Lecture 5: Solving LPs - the Simplex method

Sondre Pedersen

January 23, 2025

1 Recap from last time

General linear programming problem looks like

$$\min_{x\in\mathbb{R}^n} c^\top x \qquad \text{subject to} \qquad \begin{cases} a_i^\top x = b_i, \ i \in \mathcal{E} \\ a_i^\top x \geq b_i, \ i \in \mathcal{I} \end{cases}.$$

These can always be reformulated into standard form

$$\min_{x\in\mathbb{R}^n} c^\top x \qquad \text{subject to} \qquad \begin{cases} Ax = b \\ \ x \geq 0 \end{cases} \qquad \text{where } A \in \mathbb{R}^{m\times n}, m < n.$$

The lagrangian is now

$$\mathcal{L}(x, \lambda, s) = c^\top x - \lambda^\top (Ax - b) - s^\top x.$$

The KKT conditions for linear programming:

$$\begin{aligned} A^\top \lambda^* + s^* &= c \\ Ax^* &= b \\ x^* &\geq 0 \\ s^* &\geq 0 \\ x_i^* s_i^* &= 0 \qquad i = 1, 2, \ldots, n \end{aligned}$$

Duality - Primal and dual have the same KKT conditions.

Primal problem

$$\begin{aligned} \min_{x} \quad & c^\top x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

Dual problem

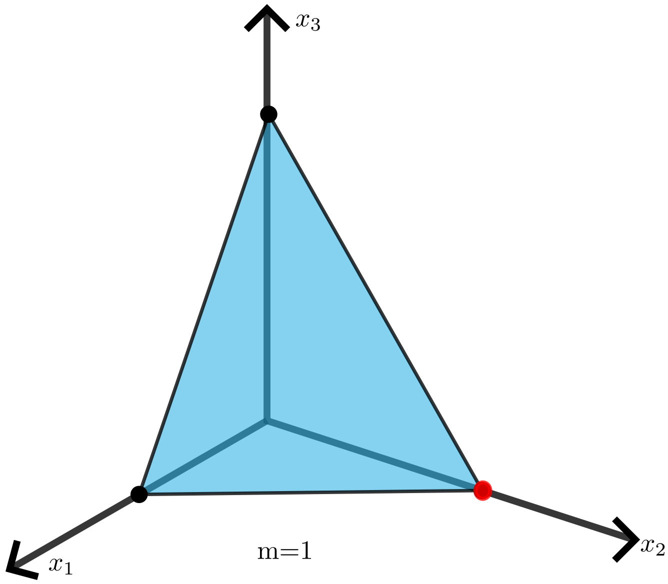
$$\begin{aligned} \max_{\lambda, s} \quad & b^\top \lambda \\ \text{s.t.} \quad & A^\top \lambda + s = c \\ & s \geq 0 \end{aligned}$$

2 Basic feasible points

A point x is a basic feasible point if

- x is feasible
- There is an index set $\mathcal{B}(x) \subset \{1, 2, , \ldots, n\}$ such that
  - $\mathcal{B}(x)$ contains m indices
  - $i \notin \mathcal{B}(x) \implies x_i = 0$
  - $B = \left[A_i\right]_{i\in\mathcal{B}(x)}$ is non-singular, $B \in \mathbb{R}^{m\times m}$
- $\mathcal{B}(x)$ is called a basis for the LP
- The indices not in $\mathcal{B}(x)$ are called $\mathcal{N}(x)$

These terms are hard to get a grasp on, so here is an example.



If $\mathcal{B}(x) = 2$ (the red point). We see that x_1 and x_3 is 0 at this point, but x_2 is not. $A = \begin{bmatrix} A_1 & A_2 & A_3 \end{bmatrix} \in \mathbb{R}^{1\times 3}$ and $B = A_2$, a scalar in this case.

For a different example, with $n = 5$ (five variables) and $m = 2$ (two constraints). Say $\mathcal{B}(x) = \{2, 3\}$ Now

$$Ax = b$$

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The dots indicate a non-zero value. The second and third column in A make up B, $B = \begin{bmatrix} A_i \end{bmatrix}_{i \in \mathcal{B}(x)}$

3 Simplex method

Theorem 1. *(Fundamental theorem of Linear Programming) For standard form LP*

- *If there is a feasible point, there is a BFP*
- *If the LP has a solution, one solution is a BOP*
- *If the LP is feasible and bounded, there is a solution*

Theorem 2. *All vertices of the feasible polytope*

$$\{x | Ax = b,\ x \geq \}.$$

are BFPs (and all BFPs are vertices)

One problem that can occur - Degeneracy. This is when a BFP x with $x_i = 0$ for some $i \in \mathcal{B}(x)$

Theorem 3. *If an LP is bounded and non-degenerate, the Simplex method terminates at a BOP*

But in practice, and with a good algorithm, this is not a big problem.

LP KKT conditions (necessary & sufficient)

The Simplex method iterates BFPs until one that fulfills KKT is found. Each step is a move from a vertex to a neighboring vertex (one change in the basis), that decreases the objective. Another example

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} \end{bmatrix} \begin{bmatrix} 0 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

x is a BFP:

$$\mathcal{B}(x) = \dots \qquad \mathcal{N}(x) = \{1, \dots, n\} \backslash \mathcal{B}(x)$$

$$B = \big[A_i\big]_{i \in \mathcal{B}(x)} \qquad N = \big[A_i\big]_{i \in \mathcal{N}(x)} = \begin{bmatrix} A_{11} & A_{14} & A_{15} \\ A_{21} & A_{24} & A_{25} \end{bmatrix}$$

$$x_B = \big[x_i\big]_{i \in \mathcal{B}(x)} \qquad x_N = \big[x_i\big]_{i \in \mathcal{N}(x)} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$$

$$s_B = \big[s_i\big]_{i \in \mathcal{B}(x)} \qquad s_N = \big[s_i\big]_{i \in \mathcal{N}(x)}$$

$$c_B = \big[c_i\big]_{i \in \mathcal{B}(x)} \qquad c_N = \big[c_i\big]_{i \in \mathcal{N}(x)}$$

One step of Simplex-algorithm - Given a BFP x and $B(x)$

- KKT-2: $Ax = Bx_B + Nx_N = Bx_B = b$
- KKT-3: $x_B B^{-1} b \geq 0, \; x_N = 0$
- KKT-5: $x^\top s = x_B^\top s_b + x_N^\top s_n = x_b^\top s_B$, set $s_B = 0$
- KKT-1: $\begin{bmatrix} s_B \\ s_N \end{bmatrix} + \begin{bmatrix} B^\top \\ N^\top \end{bmatrix} \lambda = \begin{bmatrix} c_B \\ c_N \end{bmatrix} \implies \begin{cases} \lambda = (B^\top)^{-1} c_B \\ s_N = c_N - N^\top \lambda \end{cases}$
- KKT-4: Ok if $s_N \geq 0 \implies$ BFP is BOP, so we are done

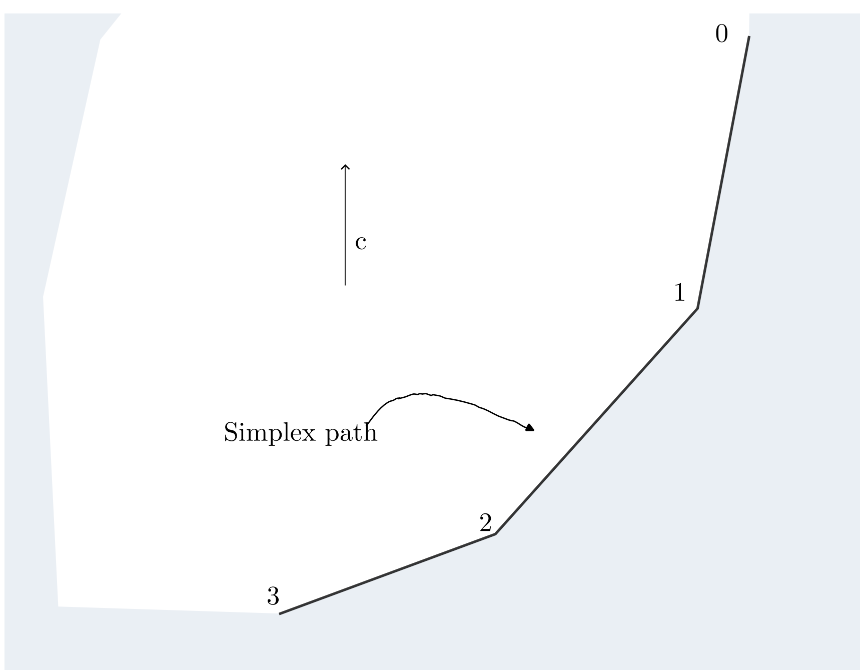
If $s_N \not\geq 0$: find a new BFP. (i) choose one index $q \in \mathcal{N}$ s.t. $s_q < 0$, and (ii) increase x_q along $Ax = b$ until a component becomes zero.

$Ax^+ = Bx_B^+ = b = Bx_B$

$x_B^+ = x_B - B^{-1} A q x_q^+$, here $d = B^{-1} A q$

$x_{B,i} - d_i x_{q,i}^\top = 0$

$x_q^+ = \min_{i, d_i \geq 0} \{ \frac{x_{B,i}}{d_i} \}$



Ex. 13.1 -

Given:

$$\begin{aligned} \min_{x\in\mathbb{R}^2} \quad & -4x_1 - 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 \le 5 \\ & 2x_1 + \frac{1}{2} \le 8 \\ & x_1 \ge 0,\ x_2 \ge 0 \end{aligned}$$

First write it on standard form:

$$\begin{aligned} \min_{x\in\mathbb{R}^4} \quad & -4x_2 - 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 + x_3 = 5 \\ & 2x_1 + \frac{1}{2}x_2 + x_4 = 8 \\ & x \ge 0 \end{aligned}$$

On matrix form:

$$c = \begin{bmatrix} -4 \\ -2 \\ 0 \\ 0 \end{bmatrix},\ A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & \frac{1}{2} & 0 & 1 \end{bmatrix},\ b = \begin{bmatrix} 5 \\ 8 \end{bmatrix}.$$

Choose first basic feasible point $x = \begin{bmatrix} 0 \\ 0 \\ 5 \\ 8 \end{bmatrix}$. We can directly read out these values from the problem:

- $x = \begin{bmatrix} 0 & 0 & 5 & 8 \end{bmatrix}^\top$
- $\mathcal{B}(x) = \{3, 4\}$
- $\mathcal{N}(x) = \{1, 2\}$
- $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- $N = \begin{bmatrix} 1 & 1 \\ 2 & \frac{1}{2} \end{bmatrix}$

Calculate some values (?):

- $x_B = B^{-1}b = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$
- $\lambda = (B^\top)^{-1}c_B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$
- $s_N = c_N - N^\top\lambda = \begin{bmatrix} -4 \\ -2 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 1 & \frac{1}{2} \end{bmatrix}\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -4 \\ -2 \end{bmatrix} \qquad \ngeq 0$

Pick smallest element of s_N: $s_1 = -4 \to q = 1$. Let $q = 1$ enter $\mathcal{B}$ (leaving $\mathcal{N}$). Increase x_1 while $Ax = b,\ x_4$ becomes zero first. Let $p = 4$ leave $\mathcal{B}$ and enter $\mathcal{N}$.

Second iteration

- $\mathcal{B}(x) = \{1, 3\}$
- $\mathcal{N}(x) = \{2, 4\}$
- $B = \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix} \to B^{-1} = \frac{1}{2}\begin{bmatrix} 0 & 1 \\ 2 & -1 \end{bmatrix}$

- $N = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & 1 \end{bmatrix}$

Calculations:

- $x_B = \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} = B^{-1}b = \frac{1}{2} \begin{bmatrix} 0 & 1 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} 5 \\ 8 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}$
- $x_N = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$
- $\lambda = (B^\top)^{-1} c_B = \cdots = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$
- $s_N = c_N - N^\top \lambda = \begin{bmatrix} -2 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -2 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$

Now the smallest value is $q = 2$, the entering index. $p = 3$ is the leaving index.

Third iteration

- $\mathcal{B}(x) = \{1, 2\}$
- $\mathcal{N}(x) = \{3, 4\}$
- $B = \begin{bmatrix} 1 & 1 \\ 2 & \frac{1}{2} \end{bmatrix} \to B^{-1} = \frac{1}{3} \begin{bmatrix} -1 & 2 \\ 4 & -2 \end{bmatrix}$
- $N = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
-

And

- $x_B = B^{-1}b = \cdots = \begin{bmatrix} 11/3 \\ 4/3 \end{bmatrix}$
- $\lambda = (B^{-1})^\top c_B = \cdots = \begin{bmatrix} -4/3 \\ -4/3 \end{bmatrix}$
- $s_N = c_N - N^\top \lambda = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -4/3 \\ -4/3 \end{bmatrix} = \begin{bmatrix} 9/3 \\ 4/3 \end{bmatrix} \geq 0$

Since s_N is greater than 0, we have an optimal solution, $x^* = \begin{bmatrix} 11/3 \\ 4/3 \\ 0 \\ 0 \end{bmatrix}$

4 Other details

In each step of the Simplex algorithm, two linear systems must be solved. $B^\top \lambda = c_B$ and $Bd = A_q$. The last one is to find the direction to check when increasing x_q. We also have $Bx_B = b$. Since x_B is not needed in the iterations, we don't need to solve this before the final iteration.

B is a general, non-singular matrix. It is guaranteed to have a solution. LU factorization is the correct method to solve both systems.

In each step of the Simplex method, one column of B is replaced. There are some tricks to save time with maintaining solutions of LU factorizations.

Finding a starting point in the Simplex method is as hard as solving the problem itself. Normally, simplex algorithms have two phases, one for finding the first feasible point, and the other to actually solve it.
This is done by making another LP with a trivial BFP, where the solution is a BFP for the original problem.

$$\min e^\top z \text{ subject to } Ax + Ez = b,\ (x, z) > 0$$

$$e = (1, 1, , \ldots, 1)^\top, \quad \text{E diagonal matrix with } \begin{cases} E_{jj} = 1 \text{ if } b_j \geq 0 \\ E_{jj} = -1 \text{ if } b_j < 0 \end{cases}$$

Complexity - the simplex method is typically very efficient (2m to 3m iterations). But the worst case is exponential complexity, if all vertices must be visited.
Interior point methods have better worst case complexity, but simplex still performs better usually.