ORF522 – Linear and Nonlinear Optimization

5. The simplex method
Ed Forum

• Can neighboring basic solutions be infeasible?

Yes!

• Is there a chance that as we move from our starting basic feasible point and check all the neighboring solutions and find none of them to be more optimal, that we miss another point (that isn’t neighboring) that could be better? Is this an issue of identifying local vs. global optima?

“More optimal” does not exist! There is no way to get better solutions there. Proof of this in previous lecture. Yes, this is due to global optimality for LPs.

• I was under the impression that solvers used a standard step size for each problem and that they did not iteratively calculate one every single step. Would this not increase computational time in a significant manner? Standard step size is not a thing for simplex and interior-point methods. It always changes.

• I'm not exactly sure why \( d_j \) is always equal to one, and how do the equations and the picture correspond exactly?

Directions can be rescaled as we please (and change theta accordingly). We set \( d_j = 1 \) to simplify the math instead of having, e.g., \( d_j = 1.947 \) (which would allow us to derive the same things).
Recap
Standard form polyhedra

**Definition**

**Standard form LP**

minimize $c^T x$

subject to $Ax = b$

$x \geq 0$

**Assumption**

$A \in \mathbb{R}^{m \times n}$ has full row rank $m \leq n$

**Interpretation**

$P$ lives in $(n - m)$-dimensional subspace

$P = \{ x \mid Ax = b, \ x \geq 0 \}$

**Standard form polyhedron**

$\alpha_1 a_1^T x + \ldots + \alpha_m a_m^T x = b \quad \alpha \in \mathbb{R}$

$x = \overline{T} y$
Standard form polyhedra

Visualization

\[ P = \{ x \mid Ax = b, \ x \geq 0 \}, \quad n - m = 2 \]

Three dimensions

Higher dimensions
Constructing basic solution

1. Choose any $m$ independent columns of $A$: $A_{B(1)}, \ldots, A_{B(m)}$
2. Let $x_i = 0$ for all $i \neq B(1), \ldots, B(m)$
3. Solve $Ax = b$ for the remaining $x_{B(1)}, \ldots, x_{B(m)}$
Constructing basic solution

1. Choose any $m$ independent columns of $A$: $A_{B(1)}, \ldots, A_{B(m)}$
2. Let $x_i = 0$ for all $i \neq B(1), \ldots, B(m)$
3. Solve $Ax = b$ for the remaining $x_{B(1)}, \ldots, x_{B(m)}$

<table>
<thead>
<tr>
<th>Basis matrix $A_B$</th>
<th>Basis columns $A_{B(1)}$ $A_{B(2)}$ $\ldots$ $A_{B(m)}$</th>
<th>Basic variables $x_B$</th>
<th>Solve $A_Bx_B = b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_B = \begin{bmatrix} A_{B(1)} &amp; A_{B(2)} &amp; \ldots &amp; A_{B(m)} \end{bmatrix}$</td>
<td>$x_B = \begin{bmatrix} x_{B(1)} \ \vdots \ x_{B(m)} \end{bmatrix}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Constructing basic solution

1. Choose any $m$ independent columns of $A$: $A_{B(1)}, \ldots, A_{B(m)}$
2. Let $x_i = 0$ for all $i \neq B(1), \ldots, B(m)$
3. Solve $Ax = b$ for the remaining $x_{B(1)}, \ldots, x_{B(m)}$

$$ A_B = \begin{bmatrix} A_{B(1)} & A_{B(2)} & \ldots & A_{B(m)} \end{bmatrix}, \quad x_B = \begin{bmatrix} x_{B(1)} \\ \vdots \\ x_{B(m)} \end{bmatrix} \quad \longrightarrow \quad \text{Solve } A_Bx_B = b $$

If $x_B \geq 0$, then $x$ is a basic feasible solution
Neighboring solutions

Two basic solutions are neighboring if their basic indices differ by exactly one variable.
Neighboring solutions

Two basic solutions are **neighboring** if their basic indices differ by exactly one variable.

**Example**

\[
\begin{bmatrix}
1 & -1 & 0 & 3 & -2 \\
2 & 0 & -1 & -1 & 0 \\
0 & 2 & 4 & -1 & 4
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5
\end{bmatrix}
= 
\begin{bmatrix}
-5 \\
-1 \\
14
\end{bmatrix}
\]
Neighboring solutions

Two basic solutions are neighboring if their basic indices differ by exactly one variable.

Example

\[
A = \begin{bmatrix}
1 & -1 & 0 & 3 & -2 \\
2 & 0 & -1 & -1 & 0 \\
0 & 2 & 4 & -1 & 4 \\
\end{bmatrix}, \quad \begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
\end{bmatrix} = \begin{bmatrix}
-5 \\
-1 \\
14 \\
\end{bmatrix}
\]

\[B = \{1, 3, 5\}, \quad x_2 = x_4 = 0\]

\[A_B x_B = b \quad \rightarrow \quad x_B = \begin{bmatrix}
x_1 \\
x_3 \\
x_5 \\
\end{bmatrix} = \begin{bmatrix}
0 \\
1 \\
2.5 \\
\end{bmatrix}\]
Neighboring solutions

Two basic solutions are neighboring if their basic indices differ by exactly one variable.

Example

\[
\begin{bmatrix}
1 & -1 & 0 & 3 & -2 \\
2 & 0 & -1 & -1 & 0 \\
0 & 2 & 4 & -1 & 4 
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 
\end{bmatrix}
=\begin{bmatrix}
-5 \\
-1 \\
14 
\end{bmatrix}
\]

\[B = \{1, 3, 5\}\quad x_2 = x_4 = 0\]

\[A_B x_B = b \quad \rightarrow \quad x_B = \begin{bmatrix}
x_1 \\
x_3 \\
x_5 
\end{bmatrix} = \begin{bmatrix}
0 \\
1 \\
2.5 
\end{bmatrix}\]

\[\bar{B} = \{1, 3, 4\}\quad y_2 = y_5 = 0\]

\[A_{\bar{B}} y_{\bar{B}} = b \quad \rightarrow \quad y_{\bar{B}} = \begin{bmatrix}
y_1 \\
y_3 \\
y_4 
\end{bmatrix} = \begin{bmatrix}
0.1 \\
3.0 \\
-1.7 
\end{bmatrix}\]
Feasible directions

Conditions

\[ P = \{ x \mid Ax = b, \ x \geq 0 \} \]

Given a basis matrix \( A_B = \begin{bmatrix} A_B(1) & \cdots & A_B(m) \end{bmatrix} \)

we have basic feasible solution \( x \):

- \( x_B \) solves \( A_B x_B = b \)
- \( x_i = 0, \ \forall i \neq B(1), \ldots, B(m) \)
Feasible directions

Conditions

\[ P = \{ x \mid Ax = b, \ x \geq 0 \} \]

Given a basis matrix \( A_B = \begin{bmatrix} A_B(1) & \cdots & A_B(m) \end{bmatrix} \)
we have basic feasible solution \( x \):

- \( x_B \) solves \( A_B x_B = b \)
- \( x_i = 0, \ \forall i \neq B(1), \ldots, B(m) \)

Let \( x \in P \), a vector \( d \) is a **feasible direction** at \( x \)
if \( \exists \theta > 0 \) for which \( x + \theta d \in P \)

**Feasible direction** \( d \)

- \( A(x + \theta d) = b \implies Ad = 0 \)
- \( x + \theta d \geq 0 \)
Feasible directions

Computation

Nonbasic indices

- \( d_j = 1 \) \( \rightarrow \) Basic direction
- \( d_k = 0, \ \forall k \notin \{j, B(1), \ldots, B(m)\} \)

Feasible direction \( d \)

- \( A(x + \theta d) = b \implies Ad = 0 \)
- \( x + \theta d \geq 0 \)
Feasible directions

Computation

Nonbasic indices

- \( d_j = 1 \)  \rightarrow  Basic direction
- \( d_k = 0, \forall k \notin \{j, B(1), \ldots, B(m)\} \)

Basic indices

\[
Ad = 0 = \sum_{i=1}^{n} A_i d_i = A_B d_B + A_j = 0 \implies d_B = -A_B^{-1} A_j
\]

Feasible direction \( d \)

- \( A(x + \theta d) = b \implies Ad = 0 \)
- \( x + \theta d \geq 0 \)
Feasible directions

Computation

Nonbasic indices
• \(d_j = 1\) → Basic direction
• \(d_k = 0, \forall k \notin \{j, B(1), \ldots, B(m)\}\)

Basic indices
\[
Ad = 0 = \sum_{i=1}^{n} A_i d_i = A_B d_B + A_j = 0 \implies d_B = -A_B^{-1} A_j
\]

Non-negativity (non-degenerate assumption)
• Non-basic variables: \(x_i = 0\). Nonnegative direction \(d_i \geq 0\)
• Basic variables: \(x_B > 0\). Therefore \(\exists \theta > 0\) such that \(x_B + \theta d_B \geq 0\)

Feasible direction \(d\)
• \(A(x + \theta d) = b \implies Ad = 0\)
• \(x + \theta d \geq 0\)
Stepsize

What happens if some $\bar{c}_j < 0$?
We can decrease the cost by bringing $x_j$ into the basis
Stepsizes

What happens if some $c_j < 0$?
We can decrease the cost by bringing $x_j$ into the basis

How far can we go?

\[
\theta^* = \max\{\theta \mid \theta \geq 0 \text{ and } x + \theta d \geq 0\}
\]

$d$ is the $j$-th basic direction
Stepsize

What happens if some $\bar{c}_j < 0$?
We can decrease the cost by bringing $x_j$ into the basis

How far can we go?

$$\theta^* = \max\{\theta \mid \theta \geq 0 \text{ and } x + \theta d \geq 0\}$$

$d$ is the $j$-th basic direction

Unbounded
If $d \geq 0$, then $\theta^* = \infty$. The LP is unbounded.
Stepsize

What happens if some $c_j < 0$?

We can decrease the cost by bringing $x_j$ into the basis

How far can we go?

$\theta^* = \max\{\theta \mid \theta \geq 0 \text{ and } x + \theta d \geq 0\}$

$d$ is the $j$-th basic direction

Unbounded

If $d \geq 0$, then $\theta^* = \infty$. The LP is unbounded.

Bounded

If $d_i < 0$ for some $i$, then

$\theta^* = \min_{\{i \mid d_i < 0\}} \left(-\frac{x_i}{d_i}\right) = \min_{\{i \in B \mid d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$

(Since $d_i \geq 0$, $i \notin B$)
Moving to a new basis

Next feasible solution

\[ x + \theta^* d \]
Moving to a new basis

Next feasible solution

\[ x + \theta^* d \]

Let \( B(\ell) \in \{B(1), \ldots, B(m)\} \) be the index such that \( \theta^* = -\frac{x_{B(\ell)}}{d_{B(\ell)}} \). Then,

\[ x_{B(\ell)} + \theta^* d_{B(\ell)} = 0 \]
Moving to a new basis

Next feasible solution

\[ x + \theta^* d \]

Let \( B(\ell) \in \{B(1), \ldots, B(m)\} \) be the index such that \( \theta^* = -\frac{x_{B(\ell)}}{d_{B(\ell)}} \). Then,

\[ x_{B(\ell)} + \theta^* d_{B(\ell)} = 0 \]

New solution

- \( x_{B(\ell)} \) becomes 0 (exits)
- \( x_j \) becomes \( \theta^* \) (enters)
Moving to a new basis

Next feasible solution

$$x + \theta^* d$$

Let $B(\ell) \in \{B(1), \ldots, B(m)\}$ be the index such that $\theta^* = -\frac{x_{B(\ell)}}{d_{B(\ell)}}$. Then,

$$x_{B(\ell)} + \theta^* d_{B(\ell)} = 0$$

New solution

- $x_{B(\ell)}$ becomes 0 (exits)
- $x_j$ becomes $\theta^*$ (enters)

New basis

$$A_B = \begin{bmatrix} A_{B(1)} & \ldots & A_{B(\ell-1)} & [\overbrace{A_j}^{A_{B(\ell+1)}}] & A_{B(\ell+1)} & \ldots & A_{B(m)} \end{bmatrix}$$
An iteration of the simplex method

Initialization
- a basic feasible solution $x$
- a basis matrix $A_B = \begin{bmatrix} A_B(1) & \cdots & A_B(m) \end{bmatrix}$

Iteration steps

1. Compute the reduced costs $\bar{c}$
   - Solve $A_B^T p = c_B$
   - $\bar{c} = c - A_B^T p$

2. If $\bar{c} \geq 0$, $x$ optimal. break

3. Choose $j$ such that $\bar{c}_j < 0$

4. Compute search direction $d$ with $d_j = 1$ and $A_B d_B = -A_j$

5. If $d_B \geq 0$, the problem is unbounded and the optimal value is $-\infty$. break

6. Compute step length $\theta^* = \min_{\{i \in B | d_i \leq 0\}} \left( -\frac{x_i}{d_i} \right)$

7. Define $y$ such that $y = x + \theta^* d$

8. Get new basis $\bar{B}$ ($i$ exits and $j$ enters)
Today’s agenda
[Chapter 3, LO]

• Find initial feasible solution
• Degeneracy
• Complexity
Find an initial point in simplex method
Initial basic feasible solution

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

How do we get an initial basic feasible solution $x$ and a basis $B$?

Does it exist?
Finding an initial basic feasible solution

minimize \( c^T x \)
subject to \( Ax = b \)
\( x \geq 0 \)
Finding an initial basic feasible solution

minimize \( c^T x \)
subject to \( Ax = b \)
\( x \geq 0 \)

Auxiliary problem
minimize \( 1^T y \)
subject to \( Ax + y = b \)
\( x \geq 0, y \geq 0 \)
Finding an initial basic feasible solution

minimize \( c^T x \)
subject to \( Ax = b \)
\( x \geq 0 \)

Auxiliary problem

minimize \( 1^T y \)
subject to \( Ax + y = b \)
\( x \geq 0, y \geq 0 \)

Minimize violations
Finding an initial basic feasible solution

minimize \( c^T x \)
subject to \( Ax = b \)
\( x \geq 0 \)

Auxiliary problem

minimize \( 1^T y \)
subject to \( Ax + y = b \)
\( x \geq 0, y \geq 0 \)

Assumption \( b \geq 0 \) w.l.o.g. (if not multiply constraint by \(-1\))

Trivial basic feasible solution: \( x = 0, y = b \)
Finding an initial basic feasible solution

**Auxiliary problem**

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad Ax = b \\
& \quad x \geq 0
\end{align*}
\]

**Auxiliary problem**

\[
\begin{align*}
\text{minimize} & \quad 1^T y \\
\text{subject to} & \quad Ax + y = b \\
& \quad x \geq 0, y \geq 0
\end{align*}
\]

**Minimize violations**

\[ A I J \{x, y\} = b \]

**Assumption** \( b \geq 0 \) w.l.o.g. (if not multiply constraint by \(-1\))

**Trivial** basic feasible solution: \( x = 0, y = b \)

**Possible outcomes**

- **Feasible problem** (cost = 0): \( y^* = 0 \) and \( x^* \) is a basic feasible solution
- **Infeasible problem** (cost > 0): \( y^* > 0 \) are the violations
Two-phase simplex method

**Phase I**
1. Construct **auxiliary problem** such that \( b \geq 0 \)
2. Solve auxiliary problem using simplex method starting from \((x, y) = (0, b)\)
3. If the optimal value is greater than 0, **problem infeasible. break.** \((P \leq +\infty)\)

**Phase II**
1. Recover original problem (drop variables \( y \) and restore original cost)
2. Solve original problem starting from the solution \( x \) and its basis \( B \).
Big-M method

minimize \quad c^T x + M 1^T y
subject to \quad Ax + y = b
\quad x \geq 0, y \geq 0
Big-M method

minimize \( c^T x + M 1^T y \)
subject to \( Ax + y = b \)
\( x \geq 0, y \geq 0 \)

Very large constant
Big-M method

minimize \( c^T x + M1^T y \)
subject to
\[
Ax + y = b \\
x \geq 0, y \geq 0
\]

Incorporate penalty in the cost

- We can still use \( y = b \geq 0 \) as initial basic feasible solution
- If the problem is feasible, \( y \) will not be in the basis.

Very large constant
Big-M method

minimize \( c^T x + M 1^T y \)
subject to \( Ax + y = b \)
\( x \geq 0, y \geq 0 \)

Incorporate penalty in the cost

• We can still use \( y = b \geq 0 \) as initial basic feasible solution
• If the problem is feasible, \( y \) will not be in the basis.

Remarks

• Pro: need to solve only one LP
• Con: it is not easy to pick \( M \) and it makes the problem badly scaled
Degeneracy
Degenerate basic feasible solutions
Inequality form polyhedron

A solution $y$ is degenerate if $|\mathcal{I}(\bar{x})| > n$

$$P = \{x \mid Ax \leq b\}$$
Degenerate basic feasible solutions

Standard form polyhedron

Given a basis matrix \( A_B = \begin{bmatrix} A_B(1) & \ldots & A_B(m) \end{bmatrix} \)
we have basic feasible solution \( x \):  

- \( A_B x_B = b \)
- \( x_i = 0, \ \forall i \neq B(1), \ldots, B(m) \)
Degenerate basic feasible solutions

Standard form polyhedron

Given a basis matrix $A_B = \begin{bmatrix} A_{B(1)} & \ldots & A_{B(m)} \end{bmatrix}$

we have basic feasible solution $x$:

- $A_B x_B = b$
- $x_i = 0, \ \forall i \neq B(1), \ldots, B(m)$

If some of the $x_B = 0$, then it is a degenerate solution
Degenerate basic feasible solutions

Standard form polyhedron

Given a basis matrix $A_B = \begin{bmatrix} A_{B(1)} & \ldots & A_{B(m)} \end{bmatrix}$
we have basic feasible solution $x$:

- $A_B x_B = b$
- $x_i = 0$, $\forall i \neq B(1), \ldots, B(m)$

If some of the $x_B = 0$, then it is a degenerate solution

$$P = \{ x \mid Ax = b, \ x \geq 0 \}$$
Degenerate basic feasible solutions

Example

\[ x_1 + x_2 + x_3 = 1 \]
\[ -x_1 + x_2 - x_3 = 1 \]
\[ x_1, x_2, x_3 \geq 0 \]
Degenerate basic feasible solutions

Example

\[
\begin{align*}
 x_1 + x_2 + x_3 &= 1 \\
 -x_1 + x_2 - x_3 &= 1 \\
 x_1, x_2, x_3 &\geq 0
\end{align*}
\]

Degenerate solutions

Basis \( B = \{1, 2\} \quad \rightarrow \quad x = (0, 1, 0) \)
Degenerate basic feasible solutions

Example

\[ \begin{align*}
    x_1 + x_2 + x_3 &= 1 \\
    -x_1 + x_2 - x_3 &= 1 \\
    x_1, x_2, x_3 &\geq 0
\end{align*} \]

Degenerate solutions

Basis \( B = \{1, 2\} \quad \rightarrow \quad x = (0, 1, 0) \)

Basis \( B = \{2, 3\} \quad \rightarrow \quad y = (0, 1, 0) \)
Cycling

Stepsize

6. Compute step length $\theta^* = \min_{\{i \in B | d_i < 0\}} \left( -\frac{x_i}{d_i} \right)$
6. Compute step length $\theta^* = \min_{\{i \in B \mid d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$

If $i \in B$, $d_i < 0$ and $x_i = 0$ (degenerate)

$\theta^* = 0$
6. Compute step length \( \theta^* = \min_{\{i \in B \mid d_i < 0\}} \left(-\frac{x_i}{d_i}\right) \)

If \( i \in B, d_i < 0 \) and \( x_i = 0 \) (degenerate)

\[ \theta^* = 0 \]

Therefore \( y = x + \theta^* x = x \) and \( B \neq \overline{B} \)

**Same** solution and cost

**Different** basis
Cycling

6. Compute step length \( \theta^* = \min_{\{i \in B | d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \)

If \( i \in B, d_i < 0 \) and \( x_i = 0 \) (degenerate)
\[ \theta^* = 0 \]

Therefore \( y = x + \theta^* x = x \) and \( B \neq \bar{B} \)

**Same** solution and cost
**Different** basis

Finite termination **no longer guaranteed!**

How can we fix it?
6. Compute step length $\theta^* = \min_{\{i \in B | d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$

If $i \in B$, $d_i < 0$ and $x_i = 0$ (degenerate)

$\theta^* = 0$

Therefore $y = x + \theta^* x = x$ and $B = \bar{B}$

**Same** solution and cost

**Different** basis

Finite termination **no longer guaranteed**!

How can we fix it?

**Pivoting rules**
Pivoting rules
Choose the index entering the basis

Simplex iterations
3. Choose $j$ such that $\bar{c}_j < 0$
Pivoting rules
Choose the index entering the basis

Simplex iterations
3. Choose $j$ such that $\bar{c}_j < 0$ \hspace{2cm} Which $j$?
Pivoting rules
Choose the index entering the basis

Simplex iterations
3. Choose $j$ such that $\bar{c}_j < 0$ → Which $j$?

Possible rules
• **Smallest subscript:** smallest $j$ such that $\bar{c}_j < 0$
• **Most negative:** choose $j$ with the most negative $\bar{c}_j$
• **Largest cost decrement:** choose $j$ with the largest $\theta^*|\bar{c}_j|$
Pivoting rules

Choose index exiting the basis

Simplex iterations

6. Compute step length \( \theta^* = \min_{\{i \in B \mid d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \)
Pivoting rules

Choose index exiting the basis

Simplex iterations

6. Compute step length \( \theta^* = \min_{\{i \in B \mid d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \)

We can have more than one \( i \) for which \( x_i = 0 \)

(next solution is degenerate)

Which \( i \)?
Pivoting rules
Choose index exiting the basis

Simplex iterations
6. Compute step length \( \theta^* = \min_{\{i \in B | d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \)

We can have more than one \( i \) for which \( x_i = 0 \)
(next solution is degenerate)

Which \( i \)?

Smallest index rule
Smallest \( i \) such that \( \theta^* = -\frac{x_i}{d_i} \)
Bland’s rule to avoid cycles

**Theorem**
If we use the **smallest index rule** for choosing both the $j$ entering the basis and the $i$ leaving the basis, then **no cycling will occur**.
Bland’s rule to avoid cycles

Theorem
If we use the smallest index rule for choosing both the $j$ entering the basis and the $i$ leaving the basis, then no cycling will occur.

Proof idea [Ch 3, Sec 4, LP][Sec 3.4, LO]
• Assume Bland’s rule is applied and there exists a cycle with different bases.
• Obtain contradiction.
Perturbation approach to avoid cycles
Perturbation approach to avoid cycles
Complexity
Complexity

**Basic operation:** one simplex iteration

**Estimate complexity of an algorithm**
- Write number of basic operations as a *function of problem dimensions*
- Simplify and keep only leading terms
Complexity

Notation

We write $g(x) \sim O(f(x))$ if and only if there exist $c > 0$ and an $x_0$ such that

$$|g(x)| \leq cf(x), \quad \forall x \geq x_0$$
Complexity

Notation
We write $g(x) \sim O(f(x))$ if and only if there exist $c > 0$ and an $x_0$ such that

$$|g(x)| \leq cf(x), \quad \forall x \geq x_0$$
\( \mathcal{P} \) and \( \mathcal{NP} \)

**Complexity class \( \mathcal{P} \)**

There exists a polynomial time algorithms to solve it
\( \mathcal{P} \) and \( \mathcal{NP} \)

**Complexity class \( \mathcal{P} \)**
There exists a polynomial time algorithms to solve it

**Complexity class \( \mathcal{NP} \)**
Given a candidate solution, there exists a polynomial time algorithm to verify it.
\( \mathcal{P} \) and \( \mathcal{NP} \)

**Complexity class** \( \mathcal{P} \)
There exists a polynomial time algorithms to solve it

**Complexity class** \( \mathcal{NP} \)
Given a candidate solution, there exists a polynomial time algorithm to verify it.

**Complexity class** \( \mathcal{NP} \)-hard
At least as hard as the hardest problem in \( \mathcal{NP} \)
\( \mathcal{P} \) and \( \mathcal{NP} \)

**Complexity class \( \mathcal{P} \)**
There exists a polynomial time algorithms to solve it.

**Complexity class \( \mathcal{NP} \)**
Given a candidate solution, there exists a polynomial time algorithm to verify it.

**Complexity class \( \mathcal{NP} \)-hard**
At least as hard as the hardest problem in \( \mathcal{NP} \).

We don’t know any polynomial time algorithm.
**P and NP**

**Complexity class P**
There exists a polynomial time algorithms to solve it

**Complexity class NP**
Given a candidate solution, there exists a polynomial time algorithm to verify it.

**Complexity class NP-hard**
At least as hard as the hardest problem in NP

---

**Million dollar problem:** $P = NP$?
- We know that $P \subset NP$
- Does it exist a polynomial time algorithm for $NP$-hard problems?
Complexity of the simplex method
Example of worst-case behavior

Innocent-looking problem

minimize \(-x_n\)
subject to \(0 \leq x \leq 1\)

\[\begin{align*}
2^n & \text{ vertices} \\
2^n/2 & \text{ vertices: cost } = 1 \\
2^n/2 & \text{ vertices: cost } = 0
\end{align*}\]
Complexity of the simplex method

Example of worst-case behavior

Innocent-looking problem

minimize $-x_n$
subject to $0 \leq x \leq 1$

$2^n$ vertices
$2^n/2$ vertices: cost $= 1$
$2^n/2$ vertices: cost $= 0$

Perturb unit cube

minimize $-x_n$
subject to $\epsilon \leq x_1 \leq 1$

$\epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}$, $i = 2, \ldots, n$
Complexity of the simplex method

Example of worst-case behavior

minimize \(-x_n\)

subject to \(\epsilon \leq x_1 \leq 1\)

\(\epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}, \quad i = 2, \ldots, n\)
Complexity of the simplex method

Example of worst-case behavior

minimize $-x_n$
subject to $\epsilon \leq x_1 \leq 1$
$\epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}, \quad i = 2, \ldots, n$

Theorem
• The vertices can be ordered so that each one is adjacent to and has a lower cost than the previous one
• There exists a pivoting rule under which the simplex method terminates after $2^n - 1$ iterations
Complexity of the simplex method

Example of worst-case behavior

minimize \(-x_n\)

subject to \(\epsilon \leq x_1 \leq 1\)
\[\epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}, \quad i = 2, \ldots, n\]

Theorem

- The vertices can be ordered so that each one is adjacent to and has a lower cost than the previous one
- There exists a pivoting rule under which the simplex method terminates after \(2^n - 1\) iterations

Remark

- A different pivot rule would have converged in one iteration.
- We have a bad example for every pivot rule.
Complexity of the simplex method

We do not know any polynomial version of the simplex method, no matter which pivoting rule we pick. Still open research question!
Complexity of the simplex method

We do not know any polynomial version of the simplex method, no matter which pivoting rule we pick. Still open research question!

Worst-case
There are problem instances where the simplex method will run an exponential number of iterations in terms of the dimensions $n$ and $m$: $O(2^n)$
Complexity of the simplex method

We do not know any polynomial version of the simplex method, no matter which pivoting rule we pick.

Still open research question!

Worst-case
There are problem instances where the simplex method will run an exponential number of iterations in terms of the dimensions \( n \) and \( m \): \( O(2^n) \)

Good news: average-case
Practical performance is very good. On average, it stops in \( O(n) \) iterations.
The simplex method

Today, we learned to:

• **Formulate** auxiliary problem to find starting simplex solutions

• **Apply** pivoting rules to avoid cycling in degenerate linear programs

• **Analyze** complexity of the simplex method
Next lecture

- Numerical linear algebra
- "Realistic" simplex implementation
- Examples