Linear optimization

The simplex method

Michel Bierlaire

Introduction to optimization and operations research

Simplex algorithm

Motivation

- Most famous optimization algorithm.
- Proposed by Dantzig in 1949.
- Solves linear optimization problems.
- Workhorse of modern optimization solvers.
- Main idea: the optimal solution lies on a vertex of the constraint polyhedron.

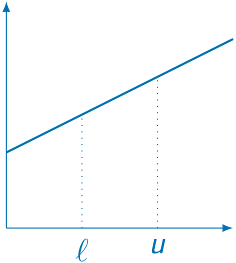
One dimension

$$\min_{x \in \mathbb{R}} ax + b$$

subject to

$$\ell \le x \le u.$$

$a > 0$



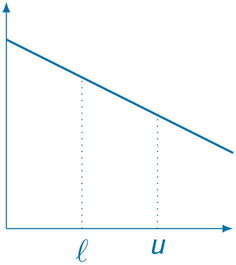
One dimension

$$\min_{x \in \mathbb{R}} ax + b$$

subject to

$$\ell \leq x \leq u.$$

$a < 0$



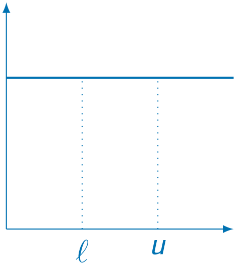
One dimension

$$\min_{x \in \mathbb{R}} ax + b$$

subject to

$$\ell \leq x \leq u.$$

$a = 0$



Several dimensions

Theorem 16.2

Consider

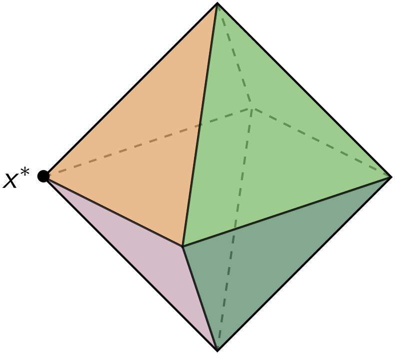
$$\min_{x \in \mathbb{R}^n} c^T x,$$

subject to

$$Ax = b,$$

$$x \geq 0,$$

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



If it has an optimal solution, there exists an optimal vertex of the constraint polyhedron.

Vertex enumeration

Geometric algorithm

- Enumerate all vertices of the polyhedron.
- For each of them, calculate $c^T x$.
- Identify the vertex with the smallest value.

Algebraic algorithm

- Enumerate all basic solutions of the polyhedron.
- For each of them, check if it is feasible, and calculate $c^T x$.
- Identify the feasible basic solution with the smallest value.

Vertex enumeration

Not practical

Number of basic solutions for a problem in standard form:

$$\frac{n!}{(n-m)!m!}.$$

- $n = 100,\ m = 50$: 10^{29}
- If one million basic solutions are treated per second,
- it will last 10^{13} (10 million million) centuries to solve.

Graphical method

Motivation

- An optimal solution of a linear optimization problem can be found on a vertex of the constraint polyhedron.
- But which one?
- In order to gain some intuition about the problem, we solve it graphically on a problem with two dimensions.

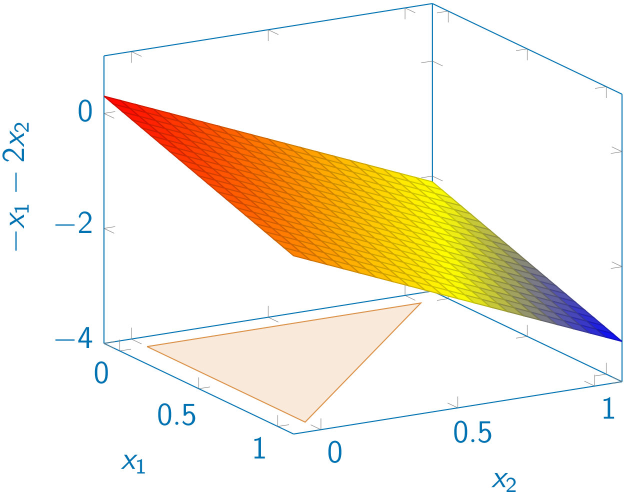
Example

$$\min_{x \in \mathbb{R}^2} -x_1 - 2x_2$$

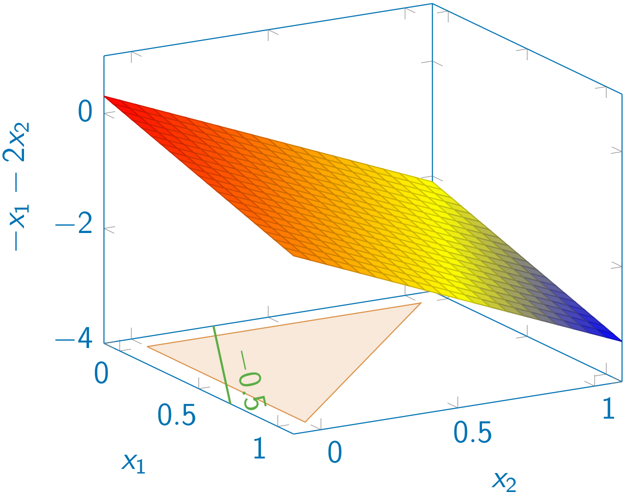
subject to

$$\begin{aligned} x_1 + x_2 &\leq 1 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

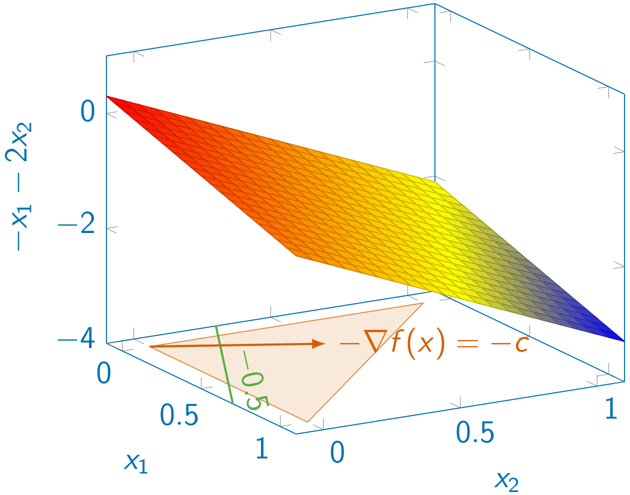
Example



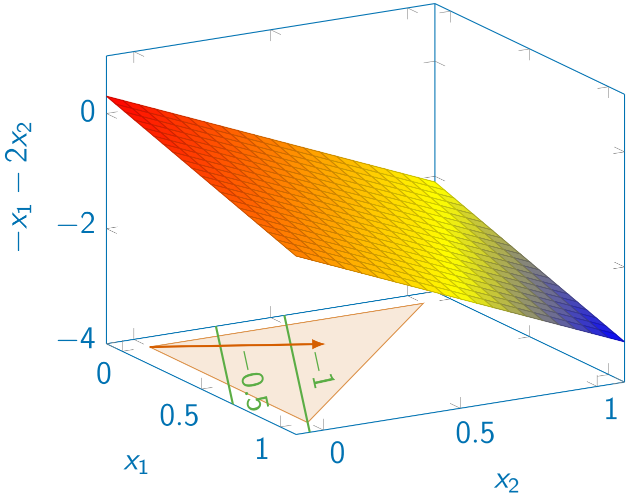
Example



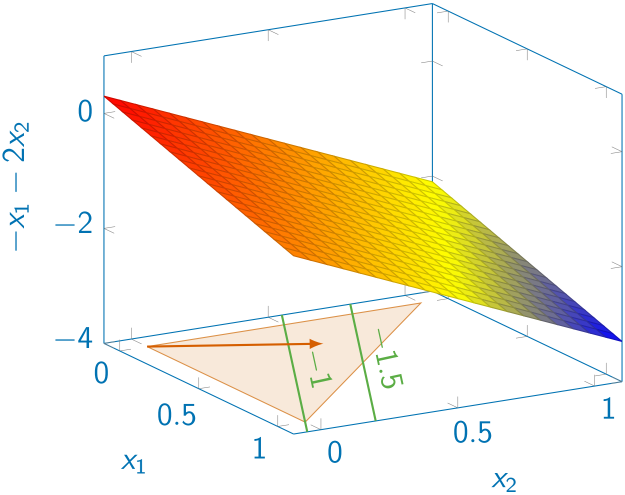
Example



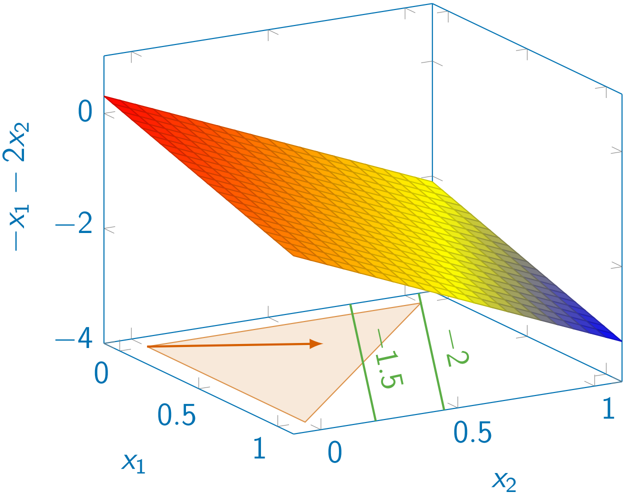
Example



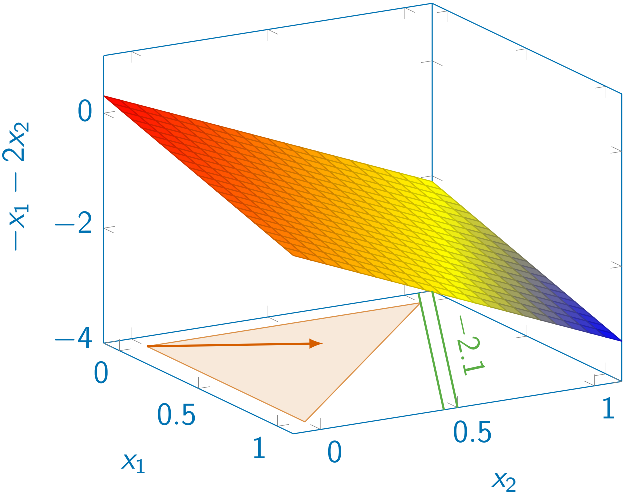
Example



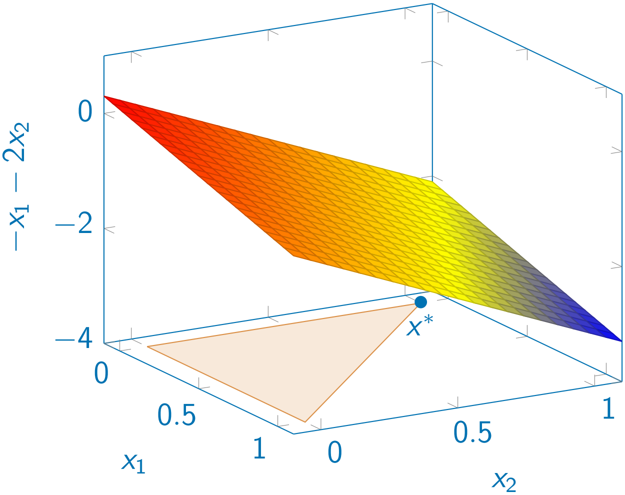
Example



Example



Example

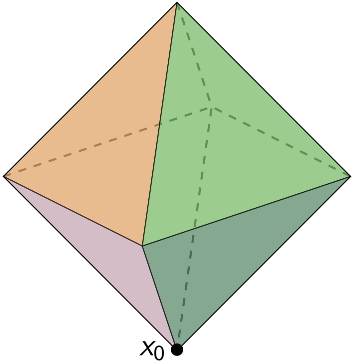


From vertex to vertex

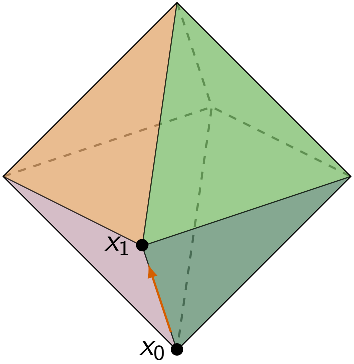
Motivation

- The optimal solution of a linear optimization problem can be found on a vertex of the constraint polyhedron.
- It is not practical to enumerate all vertices.
- Idea: start from a vertex, and move towards a neighbor vertex, that is better, in the sense of the objective function.

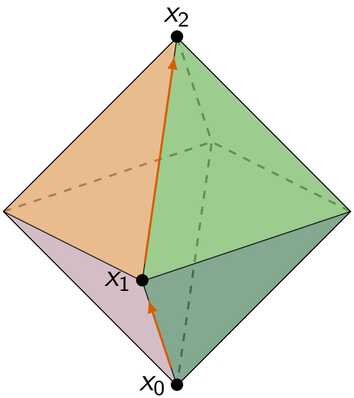
From vertex to vertex



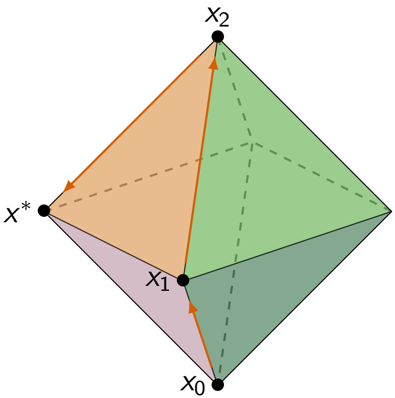
From vertex to vertex



From vertex to vertex



From vertex to vertex



Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.
- That is, if the reduced cost is negative.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.
- That is, if the reduced cost is negative.
- If so, follow the basic direction as far as possible. How far?

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.
- That is, if the reduced cost is negative.
- If so, follow the basic direction as far as possible. How far?
- Until a constraint is hit, is activated.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.
- That is, if the reduced cost is negative.
- If so, follow the basic direction as far as possible. How far?
- Until a constraint is hit, is activated.
- The corresponding variable is set to zero.

Simplex algorithm

Main ideas

- The current vertex is defined by active constraints.
- The corresponding non basic variables have been set to zero.
- Select one of them, and increase its value.
- It therefore enters the basis.
- Is it worth it? Yes, if the basic direction is descending.
- That is, if the reduced cost is negative.
- If so, follow the basic direction as far as possible. How far?
- Until a constraint is hit, is activated.
- The corresponding variable is set to zero.
- It leaves the basis.

Problem in standard form

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$\begin{aligned} Ax &= b, \\ x &\geq 0, \end{aligned}$$

where

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

Ingredients

Basic feasible solution = vertex

$$J^k = (j_1^k, \ldots, j_m^k).$$

Ingredients

Basic feasible solution = vertex

$$J^k = (j_1^k, \dots, j_m^k).$$

Basic matrix

$$B = \left(A_{j_1^k}, \dots, A_{j_m^k}\right) \text{ non singular.}$$

Ingredients

Basic feasible solution = vertex

$$J^k = (j_1^k, \ldots, j_m^k).$$

Basic matrix

$$B = \left(A_{j_1^k}, \ldots, A_{j_m^k}\right) \text{ non singular.}$$

Basic variables

$$x_B = B^{-1}b.$$

Ingredients

pth basic direction

$$d_B = -B^{-1}A_p.$$

Ingredients

pth basic direction

$$d_B = -B^{-1}A_p.$$

Reduced costs for pth basic direction

$$\bar{c}_p = c_p - c_B^T B^{-1} A_p.$$

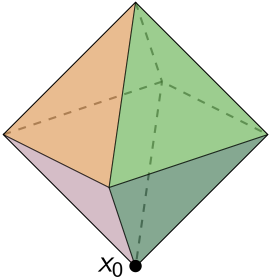
Initialization

Starting vertex

- Select a set

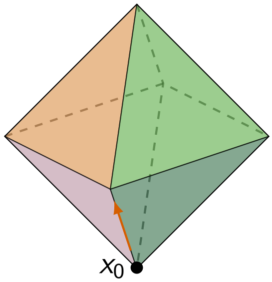
$$J^0 = (j_1^0, \ldots, j_m^0).$$

- It must correspond to a basic feasible solution.
- It means
  - B non singular, and
  - $x_B = B^{-1}b \geq 0$.
- It is in general not simple to find.



Descent direction

- Select a non basic variable p such that
$$\bar{c}_p < 0.$$
- It means that the pth basic direction is a descent direction.



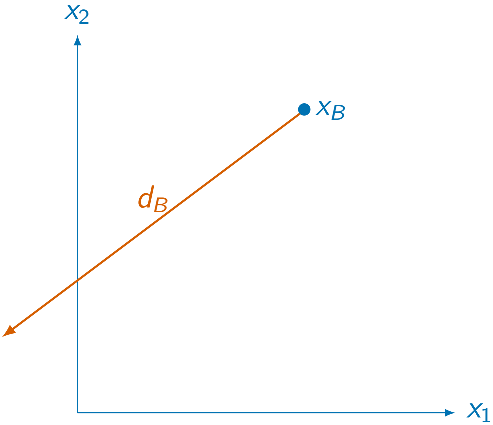
Next vertex

- Calculate the distance to each constraint, that is α_i such that

$$(x_B)_i + \alpha_i (d_B)_i = 0 \iff \alpha_i = -\frac{(x_B)_i}{(d_B)_i}.$$

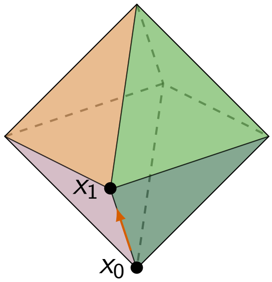
- Note: if $(d_B)_i \geq 0$, then $\alpha_i = +\infty$.
- Identify the closest constraint:

$$\alpha_q = \min_{i \in J^k} \alpha_i$$



Start a new iteration

$$J^{k+1} = J^k \cup \{p\} \setminus \{j_q^k\}.$$



Important details

- If no reduced cost is negative, we have found an optimal vertex.

Important details

- If no reduced cost is negative, we have found an optimal vertex.
- If $\alpha_{\boldsymbol{q}} = \infty$, the problem is unbounded.

Important details

- If no reduced cost is negative, we have found an optimal vertex.
- If $\alpha_q = \infty$, the problem is unbounded.
- In the presence of a degenerate basic feasible solution, it may happen that $\alpha_q = 0$.

Important details

- If no reduced cost is negative, we have found an optimal vertex.
- If $\alpha_q = \infty$, the problem is unbounded.
- In the presence of a degenerate basic feasible solution, it may happen that $\alpha_q = 0$.
- When several variables can be selected, choose the one with the smallest index (Bland's rule).

Important details

- If no reduced cost is negative, we have found an optimal vertex.
- If $\alpha_q = \infty$, the problem is unbounded.
- In the presence of a degenerate basic feasible solution, it may happen that $\alpha_q = 0$.
- When several variables can be selected, choose the one with the smallest index (Bland's rule).
- The new set of indices corresponds to a valid basic feasible solution (see Lemma 16.5 for a proof).

Simplex algorithm

Objective

Find the global optimum of a linear optimization problem in standard form.

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$Ax = b \; x \geq 0.$$

Input

- $A \in \mathbb{R}^{m \times n}$;
- $b \in \mathbb{R}^m$;
- $c \in \mathbb{R}^n$.
- $J^0 = (j_1^0, \ldots, j_m^0)$ set of indices of basic variables corresponding to a feasible basic solution.

Simplex algorithm

Outputs

- Boolean indicator U identifying an unbounded problem.
- If U is False, $J^* = (j_1^*, \ldots, j_m^*)$ the set of indices of basic variables corresponding to an optimal feasible basic solution, if it exists.

Initialization

$k = 0$.

Simplex algorithm

Iterations

1. Let $B = (A_{j_1^k}, \ldots, A_{j_m^k})$ the basic matrix with row of A corresponding to indices in J_k.
2. Select the smallest (Bland's rule) index $p \notin J^k$ such that the corresponding reduced cost

$$\bar{c}_p = c_p - c_B^T B^{-1} A_p$$

   is negative (A_p is the pth column of A). If there is none, the current solution is optimal. $J^* = J^k$, U=False. STOP.
3. Let P be the permutation matrix such that

$$AP = (B|N).$$

Simplex algorithm

Iterations (ctd)

4. Calculate
$$x_k = P \begin{pmatrix} B^{-1}b \\ 0_{\mathbb{R}^{n-m}} \end{pmatrix}.$$

Simplex algorithm

Iterations (ctd)

5. Calculate the pth basic direction

$$d_p = P \begin{pmatrix} d_{B_p} \\ d_{N_p} \end{pmatrix}$$

where $d_{B_p} = -B^{-1}A_p$, and d_{N_p} is such that

$$P^T e_p = \begin{pmatrix} 0 \\ d_{N_p} \end{pmatrix},$$

that is, all elements are 0, except the one corresponding to variable p, which is 1.

Simplex algorithm

Iterations (ctd)

6. For each basic index i, calculate the distance to the constraint $x_i \geq 0$, that is
$$\alpha_i = \begin{cases} -\frac{(x_k)_i}{(d_p)_i} & \text{if } (d_p)_i < 0 \\ +\infty & \text{otherwise.} \end{cases}$$
7. Let q be the smallest (Bland's rule) index such that
$$\alpha_q = \min_i \alpha_i.$$
8. If $\alpha_q = +\infty$, the problem is unbounded, and there is no optimal solution. U=True. STOP.
9. Index p enters the basis, and index q leaves it, i.e. $J^{k+1} = J^k \cup \{p\} \setminus q$, $k = k+1$.

Tableau

Motivation

- The simplex algorithm requires computational effort in linear algebra.
- We present here a tool to simplify the calculations.
- It is called the “simplex tableau”.

Main idea

Computational efforts

- Calculation of the reduced costs $c^T - c_B^T B^{-1} A$.
- Calculation of the current iterate $B^{-1} b$.
- Calculation of the basic direction $-B^{-1} A_p$.

Store $B^{-1} A$ and $B^{-1} b$ instead of A and b.

Tableau

Definition

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

Basic feasible solution $\widetilde{x}$

$$
\left[\begin{array}{ccc|c}
 & & & \widetilde{x}_{j_1} \\
B^{-1}A_1 & \cdots & B^{-1}A_n & \vdots \\
 & & & \widetilde{x}_{j_m} \\
\hline
\bar{c}_1 & \cdots & \bar{c}_n & -c^T\widetilde{x}
\end{array}\right]
$$

Interpretation

$$
\begin{array}{|ccc|c|}
\hline
| & & | & \widetilde{x}_{j_1} \\
B^{-1}A_1 & \cdots & B^{-1}A_n & \vdots \\
| & & | & \widetilde{x}_{j_m} \\
\hline
\bar{c}_1 & \cdots & \bar{c}_n & -c^T\widetilde{x} \\
\hline
\end{array}
$$

- Each column corresponds to a variable.
- Each row of the top part corresponds to a **basic** variable.
- Last row: reduced cost.
- If i is basic, $B^{-1}A_i$ is a column of the identity matrix.
- The only 1 identifies the corresponding row.

Example 1

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \end{pmatrix}.$$

Basic variables: x_3 and x_4

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad B^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Example 1

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix}, b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \end{pmatrix}, B^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Basic variables: x_3 and x_4

$$\begin{array}{cccc|c|l} x_1 & x_2 & x_3 & x_4 & & \\ \hline 1 & 1 & 1 & 0 & 1 & x_3, \quad \alpha_3 = 1/1 \\ 1 & -1 & 0 & 1 & 1 & x_4, \quad \alpha_4 = 1/1 \\ \hline -1 & -2 & 0 & 0 & 0 & -c^T x \\ \hline \end{array}$$

Example 2

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \end{pmatrix}.$$

Basic variables: x_2 and x_4

$$B = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}, \qquad B^{-1} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

Example 2

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix}, b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \end{pmatrix}, B^{-1} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

Basic variables: x_2 and x_4

x_1	x_2	x_3	x_4		
1	1	1	0	1	x_2
2	0	1	1	2	x_4
1	0	2	0	2	$-c^T x$

Usage in the algorithm

Reduced costs

$$c^T - c_B^T B^{-1} A.$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1} A$	$-c_B^T B^{-1} b$

Usage in the algorithm

Basic direction

$$d_B = -B^{-1}A_p \text{ and } \alpha_i = (x_B)_i/(-d_B)_i.$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

Usage in the algorithm

Solution

$$x_B = B^{-1}b \text{ and } c^T x = c_B^T x_B = c_B^T B^{-1} b.$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1} A$	$-c_B^T B^{-1} b$

Difficulties

- Prepare the tableau for the next iteration.
- Find the first tableau.

Pivoting

Motivation

- If a valid tableau is available, one iteration of the simplex algorithm is simple.
- Once the two variables to exchange in the basis have been identified, how to generate a valid tableau for the new basis?

One iteration

Before

$$B = \left(A_{j_1} \cdots A_{j_q} \cdots A_{j_m}\right)$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

After

$$\bar{B} = \left(A_{j_1} \cdots A_p \cdots A_{j_m}\right)$$

$\bar{B}^{-1}A$	$\bar{B}^{-1}b$
$c^T - c_{\bar{B}}^T \bar{B}^{-1}A$	$-c_{\bar{B}}^T \bar{B}^{-1}b$

How to transform B^{-1} into $\bar{B}^{-1}$?

Elementary row operations

Definition

- Consider row j of a matrix A.
- Multiply it by β.
- Add the result to row i.

$$\bar{a}_i \leftarrow a_i + \beta a_j \,,$$

Elementary row operations

$$\bar{A} = Q_{ij}(\beta)A, \text{ with } Q_{ij}(\beta) = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & \beta & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix}$$

Transformations

Objective

Find Q such that

$$QB^{-1} = \bar{B}^{-1}.$$

Equivalently

$$QB^{-1}\bar{B} = I$$

$$B^{-1}\bar{B} = \begin{pmatrix} 1 & 0 & & u_1 & & 0 \\ 0 & 1 & & u_2 & & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & u_q & & \vdots \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & & u_m & & 1 \end{pmatrix}$$

Pivoting

$$B^{-1}\bar{B} = \begin{pmatrix} 1 & 0 & & u_1 & & 0 \\ 0 & 1 & & u_2 & & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & u_q & & \vdots \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & & u_m & & 1 \end{pmatrix}$$

Elementary row operations

$$Q = Q_{qq}(1/u_q) \prod_{i \neq q} Q_{iq}(-u_i/u_q).$$

Pivoting

$$QB^{-1}\bar{B} = \begin{pmatrix} 1 & 0 & & u_1 - \frac{u_1}{u_q}u_q & & 0 \\ 0 & 1 & & u_2 - \frac{u_2}{u_q}u_q & & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & u_q/u_q & & \vdots \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & & u_m - \frac{u_m}{u_q}u_q & & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & & 0 & & 0 \\ 0 & 1 & & 0 & & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & 1 & & \vdots \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & & 0 & & 1 \end{pmatrix}$$

Pivoting

$$
\begin{aligned}
QB^{-1}\bar{B} &= I \\
QB^{-1} &= \bar{B}^{-1} \\
QB^{-1}A &= \bar{B}^{-1}A \\
QB^{-1}b &= \bar{B}^{-1}b
\end{aligned}
$$

One iteration

Before

$$B = \left(A_{j_1} \cdots A_{j_q} \cdots A_{j_m}\right)$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

After

$$\bar{B} = \left(A_{j_1} \cdots A_p \cdots A_{j_m}\right)$$

$QB^{-1}A$	$QB^{-1}b$
?	?

Last row: same elementary row operation.

Simplex tableau algorithm

Objective

Find the global minimum of a linear optimization problem in standard form.

Input

T_0, the simplex tableau corresponding to a feasible basic solution.

Outputs

- A boolean indicator U identifying an unbounded problem.
- If U is False, T^*, the simplex tableau corresponding to an optimal solution.

Initialization

$k = 0$.

Simplex tableau algorithm

Iterations

1. Find the reduced costs in the left part of the last row of T_k. If they are all non negative, the tableau is optimal. $T^* = T_k$, U=False. STOP.
2. Let p the index corresponding to the leftmost column with a negative reduced cost.
3. For each row i, calculate the distance to the constraint $x_i \geq 0$, that is

$$\alpha_i = \begin{cases} T(i, n+1)/T(i,p) & \text{if } T(i,p) > 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Simplex tableau algorithm

Iterations (ctd)

4. Let q be the uppermost index such that
$$\alpha_q = \min_i \alpha_i.$$
5. If $\alpha_q = +\infty$, the problem is unbounded, and there is no optimal solution. U=True. STOP.
6. Index p enters the basis, and index q leaves it. Pivot the tableau T_k to obtain T_{k+1}. $k = k + 1$.

Example

x_1	x_2	x_3	x_4	x_5	x_6		
1	2	2	1	0	0	20	$\alpha_4 = 20$
2	1	2	0	1	0	20	$\alpha_5 = 10$
2	2	1	0	0	1	20	$\alpha_6 = 10$
-10	-12	-12	0	0	0	0	

x_1	x_2	x_3	x_4	x_5	x_6	
0	1.5	1	1	-0.5	0	10
1	0.5	1	0	0.5	0	10
0	1	-1	0	-1	1	0
0	-7	-2	0	5	0	100

Example

x_1	x_2	x_3	x_4	x_5	x_6	
0	1.5	1	1	-0.5	0	10
1	0.5	1	0	0.5	0	10
0	1	-1	0	-1	1	0
0	-7	-2	0	5	0	100

$\alpha_4 = 20/3$
$\alpha_1 = 20$
$\alpha_6 = 0$

x_1	x_2	x_3	x_4	x_5	x_6	
0	0	2.5	1	1	-1.5	10
1	0	1.5	0	1	-0.5	10
0	1	-1	0	-1	1	0
0	0	-9	0	-2	7	100

Example

x_1	x_2	x_3	x_4	x_5	x_6		
0	0	2.5	1	1	-1.5	10	$\alpha_4 = 4$
1	0	1.5	0	1	-0.5	10	$\alpha_1 = 20/3$
0	1	-1	0	-1	1	0	$\alpha_2 = +\infty$
0	0	-9	0	-2	7	100	

x_1	x_2	x_3	x_4	x_5	x_6		
0	0	1	0.4	0.4	-0.6	4	x_3
1	0	0	-0.6	0.4	0.4	4	x_1
0	1	0	0.4	-0.6	0.4	4	x_2
0	0	0	3.6	1.6	1.6	136	

Optimal solution: $x^* = (4, 4, 4, 0, 0, 0)^T$, $c^T x^* = -136$.

Initial tableau: the simple case

Motivation

- The last ingredient to obtain a complete algorithm is the generation of the first tableau.
- We start by the easy case.
- We'll follow with the general case.

Motivation

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

First tableau

- Avoid trials and errors.
- Avoid the calculation of B^{-1}.

Inequality constraints

$$\min c^T x$$

subject to

$$Ax \le b,$$

$$x \ge 0.$$

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

$$b \ge 0.$$

$$\min c^T x + 0^T x_s$$

subject to

$$Ax + Ix^s = b,$$

$$x \ge 0.$$

$$x^s \ge 0.$$

$$x^s \in \mathbb{R}^m$$

Feasible solution

$$\min c^T x + 0^T x_s$$

subject to

$$\begin{aligned} Ax + Ix^s &= b, \\ x &\geq 0, \\ x^s &\geq 0. \end{aligned}$$

- $x = 0$, $x^s = b \geq 0$.
- Basic variables: x^s.
- Basic matrix: $B = I$.
- Tableau:

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1} A$	$-c_B^T B^{-1} b$

Initial tableau: the general case

Motivation

- Finding a feasible vertex of the constraint polyhedron and the corresponding tableau is not an easy task in the general case.
- This problem can actually be formulated as a linear optimization problem.
- And this optimization problem is solved using the simplex algorithm.

Problem in standard form

Problem $\mathcal{P}$

$$\min_{x} c^T x$$

subject to

$$Ax = b,$$

$$x \geq 0.$$

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

$$b \geq 0.$$

$$\min_{x, x^a} 0x + 1^T x^a$$

subject to

$$Ax + Ix^a = b,$$

$$x \geq 0.$$

$$x^a \geq 0.$$

$$x^a \in \mathbb{R}^m$$

Auxiliary problem

Problem $\mathcal{A}$

$$\min_{x,x_a} 1^T x^a = \sum_{i=1}^{m} x_i^a$$

subject to

$$\begin{aligned} Ax + Ix^a &= b, \\ x &\geq 0, \\ x^a &\geq 0, \end{aligned}$$

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

$$b \geq 0.$$

x_0 feasible for $\mathcal{P}$

- $Ax_0 = b$, $x_0 \geq 0$.
- $x = x_0$, $x^a = 0$ is feasible for $\mathcal{A}$.
- If it also optimal.
- Contrapositive: if optimal > 0, no feasible solution in $\mathcal{P}$

Initial tableau for the auxiliary problem

Problem $\mathcal{A}$

$$\min_{x,x_a} 1^T x^a = \sum_{i=1}^{m} x_i^a$$

subject to

$$Ax + Ix^a = b,$$
$$x \geq 0,$$
$$x^a \geq 0,$$

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

$$b \geq 0.$$

$B^{-1}A$	$B^{-1}b$
$c^T - c_B^T B^{-1}A$	$-c_B^T B^{-1}b$

$B \to I$, $A \to A|I$, $c_B \to 1$, $c_N \to 0$.
Reduced cost of aux. var$= 0$.
Reduced cost of orig. var $= 0$ - $c_B^T B^{-1} A_j$

$A \quad \vert I$	b
$-1^T A \vert 0$	$-1^T b$

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.
- If some auxiliary variables are in the basis, pivot them out.

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.
- If some auxiliary variables are in the basis, pivot them out.
- If all auxiliary variables are out of the basis, remove the corresponding columns.

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.
- If some auxiliary variables are in the basis, pivot them out.
- If all auxiliary variables are out of the basis, remove the corresponding columns.
- To solve $\mathcal{P}$, the last row of the tableau must be recalculated.

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.
- If some auxiliary variables are in the basis, pivot them out.
- If all auxiliary variables are out of the basis, remove the corresponding columns.
- To solve $\mathcal{P}$, the last row of the tableau must be recalculated.

Clean the tableau

- Consider the optimal tableau of the auxiliary problem.
- If some auxiliary variables are in the basis, pivot them out.
- If all auxiliary variables are out of the basis, remove the corresponding columns.
- To solve $\mathcal{P}$, the last row of the tableau must be recalculated.

Note

- If matrix A is not full rank, it may not be possible to pivot all variables out.
- In that case, redundant constraints can be eliminated.
- See example 16.15, and the discussion on p. 390.

Procedure

- Write problem $\mathcal{P}$ in standard form such that $b \geq 0$.
- Consider the auxiliary problem $\mathcal{A}$.
- Solve $\mathcal{A}$ with the simplex algorithm.
- If one of the auxiliary variables is not zero at the solution, $\mathcal{P}$ is infeasible.
- Otherwise, x^* is a feasible solution for $\mathcal{P}$.
- Clean the tableau.
- Solve $\mathcal{P}$ with the simplex algorithm.

Summary

- Solution on a vertex.
- Graphical method.
- Simplex algorithm: from vertex to vertex.
- Simplex tableau.
- Pivoting.
- Initial tableau and the auxiliary problem.