ORF522 – Linear and Nonlinear Optimization

5. The simplex method
Ed forum

• What is the geometric picture of the standard form? Given a standard form P, can we always convert it back to the version defined by halfspaces? (next slides)

• Extent to which these methods are generalizable to infinite-dimensional restrictions e.g. linear difference equations when t goes to infinity.

• Efficient way to deal with inverses? (next lecture)

• How to pick index entering the basis? (this lecture)

• Adjacent solutions why defined that way? (Same active constraints except 1)

• Feasibility LP condition in no strong arbitrage from Arrow-Debreu theory (That’s correct! We will discuss feasibility in duality lectures)
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
Standard form polyhedra

Visualization

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

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 $x_{B(1)}, \ldots, x_{B(m)}$

Basis matrix

$B = \begin{bmatrix}
A_{B(1)} & A_{B(2)} & \cdots & A_{B(m)}
\end{bmatrix}$

Basis columns

Basic variables

$x_B = \begin{bmatrix}
x_{B(1)} \\
\vdots \\
x_{B(m)}
\end{bmatrix}$

$x_B = B^{-1}b$
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 \( x_{B(1)}, \ldots, x_{B(m)} \)

\[
B = \begin{bmatrix}
A_{B(1)} & A_{B(2)} & \cdots & A_{B(m)}
\end{bmatrix}, \quad x_B = \begin{bmatrix}
x_{B(1)} \\
\vdots \\
x_{B(m)}
\end{bmatrix} \quad \rightarrow \quad x_B = B^{-1}b
\]

If \( x_B \geq 0 \), then \( x \) is a basic feasible solution
Feasible directions

Conditions

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

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

- \( x_B = B^{-1}b \)
- \( x_i = 0, \ \forall i \neq B(1), \ldots, B(m) \)
Feasible directions

Conditions

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

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

Feasible direction $d$

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

Computation

Nonbasic indices

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

Computation

Nonbasic indices

• $d_j = 1$  \quad \rightarrow \quad \text{Basic direction}

• $d_k = 0, \ \forall k \notin \{j, B(1), \ldots, B(m)\}$

Basic indices

$$Ad = 0 = \sum_{i=1}^{n} A_i d_i = Bd_B + A_j = 0 \implies d_B = -B^{-1} A_j$$
Feasible directions

Computation

**Nonbasic indices**

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

**Basic indices**

$Ad = 0 = \sum_{i=1}^{n} A_i d_i = Bd_B + A_j = 0 \implies d_B = -B^{-1}A_j$

**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 $x_B + \theta d_B \geq 0$
Stepsize

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

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

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

$$\left( \text{Since } d_i \geq 0, \ i \in N \right)$$
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

\[ \tilde{B} = \begin{bmatrix} A_{B(1)} & \cdots & A_{B(\ell-1)} & A_j & A_{B(\ell+1)} & \cdots & A_{B(m)} \end{bmatrix} \]
An iteration of the simplex method

First part

We start with a basic feasible solution $x$ and a basis matrix $B = \begin{bmatrix} A_{B(1)} & \cdots & A_{B(m)} \end{bmatrix}$
An iteration of the simplex method
First part

We start with a basic feasible solution $x$ and a basis matrix $B = \begin{bmatrix} A_{B(1)} & \cdots & A_{B(m)} \end{bmatrix}$

1. Compute the reduced costs $\bar{c}_j = c_j - c_B^T B^{-1} A_j$ for $j \in N$

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

3. Choose $j$ such that $\bar{c}_j < 0$
An iteration of the simplex method
Second part

4. Compute search direction components $d_B = -B^{-1}A_j$

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

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

7. Define $y$ such that $y = x + \theta^* d$
Today’s agenda

• 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 \textbf{basic feasible solution} \( x \) and a \textbf{basis} \( B \) ?

Does it \textbf{exist}?
Finding an initial basic feasible solution

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

Auxiliary problem

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

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

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

Minimize violations

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

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.

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

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

minimize \[ c^T x + M1^T y \]
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.
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

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

A solution $\bar{x}$ is degenerate if $|I(\bar{x})| > n$
Degenerate basic feasible solutions
Definition

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

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

Definition

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

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

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

Definition

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

- \( x_B = B^{-1}b \)
- \( x_i = 0, \forall i \neq B(1), \ldots, B(m) \)

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

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

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

Stepsize

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

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

\[ \theta^* = 0 \]
Cycling

Stepsize

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

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

$$\theta^* = 0$$

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

Same solution and cost

Different basis
Cycling

Stepsizes

\[ \theta^* = \min_{\{i \in B | d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \quad \text{If } i \in B, \ d_i < 0 \text{ and } x_i = 0 \text{ (degenerate)} \]

\[ \theta^* = 0 \]

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

Same solution and cost
Different basis

Finite termination no longer guaranteed!

How can we fix it?
Cycling

Stepsize

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

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

\[ \theta^* = 0 \]

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

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 $c_j < 0$
Pivoting rules
Choose the index entering the basis

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

Simplex iterations
3. Choose $j$ such that $\bar{c}_j < 0$  \hspace{1cm} \text{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 $\theta^* = \min_{\{i \in B | d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$
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 \( \theta^* = \min_{\{i \in B \mid d_i < 0\}} \left( -\frac{x_i}{d_i} \right) \)

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

Which \( i \)?

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 (left as exercise)
- Assume that Bland’s rule is applied and there exists a cycle with different bases.
- Obtain same basis.
Perturbation approach to avoid cycles
Perturbation approach to avoid cycles

\[ Ax = b + \epsilon \]
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 algorithm to solve it.
$\mathcal{P}$ and $\mathcal{NP}$

**Complexity class $\mathcal{P}$**
There exists a polynomial time algorithm to solve it.

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

\textbf{Complexity class P}\newline
There exists a polynomial time algorithm to solve it.

\textbf{Complexity class NP}\newline
Given a candidate solution, there exists a polynomial time algorithm to verify it.

\textbf{Complexity class \textit{NP}}\textbf{-hard}\newline
The problem is at least as hard as the hardest problem in \textit{NP}. 
$P$ and $NP$

**Complexity class $P$**
There exists a polynomial time algorithm to solve it.

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

**Complexity class $NP$-hard**
The problem is at least as hard as the hardest problem in $NP$.

\[\downarrow\]

We don’t know any polynomial time algorithm
\( \mathcal{P} \) and \( \mathcal{NP} \)

**Complexity class** \( \mathcal{P} \)
There exists a polynomial time algorithm 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
The problem is at least as hard as the hardest problem in \( \mathcal{NP} \).

\[ \downarrow \]

We don’t know any polynomial time algorithm

**Million dollar problem:** \( \mathcal{P} = \mathcal{NP} \)?
- We know that \( \mathcal{P} \subset \mathcal{NP} \)
- Does it exist a polynomial time algorithm for \( \mathcal{NP} \)-hard problems?
Complexity of the simplex method

Example of worst-case behavior

Innocent-looking problem

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

2\(^n\) vertices
\(2^n/2\) vertices: cost = 1
\(2^n/2\) vertices: cost = 0
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 $\epsilon \leq x_1 \leq 1$

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

minimize \(-x_n\)
subject to \(\epsilon \leq x_1 \leq 1\)
\(\epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}, \quad i = 2, \ldots, n\)
Complexity of the simplex method

Example of worst-case behavior

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

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 $2^n - 4$ iterations $O(2^n)$
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 \(O(2^n)\)

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