Introduction to

Linear and Combinatorial Optimization

5 Simplex Implementation

5.1 Revised Simplex Method

Observation 5.1 To execute one iteration of the simplex method efficiently, it suffices to know B, the current solution x, and the input data A, b, and c. It is then easy to compute:

$$\bar{c}^\top = c^\top - c_B^\top \cdot A_B^{-1} \cdot A; \qquad u = A_B^{-1} \cdot A_j; \qquad \theta^* = \min_{i\,:\,u_i>0} \frac{x_{B(i)}}{u_i} = \frac{x_{B(\ell)}}{u_\ell}$$

The new basis matrix is then

$$A_{\bar{B}} = \left[A_{B(1)}, \ldots, A_{B(\ell-1)}, A_j, A_{B(\ell+1)}, \ldots, A_{B(m)}\right].$$

and the new solution y is given by $y_j = \theta^*, \quad y_B = x_B - \theta^* u$

- The above steps can be done (rather) efficiently if we know the inverse matrix A_B^{-1}
- We have not discussed initialization so far, but we will see later that we always start with a identity-basis: $A_B = A_B^{-1} = I$

Critical question: How to obtain $A_{\bar{B}}^{-1}$ efficiently?

Computing the Inverse of the Basis Matrix

- notice that $A_B^{-1} \cdot A_{\bar{B}} = (e_1, \dots, e_{\ell-1}, u, e_{\ell+1}, \dots, e_m)$
- Recall that the inverse of matrix M can be found by applying elementary row operations that transform M to I:

$$(\,M \,|\, I\,) \quad \overset{\text{row operations}}{\rightsquigarrow} \quad (\,I \,|\, M^{-1}\,)$$

- Row operations are equivalent to left multiplication by some invertible matrix.
- So we start left-multiplying with A_B^{-1}:

$$(\,A_{\bar{B}} \,|\, I\,) \quad \overset{\text{left mult. } A_B^{-1}}{\rightsquigarrow} \quad (\,(e_1, \dots, e_{\ell-1}, u, e_{\ell+1}, \dots, e_m) \,|\, A_B^{-1}\,)$$

- It simply remains to bring u to the unit vector e_ℓ. This is done as follows:
  - multiply ℓth row with $1/u_\ell$;
  - for $i \neq \ell$, subtract u_i times resulting ℓth row from ith row.

Obtaining $A_{\bar{B}}^{-1}$ from A_B^{-1}

Apply elementary row operations to the matrix $(A_B^{-1} \mid u)$ to make the last column equal to the unit vector e_ℓ. The first m columns of the resulting matrix form the inverse $A_{\bar{B}}^{-1}$ of the new basis matrix $A_{\bar{B}}$.

An Iteration of the Revised Simplex Method

Given: basis B, corresponding basic feasible solution x, and A_B^{-1}.

1 Let $p^\top := c_B^\top \cdot A_B^{-1}$ and $\bar{c}_j := c_j - p^\top \cdot A_j$, $j \in N$;

if $\bar{c} \geq 0$, then STOP; else choose j with $\bar{c}_j < 0$.

2 Let $u := A_B^{-1} \cdot A_j$. If $u \leq 0$, then STOP (optimal cost is $-\infty$).

3 Let $\theta^* := \min_{i : u_i > 0} \frac{x_{B(i)}}{u_i} = \frac{x_{B(\ell)}}{u_\ell}$ for some $\ell \in \{1, \ldots, m\}$.

4 Form new basis by replacing $A_{B(\ell)}$ with A_j; corresponding basic feasible solution y is given by

$$y_j := \theta^*, \qquad y_B = x_B - \theta^* u, \qquad y_{N \setminus \{j\}} = 0.$$

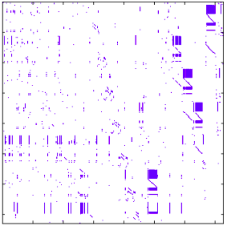
5 Apply elementary row operations to the matrix $(A_B^{-1} \mid u)$ to make the last column equal to the unit vector e_ℓ.

The first m columns of the resulting matrix yield $A_{\bar{B}}^{-1}$.

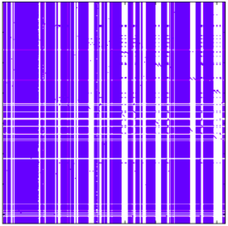
Is it a good idea to store A_B^{-1} ?

In practice, A is often sparse !

- In real-worl problems, A typically has only a few non-zero entries: much more efficient to store only non-zero coordinates.
- But the inverse base matrix is not always sparse!



A_B of size 363×363, density=2.5%



A_B^{-1} of size 363×363, density=61%

- We do not need the matrix inverse. We just need to solve linear systems

$$u = A_B^{-1} \cdot A_j \iff A_B \cdot u = A_j$$
$$p^\top = c_B^\top \cdot A_B^{-1} \iff A_B^\top \cdot p = c_B$$

Is it a good idea to store A_B^{-1}? (cont.)

- So, in pratice, in every iteration we nee to solve two linear systems, one involving the matrix A_B and one involving the matrix $A_B^\top$

- Assume we have a LU factorization of the matrix A_B, i.e. $A_B = L \cdot U$ with L lower triangular, U upper triangular.

- Then, both systems are of the form $L \cdot U \cdot z = h$, and can be solved by:
  - finding a vector v sucht that $L \cdot v = h$ using forward substitutions
  - finding a vector z such that $U \cdot z = v$ using backward substitutions

- The above operations are very fast if L and U are sparse: $O\big(\mathrm{nz}(L) + \mathrm{nz}(U)\big)$.

- In practice, LU decomposition is done by Gaussian Elimination (GE), and does not necessarily give sparse L and U. But, special GE algorithms can identify permutation matrices P, Q and sparse triangular matrices L, U such that $P \cdot A_B \cdot Q = L \cdot U$. Then,

$$A_B \cdot z = h \iff (\, L \cdot v = P \cdot h, \qquad U \cdot w = v, \qquad z = Q \cdot w \,)$$

Practical Performance Enhancements

- State-of-the art implementations update the LU factorization after an iteration, similarly as we did for A_B^{-1}.
- In fact, if $A_B = L \cdot U$, it can be seen that $A_{\bar{B}} = L \cdot R$, where R is "nearly upper triangular" (except for one column). Triangularity can be restored by decomposing $R = \tilde{L} \cdot \tilde{U}$; Then, $A_{\bar{B}} = L \cdot \tilde{L} \cdot \tilde{U}$.
- Computing $\tilde{L}$ and $\tilde{U}$ only takes $O(m)$, we only have to specify one row/column.
- After k iterations, we have a factorization of the form $A_B^{(k)} = L \cdot L_1 \cdot L_2 \cdots L_k \cdot U_k$, which can be used to solve the linear systems.

Numerical stability

The most critical issue when implementing the (revised) simplex method is numerical stability. In order to deal with this, a number of additional ideas from numerical linear algebra are needed.

- Every update of A_B^{-1} (or of the LU factorization) introduces roundoff or truncation errors which accumulate and might eventually lead to highly inaccurate results. Moreover the size of the LU factorization grows

  Solution: Compute the matrix A_B^{-1} (or $A_B = L \cdot U$) from scratch once in a while.

Introduction to

Linear and Combinatorial Optimization

5 Simplex Implementation

5.2 Full Tableau Implementation

Full Tableau Implementation

Main idea

Instead of maintaining and updating the matrix A_B^{-1}, we maintain and update the $m \times (n+1)$-matrix

$$A_B^{-1} \cdot (b \mid A) = (A_B^{-1} \cdot b \mid A_B^{-1} \cdot A)$$

which is called simplex tableau.

- The zeroth column $A_B^{-1} \cdot b$ contains x_B.
- For $i = 1, \ldots, n$, the ith column of the tableau is $A_B^{-1} \cdot A_i$.
- The column $u = A_B^{-1} \cdot A_j$ corresponding to the variable x_j that is about to enter the basis is the pivot column.
- If the ℓth basic variable $x_{B(\ell)}$ exits the basis, the ℓth row of the tableau is the pivot row.
- The element $u_\ell > 0$ is the pivot element.

Full Tableau Implementation (Cont.)

Notice: The simplex tableau $A_B^{-1} \cdot (b \mid A)$ represents the linear equation

$$A_B^{-1} \cdot b = A_B^{-1} \cdot A \cdot x$$

which is equivalent to $A \cdot x = b$.

Updating the simplex tableau

- At the end of an iteration, the simplex tableau $A_B^{-1} \cdot (b \mid A)$ has to be updated to $A_{\bar{B}}^{-1} \cdot (b \mid A)$.
- $A_{\bar{B}}^{-1}$ can be obtained from A_B^{-1} by elementary row operations, i.e., $A_{\bar{B}}^{-1} = Q \cdot A_B^{-1}$ where Q is a product of elementary matrices.
- Thus, $A_{\bar{B}}^{-1} \cdot (b \mid A) = Q \cdot A_B^{-1} \cdot (b \mid A)$, and new tableau $A_{\bar{B}}^{-1} \cdot (b \mid A)$ can be obtained by applying the same elementary row operations.

Zeroth Row of the Simplex Tableau

In order to keep track of the objective function value and the reduced costs, we consider the following augmented simplex tableau:

$-c_B^\top A_B^{-1} b$	$c^\top - c_B^\top A_B^{-1} A$
$A_B^{-1} b$	$A_B^{-1} A$

or in more detail

$-c_B^\top x_B$	$\bar{c}_1$	$\cdots$	$\bar{c}_n$
$x_{B(1)}$	$\vert$		$\vert$
$\vdots$	$A_B^{-1} A_1$	$\cdots$	$A_B^{-1} A_n$
$x_{B(m)}$	$\vert$		$\vert$

Updating the Zeroth row

- zeroth row is $\left[-c_B^\top A_B^{-1} b, c^\top - c_B^\top A_B^{-1} A\right] = \left[0, c^\top\right] - c_B^\top A_B^{-1}\left[b, A\right]$
- let $h^\top$ be the ℓth row of A_B^{-1}, then pivot row ℓ is $h^\top[b, A]$
- adding multiple λ of pivot row to zeroth row yields new zeroth row

$$[-c_B^\top x_B + \lambda h^\top b,\ \bar{c}^\top + \lambda h^\top A] = [0, c^\top] - (c_B^\top A_B^{-1} - \lambda h^\top)[b, A] =: [p_0, p^\top],$$

- for every $i \in B \setminus \{\ell\}$, we have $p_i = 0$ since $\bar{c}_i = 0$ and $h^\top A_i = (e_i)_\ell = 0$
- choose λ such that $p_j = 0$
- then $p_{\bar{B}} = c_{\bar{B}}^\top - (c_B^\top A_B^{-1} - \lambda h^\top)A_{\bar{B}} = 0$, i.e., $(c_B^\top A_B^{-1} - \lambda h^\top) = c_{\bar{B}}^\top A_{\bar{B}}^{-1}$

Update of zeroth row

The zeroth row is updated by adding a multiple of the pivot row to the zeroth row to set the reduced cost of the entering variable to zero.

An Iteration of the Full Tableau Implementation

Given: Simplex tableau corresp. to feasible basis $B = (B(1), \dots, B(m))$.

1. If $\bar{c} \geq 0$ (zeroth row), then STOP; else choose pivot column j with $\bar{c}_j < 0$.
2. If $u = A_B^{-1} A_j \leq 0$ (jth column), STOP (optimal cost is $-\infty$).
3. Choose pivot row ℓ with $\min\limits_{i\,:\,u_i > 0} \frac{x_{B(i)}}{u_i} = \frac{x_{B(\ell)}}{u_\ell}$ (compare columns 0 and j).
4. Apply elementary row operations to the simplex tableau so that u_ℓ (pivot element) becomes one and all other entries of the pivot column become zero (including zeroth row). The resulting tableau corresponds to new basis $\bar{B}$ in which $A_{B(\ell)}$ is replaced with A_j.

Full Tableau Implementation: An Example

A simple linear programming problem:

$$
\begin{array}{lrcrcrcr}
\min & -10\,x_1 & - & 12\,x_2 & - & 12\,x_3 & & \\
\text{s.t.} & x_1 & + & 2\,x_2 & + & 2\,x_3 & \leq & 20 \\
 & 2\,x_1 & + & x_2 & + & 2\,x_3 & \leq & 20 \\
 & 2\,x_1 & + & 2\,x_2 & + & x_3 & \leq & 20 \\
 & & & & & x_1, x_2, x_3 & \geq & 0
\end{array}
$$

Set of Feasible Solutions

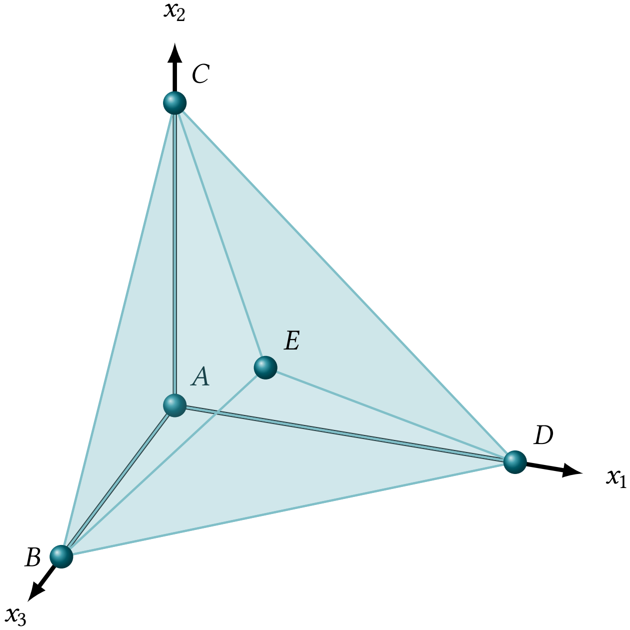
$A = (0,0,0)^\top$

$B = (0,0,10)^\top$

$C = (0,10,0)^\top$

$D = (10,0,0)^\top$

$E = (4,4,4)^\top$



Introducing Slack Variables

$$
\begin{array}{lrcrcrcr}
\min & -10\,x_1 & - & 12\,x_2 & - & 12\,x_3 & & \\
\text{s.t.} & x_1 & + & 2\,x_2 & + & 2\,x_3 & \le & 20 \\
 & 2\,x_1 & + & x_2 & + & 2\,x_3 & \le & 20 \\
 & 2\,x_1 & + & 2\,x_2 & + & x_3 & \le & 20 \\
 & & & & & x_1, x_2, x_3 & \ge & 0
\end{array}
$$

LP in standard form

$$
\begin{array}{lrcrcrcrcrcrcr}
\min & -10\,x_1 & - & 12\,x_2 & - & 12\,x_3 & & & & & & & & \\
\text{s.t.} & x_1 & + & 2\,x_2 & + & 2\,x_3 & + & x_4 & & & & & = & 20 \\
 & 2\,x_1 & + & x_2 & + & 2\,x_3 & & & + & x_5 & & & = & 20 \\
 & 2\,x_1 & + & 2\,x_2 & + & x_3 & & & & & + & x_6 & = & 20 \\
 & & & & & & & & & & & x_1, \ldots, x_6 & \ge & 0
\end{array}
$$

Observation

The right-hand side of the system is non-negative. Therefore the point $(0, 0, 0, 20, 20, 20)^\top$ is a basic feasible solution and we can start the simplex method with basis $B(1) = 4, B(2) = 5, B(3) = 6$.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	0	−10	−12	−12	0	0	0
$x_4 =$	20	1	2	2	1	0	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6	$\frac{x_{B(i)}}{u_i}$
	0	-10	-12	-12	0	0	0	
$x_4 =$	20	1	2	2	1	0	0	20
$x_5 =$	20	2	1	2	0	1	0	10
$x_6 =$	20	2	2	1	0	0	1	10

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6	$\frac{x_{B(i)}}{u_i}$
	0	−10	−12	−12	0	0	0	
$x_4 =$	20	1	2	2	1	0	0	20
$x_5 =$	20	2	1	2	0	1	0	10
$x_6 =$	20	2	2	1	0	0	1	10

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	0	−10	−12	−12	0	0	0
$x_4 =$	20	1	2	2	1	0	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\implies$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	20	1	2	2	1	0	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	20	1	2	2	1	0	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\implies$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\implies$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	0	0	1	−1	0	−1	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_5 =$	20	2	1	2	0	1	0
$x_6 =$	0	0	1	−1	0	−1	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\implies$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.

Setting Up the Simplex Tableau

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

- Determine pivot column (e.g., take smallest subscript rule).
- $\bar{c}_1 < 0$ and x_1 enters the basis.
- Find pivot row with $u_i > 0$ minimizing $\frac{x_{B(i)}}{u_i}$.
- Rows 2 and 3 both attain the minimum.
- Choose $i = 2$ with $B(i) = 5$. $\Longrightarrow$ x_5 leaves the basis.
- Perform basis change: Eliminate other entries in the pivot column.
- Obtain new basic feasible solution $(10, 0, 0, 10, 0, 0)^\top$ with cost -100.

Geometric Interpretation

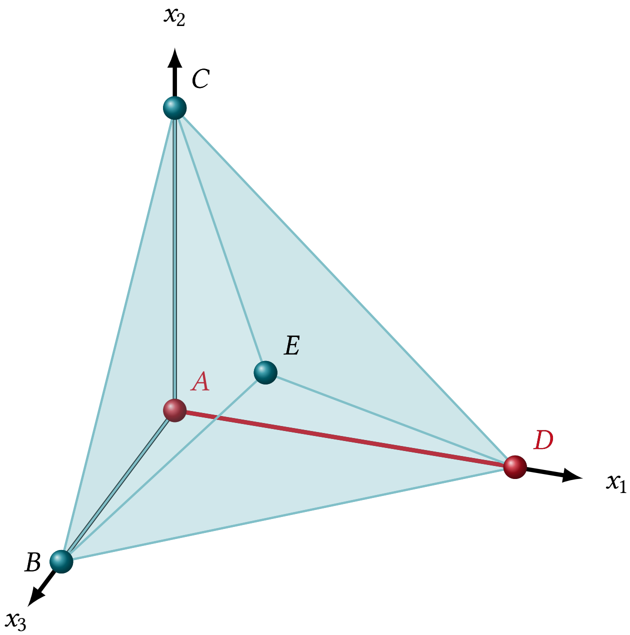
$A = (0, 0, 0)^\top$

$B = (0, 0, 10)^\top$

$C = (0, 10, 0)^\top$

$D = (10, 0, 0)^\top$

$E = (4, 4, 4)^\top$



Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6	$\frac{x_{B(i)}}{u_i}$
	100	0	−7	−2	0	5	0	
$x_4 =$	10	0	1.5	1	1	−0.5	0	10
$x_1 =$	10	1	0.5	1	0	0.5	0	10
$x_6 =$	0	0	1	−1	0	−1	1	–

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	-4	0	2	4	0
$x_4 =$	10	0	1.5	1	1	-0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	-1	0	-1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	0	1	−1	0	−1	1	0
$x_6 =$	0	0	1	−1	0	−1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	0	1	−1	0	−1	1	0
$x_6 =$	0	0	1	−1	0	−1	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
$x_4 =$	10	0	1.5	1	1	−0.5	0
$x_1 =$	0	1	−1	0	−1	1	0
$x_6 =$	10	0	2.5	0	1	−1.5	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
x_3 =	10	0	1.5	1	1	−0.5	0
x_1 =	0	1	−1	0	−1	1	0
x_6 =	10	0	2.5	0	1	−1.5	1

- $\bar{c}_2, \bar{c}_3 < 0 \implies$ two possible choices for pivot column.
- Choose x_3 to enter the new basis.
- $u_3 < 0 \implies$ third row cannot be chosen as pivot row.
- Choose x_4 to leave basis.
- New basic feasible solution $(0, 0, 10, 0, 0, 10)^\top$ with cost -120, corresponding to point B in the original polyhedron.

Geometric Interpretation

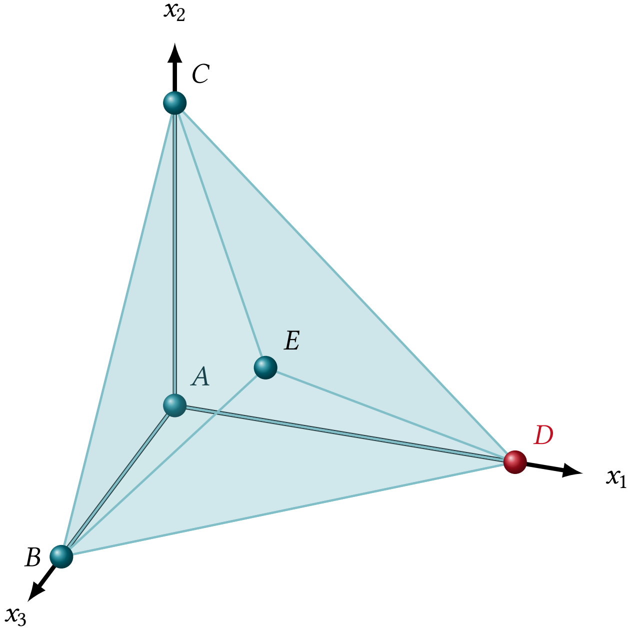
$A = (0, 0, 0)^\top$

$B = (0, 0, 10)^\top$

$C = (0, 10, 0)^\top$

$D = (10, 0, 0)^\top$

$E = (4, 4, 4)^\top$



Geometric Interpretation

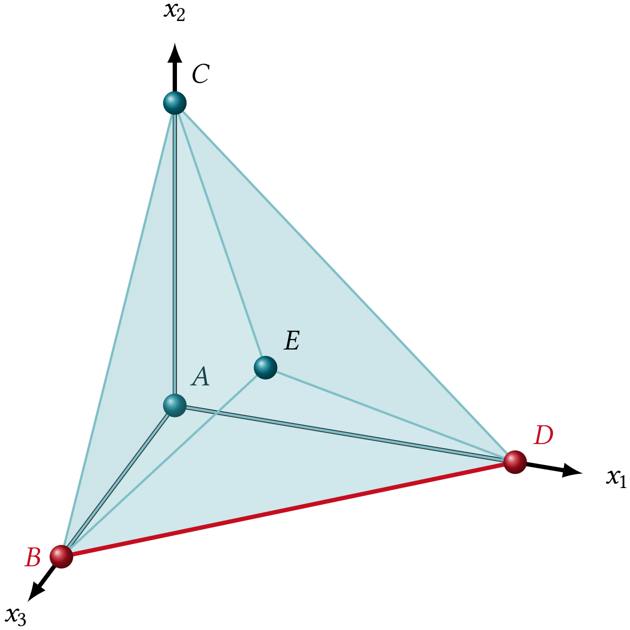
$A = (0, 0, 0)^\top$

$B = (0, 0, 10)^\top$

$C = (0, 10, 0)^\top$

$D = (10, 0, 0)^\top$

$E = (4, 4, 4)^\top$



Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6	$\frac{x_{B(i)}}{u_i}$
	120	0	−4	0	2	4	0	
$x_3 =$	10	0	1.5	1	1	−0.5	0	$\frac{20}{3}$
$x_1 =$	0	1	−1	0	−1	1	0	–
$x_6 =$	10	0	2.5	0	1	−1.5	1	4

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6	$\frac{x_{B(i)}}{u_i}$	
	120	0	-4	0	2	4	0		
$x_3 =$	10	0	1.5	1	1	-0.5	0	$\frac{20}{3}$	
$x_1 =$	0	1	-1	0	-1	1	0	$-$	
$x_6 =$	10	0	2.5	0	1	-1.5	1	4	$< \frac{20}{3}$

Next Iterations

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	−4	0	2	4	0
x_3 =	10	0	1.5	1	1	−0.5	0
x_1 =	0	1	−1	0	−1	1	0
x_6 =	10	0	2.5	0	1	−1.5	1

x_2 enters the basis, x_6 leaves it. We get

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

and the reduced costs are all non-negative.

Thus $(4, 4, 4, 0, 0, 0)$ is an optimal solution with cost -136, corresponding to point $E = (4, 4, 4)$ in the original polyhedron.

All Iterations from Geometric Point of View

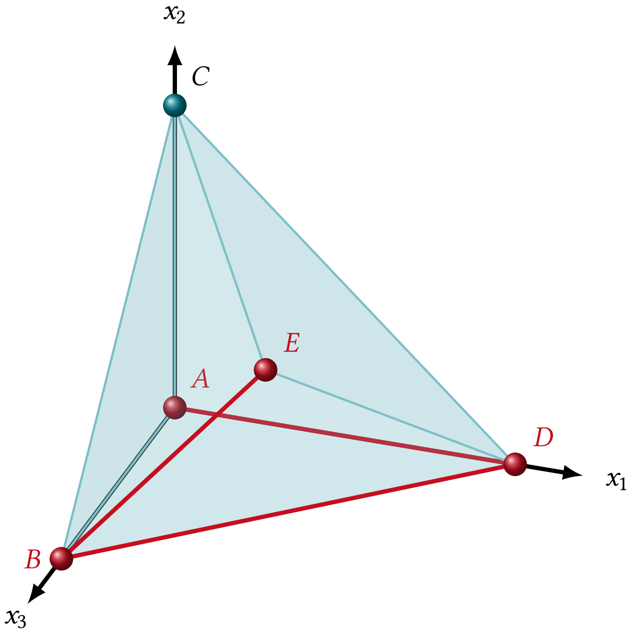
$A = (0, 0, 0)^\top$

$B = (0, 0, 10)^\top$

$C = (0, 10, 0)^\top$

$D = (10, 0, 0)^\top$

$E = (4, 4, 4)^\top$



Comparison

The following table gives the computational cost of one iteration of the simplex method for the two variants introduced above.

	full tableau	revised simplex	revised simplex sparse A	revised simplex LU factorization
memory	$O(mn)$	$O(mn)$	$O(m^2 + \mathrm{nz}(A))$	$O(m^2 + \mathrm{nz}(A))$
worst-case time	$O(mn)$	$O(mn)$	$O(m^2 + \mathrm{nz}(A))$	$O(m^2 + \mathrm{nz}(A))$
best-case time	$O(mn)$	$O(m^2)$	$O(m^2)$	$O(\mathrm{nz}(L) + \mathrm{nz}(U))$

Conclusion

- For implementation purposes, the revised simplex method is preferable due to its smaller memory requirement and average running time.
- The full tableau method is convenient for solving small LP instances by hand since all necessary information is readily available.