# The Simplex Algorithm: Phase II

Optimization Methods in Management Science
Master in Management
HEC Lausanne

Dr. Rodrigue Oeuvray

Fall 2019 Semester

# The Simplex Algortithm in a Tableau

Simplex Algorithm (phase II)

- Maximizing the objective function with pivoting
- Optimal and non-bounded tableaus
- Phase II
- Degeneracy
- Bland's rule
- Performance of the simplex algorithm
- Polynomial algorithms

# How to Maximize $z$ with Pivotings ?

Let's consider a tableau $\boldsymbol{T_B}$ associated with a feasible basis $\boldsymbol{B}$ :

$$\boldsymbol{T_B} \overset{P}{=} \begin{array}{c|c|c|c||c|} \multicolumn{1}{c}{\boldsymbol{x_N}} & \multicolumn{1}{c}{\boldsymbol{x_B}} & \multicolumn{1}{c}{z} & \multicolumn{2}{c}{} \\ \hline \boldsymbol{B}^{-1}\boldsymbol{N} & \boldsymbol{I} & 0 & & \beta \\ \hline -\gamma_{\boldsymbol{N}} & 0 & 1 & & \zeta \\ \hline \end{array}$$

The basic solution associated with $\boldsymbol{T_B}$ is

$$\boldsymbol{x_B} = \beta = \boldsymbol{B}^{-1}\boldsymbol{b}, \;\; \boldsymbol{x_N} = 0 \text{ and } z = \zeta = \boldsymbol{c_B}\boldsymbol{B}^{-1}\boldsymbol{b} \text{ (since } \boldsymbol{x_N} = 0)$$

The last row of the tableau contains the reduced costs $-\gamma_{\boldsymbol{N}}$ :

$$-\gamma_{\boldsymbol{N}} = \boldsymbol{c_B}\boldsymbol{B}^{-1}\boldsymbol{N} - \boldsymbol{c_N}$$

# How to Maximize $z$ with Pivotings ? (Cont'd)

- In basis $\boldsymbol{B}$, the objective function can be written as:

$$z = \zeta + \boldsymbol{\gamma_N x_N} = \zeta + \sum_{k \in \mathcal{N}} \gamma_k x_k = \zeta - \sum_{k \in \mathcal{N}} (-\gamma_k) x_k,$$

  where $\mathcal{N}$ is the set of non-basic variables

- **Reduced costs** measure the sensitivity of the objective function to a unit change in non-basic variables (if we assume that current basis keeps optimal)

- if $-\gamma_k < 0$, a unit increase in variable $x_k$ will increase the objective function of $\gamma_k > 0$ (if we assume that current basis keeps optimal)

- So when we consider a **maximization** problem, negative reduced costs represent potential increases in terms of the objective function !

# How to Maximize $z$ with Pivotings ? (Cont'd)

All the feasible solutions can be obtained by choosing $\boldsymbol{x_N} = \boldsymbol{s} \geq \mathbf{0}$ and by computing:

$$\boldsymbol{x_B} = \beta - \boldsymbol{B}^{-1}\boldsymbol{Ns},$$

with $\boldsymbol{x_B} \geq 0$ and $\boldsymbol{x_N} = \boldsymbol{s} \geq 0$

Case 1: If $-\gamma_r \geq 0 \;\; \forall \; r \in \mathcal{N}$, any increase of the non-basic variables causes a **decrease in the value of $z$**. The current basic solution is **optimal**:

$$\boldsymbol{x_B^*} = \beta, \qquad \boldsymbol{x_N^*} = \mathbf{0} \qquad \text{and} \qquad z^* = \zeta$$

# How to Maximize $z$ with Pivotings ? (Cont'd)

Case 2: If $\exists\, r \in \mathcal{N}$ such that $-\gamma_r < 0$, any increase of $x_r$ causes an **increase in $z$**

If all the variables that are not in the basis, except $x_r$, are kept to 0, then the solution of the system of constraints is

$$\boldsymbol{x_B} = \boldsymbol{\beta} - \boldsymbol{B}^{-1}\boldsymbol{Ns} = \boldsymbol{\beta} - \boldsymbol{\alpha}^{(r)} x_r$$

where $\boldsymbol{s}^T = (0 \ldots x_r \ldots 0), \boldsymbol{\alpha}^{(r)} = \left(\boldsymbol{B}^{-1}\boldsymbol{N}\right)_{\cdot r}$ is the column of $\boldsymbol{T_B}$ associated to $x_r$

# Case 2: $-\gamma_r < 0$

We are in this case :

| $x_1$ | $\dots$ | $x_k$ | $\dots$ | $x_{n+m}$ | $z$ | |
|---|---|---|---|---|---|---|
| | | $\alpha_{1r}$ | | | $0$ | $\oplus$ |
| | | $\vdots$ | | | $\vdots$ | $\vdots$ |
| | | $\alpha_{mr}$ | | | $0$ | $\oplus$ |
| $*$ | $\dots$ | $-$ | $\dots$ | $*$ | $1$ | $*$ |

# Case 2: $-\gamma_r < 0$

What is the impact of increasing $x_r$ on the basis variables ?

- If $\alpha_{ir} = \boldsymbol{\alpha}_i^{(r)} \leq 0$, it causes an **increase** (or at least no decrease) of the $i$th basis variable $x_{\sigma(i)}$ since $\boldsymbol{x_B} = \boldsymbol{\beta} - \boldsymbol{\alpha}^{(r)} x_r$

- If $\alpha_{ir} = \boldsymbol{\alpha}_i^{(r)} > 0$, it causes a **decrease** of $x_{\sigma(i)}$ since $\boldsymbol{x_B} = \boldsymbol{\beta} - \boldsymbol{\alpha}^{(r)} x_r$

# Case 2: -$\gamma_r < 0$

Case 2.a: $\boldsymbol{\alpha}^{(r)} \leq \mathbf{0}$ (the vector !), then the problem is **non-bounded** :

$$\boldsymbol{x_B} = \boldsymbol{\beta} - \boldsymbol{\alpha}^{(r)} x_r \geq 0 \qquad \forall\, x_r \geq 0$$

and as $z = \zeta + \boldsymbol{\gamma_N x_N} = \zeta + \sum_{k \in \mathcal{N}} \gamma_k x_k$:

$$\lim_{x_r \to \infty} z = \zeta + \lim_{x_r \to \infty} \gamma_r x_r = \infty$$

A tableau where $\exists\, r \in \mathcal{N}$ such that $-\gamma_r < 0$ and $\boldsymbol{\alpha}^{(r)} \leq \mathbf{0}$ is called **non-bounded** or **unbounded**

# Case 2.b: -$\gamma_r < 0$ and $\alpha_{ir} > 0$

**Case 2.b**: it exists $i$ such that $\alpha_{ir} > 0$. In this case:

$$x_{\sigma(i)} = \beta_i - \alpha_{ir} x_r = 0 \iff x_r = \beta_i / \alpha_{ir}, \qquad i \in \mathcal{B}$$

- By computing the ratios $\beta_i / \alpha_{ir}$ for each $i \in \mathcal{B}$ for which $\alpha_{ir} > 0$ and if we select the **smallest one**, then all the variables in $\boldsymbol{B}$ keep **non - negative**
- Let $j$ be the index of the row having the smallest ratio. The solution obtained by fixing $x_r = \beta_j / \alpha_{jr}$ corresponds to the new basic solution. The variable $x_r$ replaces $x_{\sigma(j)}$ in the new basis $\boldsymbol{B}'$

# Case 2.b: -$\gamma_r < 0$ and $\alpha_{ir} > 0$ (Cont'd)

- We move from tableau $\boldsymbol{T_B}$ to tableau $\boldsymbol{T_{B'}}$ by pivoting around $\alpha_{jr}$
- The new basis $\boldsymbol{B}'$ is feasible and the value of its basic solution is

$$\zeta' = \zeta + \gamma_r x_r = \zeta + \gamma_r \frac{\beta_j}{\alpha_{jr}} \geq \zeta$$

  since $\gamma_r > 0$, $\alpha_{jr} > 0$ and $\beta_j \geq 0$
- This just means that we have **improved** the current basic solution by performing a pivoting with these rules !
- These developments are the foundations of the simplex algorithm

# Summary: Representations of $\boldsymbol{T_B}$

Let $\boldsymbol{B}$ be a feasible basis and $\boldsymbol{T_B}$ its associated tableau. There are two representations for $\boldsymbol{T_B}$:

$$\boldsymbol{T_B} \;=\; \begin{array}{c|c|c||c} \boldsymbol{x_D} & \boldsymbol{x_E} & z & \\ \hline \boldsymbol{B}^{-1}\boldsymbol{A} & \boldsymbol{B}^{-1} & \mathbf{0} & \beta \\ \hline -\gamma_{\boldsymbol{D}} & -\gamma_{\boldsymbol{E}} & 1 & \zeta \end{array} \qquad \beta = \boldsymbol{B}^{-1}\boldsymbol{b} \geq \mathbf{0}$$

$$\boldsymbol{T_B} \;\overset{P}{=}\; \begin{array}{c|c|c||c} \boldsymbol{x_N} & \boldsymbol{x_B} & z & \\ \hline \boldsymbol{B}^{-1}\boldsymbol{N} & \boldsymbol{I} & \mathbf{0} & \beta \\ \hline -\gamma_{\boldsymbol{N}} & \mathbf{0} & 1 & \zeta \end{array} \qquad \beta = \boldsymbol{B}^{-1}\boldsymbol{b} \geq \mathbf{0}$$

# Summary: Optimal Tableau Signature

**Lemma**

*If $-\gamma_r \geq 0 \;\; \forall\, r \in \mathcal{N}$, then the tableau is* ***optimal***

**Signature** of an **optimal** tableau:

$$
\begin{array}{cccc|c||c}
x_1 & \dots & \dots & x_{n+m} & z & \\
\hline
 & & & & 0 & \oplus \\
 & & & & \vdots & \vdots \\
 & & & & 0 & \oplus \\
\hline
\oplus & \dots & \dots & \oplus & 1 & *
\end{array}
$$

# Summary: Unbounded Tableau Signature

**Lemma**

*If $\exists\ r \in \mathcal{N}$ so that $-\gamma_r < 0$ and $\boldsymbol{\alpha}^{(r)} \leq \mathbf{0}$, then the tableau is* ***unbounded***

**Signature** of an **unbounded** tableau:

$$
\begin{array}{ccccc|c||c}
x_1 & \dots & x_k & \dots & x_{n+m} & z & \\
\hline
 & & \ominus & & & 0 & \oplus \\
 & & & & & \vdots & \vdots \\
 & & \ominus & & & & \\
 & & \ominus & & & 0 & \oplus \\
\hline
* & \dots & - & \dots & * & 1 & *
\end{array}
$$

# Summary: Choosing the Next Pivot

If the conditions from the two previous lemmas are not satisfied, then one can perform a new pivoting resulting in a new feasible tableau $\boldsymbol{T_{B'}}$ where the value of the **new basic solution** is **equal or larger** than the one associated to $\boldsymbol{T_B}$

The new pivot $\alpha_{jr}$ is chosen

1. in a column $r \in \mathcal{N}$ such that $-\gamma_r < 0$,
2. in a row $j$ such that $\frac{\beta_j}{\alpha_{jr}} = \min \left\{ \frac{\beta_i}{\alpha_{ir}} \mid \alpha_{ir} > 0 \right\}$

The value of the new basic solution is:

$$\zeta' = \zeta + \gamma_r \frac{\beta_j}{\alpha_{jr}} \geq \zeta$$

# Primal Simplex Algorithm - Phase II

**Input**: a **feasible** tableau **Output**: an **optimal** or an **unbounded** tableau

(1) **Entering** column: choose a non-basis column $r$ with a negative reduced cost:

$$r \in \{k \in \mathcal{N} \mid -\gamma_k < 0\}.$$

It this column does not exist, then STOP: the current tableau is optimal

(2) **Exiting** column: choose a row $j$ minimizing the ratio

$$j \in \left\{k \in \{1, \ldots, m\} \mid \frac{\beta_k}{\alpha_{kr}} = \min\left\{\frac{\beta_i}{\alpha_{ir}} \mid \alpha_{ir} > 0\right\}\right\}$$

If this row doesn't exist, then STOP: the current tableau is non-bounded

(3) Update the basis and the tableau: pivot around $\alpha_{jr}$ and go back to (1)

# Example: Resolution with the Simplex Algorihm

We consider the following LP:

$$
\begin{array}{lrcrcr}
\text{Max} \quad z = & 800x_1 & + & 300x_2 & & \\
\text{s.t.} & 2x_1 & + & x_2 & \leq & 400 \\
 & x_1 & & & \leq & 150 \\
 & & & x_2 & \leq & 200 \\
 & x_1 & , & x_2 & \geq & 0
\end{array}
$$

# Example - Tableau $T_0$

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $z$ | | ratio | |
|---|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 0 | 0 | 0 | 400 | 200 | |
| **1** | 0 | 0 | 1 | 0 | 0 | 150 | 150 | $\leftarrow$ |
| 0 | 1 | 0 | 0 | 1 | 0 | 200 | - | |
| -800 | -300 | 0 | 0 | 0 | 1 | 0 | | |
| $\uparrow$ | | | | | | | | |

# Example - Tableau $T_1$

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $z$ | | ratio | |
|---|---|---|---|---|---|---|---|---|
| 0 | **1** | 1 | -2 | 0 | 0 | 100 | 100 | $\leftarrow$ |
| 1 | 0 | 0 | 1 | 0 | 0 | 150 | - | |
| 0 | 1 | 0 | 0 | 1 | 0 | 200 | 200 | |
| 0 | -300 | 0 | 800 | 0 | 1 | 120000 | | |
| | $\uparrow$ | | | | | | | |

# Example - Tableau $T_2$

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $z$ | |
|---|---|---|---|---|---|---|
| 0 | 1 | 1 | -2 | 0 | 0 | 100 |
| 1 | 0 | 0 | 1 | 0 | 0 | 150 |
| 0 | 0 | -1 | 2 | 1 | 0 | 100 |
| 0 | 0 | 300 | 200 | 0 | 1 | 150000 |

This tableau is **optimal**. The optimal solution is given by $x_1^* = 150$, $x_2^* = 100$. The optimal value is $z^* = 150000$

# Does this Algorithm Work ?

- A sufficient condition would be to demonstrate that the sequence of values associated to basic solutions is **strictly** increasing. Following a pivoting, the new value of the objective function is

$$\zeta' = \zeta + \gamma_r \frac{\beta_j}{\alpha_{jr}}$$

- $\gamma_r > 0$ ( $\iff -\gamma_r < 0$) due to the choice of column $r$ , $\alpha_{jr} > 0$ due to the choice of row $j$ and $\beta_j \geq 0$ since the tableau is feasible
- If $\beta_j > 0$, we have that $\zeta' > \zeta$ and the **new** basis has never been **visited** yet
- If $\beta_j = 0$, the pivoting is **stationary** (meaning that $\zeta' = \zeta$) and after several stationary pivotings, **it is possible to visit a basis more than once**
- In the latter situation, the algorithm may **cycle** and never produce the optimal solution

# Degeneracy

- A tableau and its associated basic solution are called degenerated if **at least** one of the components of vector $\beta$ is null
- A standard LP is called degenerated if it has at least one **degenerated** basic solution
- Degeneracy occurs if, in the space of decision variables (i.e. $\mathbb{R}^n$), it exists more than $n$ hyperplans defined by the the constraints (including bound constraints) **intersecting at the same point**
- It may create cycles in algorithms using pivotings

# Bland's Rule

To avoid **cycling** and to be sure that the algorithm stops after a **finite** number of iterations, Bland has defined a very simple rule:

**Rule**

When several candidates may **enter** or **exit** a basis, **always** choose the variable $x_r$ with the **smallest index**

We can apply this rule to **degenerated** tableaus only

# Example: Bland's Rule

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ | $z$ | |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | $-1$ | 0 | 0 | 0 | 3 |
| 0 | **1** | 0 | $-1$ | 1 | 0 | 0 | 1 |
| 0 | 1 | $-1$ | 0 | 0 | 1 | 0 | 1 |
| 0 | $-1$ | $-1$ | 1 | 0 | 0 | 1 | $-3$ |

- Two candidates for the **entering** variable: $x_2$ and $x_3$ (reduced costs are $< 0$)
- We take the one with the **smallest** index: $x_2$
- Then two candidates for the pivots (both have a ratio of 1)
  - If we choose $\alpha_{22}$, then $x_5$ will leave the basis
  - If we choose $\alpha_{32}$, then $x_6$ will leave the basis
- As $x_5$ has a smaller index that $x_6$, then we choose $\alpha_{22}$ as the **next pivot**

# Theoretical Performance of the Simplex Algorithm

**Complexity Theory**

One of the key issue in **complexity theory** is to determine if, for a specific problem, there exists an algorithm whose **execution time** is bounded by a **polynomial** in the size of the instance of the problem we consider

If such an algorithm exists, it is said to be **polynomial** and the problem can generally be solved in an **effective** way even for instances of important size

# Theoretical Performance of the Simplex Algorithm

**Question**

Is the simplex algorithm polynomial and, if it is not the case, does it exist polynomial algorithms for linear programming ?

# Klee-Minty Cube (1972)

- In 1972, Klee and Minty showed that after perturbating the corner of an hypercube $C$:

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid 0 \leq x_i \leq 1, i = 1, \ldots, n\},$$

  the simplex algorithm visited successively all the vertices of the feasible region before reaching the optimum

- The number of necessary pivoting is $2^n - 1$ in that case !

- The simplex algorithm is **exponential** time in the **worst** case !

# Empirical Performance of the Simplex Algorithm

- However, in many situations, the simplex method is remarkably **efficient** in practice
- It is the reason why many numerical packages include the simplex algorithm

# Polynomial Algorithms for Linear Programming

- The ellipsoid algorithm from Khachiyan. He was the first researcher to prove the **polynomial-time** solvability of linear programs. But this algorithm converges very slowly and is rarely used in practice

- Karmarkar's algorithm falls within the class of interior point methods: the current guess for the solution does not follow the **boundary of the feasible set** as in the simplex method, but it moves through the **interior of the feasible region**, improving the approximation of the optimal solution by a definite fraction with every iteration, and converging to an optimal solution

# Karmarkar's Algorithm