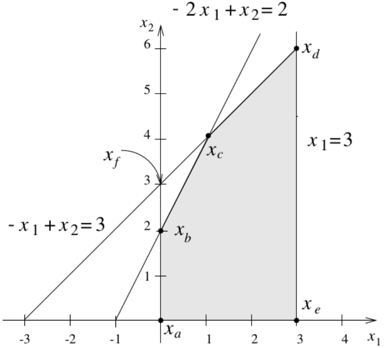
$$x = (x_1, x_2, x_3, x_4, s_1, s_2)^\top$$

$$b = (5, 4)^\top$$

$$Ax = b \text{ where } x \geq 0$$

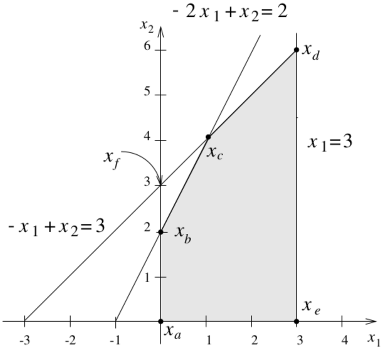
$$c = (-5, 5, -3, 3)^\top$$

Example



$$
\begin{aligned}
\text{minimize} \quad & z = -x_1 - 2x_2 \\
\text{subject to} \quad & -2x_1 + x_2 - 2 \leq 0 \\
& -x_1 + x_2 - 3 \leq 0 \\
& x_1 - 3 \leq 0 \\
& x_1, x_2 \geq 0.
\end{aligned}
$$

Example



In the standard form:

$$
\begin{array}{rr}
\text{minimize} & z = -x_1 - 2x_2 \\
\text{subject to} & -2x_1 + x_2 + s_1 = 2 \\
& -x_1 + x_2 + s_2 = 3 \\
& x_1 + x_5 = 3 \\
& x_1, x_2, s_1, s_2, x_5 \geq 0
\end{array}
$$

Assumptions

Consider a linear programming problem in the standard form:

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

Assumptions

Consider a linear programming problem in the standard form:

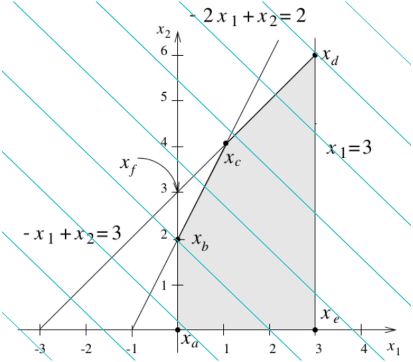
$$\begin{aligned} \text{minimize} \quad & c^\top x \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

In what follows, we will use the following shorthand: Given two column vectors x, x', we write $[x, x']$ to denote the vector resulting from stacking x on top of x'.

Solutions

There are (typically) infinitely many solutions to the constraints.

Are there some distinguished ones? How do you find minimizers?



Here, the blue lines are contours of $-x_1 - x_2$.

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Denote by N the set of indices of columns not in B.

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Denote by N the set of indices of columns not in B.

Given $x \in \mathbb{R}^n$, we let

- $x_B \in \mathbb{R}^m$ consist of components of x with indices in B
- $x_N \in \mathbb{R}^{n-m}$ consist of components of x with indices in N

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Denote by N the set of indices of columns not in B.

Given $x \in \mathbb{R}^n$, we let

- $x_B \in \mathbb{R}^m$ consist of components of x with indices in B
- $x_N \in \mathbb{R}^{n-m}$ consist of components of x with indices in N

Abusing notation, we denote by B and N the submatrices of A consisting of columns with indices in B and N, resp.

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Denote by N the set of indices of columns not in B.

Given $x \in \mathbb{R}^n$, we let

- $x_B \in \mathbb{R}^m$ consist of components of x with indices in B
- $x_N \in \mathbb{R}^{n-m}$ consist of components of x with indices in N

Abusing notation, we denote by B and N the submatrices of A consisting of columns with indices in B and N, resp.

Now, by appropriately shifting columns of A, we may write:

$$A = (B\ N) \qquad x = [x_B, x_N]$$

Basic Solutions

Assume that the matrix A has full row rank (w.l.o.g).

Let B be a set of m indices of columns of A for a linearly independent set. Such a B is called a *basis*.

Denote by N the set of indices of columns not in B.

Given $x \in \mathbb{R}^n$, we let

- $x_B \in \mathbb{R}^m$ consist of components of x with indices in B
- $x_N \in \mathbb{R}^{n-m}$ consist of components of x with indices in N

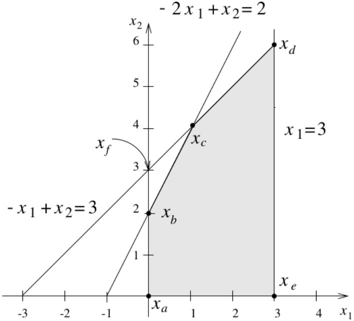
Abusing notation, we denote by B and N the submatrices of A consisting of columns with indices in B and N, resp.

Now, by appropriately shifting columns of A, we may write:

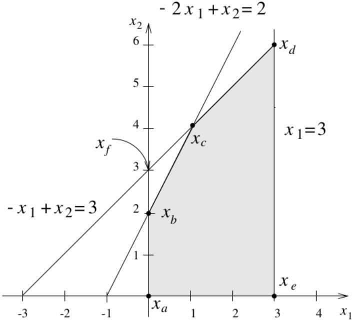
$$A = (B\ N) \qquad x = [x_B, x_N]$$

$x = [x_B, x_N] \in \mathbb{R}^n$ is a *basic solution w.r.t. the basis B* if $Ax = b$ and $x_N = 0$. Components of x_B are *basic variables*.
A basic solution x is *feasible* if $x \geq 0$.

$$
\begin{aligned}
-2x_1 + x_2 + x_3 &= 2 \\
-x_1 + x_2 + x_4 &= 3 \\
x_1 + x_5 &= 3 \\
x_1, x_2, x_3, x_4, x_5 &\geq 0
\end{aligned}
$$

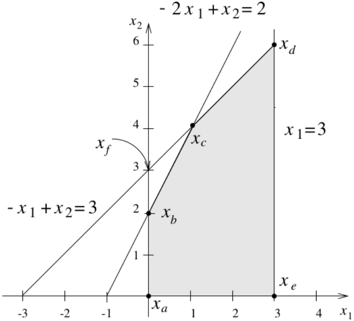


$$
\begin{aligned}
-2x_1 + x_2 + x_3 &= 2 \\
-x_1 + x_2 + x_4 &= 3 \\
x_1 + x_5 &= 3 \\
x_1, x_2, x_3, x_4, x_5 &\geq 0
\end{aligned}
$$



$$
A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}
$$

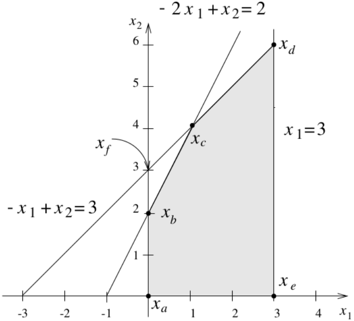
$$
\begin{aligned}
-2x_1 + x_2 + x_3 &= 2 \\
-x_1 + x_2 + x_4 &= 3 \\
x_1 + x_5 &= 3 \\
x_1, x_2, x_3, x_4, x_5 &\geq 0
\end{aligned}
$$



$$
A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}
$$

$$
x = (x_1, x_2, x_3, x_4, x_5)^\top
$$

$$
\begin{aligned}
-2x_1 + x_2 + x_3 &= 2 \\
-x_1 + x_2 + x_4 &= 3 \\
x_1 + x_5 &= 3 \\
x_1, x_2, x_3, x_4, x_5 &\geq 0
\end{aligned}
$$

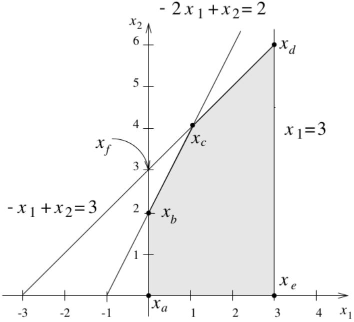


$$
A = (u_1 \; u_2 \; u_3 \; u_4 \; u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}
$$

$$
x = (x_1, x_2, x_3, x_4, x_5)^\top
$$

$$
b = (2, 3, 3)^\top
$$

$$
\begin{aligned}
-2x_1 + x_2 + x_3 &= 2 \\
-x_1 + x_2 + x_4 &= 3 \\
x_1 + x_5 &= 3 \\
x_1, x_2, x_3, x_4, x_5 &\geq 0
\end{aligned}
$$



$$
A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}
$$

$$
x = (x_1, x_2, x_3, x_4, x_5)^\top
$$

$$
b = (2, 3, 3)^\top
$$

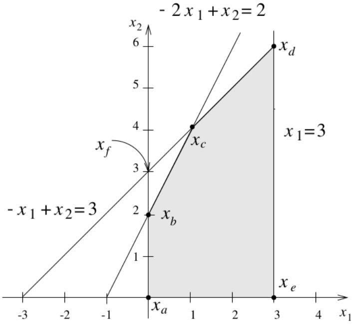
$Ax = b$ where $x \geq 0$

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_3, x_4, x_5\}$ with

$$B = (u_3\ u_4\ u_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

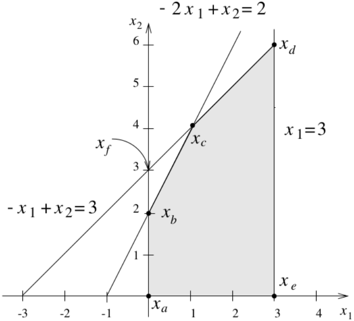
What is x_B satisfying $Bx_B = b$?

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_3, x_4, x_5\}$ with

$$B = (u_3\ u_4\ u_5) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

What is x_B satisfying $Bx_B = b$? $\quad x_B = (x_3, x_4, x_5)^\top = (2, 3, 3)^\top$.

The corresponding basic solution is

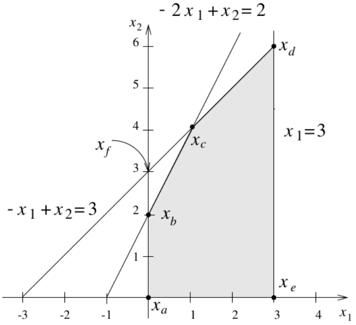
$$x = (x_1, x_2, x_3, x_4, x_5)^\top = (0, 0, 2, 3, 3)^\top = x_a \quad \text{Feasible!}$$

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_2, x_3, x_5\}$ with

$$B = (a_2\ a_3\ a_5) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

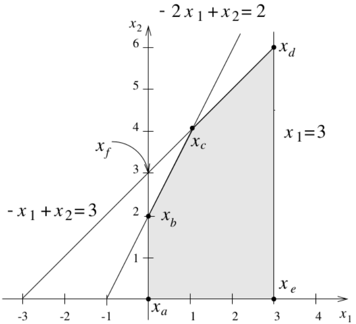
What is x_B satisfying $Bx_B = b$?

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_2, x_3, x_5\}$ with

$$B = (a_2\ a_3\ a_5) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

What is x_B satisfying $Bx_B = b$? $x_B = (x_2, x_3, x_5)^\top = (3, -1, 3)^\top$.

The corresponding basic solution is

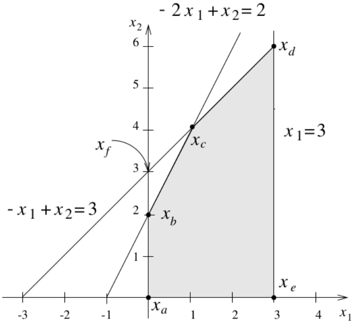
$$x = (x_1, x_2, x_3, x_4, x_5)^\top = (0, 3, -1, 0, 3)^\top = x_f \quad \text{Not feasible!}$$

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_1, x_2, x_3\}$ with

$$B = (u_1\ u_2\ u_3) = \begin{pmatrix} -2 & 1 & 1 \\ -1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

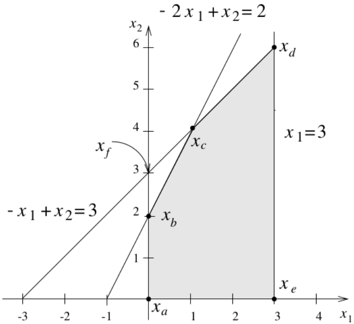
What is x_B satisfying $Bx_B = b$?

$$A = (u_1\ u_2\ u_3\ u_4\ u_5) = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4, x_5)^\top$$

$$Ax = b \text{ where } x \geq 0$$

$$b = (2, 3, 3)^\top$$



Consider a basis $\{x_1, x_2, x_3\}$ with

$$B = (u_1\ u_2\ u_3) = \begin{pmatrix} -2 & 1 & 1 \\ -1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

What is x_B satisfying $Bx_B = b$? $x_B = (x_1, x_2, x_3)^\top = (3, 6, 2)^\top$.

The corresponding basic solution is

$$x = (x_1, x_2, x_3, x_4, x_5)^\top = (3, 6, 2, 0, 0)^\top = x_d \quad \text{Feasible!}$$

Existence of Basic Feasible Solutions

Theorem 1 (Fundamental Theorem of LP)

Consider a linear program in standard form.

1. *If a feasible solution exists, then a basic feasible solution exists.*
2. *If an optimal feasible solution exists, then an optimal basic feasible solution exists.*

Existence of Basic Feasible Solutions

Theorem 1 (Fundamental Theorem of LP)

Consider a linear program in standard form.

1. *If a feasible solution exists, then a basic feasible solution exists.*
2. *If an optimal feasible solution exists, then an optimal basic feasible solution exists.*

Note that the theorem reduces solving a linear programming problem to searching for basic feasible solutions.

There are finitely many of them, which implies decidability.

Existence of Basic Feasible Solutions

Theorem 1 (Fundamental Theorem of LP)

Consider a linear program in standard form.

1. *If a feasible solution exists, then a basic feasible solution exists.*
2. *If an optimal feasible solution exists, then an optimal basic feasible solution exists.*

Note that the theorem reduces solving a linear programming problem to searching for basic feasible solutions.

There are finitely many of them, which implies decidability.

However, the enumeration of all basic feasible solutions would be impractical; the number of basic feasible solutions is potentially

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}$$

For $n = 100$ and $m = 10$, we get $535,983,370,403,809,682,970$.

Extreme Points

Note that the set Θ of points x satisfying $Ax = b, x \geq 0$ is *convex polyhedron*.

By definition, a convex hull of a finite set of points.

Extreme Points

Note that the set Θ of points x satisfying $Ax = b, x \geq 0$ is *convex polyhedron*.

By definition, a convex hull of a finite set of points.

A point $x \in \Theta$ is an *extreme point* of Θ if there are no two points x' and x'' such that $x = \alpha x' + (1 - \alpha)x''$ for some $\alpha \in (0, 1)$.

Extreme Points

Note that the set Θ of points x satisfying $Ax = b, x \geq 0$ is *convex polyhedron*.

By definition, a convex hull of a finite set of points.

A point $x \in \Theta$ is an *extreme point* of Θ if there are no two points x' and x'' such that $x = \alpha x' + (1 - \alpha)x''$ for some $\alpha \in (0, 1)$.

Theorem 2

Let Θ be the convex set consisting of all feasible solutions that is, all $x \in \mathbb{R}^n$ satisfying:

$$Ax = b, \quad x \geq 0,$$

where $A \in \mathbb{R}^{m \times n}, m < n$, $rank(A) = m$.
Then, x is an extreme point of Θ if and only if x is a basic feasible solution to $Ax = b, x \geq 0$.

Extreme Points

Note that the set Θ of points x satisfying $Ax = b, x \geq 0$ is *convex polyhedron*.

By definition, a convex hull of a finite set of points.

A point $x \in \Theta$ is an *extreme point* of Θ if there are no two points x' and x'' such that $x = \alpha x' + (1-\alpha)x''$ for some $\alpha \in (0,1)$.

Theorem 2

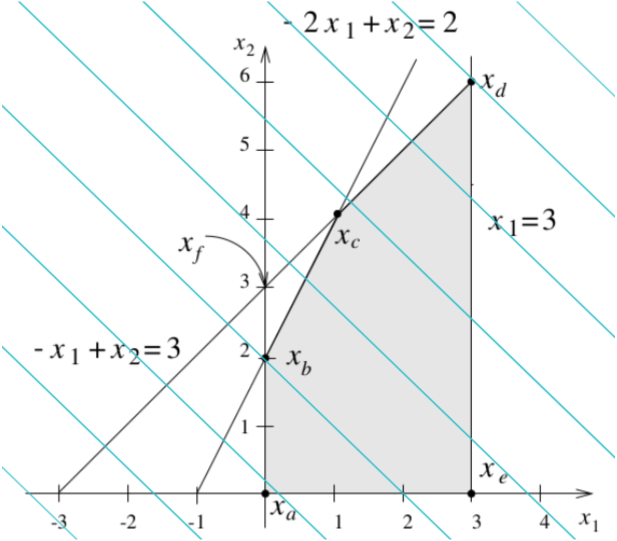
Let Θ be the convex set consisting of all feasible solutions that is, all $x \in \mathbb{R}^n$ satisfying:

$$Ax = b, \quad x \geq 0,$$

where $A \in \mathbb{R}^{m \times n}, m < n$, $rank(A) = m$.
Then, x is an extreme point of Θ if and only if x is a basic feasible solution to $Ax = b, x \geq 0$.

Thus, as a corollary, we obtain that to find an optimal solution to the linear optimization problem, we need to consider only extreme points of the feasibility region.

Optimal Solutions



Here, the blue lines are contours of $-x_1 - x_2$. The minimizer is x_d.

Degenerate Basic Solutions

A basic solution $x = [x_B, x_N] \in \mathbb{R}^n$ is *degenerate* if at least one component of x_B is 0.

Degenerate Basic Solutions

A basic solution $x = [x_B, x_N] \in \mathbb{R}^n$ is *degenerate* if at least one component of x_B is 0.

Two different bases can correspond to the same point. To see this, consider the constraints defined by

$$Ax = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 \\ 4 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 6 \\ 13 \\ 12 \end{pmatrix} = b.$$

Degenerate Basic Solutions

A basic solution $x = [x_B, x_N] \in \mathbb{R}^n$ is *degenerate* if at least one component of x_B is 0.

Two different bases can correspond to the same point. To see this, consider the constraints defined by

$$Ax = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 \\ 4 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 6 \\ 13 \\ 12 \end{pmatrix} = b.$$

There are two bases

$\{x_1, x_2, x_3\}$ giving

$$B = \begin{pmatrix} 2 & 1 & 0 \\ 3 & 0 & 1 \\ 4 & 0 & 0 \end{pmatrix}$$

$\{x_1, x_3, x_4\}$ giving

$$B' = \begin{pmatrix} 2 & 0 & 0 \\ 3 & 1 & 0 \\ 4 & 0 & 1 \end{pmatrix}$$

Each gives the same *degenerate* basic solution $x = (3, 0, 4, 0)^\top$.

Simplex Algorithm

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.
- Move to each of the neighboring vertices and check whether it is better from the point of view of the objective.

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.
- Move to each of the neighboring vertices and check whether it is better from the point of view of the objective.
- If yes, move to such a neighbor (there may be more than one better than the current one; choose one of them).

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.
- Move to each of the neighboring vertices and check whether it is better from the point of view of the objective.
- If yes, move to such a neighbor (there may be more than one better than the current one; choose one of them).
- If there is no better neighbor, the algorithm stops.

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.
- Move to each of the neighboring vertices and check whether it is better from the point of view of the objective.
- If yes, move to such a neighbor (there may be more than one better than the current one; choose one of them).
- If there is no better neighbor, the algorithm stops.
- (It may happen that the polyhedron is unbounded if the algorithm finds out that the objective may be infinitely improved.)

Intuition

The algorithm proceeds as follows:

- Start in a vertex of the polyhedron defined by the constraints.
- Move to each of the neighboring vertices and check whether it is better from the point of view of the objective.
- If yes, move to such a neighbor (there may be more than one better than the current one; choose one of them).
- If there is no better neighbor, the algorithm stops.
- (It may happen that the polyhedron is unbounded if the algorithm finds out that the objective may be infinitely improved.)

Now, how do you move from one vertex to another one algebraically?

First, we consider LP problems where each basic solution is non-degenerate.

Later we drop this assumption.

Changing Basis (Non-Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \ldots u_m\, u_{m+1} \ldots u_n)$ where $B = (u_1 \ldots u_m)$ and $N = (u_{m+1} \ldots u_n)$.

Note that each u_i is a column vector of dimension m.

Changing Basis (Non-Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \dots u_m\, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots x_m u_m = b$$

For a non-degenerate case, we have $x_j > 0$ for all $j = 1, \dots, m$.

Changing Basis (Non-Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \ldots u_m\, u_{m+1} \ldots u_n)$ where $B = (u_1 \ldots u_m)$ and $N = (u_{m+1} \ldots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots x_m u_m = b$$

For a non-degenerate case, we have $x_j > 0$ for all $j = 1, \ldots, m$.

Now as B is a basis, we have that for each $i \in \{m+1, \ldots, n\}$ there are coefficients $y_1, \ldots, y_m$ such that $y_1 u_1 + \cdots + y_m u_m = u_i$.

Changing Basis (Non-Degenerate Case)

Consider a basis B and write $A = (B \ N) = (u_1 \dots u_m \, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B \ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots x_m u_m = b$$

For a non-degenerate case, we have $x_j > 0$ for all $j = 1, \dots, m$.

Now as B is a basis, we have that for each $i \in \{m+1, \dots, n\}$ there are coefficients $y_1, \dots, y_m$ such that $y_1 u_1 + \cdots + y_m u_m = u_i$. Then

$$\begin{aligned} b &= x_1 u_1 + \cdots x_m u_m \\ &= x_1 u_1 + \cdots x_m u_m - \alpha u_i + \alpha u_i \\ &= x_1 u_1 + \cdots x_m u_m - \alpha (y_1 u_1 + \cdots + y_m u_m) + \alpha u_i \\ &= (x_1 - \alpha y_1) u_1 + \cdots + (x_m - \alpha y_m) u_m + \alpha u_i \end{aligned}$$

Changing Basis (Non-Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \ldots u_m\, u_{m+1} \ldots u_n)$ where $B = (u_1 \ldots u_m)$ and $N = (u_{m+1} \ldots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots x_m u_m = b$$

For a non-degenerate case, we have $x_j > 0$ for all $j = 1, \ldots, m$.

Now as B is a basis, we have that for each $i \in \{m+1, \ldots, n\}$ there are coefficients $y_1, \ldots, y_m$ such that $y_1 u_1 + \cdots + y_m u_m = u_i$. Then

$$\begin{aligned} b &= x_1 u_1 + \cdots x_m u_m \\ &= x_1 u_1 + \cdots x_m u_m - \alpha u_i + \alpha u_i \\ &= x_1 u_1 + \cdots x_m u_m - \alpha (y_1 u_1 + \cdots + y_m u_m) + \alpha u_i \\ &= (x_1 - \alpha y_1) u_1 + \cdots + (x_m - \alpha y_m) u_m + \alpha u_i \end{aligned}$$

Now consider maximum $\alpha > 0$ such that $x_j - \alpha y_j \geq 0$ for all j.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\} > 0$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\} > 0$$

There would be a *unique* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.

The uniqueness follows from non-degeneracy because otherwise, we would move to a basis giving a degenerate solution.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\} > 0$$

There would be a *unique* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
The uniqueness follows from non-degeneracy because otherwise, we would move to a basis giving a degenerate solution.

Note that such j can be computed using:

$$j = \text{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\} > 0$$

There would be a *unique* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
The uniqueness follows from non-degeneracy because otherwise, we would move to a basis giving a degenerate solution.

Note that such j can be computed using:

$$j = \text{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Obtain a basis $B_{j \to i} = B \setminus \{j\} \cup \{i\}$ and a basic feasible solution

$$x_{j \to i} = (x'_1, \ldots, x'_{j-1}, 0, x'_{j+1}, \ldots, x'_m, 0, \ldots, 0, \alpha, 0, \ldots, 0)^\top$$

Here $x'_k = x_k - \alpha y_k$ for each $k \in \{1, \ldots, j-1, j+1, \ldots, m\}$.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\} > 0$$

There would be a *unique* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
The uniqueness follows from non-degeneracy because otherwise, we would move to a basis giving a degenerate solution.

Note that such j can be computed using:

$$j = \text{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Obtain a basis $B_{j\to i} = B \setminus \{j\} \cup \{i\}$ and a basic feasible solution

$$x_{j\to i} = (x'_1, \ldots, x'_{j-1}, 0, x'_{j+1}, \ldots, x'_m, 0, \ldots, 0, \alpha, 0, \ldots, 0)^\top$$

Here $x'_k = x_k - \alpha y_k$ for each $k \in \{1, \ldots, j-1, j+1, \ldots, m\}$.
We say that we *pivot about* (j, i).

Algorithm 1 Simplex - **Non-degenerate**

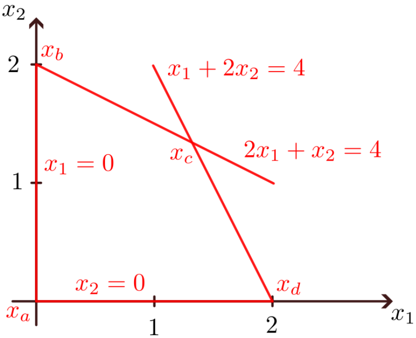
Choose a starting basis $B = (u_1 \dots u_m)$ (here $A = (B\ N)$)
repeat
  Compute the basic solution x for the basis B
  for $i \in \{m+1, \dots, n\}$ **do**
    Solve $B(y_1, \dots, y_m)^\top = u_i$
    if $y_k \leq 0$ for all $k \in \{1, \dots, m\}$ **then**
      Stop, unbounded problem.
    end if
    Select $j = \text{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \dots, m\}$
    Compute $x_{j \to i}$
  end for
  if $c^\top(x_{j \to i} - x) \geq 0$ for all $i \in \{m+1, \dots, n\}$ **then**
    Stop, we have an optimal solution.
  end if
  Select $i \in \{m+1, \dots, n\}$ such that $c^\top(x_{j \to i} - x) < 0$
  $B \leftarrow B_{j \to i}$
until convergence

$$A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4)^\top$$

$$b = (4, 4)^\top$$

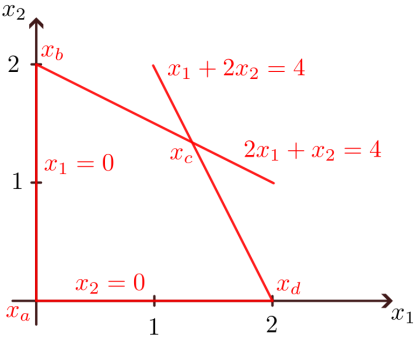
$$c = (-1, -1, 0, 0)^\top$$



minimize $c^\top x$ subject to $Ax = b$ where $x \geq 0$

$$A = (u_1\ u_2\ u_3\ u_4)$$
$$= \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix}$$

$$x = (x_1, x_2, x_3, x_4)^\top$$
$$b = (4, 4)^\top$$
$$c = (-1, -1, 0, 0)^\top$$



minimize $c^\top x$ subject to $Ax = b$ where $x \geq 0$

Consider a basis

$$B = (a_3\ a_4) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The basic solution is $x = (x_1, x_2, x_3, x_4)^\top = (0, 0, 4, 4)^\top$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Now $x_4/y_4 = 4/2 < 4/1 = x_3/y_3$, pivot about $(4, 1)$ and $\alpha = x_4/y_4 = 2$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Now $x_4/y_4 = 4/2 < 4/1 = x_3/y_3$, pivot about $(4, 1)$ and $\alpha = x_4/y_4 = 2$.

$$x_{4\to 1} = (\alpha, 0, (x_3 - \alpha y_3), (x_4 - \alpha y_4)) = (2, 0, 2, 0)$$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Now $x_4/y_4 = 4/2 < 4/1 = x_3/y_3$, pivot about $(4, 1)$ and $\alpha = x_4/y_4 = 2$.

$$x_{4\to 1} = (\alpha, 0, (x_3 - \alpha y_3), (x_4 - \alpha y_4)) = (2, 0, 2, 0)$$

As a result we get the basis $\{x_1, x_3\}$ and the basic solution $(2, 0, 2, 0)$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Now $x_4/y_4 = 4/2 < 4/1 = x_3/y_3$, pivot about $(4, 1)$ and $\alpha = x_4/y_4 = 2$.

$$x_{4\to1} = (\alpha, 0, (x_3 - \alpha y_3), (x_4 - \alpha y_4)) = (2, 0, 2, 0)$$

As a result we get the basis $\{x_1, x_3\}$ and the basic solution $(2, 0, 2, 0)$.

Similarly, we may also put x_2 into the basis instead of x_3 and obtain the basis $\{x_2, x_4\}$ and the basic solution $(0, 2, 0, 2)$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Start with the basis $\{x_3, x_4\}$ giving $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 4, 4)$.

Consider x_1 as a candidate to the basis, i.e., consider the first column u_1 of A expressed in the basis B:

$$u_1 = (1, 2)^\top = B\ (1, 2)^\top \text{ thus } y = (y_3, y_4) = (1, 2)$$

Now $x_4/y_4 = 4/2 < 4/1 = x_3/y_3$, pivot about $(4, 1)$ and $\alpha = x_4/y_4 = 2$.

$$x_{4\to 1} = (\alpha, 0, (x_3 - \alpha y_3), (x_4 - \alpha y_4)) = (2, 0, 2, 0)$$

As a result we get the basis $\{x_1, x_3\}$ and the basic solution $(2, 0, 2, 0)$.

Similarly, we may also put x_2 into the basis instead of x_3 and obtain the basis $\{x_2, x_4\}$ and the basic solution $(0, 2, 0, 2)$.

We have $c^\top (x_{4\to 1} - x) = -2 < 0$
So let us move to the basis $\{x_1, x_3\}$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Consider the basis $\{x_1, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (2, 0, 2, 0)$.

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Consider the basis $\{x_1, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (2, 0, 2, 0)$.

Consider x_2 as a candidate for the basis, i.e., consider the second column u_2 of A expressed in the basis B:

$$u_2 = (2, 1)^\top = B\ (1/2, 3/2)^\top \text{ thus } y = (y_1, y_3) = (1/2, 3/2)$$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Consider the basis $\{x_1, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (2, 0, 2, 0)$.

Consider x_2 as a candidate for the basis, i.e., consider the second column u_2 of A expressed in the basis B:

$$u_2 = (2, 1)^\top = B\ (1/2, 3/2)^\top \text{ thus } y = (y_1, y_3) = (1/2, 3/2)$$

Now $\alpha = x_3/y_3 = 4/3 < 2/(1/2) = 4 = x_1/y_1$, pivot about $(3, 2)$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Consider the basis $\{x_1, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (2, 0, 2, 0)$.

Consider x_2 as a candidate for the basis, i.e., consider the second column u_2 of A expressed in the basis B:

$$u_2 = (2, 1)^\top = B\ (1/2, 3/2)^\top \text{ thus } y = (y_1, y_3) = (1/2, 3/2)$$

Now $\alpha = x_3/y_3 = 4/3 < 2/(1/2) = 4 = x_1/y_1$, pivot about $(3, 2)$

$$x_{3\to 2} = ((x_1 - \alpha y_1), \alpha, (x_3 - \alpha y_3), 0) = (4/3, 4/3, 0, 0)$$

Non-Degenerate Example

$$c = (-1, -1, 0, 0) \quad A = (u_1\, u_2\, u_3\, u_4) = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

Consider the basis $\{x_1, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (2, 0, 2, 0)$.

Consider x_2 as a candidate for the basis, i.e., consider the second column u_2 of A expressed in the basis B:

$$u_2 = (2, 1)^\top = B\ (1/2, 3/2)^\top \text{ thus } y = (y_1, y_3) = (1/2, 3/2)$$

Now $\alpha = x_3/y_3 = 4/3 < 2/(1/2) = 4 = x_1/y_1$, pivot about $(3, 2)$

$$x_{3\to 2} = ((x_1 - \alpha y_1), \alpha, (x_3 - \alpha y_3), 0) = (4/3, 4/3, 0, 0)$$

$$c^\top (x_{3\to 2} - x) = c(-2/3, 4/3)^\top = -2/3 < 0$$

We have reached a minimizer. All changes would lead to a higher objective value.

We may exchange x_1 with x_4, but this would give us the initial basis with a higher objective value.

Non-Degenerate Case Convergence

Theorem 3

Suppose that the simplex method is applied to a linear program and that every basic variable is strictly positive at every iteration. Then, in a finite number of iterations, the method either terminates at an optimal basic feasible solution or determines that the problem is unbounded.

Non-Degenerate Case Convergence

Theorem 3

Suppose that the simplex method is applied to a linear program and that every basic variable is strictly positive at every iteration. Then, in a finite number of iterations, the method either terminates at an optimal basic feasible solution or determines that the problem is unbounded.

However, what happens if we meet a degenerate solution?

Non-Degenerate Case Convergence

Theorem 3

Suppose that the simplex method is applied to a linear program and that every basic variable is strictly positive at every iteration. Then, in a finite number of iterations, the method either terminates at an optimal basic feasible solution or determines that the problem is unbounded.

However, what happens if we meet a degenerate solution?

So, let us drop the non-degeneracy assumption.

Changing Basis (Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \ldots u_m\, u_{m+1} \ldots u_n)$ where $B = (u_1 \ldots u_m)$ and $N = (u_{m+1} \ldots u_n)$.

Note that each u_i is a column vector of dimension m.

Changing Basis (Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \dots u_m\, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots + x_m u_m = b$$

For a degenerate case, we have $x_j \geq 0$ for all $j \in \{1, \dots, m\}$, and *may have* $x_i = 0$ *for some* $j \in \{1, \dots, m\}$.

Changing Basis (Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \dots u_m\, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \dots + x_m u_m = b$$

For a degenerate case, we have $x_j \geq 0$ for all $j \in \{1, \dots, m\}$, and *may have* $x_i = 0$ *for some* $j \in \{1, \dots, m\}$.

Now as B is a basis, we have that for each $i \in \{m+1, \dots, n\}$ there are coefficients $y_1, \dots, y_m$ such that $y_1 u_1 + \dots + y_m u_m = u_i$.

Changing Basis (Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \dots u_m\, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \dots + x_m u_m = b$$

For a degenerate case, we have $x_j \geq 0$ for all $j \in \{1, \dots, m\}$, and *may have $x_i = 0$ for some $j \in \{1, \dots, m\}$*.

Now as B is a basis, we have that for each $i \in \{m+1, \dots, n\}$ there are coefficients $y_1, \dots, y_m$ such that $y_1 u_1 + \dots + y_m u_m = u_i$. Then

$$\begin{aligned} b &= x_1 u_1 + \dots + x_m u_m \\ &= x_1 u_1 + \dots + x_m u_m - \alpha u_i + \alpha u_i \\ &= x_1 u_1 + \dots + x_m u_m - \alpha(y_1 u_1 + \dots + y_m u_m) + \alpha u_i \\ &= (x_1 - \alpha y_1) u_1 + \dots + (x_m - \alpha y_m) u_m + \alpha u_i \end{aligned}$$

Changing Basis (Degenerate Case)

Consider a basis B and write $A = (B\ N) = (u_1 \dots u_m\, u_{m+1} \dots u_n)$ where $B = (u_1 \dots u_m)$ and $N = (u_{m+1} \dots u_n)$.

Note that each u_i is a column vector of dimension m.

Consider a basic feasible solution $x = [x_B\ x_N]$ where $x_N = 0$. Then

$$x_1 u_1 + \cdots + x_m u_m = b$$

For a degenerate case, we have $x_j \geq 0$ for all $j \in \{1, \dots, m\}$, and *may have* $x_i = 0$ *for some* $j \in \{1, \dots, m\}$.

Now as B is a basis, we have that for each $i \in \{m+1, \dots, n\}$ there are coefficients $y_1, \dots, y_m$ such that $y_1 u_1 + \cdots + y_m u_m = u_i$. Then

$$\begin{aligned} b &= x_1 u_1 + \cdots + x_m u_m \\ &= x_1 u_1 + \cdots + x_m u_m - \alpha u_i + \alpha u_i \\ &= x_1 u_1 + \cdots + x_m u_m - \alpha(y_1 u_1 + \cdots + y_m u_m) + \alpha u_i \\ &= (x_1 - \alpha y_1) u_1 + \cdots + (x_m - \alpha y_m) u_m + \alpha u_i \end{aligned}$$

Now consider maximum $\alpha \geq 0$ such that $x_j - \alpha y_j \geq 0$ for all j.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Otherwise, there *exists* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
j DOES NOT have to be unique in a degenerate case.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \land k = 1, \ldots, m\}$$

Otherwise, there *exists* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
j DOES NOT have to be unique in a degenerate case.

Note that such j can be computed using:

$$j \in \operatorname{argmin}\{x_k/y_k \mid y_k > 0 \land k = 1, \ldots, m\}$$

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Otherwise, there *exists* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
j DOES NOT have to be unique in a degenerate case.

Note that such j can be computed using:

$$j \in \mathrm{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Obtain a basis $B_{j \to i} = B \smallsetminus \{j\} \cup \{i\}$ and a basic feasible solution

$$x_{j \to i} = (x'_1, \ldots, x'_{j-1}, 0, x'_{j+1}, \ldots, x'_m, 0, \ldots, 0, \alpha, 0, \ldots, 0)^\top$$

Here $x'_k = x_k - \alpha y_k$ for each $k \in \{1, \ldots, j-1, j+1, \ldots, m\}$.
Note that if $\alpha = 0$, the solution does not change. The basis, however, changes.

$$b = (x_1 - \alpha y_1)u_1 + \cdots + (x_m - \alpha y_m)u_m + \alpha u_i$$

If all $y_j \leq 0$, the problem is unbounded because one component grows indefinitely and others do not decrease with $\alpha \to \infty$.

Otherwise, we put

$$\alpha = \min\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Otherwise, there *exists* $j \in \{1, \ldots, m\}$ such that $x_j - \alpha y_j = 0$.
j DOES NOT have to be unique in a degenerate case.

Note that such j can be computed using:

$$j \in \operatorname{argmin}\{x_k/y_k \mid y_k > 0 \wedge k = 1, \ldots, m\}$$

Obtain a basis $B_{j \to i} = B \smallsetminus \{j\} \cup \{i\}$ and a basic feasible solution

$$x_{j \to i} = (x'_1, \ldots, x'_{j-1}, 0, x'_{j+1}, \ldots, x'_m, 0, \ldots, 0, \alpha, 0, \ldots, 0)^\top$$

Here $x'_k = x_k - \alpha y_k$ for each $k \in \{1, \ldots, j-1, j+1, \ldots, m\}$.
Note that if $\alpha = 0$, the solution does not change. The basis, however, changes.
We say that we *pivot about* (j, i).

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\; u_2\; u_3\; u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 1$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 1$

$$x_{2\to 4} = (0, (x_2 - \alpha y_2), (x_3 - \alpha y_3), \alpha)^\top = (0, 0, 1, 1)^\top$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\; u_2\; u_3\; u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 1$

$$x_{2\to 4} = (0, (x_2 - \alpha y_2), (x_3 - \alpha y_3), \alpha)^\top = (0, 0, 1, 1)^\top$$

Note that $c^\top x_{2\to 4} = 0$.

Thus **no effect on the objective value!**

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Pivot about $(3, 1)$, that is x_3 exchanges with x_1 and $\alpha = x_3/y_3 = 0$.

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Pivot about $(3, 1)$, that is x_3 exchanges with x_1 and $\alpha = x_3/y_3 = 0$.

$$x_{3\to 1} = (\alpha, (x_2 - \alpha y_2), (x_3 - \alpha y_3), 0)^\top = (0, 1, 0, 0)^\top$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Pivot about $(3, 1)$, that is x_3 exchanges with x_1 and $\alpha = x_3/y_3 = 0$.

$$x_{3\to1} = (\alpha, (x_2 - \alpha y_2), (x_3 - \alpha y_3), 0)^\top = (0, 1, 0, 0)^\top$$

No change in the basic solution, and thus $c^\top x_{3\to1} = c^\top x = 0$.

Thus **no effect on the objective value either!**

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4)^\top = (0, 1, 0, 0)^\top$ with $c^\top x = 0$.

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Pivot about $(3, 1)$, that is x_3 exchanges with x_1 and $\alpha = x_3/y_3 = 0$.

$$x_{3\to 1} = (\alpha, (x_2 - \alpha y_2), (x_3 - \alpha y_3), 0)^\top = (0, 1, 0, 0)^\top$$

No change in the basic solution, and thus $c^\top x_{3\to 1} = c^\top x = 0$.

Thus **no effect on the objective value either!**

Which variable should go to the basis?!

Reduced Cost

Given a basis B, we denote by c_B the vector of components of c that correspond to the variables of B.

Reduced Cost

Given a basis B, we denote by c_B the vector of components of c that correspond to the variables of B.

One can prove that for every $i \in \{m+1, \ldots, n\}$ we have

$$c^\top x_{j \to i} - c^\top x = (c_i - c_B^\top y)\alpha$$

Here $y = (y_1, \ldots, y_m)^\top$ where $By = u_i$.

Reduced Cost

Given a basis B, we denote by c_B the vector of components of c that correspond to the variables of B.

One can prove that for every $i \in \{m+1, \ldots, n\}$ we have

$$c^\top x_{j \to i} - c^\top x = (c_i - c_B^\top y)\alpha$$

Here $y = (y_1, \ldots, y_m)^\top$ where $By = u_i$.

For non-degenerate case, we have $\alpha > 0$ and thus

$$c^\top x_{j \to i} < c^\top x \quad \text{iff} \quad c_i - c_B^\top y < 0$$

For the degenerate case, we may have $\alpha = 0$ and $c_i - c_B y < 0$.

Reduced Cost

Given a basis B, we denote by c_B the vector of components of c that correspond to the variables of B.

One can prove that for every $i \in \{m+1, \ldots, n\}$ we have

$$c^\top x_{j \to i} - c^\top x = (c_i - c_B^\top y)\alpha$$

Here $y = (y_1, \ldots, y_m)^\top$ where $By = u_i$.

For non-degenerate case, we have $\alpha > 0$ and thus

$$c^\top x_{j \to i} < c^\top x \quad \text{iff} \quad c_i - c_B^\top y < 0$$

For the degenerate case, we may have $\alpha = 0$ and $c_i - c_B y < 0$.

Define the *reduced cost* by

$$r_i = c_i - c_B^\top y$$

Intuitively, c_i is the cost of x_i in the new basis and $c_B^\top y$ in the old one.

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

The reduced cost is:

$$r_4 = c_4 - (c_2 y_2 + c_3 y_3) = 0 - (0 \cdot (-1) + 0 \cdot 2) = 0$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

The reduced cost is:

$$r_4 = c_4 - (c_2 y_2 + c_3 y_3) = 0 - (0 \cdot (-1) + 0 \cdot 2) = 0$$

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\; u_2\; u_3\; u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

The reduced cost is:

$$r_4 = c_4 - (c_2 y_2 + c_3 y_3) = 0 - (0 \cdot (-1) + 0 \cdot 2) = 0$$

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

The reduced cost is

$$r_1 = c_1 - (c_2 y_2 + c_3 y_3) = -1 - (0 \cdot (-1) + 0 \cdot 2) = -1 < 0$$

Degenerate Example

$$c = (-1, 0, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Start with the basis $\{x_2, x_3\}$ giving $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $cx = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(1, -1)^\top \text{ thus } y = (y_2, y_3) = (1, -1)$$

The reduced cost is:

$$r_4 = c_4 - (c_2 y_2 + c_3 y_3) = 0 - (0 \cdot (-1) + 0 \cdot 2) = 0$$

Consider x_1 as a candidate for the basis:

$$u_1 = (1, -1)^\top = B(-1, 2)^\top \text{ thus } y = (y_2, y_3) = (-1, 2)$$

The reduced cost is

$$r_1 = c_1 - (c_2 y_2 + c_3 y_3) = -1 - (0 \cdot (-1) + 0 \cdot 2) = -1 < 0$$

So we should put x_1 into the basis (the reduced cost gets smaller).

Algorithm 2 Simplex

Choose a starting basis $B = (u_1 \dots u_m)$ (here $A = (B\ N)$)
repeat
  Compute the basic solution x for the basis B
  for $i \in \{m+1, \dots, n\}$ **do**
    Solve $B\,(y_1, \dots, y_m)^\top = u_i$
    if $y_k \leq 0$ for all $k \in \{1, \dots, m\}$ **then**
      Stop, unbounded problem.
    end if
    Select $j \in \text{argmin}\{x_k / y_k \mid y_k > 0 \wedge k = 1, \dots, m\}$
    Compute $r_i = c_i - c_B^\top y$ where $y = (y_1, \dots, y_m)^\top$
  end for
  if $r_i \geq 0$ for all $i \in \{m+1, \dots, n\}$ **then**
    Stop, we have an optimal solution.
  end if
  Select $i \in \{m+1, \dots, n\}$ such that $r_i < 0$
  $B \leftarrow B_{j \to i}$
until convergence

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(-1/2, 1/2)^\top \text{ thus } y = (y_1, y_2) = (-1/2, 1/2)$$

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(-1/2, 1/2)^\top \text{ thus } y = (y_1, y_2) = (-1/2, 1/2)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 2$

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(-1/2, 1/2)^\top \text{ thus } y = (y_1, y_2) = (-1/2, 1/2)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 2$

$$x_{2\to 4} = ((x_1 - \alpha y_1), (x_2 - \alpha y_2), 0, \alpha) = (1, 0, 0, 2)$$

This is the minimizer!

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(-1/2, 1/2)^\top \text{ thus } y = (y_1, y_2) = (-1/2, 1/2)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 2$

$$x_{2\to 4} = ((x_1 - \alpha y_1), (x_2 - \alpha y_2), 0, \alpha) = (1, 0, 0, 2)$$

This is the minimizer!

Does this always work?

Degenerate Example (Cont.)

$$c = (-1, 1, 0, 0)^\top \quad A = (u_1\ u_2\ u_3\ u_4) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

After following the reduced cost from the basis $\{x_2, x_3\}$, we end up in the basis $\{x_1, x_2\}$ giving $B = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ and the basic solution $x = (x_1, x_2, x_3, x_4) = (0, 1, 0, 0)$ with $c^\top x = 0$.

Consider x_4 as a candidate for the basis:

$$u_4 = (0, 1)^\top = B\,(-1/2, 1/2)^\top \text{ thus } y = (y_1, y_2) = (-1/2, 1/2)$$

Pivot about $(2, 4)$, that is x_2 exchanges with x_4 and $\alpha = x_2/y_2 = 2$

$$x_{2\to 4} = ((x_1 - \alpha y_1), (x_2 - \alpha y_2), 0, \alpha) = (1, 0, 0, 2)$$

This is the minimizer!

Does this always work? Unfortunately, NO!

Degenerate Case - Looping

Consider the following linear program:

$$
\begin{array}{ll}
\text{minimize} & z = -\frac{3}{4}x_1 + 150x_2 - \frac{1}{50}x_3 + 6x_4 \\
\text{subject to} & \frac{1}{4}x_1 - 60x_2 - \frac{1}{25}x_3 + 9x_4 + x_5 = 0 \\
& \frac{1}{2}x_1 - 90x_2 - \frac{1}{50}x_3 + 3x_4 + x_6 = 0 \\
& x_3 + x_7 = 1 \\
& x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0
\end{array}
$$

Executing the simplex method on this program starting with the basis $\{x_5, x_6, x_7\}$ and always choosing i minimizing the reduced cost at line 15, eventually ends up back in the basis $\{x_5, x_6, x_7\}$.

In other words, even though the reduced cost is always negative, the overall effect on the objective is 0.

Convergence of Simplex Method

A solution is to use Bland's rule:

- Select the smallest index j at line 9.
- Select the smallest index i at line 15.

Theorem 4

If the simplex method is implemented using Bland's rule to select the entering and leaving variables, then the simplex method is guaranteed to terminate.

Simplex Convergence Summary

In a non-degenerate case:

- There is always a unique j to be selected at line 9.
- The objective of the basic solution decreases with each step.

Thus, we have a deterministic algorithm that always terminates in a non-degenerate case.

Simplex Convergence Summary

In a non-degenerate case:

- There is always a unique j to be selected at line 9.
- The objective of the basic solution decreases with each step.

Thus, we have a deterministic algorithm that always terminates in a non-degenerate case.

In a degenerate case:

- We may have several j from which to select at line 9.
- Even though the reduced cost is negative, the basic solution may remain the same.

The simplex algorithm may cycle!

Using Bland's rule, the simplex method always converges to a minimizer or detects an unbounded LP.

Two-Phase Simplex Algorithm

A Simplex algorithm is initialized with a basic feasible solution.

Two-Phase Simplex Algorithm

A Simplex algorithm is initialized with a basic feasible solution.

How do we obtain such a solution? Given a standard form LP

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

Two-Phase Simplex Algorithm

A Simplex algorithm is initialized with a basic feasible solution.

How do we obtain such a solution? Given a standard form LP

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

We construct an artificial LP problem.

$$\begin{aligned} \text{minimize} \quad & y_1 + y_2 + \cdots + y_m \\ \text{subject to} \quad & (A \; I_m) \begin{pmatrix} x \\ y \end{pmatrix} = b \\ & \begin{pmatrix} x \\ y \end{pmatrix} \geq 0 \end{aligned}$$

Here $y = (y_1, \ldots, y_m)^\top$ is a vector of artificial variables, I_m is the identity matrix of dimensions $m \times m$.

Two-Phase Simplex Algorithm

A Simplex algorithm is initialized with a basic feasible solution.

How do we obtain such a solution? Given a standard form LP

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

We construct an artificial LP problem:

$$\begin{aligned} \text{minimize} \quad & y_1 + y_2 + \cdots + y_m \\ \text{subject to} \quad & [A \ I_m] \begin{pmatrix} x \\ y \end{pmatrix} = b \\ & \begin{pmatrix} x \\ y \end{pmatrix} \geq 0 \end{aligned}$$

Here $y = (y_1, \ldots, y_m)^\top$ is a vector of artificial variables, I_m is the identity matrix of dimensions $m \times m$.

Two-Phase Simplex Algorithm

Solve the artificial LP problem:

$$
\begin{aligned}
\text{minimize} \quad & y_1 + y_2 + \cdots + y_m \\
\text{subject to} \quad & [A \ I_m] \begin{pmatrix} x \\ y \end{pmatrix} = b \\
& \begin{pmatrix} x \\ y \end{pmatrix} \geq 0
\end{aligned}
$$

Proposition 1

The original LP problem has a basic feasible solution iff the associated artificial LP problem has an optimal feasible solution with the objective function 0.

If we solve the artificial problem with $y = 0$, we obtain x such that $Ax = b, x \geq 0$ is a basic feasible solution for the original problem.

If there is no such a solution to the artificial problem, there is no basic feasible solution, and hence no feasible solution, to the original problem.

Linear Programming

Properties

LP Complexity

Iterations of the simplex algorithm can be implemented to compute the first step using $\mathcal{O}(m^2 n)$ arithmetic operations and each next step $\mathcal{O}(mn)$.

LP Complexity

Iterations of the simplex algorithm can be implemented to compute the first step using $\mathcal{O}(m^2 n)$ arithmetic operations and each next step $\mathcal{O}(mn)$.

There are as many as $\binom{n}{m}$ basic solutions (many of them likely infeasible). How large are these numbers?

m	$\binom{2m}{m}$
1	2
5	252
10	184756
20	1×10^{11}
50	1×10^{29}
100	9×10^{58}
200	1×10^{119}
300	1×10^{179}
400	2×10^{239}
500	3×10^{299}

The number of iterations may be proportional to $\binom{n}{m}$ that is EXPTIME.

Linear Programming Complexity

Complexity of the simplex algorithm:

- In the worst case, the time complexity of the simplex algorithm is exponential. This holds for any deterministic pivoting rule.
  For details, see "How good is the simplex algorithm?" by Klee, Victor, and Minty, George J. Inequalities 1972.

Linear Programming Complexity

Complexity of the simplex algorithm:

- In the worst case, the time complexity of the simplex algorithm is exponential. This holds for any deterministic pivoting rule.
  For details, see "How good is the simplex algorithm?" by Klee, Victor, and Minty, George J. Inequalities 1972.
- There is a theory that shows that examples with exponential complexity are rare. More precisely (but still very imprecisely)

Linear Programming Complexity

Complexity of the simplex algorithm:

- In the worst case, the time complexity of the simplex algorithm is exponential. This holds for any deterministic pivoting rule.
  For details, see "How good is the simplex algorithm?" by Klee, Victor, and Minty, George J. Inequalities 1972.
- There is a theory that shows that examples with exponential complexity are rare. More precisely (but still very imprecisely)
  - Consider small random perturbations of the coefficients in the LP (use Gaussian noise with a small variance)

Linear Programming Complexity

Complexity of the simplex algorithm:

- In the worst case, the time complexity of the simplex algorithm is exponential. This holds for any deterministic pivoting rule.

  For details, see "How good is the simplex algorithm?" by Klee, Victor, and Minty, George J. Inequalities 1972.

- There is a theory that shows that examples with exponential complexity are rare. More precisely (but still very imprecisely)
  - Consider small random perturbations of the coefficients in the LP (use Gaussian noise with a small variance)
  - Then, the expected computation time for the resulting instances of LP is polynomial.

  For details, see "Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time" by Daniel A. Spielman and Shang-Hua Teng in JACM 2004.

Is there a deterministic polynomial time algorithm for solving LP?

Linear Programming Complexity

We assume that all coefficients are encoded in binary (more precisely, as fractions of two integers encoded in binary).

Linear Programming Complexity

We assume that all coefficients are encoded in binary (more precisely, as fractions of two integers encoded in binary).

Theorem 5 (Khachiyan, Doklady Akademii Nauk SSSR, 1979)

There is an algorithm that, for any linear program, computes an optimal solution in polynomial time.

The algorithm uses so-called ellipsoid method.

Linear Programming Complexity

We assume that all coefficients are encoded in binary (more precisely, as fractions of two integers encoded in binary).

Theorem 5 (Khachiyan, Doklady Akademii Nauk SSSR, 1979)

There is an algorithm that, for any linear program, computes an optimal solution in polynomial time.

The algorithm uses so-called ellipsoid method.

In practice, the Khachiyan's is not used.

Linear Programming Complexity

We assume that all coefficients are encoded in binary (more precisely, as fractions of two integers encoded in binary).

Theorem 5 (Khachiyan, Doklady Akademii Nauk SSSR, 1979)

There is an algorithm that, for any linear program, computes an optimal solution in polynomial time.

The algorithm uses so-called ellipsoid method.

In practice, the Khachiyan's is not used.

There is also a polynomial time algorithm (by Karmarkar) that has lower complexity upper bounds than the Khachiyan's and sometimes works even better than the simplex.

Linear Programming in Practice

Heavily used tools for solving practical problems.

Several advanced linear programming solvers (usually parts of larger optimization packages) implement various heuristics for solving large-scale problems, such as sensitivity analysis.

See an overview of tools here:

http://en.wikipedia.org/wiki/Linear_programming#Solvers_and_scripting_.28programming.29_languages

For example, the well-known Gurobi solver uses the simplex algorithm to solve LP problems.