Lecture 2

Revised simplex algorithm, bounded variables

Taha sections
- 7.1, 7.2, 7.3
- all examples can be read briefly

Terminology

<table>
<thead>
<tr>
<th>j' th column in A</th>
<th>Taha</th>
<th>INTOPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>basis</td>
<td>P_j</td>
<td>A_j</td>
</tr>
<tr>
<td>reduced cost</td>
<td>z_j - c_j</td>
<td>\tau_j</td>
</tr>
</tbody>
</table>

Linear Programming (Taha example 3.2.1)

maximize \[ 2x_1 + 3x_2 \]
subject to \[ 2x_1 + x_2 \leq 4 \]
\[ x_1 + 2x_2 \leq 5 \]
\[ x_1, x_2 \geq 0 \]

Add slack variables

maximize \[ 2x_1 + 3x_2 + x_3 \]
subject to \[ 2x_1 + x_2 + x_3 = 4 \]
\[ x_1 + 2x_2 + x_4 = 5 \]
\[ x_1, x_2, x_3, x_4 \geq 0 \]

The set of constraints form a polyhedral.

<table>
<thead>
<tr>
<th>Non-basic</th>
<th>Basic</th>
<th>Basic Solution</th>
<th>Corner Point</th>
<th>Feasible</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_1, x_2)</td>
<td>(x_1, x_4)</td>
<td>(x_2, x_4)</td>
<td>(x_1, x_3)</td>
<td>( x_1, x_2 )</td>
<td>( x_3, x_4 )</td>
</tr>
<tr>
<td>(x_1, x_3)</td>
<td>(x_2, x_3)</td>
<td>(x_2, x_4)</td>
<td>(x_1, x_4)</td>
<td>( x_1, x_3 )</td>
<td>( x_2, x_4 )</td>
</tr>
<tr>
<td>(x_1, x_4)</td>
<td>(x_2, x_1)</td>
<td>(x_2, x_3)</td>
<td>(x_1, x_4)</td>
<td>( x_2, x_1 )</td>
<td>( x_2, x_3 )</td>
</tr>
<tr>
<td>(x_2, x_3)</td>
<td>(x_1, x_2)</td>
<td>(x_2, x_4)</td>
<td>(x_1, x_3)</td>
<td>( x_1, x_2 )</td>
<td>( x_2, x_3 )</td>
</tr>
<tr>
<td>(x_2, x_4)</td>
<td>(x_1, x_1)</td>
<td>(x_2, x_3)</td>
<td>(x_1, x_4)</td>
<td>( x_2, x_1 )</td>
<td>( x_2, x_3 )</td>
</tr>
<tr>
<td>(x_3, x_4)</td>
<td>(x_1, x_2)</td>
<td>(x_3, x_4)</td>
<td>(x_1, x_3)</td>
<td>( x_3, x_4 )</td>
<td>( x_1, x_2 )</td>
</tr>
</tbody>
</table>

Basis, basis feasible solution

Since we have added slack variables, the number of variables \( n \) is larger than the number of constraints \( m \).

maximize \[ cx \]
subject to \[ Ax = b \]
\[ x \geq 0 \]

Choose \( m \) linearly independent columns from \( A \). The corresponding set \( B = \{ i_1, i_2, \ldots, i_m \} \) is called a basis.

Reformulation

maximize \[ c_B x_B + c_N x_N \]
subject to \[ A_B x_B + A_N x_N = b \]
\[ x \geq 0 \]

A Basis feasible solution is obtained by setting \( x_N = 0 \).

\[ A_B x_B + A_N 0 = b \]
\[ x_B = A_B^{-1} b \]

\( x_B \) is well defined since \( A_B \) is an \( m \times m \) matrix and columns are linearly independent.
**Algorithm** Search through all corner points
Basis can be chosen in \( C_m^n = \frac{n!}{m!(n-m)!} \) ways

**Adjacent basis feasible solutions** Two basis feasible solutions \( x^1 \) and \( x^2 \) are adjacent if \( B^1 \) and \( B^2 \) have \( m-1 \) common elements.
(one entering, one leaving variable)

**Simplex algorithm** is a greedy algorithm which works as follows: Move from basis feasible solution to adjacent basis feasible solution such that objective function is “increased most possible” in each step.

---

**Canonical form (Taha notation)**

Objective function \( \bar{c}x \) expressed in nonbasis variables only (canonical form)

\[
\bar{c}_B = 0
\]

This can be obtained by considering the objective function as an ordinary constraint

\[
\begin{align*}
z - cx &= 0 \\
Ax &= b \\
x &\geq 0
\end{align*}
\]

Multiplying any constraint \( i \) by a real number \( \pi_i \) and adding it to some other constraint \( j \) does not change the problem. In particular, we can add any constraint to the objective function.

\[
z - cx + \pi_i(Ax - b) = 0
\]

\[
z + (\pi_iA - c)x = \pi_i b
\]

\[
z + (\bar{\pi}_A - c_B)x_N + (\pi A_N - c_N)x_N = \pi b
\]

To have a canonical form \( \bar{c}_B = 0 \) so we must have \( \pi A_B - c_B = 0 \). From this we can determine \( \pi \) as

\[
\pi = c_B A_B^{-1}
\]

Thus the objective function becomes

\[
z - (c_B A_B^{-1} A_N - c_N) x_N = (c_B A_B^{-1}) b
\]

\[
z + \bar{c}_N x_N = z_0
\]

where \( \bar{c}_N = c_B A_B^{-1} A_N - c_N \) are the reduced costs.

---

**Simplex, example 3.2.1**

<table>
<thead>
<tr>
<th></th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>1 2 -3 0 0</td>
<td>0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>0 0 1 0 0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>0 0 1 0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Entering variable \( x_2 \)
maximum value of entering variable \( \min \{\frac{4}{7}, \frac{5}{7}\} = \frac{5}{7} \)
leaving variable is \( x_4 \)

**Pivot row 3 with \( \left( \frac{2}{7}, -\frac{1}{7}, \frac{1}{7} \right) \)**

<table>
<thead>
<tr>
<th></th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>1 -\frac{2}{7} 0 0</td>
<td>0</td>
<td>\frac{5}{7}</td>
<td>\frac{15}{7}</td>
<td>\frac{5}{7}</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>0 \frac{2}{7} 0 1 -\frac{1}{7}</td>
<td>\frac{1}{7}</td>
<td>\frac{3}{7}</td>
<td>\frac{5}{7}</td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0 \frac{2}{7} 1 0 1</td>
<td>\frac{3}{7}</td>
<td>\frac{5}{7}</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Entering variable \( x_1 \)
maximum value of entering variable \( \min \{\frac{3}{7}, \frac{5}{7}\} = 1 \)
leaving variable is \( x_3 \)

**Pivot row 2 with \( \left( \frac{4}{7}, -\frac{2}{7}, -\frac{1}{7} \right) \)**

<table>
<thead>
<tr>
<th></th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>1 0 0 \frac{1}{7}</td>
<td>\frac{2}{7}</td>
<td>\frac{3}{7}</td>
<td>\frac{8}{7}</td>
<td>\frac{8}{7}</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>0 1 0 \frac{3}{7} -\frac{1}{7}</td>
<td>1</td>
<td>\frac{3}{7}</td>
<td>\frac{8}{7}</td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0 0 1 -\frac{1}{7} \frac{2}{7}</td>
<td>2</td>
<td>\frac{3}{7}</td>
<td>\frac{8}{7}</td>
<td></td>
</tr>
</tbody>
</table>

All reduced costs in objective are positive, hence stop

---

**Canonical form (Cormen notation)**

The objective function \( \bar{c}x \) is in canonical form if

\[
\bar{c}_B = 0
\]

This can be obtained by considering the objective function as an ordinary constraint

\[
\begin{align*}
z &= cx \\
Ax &= b \\
x &\geq 0
\end{align*}
\]

Multiplying any constraint \( i \) by a real number \( \pi_i \) and adding it to some other constraint \( j \) does not change the problem. In particular, we can add any constraint to the objective function.

\[
z - cx + \pi_i(Ax - b) = 0
\]

\[
z + (\pi_iA - c)x = \pi_i b
\]

\[
z + (c_B - \pi_i A_B)x_N + (\pi A_N - c_N)x_N = \pi b
\]

To have a canonical form \( \bar{c}_B = 0 \) so we must have \( c_B - \pi A_B = 0 \). From this we can determine \( \pi \) as

\[
\pi = c_B A_B^{-1}
\]

Thus the objective function becomes

\[
z - (c_B A_B^{-1} A_N - c_N) x_N = (c_B A_B^{-1}) b
\]

\[
z + \bar{c}_N x_N = z_0
\]

where \( \bar{c}_N = c_B A_B^{-1} A_N - c_N \) are the reduced costs.
Current solution value

The basis feasible solution is found by setting \( x_N = 0 \). In this case the objective value becomes:

\[ z + \bar{c}_N x_N = z_0 \]

Reduced costs

The reduced costs \( \bar{c}_j \) represent the gain by increasing the value of a non-basis variable.

Iterative step

Choose the variable \( s \in N \) with largest negative value of \( \bar{c}_s \) to enter basis.

Optimality criteria

If all the reduced costs \( \bar{c}_j \geq 0 \) for a given basis feasible solution \( x \), then \( x \) is an optimal solution.

<table>
<thead>
<tr>
<th>basic</th>
<th>( z )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>1</td>
<td>(-\frac{1}{2})</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
<td>(-\frac{1}{2} )</td>
<td>( 1 )</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
<td>0</td>
<td>1</td>
<td>(-\frac{1}{2} )</td>
<td>( \frac{1}{2} )</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
<td>1</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
<td>( 2 )</td>
</tr>
</tbody>
</table>

Iteration 1:

Most promising variable \( x_1 \)

Keeping all other nonbasic variables at 0, constraints

\[ x_3 \geq 0, \quad x_3 = \frac{3}{4} - \frac{3}{4} x_1 \Rightarrow \frac{3}{4} \geq \frac{3}{4} x_1 \geq 0 \]

implying \( x_1 \leq 1 \).

When \( x_1 = 1 \) we have \( x_3 = 0 \) (\( x_3 \) leaves basis)

Revised simplex algorithm, maximization (Taha 7.2.2)

0) Construct a starting basis feasible solution and let \( A_B \) and \( c_B \) be its associated basis and objective.

1) Compute the inverse \( A_B^{-1} \).

2) For each nonbasis variable \( j \in N \) compute

\[ \bar{c}_j = z_j - c_j = c_B A_B^{-1} A_j - c_j \]

if \( \bar{c}_j \geq 0 \) for all nonbasis \( j \in N \) stop; optimal solution

\[ x_B = A_B^{-1} b, \quad z = c_B x_B \]

Else, apply optimality condition to find entering variable \( s \)

\[ s = \arg \min_{j \in N} \{ \bar{c}_j \} \]

3) Compute \( \bar{A}_s = A_B^{-1} A_s \).

If \( \bar{A}_s \leq 0 \) the problem is unbounded, stop.

Else, compute \( \bar{b} = A_B^{-1} b \).

Feasibility check

\[ k = \arg \min_{i=1,...,m} \left\{ \frac{\bar{b}_i}{a_{is}} \middle| a_{is} > 0 \right\} \]

Leaving variable: basis variable corresponding to row \( k \), \( r = B_k \).

4) New basis is \( B := B \cup \{ s \} \setminus \{ r \} \). Go to step 1.

When Simplex terminates

Assume that problem is bounded and feasible. When simplex terminates we have

- solution value

\[ z^* = c_B A_B^{-1} b \]

- objective function

\[ z - (c_B A_B^{-1} A_N - c_N) x_N = (c_B A_B^{-1}) b \]

- reduced costs

\[ \bar{c} = c_B A_B^{-1} A_N - c_N \geq 0 \]

- basis equations

\[ x_B + A_B^{-1} A_N x_N = A_B^{-1} b \]

- nonnegativity of all variables

\[ x \geq 0 \]
Advantages of revised simplex

- Simplex table expressed in original variables
- Valuable interpretation of all terms in simplex table
- Can avoid to write $A$ explicitly (e.g. delayed column generation)

All commercial simplex algorithms use revised simplex

- Fewer calculations needed, since only maintain $A_B^{-1}$ and right-hand side
- Less storage, due to same arguments
- Rest of $A$ can be stored in compact form (only storing non-zero elements, low precision)
- Calculations made on $A_B^{-1}$ with high precision

Bigger example, The Reddy Mikks Company

Problem formulation in standard form

\[
\begin{align*}
\text{maximize} & \quad 5x_1 + 4x_2 \\
\text{subject to} & \quad 6x_1 + 4x_2 \leq 24 \\
& \quad x_1 + 2x_2 \leq 6 \\
& \quad -x_1 + x_2 \leq 1 \\
& \quad 2x_1 + x_2 \leq 2 \\
& \quad x_1, x_2 \geq 0
\end{align*}
\]

Add slack variables $x_3, x_4, x_5, x_6$

\[
\begin{align*}
\text{maximize} & \quad 5x_1 + 4x_2 \\
\text{subject to} & \quad 6x_1 + 4x_2 + x_3 = 24 \\
& \quad x_1 + 2x_2 + x_4 = 6 \\
& \quad -x_1 + x_2 + x_5 = 1 \\
& \quad x_2 + x_6 = 2 \\
& \quad x_1, x_2, x_3, x_4, x_5, x_6 \geq 0
\end{align*}
\]

Revised simplex algorithm

Example, Reddy Mikks.

\[
\max \ z = (5, 4, 0, 0, 0, 0)(x_1, x_2, x_3, x_4, x_5, x_6)^T
\]

subject to

\[
\begin{pmatrix}
6 & 4 & 1 & 0 & 0 & 0 \\
1 & 2 & 0 & 1 & 0 & 0 \\
-1 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{pmatrix}
= \begin{pmatrix} 24 \\ 6 \\ 1 \\ 2 \end{pmatrix}
\]
Iteration 0

\[B = \{3, 4, 5, 6\}\]
\[N = \{1, 2\}\]
\[c_B = (0, 0, 0, 0)\]
\[A_B = (A_3, A_4, A_5, A_6) = I\]
\[A_B^{-1} = I\]

Thus
\[x_B = A_B^{-1}b = (24, 6, 1, 2)^T\]
\[z = c_Bx_B = 0\]

Optimality computation

\[c_BA_B^{-1} = (0, 0, 0, 0)\]
\[\{\overline{e}_j\}_{j \in N} = c_BA_B^{-1}(A_1, A_2) - (c_1, c_2) = (-5, -4)\]

entering variable
\[s = \arg\min_{j \in N} \{\overline{e}_j\} = \arg\min_{j = 1, 2} \{-5, -4\} = 1\]

Feasibility computation

\[\overline{b} = A_B^{-1}b = (24, 6, 1, 2)^T\]
\[\overline{A}_s = A_B^{-1}A_s = (6, 1, -1, 0)^T\]

Feasibility computation

\[\overline{b} = (x_1, x_2, x_3, x_4, x_5, x_6)^T = (4, 2, 5, 2)^T\]
\[\overline{A}_s = A_B^{-1}A_s = (\frac{2}{3}, \frac{4}{3}, \frac{5}{3}, 1)^T\]

leaving variable

\[k = \arg\min_{i = 1, \ldots, m} \left\{ \frac{e_i}{\overline{a}_{ii}} \right\} a_{ii} > 0 \]
\[= \arg\min \left\{ \frac{4}{3}, \frac{2}{3}, \frac{5}{3}, 1 \right\} \]
\[= \arg\min \{6, \frac{3}{2}, 3, 2\} = 2\]

hence, leaving variable is second element in \(B = \{1, 4, 5, 6\}\), so leaving variable is \(r = 4\).
Iteration 2

\[ B = \{1, 2, 5, 6\} \]
\[ N = \{3, 4\} \]
\[ c_B = (5, 4, 0, 0) \]
\[ A_B = (A_1, A_2, A_5, A_6) = \begin{pmatrix}
6 & 4 & 0 & 0 \\
1 & 2 & 0 & 0 \\
-1 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 
\end{pmatrix} \]

we find
\[
A_B^{-1} = \frac{1}{8} \begin{pmatrix}
2 & -4 & 0 & 0 \\
-1 & 6 & 0 & 0 \\
3 & -10 & 8 & 0 \\
1 & -6 & 0 & 8 
\end{pmatrix}
\]

thus
\[ x_B = A_B^{-1} b = (3, 3, 5, 1)^T \]
\[ z = c_B x_B = 21 \]

Optimality computation
\[ c_B A_B^{-1} = \begin{pmatrix}
(3, 1, 5, 0, 0) \\
\{\tilde{e}_j\}_{j \in N} = c_B A_B^{-1} (A_3, A_4) - (c_3, c_4) = (\frac{3}{4}, \frac{1}{2})
\end{pmatrix} \]

Thus \( B \) is optimal, stop.

Optimal solution
\[
\begin{align*}
x_1 &= 3 \\
x_2 &= \frac{3}{4} \\
z &= 21
\end{align*}
\]

Applications of OR

Example 6.3.2

- weighted graph \( G = (V, E) \)
- each edge \((i, j)\) has probability of success \( p_{ij} \)
- find most reliable route \( s \to t \)

Introduce \( x_{ij} = 1 \) iff edge \((i, j)\) is used

- flow conservation: \( \sum_{j \in V} x_{ij} - \sum_{j \in V} x_{ji} = 0 \)
- one edge leaving \( s \), one edge entering \( t \)
- objective

\[
\max \prod_{x_{ij}=1} p_{ij}
\]

<table>
<thead>
<tr>
<th>( x_{12} )</th>
<th>( x_{13} )</th>
<th>( x_{23} )</th>
<th>( x_{24} )</th>
<th>( x_{34} )</th>
<th>( x_{35} )</th>
<th>( x_{45} )</th>
<th>( x_{46} )</th>
<th>( x_{57} )</th>
<th>( x_{67} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ s \]
\[ t \]
\[ 1 \]
\[ 2 \]
\[ 3 \]
\[ 4 \]
\[ 5 \]
\[ 6 \]

Applications of OR

Objective
\[
\max \log \prod_{x_{ij}=1} p_{ij} = \max \sum_{(i, j) \in E} \log p_{ij} x_{ij} = \min \sum_{(i, j) \in E} -\log p_{ij} x_{ij}
\]

Size of input
- \( V \) nodes, \( E \) edges with cost

Size of model
- \( E = O(V^2) \) variables
- \( O(V) \) constraints
- Size of \( A \)-matrix \( O(V^3) \)

Revised simplex
- Only store \( A_B \) of size \( O(V^2) \)
- Columns in \( A \) can be generated “on the fly” from graph

Note

Cormen is using a better LP-formulation of shortest-path
Advanced comments

Pivot operation corresponds to multiplying current $A_B^{-1}$ with

$$E = \begin{pmatrix}
1 & \nu_1 \\
\vdots & \vdots \\
1 & \nu_r \\
\vdots & \vdots \\
1 & \nu_m
\end{pmatrix}
$$

If current basis inverse is $A_B^{-1}$ and right-hand side is $\bar{b}$ then new basis and right-hand side is

$$EA_B^{-1} \quad \bar{b}
$$

If initial basis is $I$ and operations $E_1, E_2, \ldots, E_k$ then

$$A_B^{-1} = E_k \ldots E_2 E_1 I \quad \bar{b} = E_k \ldots E_2 E_1 b$$

Advanced algorithms

- Maintain $A_B^{-1}$ in product form
- Only column $v_1, \ldots, v_m$ is stored from $E$
- When $A_B^{-1} = E_k \ldots E_2 E_1$ becomes too complex to calculate, store result, start over again.

Simplex, example 3.2.1

<table>
<thead>
<tr>
<th>Iteration 0:</th>
<th>basic</th>
<th>$z$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$z$</td>
<td>1</td>
<td>-2</td>
<td>-3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>$x_4$</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
</tbody>
</table>

Entering variable $x_2$

maximum value of entering variable $\min\{4, 5\} = 5$

leaving variable is $x_4$

Pivot row 3 with $(\frac{3}{2}, -\frac{1}{2}, \frac{1}{2})$

<table>
<thead>
<tr>
<th>Iteration 1:</th>
<th>basic</th>
<th>$z$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$z$</td>
<td>1</td>
<td>$\frac{4}{2}$</td>
<td>0</td>
<td>$\frac{5}{2}$</td>
<td>$\frac{3}{2}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$x_3$</td>
<td>0</td>
<td>$\frac{7}{2}$</td>
<td>0</td>
<td>$\frac{7}{2}$</td>
<td>$\frac{7}{2}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$x_2$</td>
<td>0</td>
<td>$\frac{7}{2}$</td>
<td>1</td>
<td>$\frac{7}{2}$</td>
<td>$\frac{7}{2}$</td>
<td></td>
</tr>
</tbody>
</table>

Entering variable $x_1$

maximum value of entering variable $\min\{\frac{3}{2}, \frac{5}{2}\} = 1$

leaving variable is $x_3$

Pivot row 2 with $(\frac{1}{3}, \frac{2}{3}, -\frac{1}{3})$

<table>
<thead>
<tr>
<th>Iteration 2:</th>
<th>basic</th>
<th>$z$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$z$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>$\frac{4}{3}$</td>
<td>$\frac{3}{3}$</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>$x_1$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>$\frac{2}{3}$</td>
<td>$-\frac{1}{3}$</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$x_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>$-\frac{1}{3}$</td>
<td>$\frac{2}{3}$</td>
<td>2</td>
</tr>
</tbody>
</table>

All reduced costs in objective are positive, hence stop

Example

iteration 1

$$E_1 = \begin{pmatrix}
1 & -\frac{1}{2} \\
0 & 1/2
\end{pmatrix}
$$

iteration 2

$$E_2 = \begin{pmatrix}
2 & 0 \\
-\frac{1}{3} & 1
\end{pmatrix}
$$

we have

$$A_B^{-1} = E_2 E_1 = \begin{pmatrix}
\frac{2}{3} & 0 \\
-\frac{1}{3} & 1
\end{pmatrix} \begin{pmatrix}
1 & -\frac{1}{2} \\
0 & 1/2
\end{pmatrix} = \begin{pmatrix}
\frac{2}{3} & -\frac{1}{2} \\
\frac{1}{3} & \frac{1}{2}
\end{pmatrix}
$$

Bounded variables

In production planning, branch-and-bound we frequently have

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

Lower bound

$$x \geq \ell$$

substitute

$$x = \ell + x'$$

$$x' \geq 0$$

solve problem in terms of $x'$. Back-substitute original variables $x = x' + \ell$
Bounded variables

Upper bound
Handle when computing max value of entering variable.

\[ x_j \leq \arg \min_{i=1,\ldots,m} \left\{ \frac{b_i}{a_{is}} \mid a_{is} > 0 \right\} \]

New constraints
We need to ensure \( x_i \geq 0 \) for \( i \in B \)

\[ x_j \leq \theta_j^i = \min_{i=1,\ldots,m} \left\{ \frac{b_i - u_i}{a_{is}} \mid a_{is} > 0 \right\} \]

and to ensure that \( x_i \leq u_i \) for \( i \in B \)

\[ x_j \leq \theta_j^i = \min_{i=1,\ldots,m} \left\{ \frac{b_i - u_i}{a_{is}} \mid a_{is} > 0 \right\} \]

combining the three restrictions

\[ x_j = \min(\theta_j^1, \theta_j^2, u_j) \]

Complexity of Simplex

Klee and Minty (1975) proved that the Simplex algorithm may use exponential time

\[
\begin{align*}
\text{maximize} & \quad 2^{n-1}x_1 + 2^{n-2}x_2 + \ldots + 2x_{n-1} + 1x_n \\
\text{subject to} & \quad 1x_1 + \ldots + 1x_n \leq 5 \\
& \quad 4x_1 + 1x_2 + \ldots + 1x_n \leq 5^2 \\
& \quad 8x_1 + 4x_2 + 1x_3 + \ldots + 1x_n \leq 5^3 \\
& \quad \vdots \\
& \quad 2^nx_1 + 2^{n-1}x_2 + \ldots + 4x_{n-1} + 1x_n \leq 5^n \\
x_i \geq 0, i = 1, \ldots, n
\end{align*}
\]

The problem has
\begin{itemize}
\item \( n \) variables
\item \( n \) constraints
\item \( 2^n \) extreme points
\item Simplex, starting at \( x = (0, \ldots, 0) \), visits all extreme points
\item optimal solution \((0, 0, \ldots, 0, 5^n)\)
\end{itemize}

Pitfalls:
\begin{itemize}
\item Since upper bounds \( x_j \leq u_j \) are handled implicit, no dual variable are calculated corresponding to the constraint
\item Duality theorem, complementary slackness seem to not work
\end{itemize}

Only use bounds if you know what you are doing, otherwise use explicit constraint

Complexity of Simplex

For \( n = 3 \) simplex visits \( 2^3 = 8 \) extreme points

Assume \((s_1, s_2, s_3)\) slack variables:

<table>
<thead>
<tr>
<th>basis</th>
<th>nonbasis x_1</th>
<th>x_2</th>
<th>x_3</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>-4</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td>( s_1 )</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>( s_2 )</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>( s_3 )</td>
<td>8</td>
<td>4</td>
<td>125</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>basis</th>
<th>nonbasis x_1</th>
<th>x_2</th>
<th>x_3</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>-4</td>
<td>2</td>
<td>-1</td>
<td>30</td>
</tr>
<tr>
<td>( s_1 )</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>( s_2 )</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>( s_3 )</td>
<td>8</td>
<td>4</td>
<td>65</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>basis</th>
<th>nonbasis x_1</th>
<th>x_2</th>
<th>x_3</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>-4</td>
<td>2</td>
<td>-1</td>
<td>75</td>
</tr>
<tr>
<td>( s_1 )</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>( s_2 )</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>( s_3 )</td>
<td>8</td>
<td>4</td>
<td>65</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>basis</th>
<th>nonbasis x_1</th>
<th>x_2</th>
<th>x_3</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>-4</td>
<td>2</td>
<td>1</td>
<td>105</td>
</tr>
<tr>
<td>( s_1 )</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>( s_2 )</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>( s_3 )</td>
<td>8</td>
<td>4</td>
<td>85</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>basis</th>
<th>nonbasis x_1</th>
<th>x_2</th>
<th>x_3</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>-4</td>
<td>2</td>
<td>1</td>
<td>125</td>
</tr>
<tr>
<td>( s_1 )</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>( s_2 )</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>( s_3 )</td>
<td>8</td>
<td>4</td>
<td>125</td>
<td></td>
</tr>
</tbody>
</table>
Complexity of Simplex

- Worst-case complexity is exponential (instances have been constructed which "fools" the greedy strategy to visit nearly all corner points).
- Several heuristics are used in commercial simplex implementations

<table>
<thead>
<tr>
<th>m \ n</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>9.4</td>
<td>14.2</td>
<td>17.4</td>
<td>19.4</td>
<td>20.2</td>
</tr>
<tr>
<td>20</td>
<td>25.2</td>
<td>30.7</td>
<td>38.0</td>
<td>41.5</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>44.4</td>
<td>52.7</td>
<td>62.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>67.6</td>
<td>78.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>95.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: Avis and Chvatal (1978).