Linear Programming: The Simplex Method

Saurav Samantaray

Department of Mathematics

Indian Institute of Technology Madras

April 22, 2024

Linear programming and simplex method

- Today, linear programming and the simplex method continue to hold sway as the most widely used of all optimisation tools.
- The technique is to formulate linear models and solve them with simplex-based software.
- Often, the situations they model are actually non-linear.
- But linear programming is appealing,
  - advanced state of the software,
  - guaranteed convergence to a global minimum,
  - uncertainty in the model makes a linear model more appropriate than an overly complex non-linear model.

Non-linear Programming Might be the Future !!!

- Non-linear programming may replace linear programming as the method of choice in some applications as the non-linear software improves.
- A new class of methods known as interior-point methods has proved to be faster for some linear programming problems.
- But the continued importance of the simplex method is assured for the foreseeable future.

LINEAR PROGRAMMING

Linear programs have:

- linear objective function;
- linear constraints;
- which may include both equalities and inequalities.
- The feasible set is a polytope, a convex, connected set with flat, polygonal faces.
- Owing to the linearity of the objective function its contours are planar.
- Figure below depicts a linear program in two-dimensional space, in which the contours of the objective function are indicated by dotted lines.

LINEAR PROGRAMMING

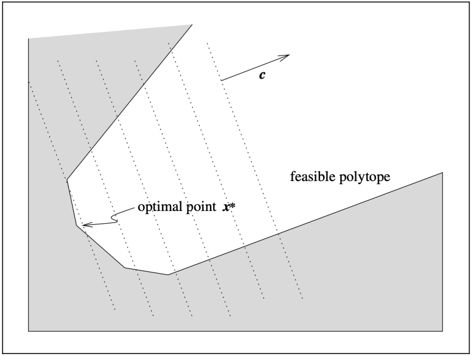

Figure: A linear program in two dimensions with solution at x^*

- The solution in this case is unique-a single vertex.

Solution to Linear Programs

- The solution to a linear program could be non-unique as well.
- It could be an entire edge instead of just one vertex.
- In higher dimensions, the set of optimal points can be a single vertex, an edge or face, or even the entire feasible set.
- The problem has no solution if the feasible set is empty (infeasible case);
- or if the objective function is unbounded below on the feasible region (the unbounded case)

Standard Form of Linear Programs

Linear programs are usually stated and analysed in the following standard form:

Linear Program

$$\min\ c^T x, \qquad \text{subject to } Ax = b,\ x \geq 0, \tag{1}$$

where

- c and x are vectors in $\mathbb{R}^n$,
- b is a vector in $\mathbb{R}^m$ and A is an $m \times n$ matrix

Transforming to Standard Form

- Consider the form:

$$\min\ c^T x, \qquad \text{subject to } Ax \leq b \tag{2}$$

  without any bounds on x.
- By introducing a vector of slack variables z the inequality constraints can be converted to equalities.
-

$$\min\ c^T x, \qquad \text{subject to } Ax + z = b,\ z \geq 0, \tag{3}$$

- Still not all variables (x) are constrained to be non-negative as in the standard form.

Transforming to Standard Form

- It is dealt by splitting x into non-negative and non-positive parts.

$$x = x^{+} - x^{-}, \; x^{+} = \max(x, 0) \geq 0 \text{ and } x^{-} = \max(-x, 0)$$

- Now the above considered problem can be written as:

$$\min \begin{bmatrix} c \\ -c \\ 0 \end{bmatrix} \begin{bmatrix} x^{+} \\ x^{-} \\ z \end{bmatrix} \text{ s.t. } \begin{bmatrix} A & -A & I \end{bmatrix} \begin{bmatrix} x^{+} \\ x^{-} \\ z \end{bmatrix} = b, \begin{bmatrix} x^{+} \\ x^{-} \\ z \end{bmatrix} \geq 0,$$

- The above system is now in the standard form.

Transforming to Standard Form

- Inequality constraints of the form $x \leq u$ and $Ax \geq b$ can be converted to equality constraints by adding or subtracting slack variables.
- $$x \leq u \Leftrightarrow \ x + w = u, \ \ w \geq 0,$$
  $$Ax \geq b \Leftrightarrow \ Ax - y = b \ y \geq 0$$
- We subtract the variables from the left hand side, they are known as surplus variables.
- We add the variables to the left hand side, they are known as deficit variables.
- By simply negating c "maximise" objective $\max c^T x$ can be converted to "minimise" form $\min \ -c^T x$.

LINEAR PROGRAMMING

- The linear program is said to be infeasible if the feasible set is empty.
- The problem is considered to be unbounded if the objective function is unbounded below on the feasible region.
- That is, there is a sequence of points x_k in the feasible region such that $c^T x_K \downarrow -\infty$.
- Unbounded problems have no solution.
- For the standard formulation , we will assume throughout that $m < n$.
- Otherwise, the system $Ax = b$ contains redundant rows, or is infeasible, or defines a unique point.
- When $m \geq n$, factorisations such as the QR or LU factorisation can be used to transform the system $Ax = b$ to one with a coefficient matrix of full row rank.

OPTIMALITY CONDITIONS

- Optimality conditions can be derived from the first-order conditions, the Karush–Kuhn–Tucker (KKT) conditions.
- Convexity of the problem ensures that these conditions are sufficient for a global minimum.
- Do not need to refer to the second-order conditions, which are not informative because the Hessian of the Lagrangian is zero.
- The LICQ condition is not required to be enforced here as the KKT results continue to hold for dependent constraints provided they are linear, as is the case here.

OPTIMALITY CONDITIONS

- The Lagrange multipliers for linear problems are partitioned into two vectors λ and s.
- Where $\lambda \in \mathbb{R}^m$ is the multiplier vector for the equality constraints $Ax = b$.
- While $s \in \mathbb{R}^n$ is the multiplier vector for the bound constraints $x \geq 0$.
- Using the definition we can write the Lagrangian function:

$$\mathscr{L}(x, \lambda, s) = c^T x - \lambda^T (Ax - b) - s^T x. \tag{4}$$

OPTIMALITY CONDITIONS

- The first-order necessary conditions for x^* to be a solution of the linear programming problem (1) are, if there exists λ and s such that:

$$A^T\lambda + s = c, \tag{5}$$
$$Ax = b, \tag{6}$$
$$x \geq 0, \tag{7}$$
$$s \geq 0, \tag{8}$$
$$x_i s_i = 0,\ \ i = 1, 2, \ldots, n. \tag{9}$$

- The last condition, which is the complementarity condition, which says that at-least either one of x_i or s_i is zero, can be written alternatively as

$$x^T s = 0$$

Sufficiency of Optimality Conditions

- Let (x^*, λ^*, s^*) denote a vector triple that satisfy the KKT conditions, then

$$c^T x^* = (A^T \lambda^* + s^*)^T x^* = (Ax^*)^T \lambda^* = b^T \lambda^* \qquad (10)$$

- The first order KKT conditions for optimality for LPP is indeed sufficient.
- Let $\bar{x}$ be any other feasible point, so that $A\bar{x} = b$ and $\bar{x} \geq 0$.

$$\begin{aligned} c^T \bar{x} &= (A^T \lambda^* + s^*)^T \bar{x} \\ &= b^T \lambda^* + \bar{x}^T s^* \\ &\geq b^T \lambda^* = c^T x^* \end{aligned}$$

OPTIMALITY CONDITIONS

- The above inequality tells that no other feasible point can have a lower objective value than $c^T x^*$.
- To say more the feasible point $\bar{x}$ is optimal if and only if

$$\bar{x}^T s^* = 0$$

  otherwise the inequality is strict.
- When $s_i^* > 0$ then we must have $\bar{x}_i = 0$ for all solutions $\bar{x}$ of the LPP.

Duality

Dual problem:

- constructed from the primal problem (objective and constraints).
- related to it in certain ways,
- possibly easier to solve computationally,
- possibly gives lower bound on the optimal primal objective.
- Applies to convex problems.

Dual Problem

The Primal Problem:

- Consider only inequality constraints.
- Let the objective f and the constraints c_i are all convex.

$$\min_{x \in \mathbb{R}^n} f(x), \text{ s.t. } c_i(x) \geq 0, \ i = 1, 2, \ldots, m. \tag{11}$$

- with $c(x) = (c_1(x), \ldots, c_m(x))^T$,
- Lagrangian $\mathscr{L}(x, \lambda) = f(x) - \lambda^T c(x)$.
- Note that $\mathscr{L}(., \lambda)$ is convex for any $\lambda \geq 0$.

Dual Problem

The Dual Problem:

- The dual objective function $q : \mathbb{R}^m \to \mathbb{R}$ is given as:

$$q(\lambda) := \inf_x \mathscr{L}(x, \lambda). \tag{12}$$

$$\max_{\lambda \in \mathbb{R}^m} \; q(\lambda), \text{ s.t. } \lambda \geq 0. \tag{13}$$

- With domain $\mathscr{D} = \{\lambda : q(\lambda) > -\infty\}$

Primal to Dual

Consider the problem

$$\min_{(x_1,x_2)} 0.5(x_1^2 + x_2^2) \quad \text{s.t. } x_1 - 1 \geq 0.$$

The Lagrangian is

$$\mathscr{L}(x_1, x_2, \lambda_1) = 0.5(x_1^2 + x_2^2) - \lambda_1(x_1 - 1).$$

- If λ_1 is fixed then $\mathscr{L}$ is a convex function of $(x_1, x_2)^T$.
- The infimum with respect to $(x_1, x_2)^T$ is achieved when the partial derivatives with respect to x_1 and x_2 are zero, i.e.

$$x_1 - \lambda_1 = 0, \quad x_2 = 0$$

Primal to Dual

- By substituting the infimum values into the Lagrangian $\mathscr{L}(x_1, x_2, \lambda_1)$ we obtain the dual objective function as:

$$q(\lambda_1) = 0.5(\lambda_1^2 + 0) - \lambda_1(\lambda_1 - 1) = -0.5\lambda_1^2 + \lambda_1.$$

- The dual problem is

$$\max_{\lambda_1 \geq 0} -0.5\lambda_1^2 + \lambda_1,$$

- Which clearly has the solution $\lambda_1 = 1$.

Dual of LPP

The Dual Problem

Given c, b, and A of the primal LP problem in the standard form we introduce the following problem

$$\max b^T\lambda, \quad \text{subject to } A^T\lambda \leq c. \tag{14}$$

The above problem is called the Dual problem. It can be restated as a standard LPP problem as:

$$\max b^T\lambda, \quad \text{subject to } A^T\lambda + s = c, \quad s \geq 0, \tag{15}$$

by introducing a vector of dual slack variables s. The variables (λ, s) in this problem are jointly referred to collectively as dual variables.

Dual of LPP

- The primal and dual problems present two different viewpoints on the same data.
- The link between both can be seen via the KKT conditions.
- Let us first recast it in the form:

$$\min -b^T\lambda \quad \text{subject to } c - A^T\lambda \geq 0, \tag{16}$$

- Use $x \in \mathbb{R}^n$ to denote the Lagrange multipliers for the constraints $A^T\lambda \leq c$, the Lagrangian function is

$$\bar{\mathscr{L}}(\lambda, x) = -b^T\lambda - x^T(c - A^T\lambda).$$

Dual of LPP

- Let us write down the KKT conditions for the Dual problem:

$$Ax = b, \tag{17}$$
$$A^T\lambda \leq c, \tag{18}$$
$$x \geq 0, \tag{19}$$
$$x_i(c - A^T\lambda)_i = 0, \; i = 1, 2, \ldots n. \tag{20}$$

- Define $s = c - A^T\lambda$
- Note that the KKT conditions for both the primal and the dual are identical.

Dual of LPP

- The optimal Lagrange multipliers λ in the primal problem are the optimal variables in the dual problem.
- While the optimal Lagrange multipliers x in the dual problem are the optimal variables in the primal problem.
- Given x^* and λ^* satisfying the satisfying the KKT conditions for the dual ($(x, \lambda, s) = (x^*, \lambda^*, c - A^T\lambda^*)$ satisfy the same for the primal) .
- For any other feasible point $\bar{\lambda}$ (with $A^T\lambda \leq c$) we have:

$$
\begin{aligned}
b^T\bar{\lambda} = (x^*)^T A^T \bar{\lambda} &= (x^*)^T(A^T\bar{\lambda} - c) + c^T x^* \\
&\leq c^T x^* \qquad (A^T\bar{\lambda} - c \leq 0 \text{ and } x^* \geq 0) \\
&= b^T\lambda^*
\end{aligned}
$$

Duality Gap

The primal–dual relationship is symmetric (by taking the dual of the dual problem, we recover the primal problem).

- x be a feasible vector for the primal ($Ax = b$ and $x \geq 0$).
- (λ, s) be a feasible vector for the dual ($A^T\lambda + s = c, s \geq 0$)

$$0 \leq x^T s = x^T(c - A^T\lambda) = c^T x - b^T\lambda^T$$
$$\Leftrightarrow c^T x \geq b^T\lambda.$$

- Duality Gap: $c^T x - b^T\lambda$
- Thus, the dual objective function $b^T\lambda$ is a lower bound on the primal objective function $c^T x$.

Duality of LPP

Theorem (Strong duality)

1. If either problem (primal or dual) has a (finite) solution, then so does the other, and the objective values are equal.
2. If either problem (primal or dual) is unbounded, then the other problem is infeasible.

Duality is important in the theory of LP (and convex opt. in general) and in primal-dual algorithms; also, the dual may be easier to solve than the primal.

GEOMETRY OF THE FEASIBLE SET

- Assume that the matrix A has full row rank.
- In practice, a pre-processing is applied to remove some redundancies from the given constraints and eliminate some of the variables.
- Reformulation by adding slack, surplus, and artificial variables can also bring out the full row rank.

GEOMETRY OF THE FEASIBLE SET

Basic Feasible Point (BFP)

A vector x is a basic feasible point if it is feasible and if there exists a subset $\mathscr{B}$ of the index set 1,2,...,n such that:

- $\mathscr{B}$ contains exactly m indices;
- $i \notin \mathscr{B} \implies x_i = 0$ (that is, the bound $x_i \geq 0$ can be active only if $i \in \mathscr{B}$);
- The $m \times m$ matrix B defined by

$$B = [A_i]_{i \in \mathscr{B}}$$

  is non-singular, where A_i is the i^{th} column of A.

- A set $\mathscr{B}$ satisfying these properties is called a basis for the problem.
- The corresponding matrix B is called the basis matrix.

GEOMETRY OF THE FEASIBLE SET

- The simplex method generates a sequence of iterates x_k
  - that are BFPs;
  - if the LP has BFPs and at least one of them is a basic optimal point (= a BFP which is a minimizer), x_k converges (in a finite number of steps) to such a solution.

Theorem

1. If LPP has a non-empty feasible region, then there is at least one basic feasible point;
2. If LPP has solutions, then at least one such solution is a basic optimal point.
3. If LPP is feasible and bounded, then it has an optimal solution.

VERTICES OF THE FEASIBLE POLYTOPE

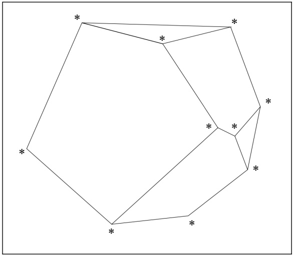

Figure: Vertices of a three-dimensional polytope (indicated by *)

- The feasible set defined by the linear constraints is a polytope.
- The vertices of this polytope are the points that do not lie on a straight line between two other points in the set.
- Geometrically, they are easily recognisable; see Figure

VERTICES OF THE FEASIBLE POLYTOPE

- Algebraically, the vertices are exactly the basic feasible points defined above.
- This important relationship between the algebraic and geometric viewpoints can be a useful aid to understanding how the simplex method works.

Theorem

All basic feasible points for LPP are vertices of the feasible polytope $\{x | Ax = b,\ x \geq 0\}$, and vice versa.

Degeneracy

A basis $\mathscr{B}$ is said to be degenerate if $x_i = 0$ for some $i \in \mathscr{B}$, where x is the basic feasible solution corresponding to $\mathscr{B}$. A linear program is said to be degenerate if it has at least one degenerate basis.

Simplex Method

- There are a number of variants of the simplex method.
- The one considered here is sometimes known as the revised simplex method.
- All iterates of the simplex method are basic feasible points for LPP and therefore vertices of the feasible polytope.
- Most steps consist of a move from one vertex to an adjacent one for which the basis $\mathscr{B}$ differs in exactly one component.
- On most steps (but not all), the value of the primal objective function $c^T x$ is decreased.
- The steps may follow an edge along which the objective function is reduced, and along which we can move infinitely far without ever reaching a vertex.

Simplex Method

- There are at most $\binom{n}{m}$ different sets of basic indices $\mathscr{B}$.
- So a brute-force way to find a solution would be to try them all and check the KKT conditions.
- The simplex algorithm does better than this.
- The major issue at each simplex iteration is to decide which index to remove from the basis $\mathscr{B}$.
- Unless the step is a direction of un-boundedness, a single index must be removed from $\mathscr{B}$ and replaced by another from outside $\mathscr{B}$.

Simplex Method

Definition

The non-basic index set $\mathscr{N}$ is the complement of $\mathscr{B}$, that is:

$$\mathscr{N} = \{1, 2, \dots, n\} \setminus \mathscr{B}$$

- As B was denoted as the basic matrix, whose columns are A_i for $i \in \mathscr{B}$, N is used to denote the non-basic matrix $N = [A_i]_{i \in \mathscr{N}}$
- Partition the n-element vectors x, s, and c according to the index sets $\mathscr{B}$ and $\mathscr{N}$

$$\begin{aligned} x_B &= [x_i]_{i \in \mathscr{B}} & x_N &= [x_i]_{i \in \mathscr{N}} \\ s_B &= [s_i]_{i \in \mathscr{B}} & s_N &= [s_i]_{i \in \mathscr{N}} \\ c_B &= [c_i]_{i \in \mathscr{B}} & c_N &= [c_i]_{i \in \mathscr{N}} \end{aligned}$$

Simplex Method

- From the second KKT condition ($Ax = b$),

$$Ax = Bx_B + Nx_N = b.$$

- The primal variable x for this simplex iterate is defined as

$$x_B = B^{-1}b, \qquad x_N = 0. \tag{21}$$

- We are dealing only with basic feasible points.
- B is non-singular.
- $x_B \geq 0$.
- So this choice of x satisfies two of the KKT conditions:
  1. the equality constraints and;
  2. the non-negativity condition.

Simplex Method

- choose s to satisfy the complementarity condition, by setting $s_B = 0$.
-

$$s + A^T\lambda = c \implies \begin{pmatrix} s_B \\ s_N \end{pmatrix} + \begin{pmatrix} B^T \\ N^T \end{pmatrix} \lambda = \begin{pmatrix} B^T\lambda \\ s_N + N^T\lambda \end{pmatrix} = \begin{pmatrix} c_B \\ c_N \end{pmatrix}$$

$$\implies \begin{cases} \lambda = (B^{-1})^T c_B \\ s_N = c_N - (B^{-1}N)^T c_B. \end{cases} \tag{22}$$

- $s \geq 0$, while s_B satisfies this, $s_N = c_N - (B^{-1}N)^T c_B$ may not.
- if it does, i.e.,$s_N \geq 0$, we have found an optimal (x, λ, s) and we have finished

Simplex Method

Thus we take out one of the indices $q \in \mathscr{N}$ for which $s_q < 0$ (there are usually several) and:

- allow x_q to increase from 0;
- fix all other components of x_N to 0;
- figure out the effect of increasing x_q on the current BFP x_B given that we want it to stay feasible wrt $Ax = b$.
- keep increasing x_q until one of components of x_B (say, that of x_p) is driven to 0.
- p leaves $\mathscr{B}$ to $\mathscr{N}$, q enters $\mathscr{B}$ from $\mathscr{N}$.
- This process of selecting entering and leaving indices, and performing the algebraic operations necessary to keep track of the values of the variables x, λ, and s is known as pivoting.

Simplex Method

- Formally, the pivoting procedure can be written in algebraic terms.
- Both the new iterate x^+ and the current iterate x satisfy $Ax = b$ and $x_N = 0$ and $x_i^+ = 0$ for $i \in \mathcal{N} \setminus \{q\}$.

$$Ax^+ = \begin{pmatrix} B & N \end{pmatrix} \begin{pmatrix} x_B^+ \\ x_N^+ \end{pmatrix} = Bx_B^+ + A_q x_q^+ = Bx_B = Ax$$

$$x_B^+ = x_B - B^{-1} A_q x_q^+ \tag{23}$$

- Geometrically speaking, (23) is usually a move along an edge of the feasible polytope that decreases $c^T x$.

Simplex Method

- Continue to move along this edge until a new vertex is encountered.
- At this vertex,a new constraint $x_p \geq 0$ must have become active,that is,one of the components x_p , $p \in \mathscr{B}$, has decreased to zero.
- Then remove this index p from the basis B and replace it by q.

$$c^T x^+ = c_B^T x_B^+ + c_q x_q^+ = c_B^T x_B - c_B^T B^{-1} A_q x_q^+ + c_q x_q^+ \quad (24)$$

- $\lambda^T = (c_B^T B^{-1})$, since $q \in \mathscr{N}$ we have $A_q^T \lambda = c_q - s_q$.

$$c_B^T B^{-1} A_q x_q^+ = \lambda^T A_q x_q^+ = (c_q - s_q) x_q^+$$

Simplex Method

- Substituting the above in (24) we obtain

$$c^T x^+ = c_B^T x_B - (c_q - s_q)x_q^+ + c_q x_q^+ = c^T x + s_q x_q^+ \quad (25)$$

- q was chosen to have $s_q < 0$.
- It follows that the step (23) produces a decrease in the primal objective function $c^T x$ whenever $x_q^+ > 0$.
- Sometimes it is possible to increase x_q^+ to ∞ without ever encountering a new vertex.
- In other words, the constraint $x_B^+ = x_B - B^{-1}A_q \geq 0$ holds for all positive values of x_q^+.
- In such cases, the linear program is unbounded; the simplex method has identified a ray that lies entirely within the feasible polytope along which the objective $c^T x$ decreases to $-\infty$.

Simplex Method

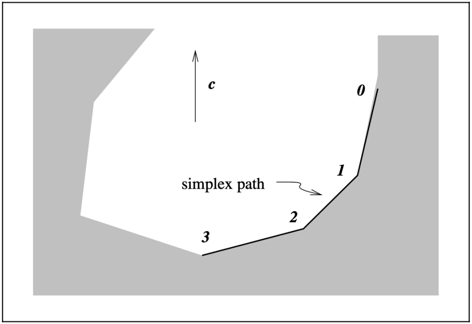

Figure: Simplex iterates for a two-dimensional problem.

- In this example, the optimal vertex x^* is found in three steps.
- If the basis $\mathscr{B}$ is non-degenerate, then its guaranteed that $x_q^+ > 0$, so it is assured to get a strict decrease in the objective function $c^T x$ at this step.

Simplex Method

If the problem is non-degenerate, it can be ensured to get a decrease in $c^T x$ at every step, and can therefore prove the following result concerning termination of the simplex method.

Theorem

Provided that the linear program is non-degenerate and bounded, the simplex method terminates at a basic optimal point.

A SINGLE STEP OF THE METHOD

Procedure 13.1 (One Step of Simplex).

Given $\mathcal{B}, \mathcal{N}, x_B = B^{-1}b \geq 0, x_N = 0$;
Solve $B^T\lambda = c_B$ for λ,
Compute $s_N = c_N - N^T\lambda$; (* pricing *)
if $s_N \geq 0$
  stop; (* optimal point found *)
Select $q \in \mathcal{N}$ with $s_q < 0$ as the entering index;
Solve $Bd = A_q$ for d;
if $d \leq 0$
  stop; (* problem is unbounded *)
Calculate $x_q^+ = \min_{i \mid d_i > 0} (x_B)_i / d_i$, and use p to denote the minimizing i;
Update $x_B^+ = x_B - dx_q^+, x_N^+ = (0, \ldots, 0, x_q^+, 0, \ldots, 0)^T$;
Change $\mathcal{B}$ by adding q and removing the basic variable corresponding to column p of B.

Example

- Consider the problem

$$\begin{aligned} \min \ -3x_1 - 2x_2 \quad & \text{subject to} \\ x_1 + x_2 + x_3 &= 5, \\ 2x_1 + (1/2)x_2 + x_4 &= 8, \\ x &\geq 0. \end{aligned}$$

- $$A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & 1/2 & 0 & 1 \end{bmatrix}$$
- $c = \begin{bmatrix} -3 & -2 & 0 & 0 \end{bmatrix}^T \qquad b = \begin{bmatrix} 5 & 8 \end{bmatrix}^T$
- The constraints that we have are:

$$Ax = b \quad \text{and} \quad x \geq 0$$

Example

- Lets start with the basis $\mathscr{B} = \{3, 4\}$ and $\mathscr{N} = \{1, 2\}$
- $$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
- we have
$$x_B = B^{-1}b \implies \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$$
- $c_B = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ $\qquad B^T\lambda = c_B$
$$\lambda = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$$
- $$N = \begin{bmatrix} 1 & 1 \\ 2 & 1/2 \end{bmatrix}$$

Example

- $c_N = \begin{bmatrix} -3 & -2 \end{bmatrix}^T$
- $$s_N = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = c_N - N^T \lambda = \begin{bmatrix} -3 & -2 \end{bmatrix}^T$$
- Value of the objective function is $c^T x = 0$.
- Both elements of s_N are negative, lets choose $q = 1$.
- Solve $Bd = A_q$ ($A_q = [1, 2]^T$) for d to get
$$d = [1, 2]^T$$
- $d > 0$
- $$x_q^+ = \min_{i | d_i > 0} \frac{(x_B)_i}{d_i} = \min \left\{ \frac{5}{1}, \frac{8}{2} \right\} = 4$$

Example

- $p = 2$, which corresponds to x_4.
- Therefore, 4 moves out of $\mathscr{B}$ and goes into $\mathscr{N}$ and 1 enters into $\mathscr{B}$ and exits $\mathscr{N}$.
- New $\mathscr{B} = \{3, 1\}$ and $\mathscr{N} = \{4, 2\}$.
- Second iteration

$$B = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} \quad B^{-1} = \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & \frac{1}{2} \end{bmatrix} \quad x_B = \begin{bmatrix} x_3 \\ x_1 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

-

$$c_B = [0, -3]^T \quad \lambda = (B^T)^{-1} c_B = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 \\ -3 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{3}{2} \end{bmatrix}$$

Example

- $$N = \begin{bmatrix} 0 & 1 \\ 1 & \frac{1}{2} \end{bmatrix} \quad c_N = [0, -2]^T$$
- $$s_N = \begin{bmatrix} s_4 \\ s_2 \end{bmatrix} = c_N - N^T\lambda = \begin{bmatrix} \frac{3}{2} \\ -\frac{5}{4} \end{bmatrix}$$
- The objective value $c^T x = -12$.
- s_N still has one negative component, corresponding to $q = 2$.
- Solve $Bd = A_q$ ($A_q = [1, 1/2]^T$) for d to get
$$d = [3/4, 1/4]^T$$
- No unboundedness.

Example

- $$x_q^+ = \min_{i|d_i>0} \frac{(x_B)_i}{d_i} = \min\left\{\frac{4}{3}, 16\right\} = \frac{4}{3}$$
- $p = 1$ corresponds to the index 3, which will leave the basis $\mathscr{B}$.
- Update the index sets to $\mathscr{B} = \{2, 1\}$ and $\mathscr{N} = \{4, 3\}$ and continue.
- At the start of the third iteration, we have

$$x_B = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} 4/3 \\ 11/3 \end{bmatrix}, \quad \lambda = \begin{bmatrix} -5/3 \\ -2/3 \end{bmatrix}, \quad s_N = \begin{bmatrix} s_4 \\ s_3 \end{bmatrix} = \begin{bmatrix} 7/3 \\ 5/3 \end{bmatrix}$$

- With an objective value of $c^T x = -41/3$, $s_N \geq 0$, the optimality test is satisfied, and we terminate.

Important Aspects for Implementation

There are three important aspects of implementation those need to be taken care of:

- Linear algebra issues—maintaining an LU factorisation of B that can be used to solve for λ and d.
- Selection of the entering index q from among the negative components of s_N. (In general, there are many such components.)
- Handling of degenerate bases and degenerate steps, in which it is not possible to choose a positive value of x_q^+ without violating feasibility.

LINEAR ALGEBRA IN THE SIMPLEX METHOD

- To solve two linear systems involving the matrix B at each step:

$$B^T\lambda = c_B, \qquad Bd = A_q. \tag{26}$$

- Do not calculate the inverse basis matrix B.
- Instead, calculate or maintain some factorisation of B, usually an LU factorisation.
- Use triangular substitutions with the factors to recover λ and d.
- The basis matrix B changes by just a single column between iterations.
- It is less expensive to update the factorisation than to calculate it afresh at each iteration.

PRICING AND SELECTION OF THE ENTERING INDEX

- There are usually many negative components of s_N at each step.
- How to choose one of these to become the index that enters the basis?
- Ideally, want to choose the sequence of entering indices q that gets to the solution x^* in the fewest possible steps.
- But rarely a global perspective is available to implement this strategy.
- Instead, more shortsighted but practical strategies that obtain a significant decrease in $c^T x$ on just the present iteration are employed.
- There is usually a tradeoff between the effort spent on finding a good entering index and the amount of decrease in $c^T x$ resulting from this choice.

STARTING THE SIMPLEX METHOD

- The simplex method requires a basic feasible starting point x and a corresponding initial basis.
- $\mathscr{B} \subset \{1, 2, \ldots, n\}$ with $|\mathscr{B}| = m$;
- the basis matrix B is non-singular and $x_B = B^{-1}b \geq 0$ and $x_N = 0$.
- The problem of finding this initial point and basis may itself be nontrivial.
- More often than not its difficulty is equivalent to that of actually solving a linear program.
- A two-phase approach is commonly used to deal with this difficulty in practical implementations.

STARTING THE SIMPLEX METHOD

- In Phase I set up an auxiliary linear program based on the data of the parent LP, and solve it with the simplex method.
- The Phase I problem is so designed that an initial basis and initial basic feasible point is trivial to find.
- It's solution gives a basic feasible initial point for the second phase.
- In Phase II, a second linear program similar to the original LP is solved, with the Phase-I solution as a starting point.
- The solution of the original LP can be extracted easily from the solution of the Phase-II problem.

Phase I

- Introduce artificial variables z into LP and redefine the objective function to be the sum of these artificial variables, as follows:

$$\min e^T z, \quad \text{subject to } Ax + Ez = b, \ (x, z) \geq 0, \tag{27}$$

- where $z \in \mathbb{R}^m$, $e = (1, 1, \ldots, 1)^T$, and E is a diagonal matrix whose diagonal elements are:

$$E_{jj} = +1 \text{ if } b_j \geq 0, \qquad E_{jj} = -1 \text{ if } b_j < 0.$$

- The point (x, z) defined by

$$x = 0, \qquad z_j = |b_j|, \qquad j = 1, 2, \ldots, m, \tag{28}$$

is a basic feasible point for (27).

Phase I

- This point satisfies the constraints in (27).
- The initial basis matrix B is simply the diagonal matrix E, which is non-singular.
- At any feasible point for (27) the artificial variables z represent the amounts by which the constraints $Ax = b$ are violated by the x component.
- The objective function is simply the sum of these violations.
- By minimising this sum we are forcing x to become feasible for the original problem.
- If there exists a vector $(\tilde{x}, \tilde{z})$ that is feasible for (27) such that $e^T \tilde{z} = 0, \implies \tilde{z} = 0$.
- Therefore $A\tilde{x} = b$ and $\tilde{x} \geq 0$, so $\tilde{x}$ is feasible for the original LP.

Phase I

- Conversely, if $\tilde{x}$ is feasible for LP, then the point $(\tilde{x}, 0)$ is feasible for (27) with an objective value of 0.
- Therefore the Phase-I problem (27) has an optimal objective value of zero if and only if the original LP is feasible.
- In Phase I, the simplex method is applied to (27) from the initial point (28).
- The objective function is bounded below by 0.
- So the simplex method will terminate at an optimal point.
- If $e^T z$ is positive at this solution, conclude by the argument above that the original LP is infeasible.
- Otherwise, the method identifies a point $(\tilde{x}, \tilde{z})$ with $e^T \tilde{z} = 0$, which is also a basic feasible point for the Phase-II problem.

Phase II

- The Phase II problem is given as:

$$\min c^T x \quad \text{subject to} \quad Ax + z = b, \quad x \geq 0, \; 0 \geq z \geq 0. \quad (29)$$

- The objective function of (29) is same as the original LP.
- Upper bounds of 0 have been imposed on z from Phase I.
- The original LP is equivalent to (29), because any solution (and indeed any feasible point) must have $z = 0$.
- Need to retain the artificial variables z in Phase II, since some components of z may still be present in the optimal basis from Phase I that are used as the initial basis for (29).
- Though off-course the values z_j of these components must be zero.

EXAMPLE

Consider the inequality-constrained LP:

$$\begin{aligned} \min 3x_1 + x_2 + x_3 \quad & \text{subject to} \\ 2x_1 + x_2 + x_3 &\leq 2, \\ x_1 - x_2 - x_3 &\leq -1, \\ x &\geq 0. \end{aligned}$$

By adding slack variables to both inequality constraints, the following equivalent problem in standard form can be obtained:

$$\begin{aligned} \min 3x_1 + x_2 + x_3 \quad & \text{subject to} \\ 2x_1 + x_2 + x_3 + x_4 &= 2, \\ x_1 - x_2 - x_3 + x_5 &= -1, \\ x &\geq 0. \end{aligned}$$

EXAMPLE

- The vector $x = (0, 0, 0, 2, 0)$ is feasible with respect to the first linear constraint and the lower bound $x \geq 0$.
- It does not satisfy the second constraint.
- Therefore, in forming the Phase-I problem, we add just a single artificial variable z_2 to the second constraint and obtain

$$
\begin{aligned}
\min z_2 \quad & \text{subject to} \\
2x_1 + x_2 + x_3 + x_4 &= 2, \\
x_1 - x_2 - x_3 + x_5 - z_2 &= -1, \\
(x, z_2) &\geq 0.
\end{aligned}
$$

- The vector $(x, z_2) = ((0, 0, 0, 2, 0), 1)$ is feasible with respect to above problem.

EXAMPLE

- It is a basic feasible point, with the corresponding basis matrix B

$$B = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- The variable x_4 plays the role of artificial variable for the first constraint.
- There was no need to add an explicit artificial variable z_1.

DEGENERATE STEPS AND CYCLING

- The simplex method may encounter situations in which for the entering index q, we cannot set x_q^+ any greater than zero without violating the non-negativity condition $x^+ \geq 0$.
- These situations arise when there is i with $(x_B)_i = 0$ and $d_i < 0$.
- Steps of this type are called degenerate steps.
- On such steps, the components of x do not change and, therefore, the objective function $c^T x$ does not decrease.
- The steps may still be useful because they change the basis $\mathscr{B}$ and the updated $\mathscr{B}$ may be closer to the optimal basis.
- In other words, the degenerate step may be laying the groundwork for reductions in $c^T x$ on later steps.

DEGENERATE STEPS AND CYCLING

- Sometimes, a phenomenon known as cycling can occur.
- After a number of successive degenerate steps, we may return to an earlier basis $\mathscr{B}$.
- If we continue to apply the algorithm from this point using the same rules for selecting entering and leaving indices,
- we will repeat the same cycle ad infinitum, never converging.
- Once thought to be a rare phenomenon, in recent times cycling is observed frequently in the large linear programs that arise as relaxations of integer programming problems.
- practical simplex codes usually incorporate a cycling avoidance strategy.

DEGENERATE STEPS AND CYCLING

- Suppose that a degenerate basis is encountered at some simplex iteration.
- The basis is $\hat{\mathscr{B}}$ and the basis matrix is $\hat{B}$.
- A modified linear program is considered in which a small perturbation to the right-hand side of the constraint of the original LP is added as follows:

$$b(\varepsilon) = b + \hat{B} \begin{bmatrix} \varepsilon \\ \varepsilon^2 \\ \cdot \\ \cdot \\ \cdot \\ \varepsilon^m \end{bmatrix}$$

where ε is a very small positive number.

DEGENERATE STEPS AND CYCLING

- This perturbation in b induces a perturbation in the components of the basic solution vector;

$$x_{\hat{B}}(\varepsilon) = x_{\hat{B}} + \begin{bmatrix} \varepsilon \\ \varepsilon^2 \\ . \\ . \\ . \\ \varepsilon^m \end{bmatrix}$$

DEGENERATE STEPS AND CYCLING

Retaining the perturbation for subsequent iterations, we see that subsequent basic solutions have the form:

$$x_B(\varepsilon) = x_B + B^{-1}\hat{B} \begin{bmatrix} \varepsilon \\ \varepsilon^2 \\ \cdot \\ \cdot \\ \cdot \\ \varepsilon^m \end{bmatrix} = x_B + \sum_{k=1}^{m} (B^{-1}\hat{B})_k \varepsilon^k,$$

where $(B^{-1}\hat{B})_k$ denote the k^{th} column of $B^{-1}\hat{B}$ and x_B is the basic solution of the unperturbed right-hand side b.

DEGENERATE STEPS AND CYCLING

- For all ε sufficiently small (but positive), $(x_{\hat{B}}(\varepsilon))_i > 0$ for all i.
- Hence, the basis is non-degenerate for the perturbed problem.
- We can perform a step of the simplex method that produces a nonzero (but tiny) decrease in the objective.
- If we retain the perturbation over all subsequent iterations, and provided that the initial choice of ε was small enough, all subsequent bases visited by the algorithm are non-degenerate.
- Therefore, we conclude that, provided the initial choice of ε is sufficiently small to ensure non-degeneracy of all subsequent bases, no basis is visited more than once by the simplex method.

DEGENERATE STEPS AND CYCLING

- The method terminates finitely at a solution of the perturbed problem.
- The perturbation can be removed in a post-processing phase, by resetting $x_B = B^{-1}b$ for the final basis B and the original right-hand side b.
- How to choose ε small enough at the point at which the original degenerate basis $\hat{\mathscr{B}}$ is encountered.
- The lexicographic strategy finesses this issue by not making an explicit choice of ε but rather keeping track of the dependence of each basic variable on each power of ε.
- When it comes to selecting the leaving variable, it chooses the index p that minimises $(x_B(\varepsilon))_i/d_i$ over all variables in the basis, for all sufficiently small ε.

Simplex

- The simplex method is very efficient in practice.
- It typically requires $2m$ to $3m$ iterations.
- But it does have a worst-case complexity that is exponential in n.