Math 373: Mathematical Programming and Optimization I

John C. Bowman and Pouria Ramazi
University of Alberta
Edmonton, Canada

November 8, 2018
## Contents

1 Linear Programming 5
   1.A Linear Programming Problems .................................. 9
      1.A.1 Standard form ................................................. 11
   1.B Reduction to standard form .................................... 12

2 The Geometry of Linear Programming 14
   2.A Polyhedra .......................................................... 14
   2.B Review of convexity for functions $f : \mathbb{R} \rightarrow \mathbb{R}$ .................. 15
   2.C Convexity for functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ...................... 16
   2.D The convex hull .................................................. 16
   2.E Piecewise linear convex objective functions ....................... 18
   2.F Problems involving absolute values ................................ 19
   2.G Extreme points ................................................... 20
   2.H Polyhedra in standard form .................................... 24
   2.I Correspondence of bases and basic solutions: ...................... 28
   2.J Degeneracy ......................................................... 28
   2.K Existence of extreme points .................................... 30
   2.L Optimality of extreme points ................................... 31

3 The Simplex Method 33
   3.A Development of the simplex method .............................. 37
   3.B An iteration of the simplex method .............................. 40
   3.C The simplex method for degenerate problems .................... 41
   3.D Implementation of the simplex method .......................... 42
      3.D.1 Naive implementation ......................................... 42
      3.D.2 Revised simplex method ...................................... 43
      3.D.3 Full tableau implementation .................................. 44
   3.E Finding an initial basic feasible solution ....................... 52
   3.F The two-phase simplex method .................................. 54

Bibliography 57

Index 57
Preface

The figures in this text were drawn with the vector graphics language Asymptote (freely available at http://asymptote.sourceforge.net).
Chapter 1
Linear Programming

Optimization is an important area of applied mathematics that is widely used in many areas of society, including industry, business, science, and economics. Some optimization problems, involve an infinite number of variables (like those that arise in the calculus of variations that you may encounter in a mechanics course). However, in this course, we focus on optimization problems that involve maximizing or minimizing a function of a finite number of variables subject to finitely many inequality (or equality) constraints. In particular, we will examine the case where both the function and the constraints are linear functions of their arguments. This subject is often confusingly called linear programming, even though it deals more with linear algebra and geometry than the technical aspects of computer programming. The term “programming” is used here in the sense of detailed military logistics planning, stemming from the work of George Danzig in the US Air Force and at the RAND Corporation just after the end of World War II. The alternative term linear optimization is perhaps a more suitable name.

To illustrate what linear programming is all about, let us begin with a simple optimization problem:

- A farmer has 25 acres of land on which to grow barley or corn. He earns $600 for each acre of barley and $500 for each acre of corn. Harvesting the barley requires 20 hours of labour per acre, and harvesting the corn requires 10 hours of labour per acre. The farmer can afford 200 hours of labor. How can he maximize his profit?

To help us answer this question, let us define the decision variables that express the choices facing the farmer: how many acres of barley to grow \( B \) and how many acres of corn \( C \) to grow.

We are told that the profit is \( f(B, C) = 600B + 500C \).

There are four constraints in the problem. First, the total number of acres \( B + C \leq 25 \). Second, the total amount of labour \( 20B + 10C \leq 200 \). Finally, since you cannot plant negative acres of grain, we know that \( B \geq 0 \) and \( C \geq 0 \).
Remark: It is helpful to illustrate this linear programming problem graphically. In the following graph, the shaded (green+gray) region in the $B$–$C$ plane depicts the constraints $B + C \leq 25$, $B \geq 0$, $C \geq 0$, which is further refined by the constraint $20B + 10C \leq 200$ to the green triangle. The shaded green triangle is the feasible set, which is the collection of all feasible solutions $(B, C)$ that satisfy all four given constraints.

Remark: Let us now consider the behaviour of the objective function $f(B, C) = 600B + 500C$ as $B$ and $C$ vary over the box $[0, 25] \times [0, 25]$ in the $B$–$C$ plane.
One notices that the function takes on its maximum value over the green region at
the vertex \((0, 20)\) and its minimum value at the vertex \((0, 0)\). The graph illustrates
that the solution to the farmer’s maximization problem occurs at \(C = 20\) and \(B = 0\).
That is, in order to maximize his profit, the farmer should only plant 20 acres of
corn; this will use up all 200 hours of his labour but reward him with a tidy profit of
$10 000.

**Remark:** Notice that the extrema of the objective function occur at vertices of the
feasible set.

**Remark:** Another feature that we notice in this example is the first constraint
\(B + C \leq 25\) was superfluous and can be removed from the problem. In some
applications, the removal of such superfluous constraints can make solving linear
programming problems much more efficient!

**Remark:** In general, an optimization problem to an objective function \(f\) of \(n\) decision
variables \(x = (x_1, \ldots, x_n) \in \mathbb{R}^n\), over the feasible set \(F \subset \mathbb{R}^n\) of solutions satisfying
all of the constraints, can be written symbolically as

\[
\max_{x \in F} f(x)
\]

or

\[
\min_{x \in F} f(x).
\]

- For example, in the problem

\[
\max_{x_1^2 + x_2^2 \leq 1} (x_1^2 + x_2^2),
\]

the feasible set is the unit disk.

**Definition:** The *optimal value* is the desired extreme (maximum or minimum) value
(if it exists) of the objective function.

**Definition:** A feasible solution \(x^*\) that extremizes the objective function is called an
*optimal feasible solution*, or simply an *optimal solution*.

**Definition:** The *optimal set* is the set of optimal solutions; that is, the set of feasible
solutions at which the objective function \(f\) takes on its optimal value (if it exists).

**Remark:** In the above problem, the optimal value is 1 and the set of optimal solutions
is all points on the unit circle.
CHAPTER 1. LINEAR PROGRAMMING

Remark: In determining optimal solutions, one considers only points $x$ that are feasible.

- For the problem
  \[
  \max_{x_1^2 + x_2^2 \leq 1, x_2 \geq 0} (x_1^2 + x_2^2),
  \]
  the optimal set is the upper half circle.

- For the problem
  \[
  \max_{x \geq -1, x \leq 1} (x^2 + 2),
  \]
  the feasible set is $[-1, 1]$, the optimal value is 3, and $x = -1$ and $x = 1$ are the optimal solutions.

- For the problem
  \[
  \min_{x \geq -1, x \leq 3, y \leq 2} (x^2 + y^2),
  \]
  the feasible set $F = \{(x, y) : x \in [-1, 3], y \in (-\infty, 2]\}$, the optimal value is 0 and the optimal solution is $(0, 0)$.

- For the problem
  \[
  \max_{x > 0} \frac{1}{x},
  \]
  the feasible set is $(0, \infty)$, we say that the optimal value is $\infty$, but there are no optimal solutions (the optimal set is empty).

- For the problem
  \[
  \min_{x > 0} \frac{1}{x},
  \]
  the feasible set is again $(0, \infty)$, we say that the optimal value is 0, but there are still no optimal solutions.

Remark: There is no need to study maximization and minimization problems separately, because maximizing $f$ is equivalent to minimizing $-f$.

Remark: In typical real-life problems, determining the optimal set is often even more important than knowing the optimal value. For example, if you are asked to solve an optimization problem to maximize profit, it doesn’t help much to know the optimal value if you don’t know how to achieve it!
Remark: Solving an optimization problem means finding the optimal value and all optimal solutions.

Remark: There are various possible outcomes for an optimization problem:

- The problem is infeasible (no feasible solutions)
- The problem is feasible
  - no optimal value (unbounded)
  - an optimal value
    - finitely many optimal solutions
    - infinitely many optimal solutions

Problem 1.1: Suppose you are asked to produce two kinds of meals: economy and deluxe. The economy meal sells for $3/kg and the deluxe version sells for $4/kg. The ingredients are rice and lamb. The economy version should contain at most 10% lamb, while the deluxe version should contain at least 50% lamb. The cost of rice is $1/kg, whereas the lamb costs $2/kg. You have available 300 kg of rice and 100 kg of lamb. Formulate a linear programming problem that investigates how much rice and lamb should you put into each dish to maximize your profit, assuming that all of the prepared dishes sell.

Let the decision variables $r_e$ and $r_d$, represent the weight of rice in the economy and deluxe dishes respectively, and $l_e$ and $l_d$ represent the corresponding amounts of lamb. The objective function is $f(r_e, r_d, l_e, l_d) = 3(r_e + l_e) + 4(r_d + l_d) - (r_e + r_d) - 2(l_e + l_d)$. One wants to maximize this function subject to the constraints

- $r_e + r_d \leq 300$,
- $l_e + l_d \leq 100$,
- $l_e \leq 0.1(r_e + l_e)$,
- $l_d \geq 0.5(r_d + l_d)$,

and

- $r_e, l_e, r_d, l_d \geq 0$.

1.A Linear Programming Problems

Definition: A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is linear if can be expressed as

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} c_i x_i,$$

for some real numbers $c_i$. 
Remark: Equivalently, a function \( f : \mathbb{R}^n \to \mathbb{R} \) is linear if
\[
f(\alpha \mathbf{x} + \beta \mathbf{y}) = \alpha f(\mathbf{x}) + \beta f(\mathbf{y})
\]
for all vectors \( \mathbf{x} \) and \( \mathbf{y} \) and real numbers \( \alpha \) and \( \beta \).

Definition: A linear constraint on the variables \( x_1, \ldots, x_n \) has one of the following forms, where \( a_1, \ldots, a_n \) and \( b \) are real numbers:

1. linear equalities, such as
\[
\sum_{i=1}^{n} a_i x_i = b;
\]
2. linear inequalities, such as
\[
\sum_{i=1}^{n} a_i x_i \geq b.
\]

Definition: Although for \( n > 1 \), \( \mathbb{R}^n \) is not an ordered field, it is nevertheless possible to introduce a partial ordering on \( \mathbb{R}^n \) component-wise: given two vectors \( \mathbf{x} = (x_1, \ldots, x_n) \) and \( \mathbf{y} = (y_1, \ldots, y_n) \) where \( x_i \leq y_i \) for all \( i \), we say \( \mathbf{x} \leq \mathbf{y} \). Similarly, \( \mathbf{x} \geq \mathbf{y} \) means that \( x_i \geq y_i \) for all \( i \). Corresponding definitions hold for strict inequalities, with \( \leq \) replaced by \( < \) and \( \geq \) replaced by \( > \).

Definition: In a linear programming problem, one seeks to minimize a linear cost function \( \mathbf{c}^\top \mathbf{x} = \sum_{i=1}^{n} c_i x_i \) over all vectors \( \mathbf{x} = (x_1, \ldots, x_n) \) in \( \mathbb{R}^n \), subject to a set of linear equality and inequality constraints. The vector \( \mathbf{c} = (c_1, \ldots, c_n) \) is called a cost vector.

We recall that in a linear programming problem of the form
\[
\begin{align*}
\text{minimize} \quad & \mathbf{c}^\top \mathbf{x} \\
\text{subject to} \quad & \mathbf{a}_i^\top \mathbf{x} \geq b_i, \quad i \in \mathcal{M}_1, \\
& \mathbf{a}_i^\top \mathbf{x} \leq b_i, \quad i \in \mathcal{M}_2, \\
& \mathbf{a}_i^\top \mathbf{x} = b_i, \quad i \in \mathcal{M}_3, \\
& x_j \leq 0, \quad j \in \mathcal{N}_1, \\
& x_j \geq 0, \quad j \in \mathcal{N}_2,
\end{align*}
\]
where \( \mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3, \mathcal{N}_1, \) and \( \mathcal{N}_2 \) are finite sets of integers, the variables \( \mathbf{x} = (x_1, \ldots, x_n) \) are called decision variables. When \( \mathbf{x} \) satisfies all of the given constraints, it is called a feasible solution or feasible vector. If there are no restrictions on the sign of \( x_j \), i.e., \( j \) does not belong to \( \mathcal{N}_1 \cup \mathcal{N}_2 \), we say that \( x_j \) is a free or unrestricted variable. The linear function \( f(\mathbf{x}) = \mathbf{c}^\top \mathbf{x} \) is called the objective function or cost function. A feasible solution \( \mathbf{x}^* \) that minimizes the objective function is called an optimal feasible solution or optimal solution. The value \( \mathbf{c}^\top \mathbf{x}^* \) is the optimal cost.
Problem 1.2: By defining an appropriate matrix $A$ and Show that (1.1) can be rewritten in the compact form

$$\begin{align*}
\text{minimize} & \quad c^\top x \\
\text{subject to} & \quad Ax \geq b.
\end{align*}$$

1.A.1 Standard form

A linear programming problem of the form

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

is said to be in *standard form*.

Remark: If we let $A_1, \ldots, A_n$ represent the column vectors of the matrix, the constraint $Ax = b$ can be written as

$$\sum_{i=1}^{n} A_i x_i = b.$$ 

The goal is to synthesize the target vector $b$ from a non-negative amount $x_i$ of each resource $A_i$, such that the cost $\sum_{i=1}^{n} c_i x_i$ is minimized, where $c_i$ is the unit cost of the $i$th resource.

- The diet problem. Suppose that there are $n$ different foods and $m$ different nutrients. The following table indicates the nutritional content of a unit of each food:

<table>
<thead>
<tr>
<th>nutrient 1</th>
<th>food 1</th>
<th>$\cdots$</th>
<th>food $n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_{11}$</td>
<td>$\cdots$</td>
<td>$a_{1n}$</td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td></td>
</tr>
<tr>
<td>nutrient $m$</td>
<td>$a_{m1}$</td>
<td>$\cdots$</td>
<td>$a_{mn}$</td>
</tr>
</tbody>
</table>

Let $A$ be the $m \times n$ matrix with entries $a_{ij}$, so that its $j$th column $A_j$ represents the nutritional content of the $j$th food. Let $b$ be the vector with the requirements of an ideal diet. The standard-form problem aims to mix non-negative quantities $x_i$ of the available foods to achieve an ideal diet at minimal cost. The condition for synthesizing the ideal diet is

$$\sum_{i=1}^{n} A_i x_i = b.$$
If instead, \( b \) represents the minimal requirements of an adequate diet, then the condition will become

\[
\sum_{i=1}^{n} A_i x_i \geq b.
\]

### 1.B Reduction to standard form

Using the following procedures, it is possible to reduce any linear programming problem to standard form:

1. **Elimination of free variables**: Each free variable \( x_j \) can be replaced by the difference \( x_j^+ - x_j^- \) of two additional non-negative variables \( x_j^+ \) and \( x_j^- \) (every real number can be as the difference of two non-negative numbers). This removes \( x_j \) from the problem but adds two new constraints: \( x_j^+ \geq 0 \) and \( x_j^- \geq 0 \).

2. **Elimination of inequality constraints**: Given an inequality constraint of the form

\[
\sum_{j=1}^{n} a_{ij} x_j \leq b_i,
\]

we introduce a non-negative slack variable \( s_i \) to obtain the standard form constraints

\[
\sum_{j=1}^{n} a_{ij} x_j + s_i = b_i,
\]

\[
s_i \geq 0.
\]

Likewise, for each constraint of the form

\[
\sum_{j=1}^{n} a_{ij} x_j \geq b_i,
\]

we introduce a non-negative slack variable \( s_i \), such that

\[
\sum_{j=1}^{n} a_{ij} x_j - s_i = b_i,
\]

\[
s_i \geq 0.
\]
For the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad x_1 + 2x_2 \\
\text{subject to} & \quad x_1 + 3x_2 \leq 2, \\
& \quad 2x_1 + x_2 = 1, \\
& \quad x_1 \geq 0,
\end{align*}
\]

an equivalent problem in standard form is

\[
\begin{align*}
\text{minimize} & \quad x_1 + 2x_2^+ - 2x_2^- \\
\text{subject to} & \quad x_1 + 3x_2^+ - 3x_2^- + x_3 = 2, \\
& \quad 2x_1 + x_2^+ - x_2^- = 1, \\
& \quad x_1 \geq 0, \\
& \quad x_2^+ \geq 0, \\
& \quad x_2^- \geq 0, \\
& \quad x_3 \geq 0,
\end{align*}
\]

For example, for the feasible solution \((x_1, x_2) = (1, -1)\) to the original problem, the standard-form problem has the corresponding feasible solution \((x_1, x_2^+, x_2^-, x_3) = (1, 0, 1, 4)\), which has the same cost. Conversely, the feasible solution \((x_1, x_2^+, x_2^-, x_3) = (2, 1, 4, 9)\) to the standard-form problem corresponds to the solution \((x_1, x_2) = (2, -3)\) to the original problem at the same cost.
Chapter 2

The Geometry of Linear Programming

2.A Polyhedra

**Definition:** A *polyhedron* is a set that can be described in the form

\[ \{ x \in \mathbb{R}^n : Ax \geq b \} , \]

where \( A \) is an \( m \times n \) matrix and \( b \) is a vector in \( \mathbb{R}^m \).

**Remark:** By the same arguments used to reduce a linear programming problem to standard form, a polyhedron can be equivalently defined as a set

\[ \{ x \in \mathbb{R}^n : Ax = b, x \geq 0 \} . \]

**Remark:** A polyhedron can either extend to infinity or can be contained in a finite region.

**Definition:** A set \( S \subset \mathbb{R}^n \) is *bounded* if there exists some constant \( K \) such that \( |x| \leq K \) for every element \( x \in S \). Typically, we will use the Euclidean norm \( |x| = \sqrt{\sum_{i=1}^{n} x_i^2} \).

**Definition:** Let \( a \in \mathbb{R}^n, a \neq 0 \) and \( b \in \mathbb{R} \). Then

1. The set \( \{ x \in \mathbb{R}^n : a^T x = b \} \) is called a *hyperplane*.

2. The set \( \{ x \in \mathbb{R}^n : a^T x \geq b \} \) is called a *half space*.

Notice that the hyperplane is the boundary of the corresponding half space and is perpendicular to \( a \).
**Definition:** The *graph* of a function $f : \mathbb{R}^n \to \mathbb{R}$ is the set of ordered pairs
\[
\{(x, f(x)) : x \in \mathbb{R}^n\}.
\]

**Remark:** The graph of $f$ is just the set of values taken on by the function $F : \mathbb{R}^n \to \mathbb{R}^{n+1}$, $F(x) = (x, f(x))$.

### 2.B Review of convexity for functions $f : \mathbb{R} \to \mathbb{R}$

**Definition:** A function $f : \mathbb{R} \to \mathbb{R}$ is *convex* (sometimes called *concave up*) on an interval $I \subseteq \mathbb{R}$ if its graph lies below or on the secant line segment joining $(p, f(p))$ and $(q, f(q))$.

**Definition:** A function $f$ is *concave* (sometimes called *concave down*) on an interval $I$ if $-f$ is convex on $I$.

**Remark:** Since the equation of the line through $(p, f(p))$ and $(q, f(q))$ is $y = f(p) + \frac{f(q) - f(p)}{q - p}(x - p)$, the definition of convex says
\[
f(x) \leq f(p) + \frac{f(q) - f(p)}{q - p}(x - p) \quad \forall x \in [p, q], \forall p, q \in I. \quad (2.1)
\]
This condition may be rewritten by re-expressing the *linear interpolation* of $f$ between $[p, q]$ on the right-hand side of (2.1):
\[
f(x) \leq \left(\frac{q - x}{q - p}\right)f(p) + \left(\frac{x - p}{q - p}\right)f(q) \quad \forall x \in [p, q], \forall p, q \in I. \quad (2.2)
\]
It is sometimes convenient to introduce the parameter $t = \frac{q - x}{q - p}$, in terms of which we may express $x = q - (q - p)t$ and
\[
\frac{x - p}{q - p} = \frac{(q - p) - (q - p)t}{q - p} = 1 - t.
\]
This allows us to restate (2.2) in *parametric form*:
\[
f(tp + (1 - t)q) \leq tf(p) + (1 - t)f(q) \quad \forall t \in [0, 1], \forall p, q \in I. \quad (2.3)
\]
2.C Convexity for functions $f : \mathbb{R}^n \to \mathbb{R}$

**Definition:** A set $S$ is convex if for every $p, q \in S$ the line segment joining $p$ and $q$ is contained entirely within $S$.

**Definition:** Given points $p, q \in \mathbb{R}^n$, the set $\{tp + (1-t)q : t \in [0, 1]\}$ describes the line segment with endpoints $p$ and $q$.

**Definition:** A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if for every $p, q$ and $t \in [0, 1]$ we have $f(tp + (1-t)q) \leq tf(p) + (1-t)f(q)$.

**Remark:** Equivalently, a function is convex if its graph lies below or on the secant line segment joining $(p, f(p))$ and $(q, f(q))$.

**Definition:** A function $f : \mathbb{R}^n \to \mathbb{R}$ is concave if for every $p, q$ and $t \in [0, 1]$ we have $f(tp + (1-t)q) \geq tf(p) + (1-t)f(q)$.

**Definition:** An affine function is a function of the form $f(x) = a^T x + c$ for some constant $c$.

**Remark:** The only functions that are both convex and concave are affine functions.

**Remark:** A real-valued function $f$ has a local minimum at $p$ if $f(q) \geq f(p)$ for all $q$ in the vicinity of $p$.

**Remark:** A real-valued function $f$ has a global minimum at $p$ if $f(q) \geq f(p)$ for all $q$.

**Theorem 2.1:** For convex functions, local minima are also global minima.

2.D The convex hull

**Definition:** Let $x_1, \ldots, x_k \in \mathbb{R}^n$, $t_1, \ldots, t_k \geq 0$, and $\sum_{i=1}^k t_i = 1$. Then

1. The vector $\sum_{i=1}^k t_i x_i$ is a convex combination of the vectors $x_1, \ldots, x_k$.

2. The convex hull of the vectors $x_1, \ldots, x_k$ is the set of all convex combinations of these vectors. Equivalently, it is the smallest convex polyhedron that contains the points $x_1, \ldots, x_k$. 
Remark: Every half space \( H = \{ x \in \mathbb{R}^n : a^T x \geq b \} \) is convex: let \( x, y \in H \). For every \( t \in [0,1] \) we see that \( a^T (tx + (1-t)y) \geq tb + (1-t)b = b \). That is, \( a^T (tx + (1-t)y) \in H \). Thus \( H \) is convex.

**Theorem 2.2** (Properties of Convex Sets):

a) *The intersection of convex sets is convex.*

b) *Every polyhedron* \( \{ x \in \mathbb{R}^n : Ax \geq b \in \mathbb{R}^m \} \) *is a convex set.*

c) *A convex combination of a finite number of elements of a convex set also belongs to that set.*

d) *The convex hull of a finite number of points is a convex set.*

**Proof:**

a) Consider a collection of convex sets \( S_i \), where \( i \) belongs to some index set \( I \). If \( x, y \in \cap_{i \in I} S_i \) then for each \( i \in I \) and \( t \in [0,1] \), we know from the convexity of \( S_i \) that \( tx + (1-t)y \in S_i \). Therefore \( tx + (1-t)y \in \cap_{i \in I} S_i \). Thus \( \cap_{i \in I} S_i \) is convex.

b) Since a polyhedron is an intersection of convex half spaces, which we have seen are convex, this follows from a).

c) By the definition of convexity, a convex combination of two elements of a convex set \( S \) lies in that set. Suppose that every convex combination of \( k \) elements of \( S \) belongs to \( S \), where \( k \in \{2,3,\ldots\} \). Given a nontrivial convex combination \( \sum_{i=1}^{k+1} t_i x_i \) of \( k+1 \) elements \( x_i \) of \( S \), where \( i = 1,\ldots,k+1 \) and \( t_i \in (0,1) \) such that \( \sum_{i=1}^{k+1} t_i = 1 \), we express

\[
\sum_{i=1}^{k+1} t_i x_i = t_{k+1} x_{k+1} + (1-t_{k+1}) \sum_{i=1}^{k} \frac{t_i}{1-t_{k+1}} x_i.
\]

The summation in the second term belongs to \( S \) since it is a linear combination of \( k \) elements of \( S \): \( \sum_{i=1}^{k} t_i = \sum_{i=1}^{k+1} t_i - t_{k+1} = 1 - t_{k+1} \). Thus \( \sum_{i=1}^{k+1} t_i x_i \) is a linear combination of two elements of \( S \) and is therefore also in \( S \). By induction, a convex combination of any finite number of elements of a convex set also belongs to that set.

d) Let \( y = \sum_{i=1}^{k} \alpha_i x_i \) and \( z = \sum_{i=1}^{k} \beta_i x_i \) be two elements of the convex hull \( S \) of points \( x_1,\ldots,x_k \) and \( \tau \in [0,1] \). Then

\[
ty + (1-t)z = t \sum_{i=1}^{k} \alpha_i x_i + (1-t) \sum_{i=1}^{k} \beta_i x_i = \sum_{i=1}^{k} [t \alpha_i + (1-t) \beta_i] x_i
\]
is a convex combination of \(x_1, \ldots, x_k\), noting that
\[
\sum_{i=1}^{k} [t\alpha_i + (1 - t)\beta_i] = t \sum_{i=1}^{k} \alpha_i + (1 - t) \sum_{i=1}^{k} \beta_i = t + (1 - t) = 1.
\]

It follows from c) that \(S\) is convex.

2.E Piecewise linear convex objective functions

Definition: A function \(f : \mathbb{R}^n \to \mathbb{R}\) is piecewise linear if it is linear on each of a finite number of intervals of \(\mathbb{R}^n\).

- The absolute value function
  \[
  |x| = \begin{cases} 
  x & \text{if } x \geq 0, \\
  -x & \text{if } x < 0.
  \end{cases}
  \]

  is piecewise linear.

Remark: Although we stated earlier that we will focus on objective functions that are linear, one can easily generalize the methods we will develop to functions that are piecewise linear. We will then be able to apply linear programming to problems of the form

\[
\text{minimize } \max_{i=1,\ldots,m} (c_i^T x + d_i)
\]

subject to \(Ax \geq b\).

Since \(\max_{i=1,\ldots,m} (c_i^T x + d_i)\) is equal to the smallest number \(M\) such that \(M \geq c_i^T x + d_i\) for all \(i\), the above optimization problem is equivalent to the linear programming problem

\[
\text{minimize } M
\]

subject to \(M \geq c_i^T x + d_i, \quad i = 1, \ldots, m,\)

\(Ax \geq b\).

Theorem 2.3 (Piecewise convex functions): Let \(f_1, \ldots, f_m : \mathbb{R}^n \to \mathbb{R}\) be convex functions. Then the function \(f(x) = \max_{i=1,\ldots,m} f_i(x)\) is also convex.

Proof: Since
\[
f(tp + (1 - t)q) = \max_{i=1,\ldots,m} f_i(tp + (1 - t)q)
\]
\[
\leq \max_{i=1,\ldots,m} [tf_i(p) + (1 - t)f_i(q)]
\]
\[
\leq t \max_{i=1,\ldots,m} f_i(p) + (1 - t) \max_{i=1,\ldots,m} f_i(q)
\]
\[
= tf(p) + (1 - t)f(q).
\]

for all \(p\) and \(q\), we see that \(f\) is indeed convex.
Since the absolute value function \( f(x) = |x| = \max\{x, -x\} \), we see from Theorem 2.3 that the absolute value function is convex (as well as piecewise linear).

**Remark:** Piecewise linear convex functions can be used to approximate more general functions.

**Remark:** In addition to now being able to handle piecewise linear convex objective functions, we can also handle piecewise linear affine constraints such as

\[
\max_{i=1,...,m} (e_i^T x + d_i) \leq b
\]

by rewriting them as \( e_i^T x + d_i \leq b \).

### 2.F Problems involving absolute values

Consider problems of the form

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} c_i |x_i| \\
\text{subject to} & \quad Ax \geq b,
\end{align*}
\]

where \( \mathbf{x} = (x_1, \ldots, x_n) \) and the cost coefficients \( c_i \) are non-negative. Although the objection function here is easily shown to be piecewise linear and convex (being the sum of piecewise linear convex functions) and therefore can be handled in the manner just described, a more efficient method is available. We can use the fact that \( |x_i| \) is the smallest number \( M_i \) that bounds both \( x_i \) and \( -x_i \) from above to rewrite the linear programming problem as:

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} c_i M_i \\
\text{subject to} & \quad Ax \geq b, \\
& \quad x_i \leq M_i, \quad i = 1, \ldots, n, \\
& \quad -x_i \leq M_i, \quad i = 1, \ldots, n.
\end{align*}
\]

An alternative method is to replace each occurrence of \( |x_i| \) with \( x_i^+ + x_i^- \) and then each of the remaining occurrences of \( x_i \) with \( x_i^+ - x_i^- \), where \( x_i^+ \) and \( x_i^- \) are new non-negative decision variables:

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} c_i (x_i^+ + x_i^-) \\
\text{subject to} & \quad A\mathbf{x}^+ - A\mathbf{x}^- \geq b, \\
& \quad \mathbf{x}^+, \mathbf{x}^- \geq 0,
\end{align*}
\]
where $x^+ = (x_1^+, \ldots, x_n^+)$ and $x^- = (x_1^-, \ldots, x_n^-)$. The equivalence of these two formulations follows from the observation that at an optimal solution, for each $i$ one of $x_i^+$ or $x_i^-$ must be zero (otherwise we could decrease both variables, preserving feasibility and reducing the cost), from which it follows that $|x_i| = x_i^+ + x_i^-.$

2.G Extreme points

Definition: Let $P$ be a polyhedron. A point $x \in P$ is an extreme point of $P$ if we cannot find two vectors $y, z \in P$, both different from $x$, and a scalar $t \in [0, 1]$ such that $x = ty + (1-t)z$. That is, a point $x \in P$ is an extreme point if it cannot be expressed as a convex combination of two other points of $P$.

Definition: Let $P$ be a polyhedron. A vector $x \in P$ is a vertex of $P$ if there exists some $c$ such that $c^T x < c^T y$ for all $y \in P, y \neq x$. That is, $x$ is a vertex of $P$ if $P$ lies entirely on one side of the hyperplane $\{y : c^T y = c^T x\}$, intersecting the hyperplane only at $x$.

Consider a polyhedron $P \in \mathbb{R}^n$ defined in terms of

- $a_i^T x \geq b_i \quad i \in M_1,$
- $a_i^T x \leq b_i \quad i \in M_2,$
- $a_i^T x = b_i \quad i \in M_3.$

Definition: If a vector $x$ satisfies $a_i^T x = b_i$ for some $i$ in $M_1, M_2$ or $M_3$, we say that the corresponding constraint is active or binding at $x$.

- Given the polyhedron $P = \{(x_1, x_2, x_3) : x_1 + x_2 + x_3 = 1, x_1, x_2, x_3 \geq 0\}$ shown in Figure 2.1, which constraints are active at the following points?

  - $A$: $x_1 + x_2 + x_3 = 1, x_2 = 0, x_3 = 0$
  - $B$: $x_1 + x_2 + x_3 = 1, x_1 = 0, x_3 = 0$
  - $C$: $x_1 + x_2 + x_3 = 1, x_1 = 0, x_2 = 0$
  - $D$: $x_1 = 0, x_2 = 0, x_3 = 0$
  - $E$: $x_1 + x_2 + x_3 = 1, x_3 = 0$

**Theorem 2.4 (Active constraints):** Let $x \in \mathbb{R}^n$ and $I = \{i : a_i^T x = b_i\}$ be the set of indices of active constraints at $x$. Then the following are equivalent:

1. There exists $n$ vectors in the set $\{a_i : i \in I\}$ that are linearly independent.
2. The span of the vectors $\{a_i : i \in I\}$ is $\mathbb{R}^n$.
3. The system of equations $a_i^T y = b_i, i \in I$, has the unique solution $y = x$. 
Figure 2.1: Polyhedron $P = \{(x_1, x_2, x_3): x_1 + x_2 + x_3 = 1, x_1, x_2, x_3 \geq 0\}$.

(a) $\iff$ (b): Suppose that the vectors $\{a_i : i \in I\}$ span $\mathbb{R}^n$. Then $n$ of these vectors form a basis of $\mathbb{R}^n$ and therefore must be linearly independent. Conversely, suppose that $n$ of the vectors $a_i, i \in I$, are linearly independent. The subspace spanned by these vectors is $n$ dimensional and must therefore span all of $\mathbb{R}^n$.

(b) $\iff$ (c): Suppose that $x_1$ and $x_2$ are solutions to the system of equations $a_i^T x = b_i, i \in I$. Then $d = x_1 - x_2 \in \mathbb{R}^n$ satisfies $a_i^T d = 0$ for all $i \in I$. That is, $d$ is orthogonal to every $a_i, i \in I$, and therefore cannot be expressed as a linear combination of these vectors. That is, $\{a_i : i \in I\}$ do not span all of $\mathbb{R}^n$. Conversely, if the vectors $a_i, i \in I$, do not span all of $\mathbb{R}^n$, choose a nonzero vector $d$ orthogonal to the subspace they span. For every solution $x$ to the system of equations $a_i^T x = b_i, i \in I$, the vector $x + d$ will then be a distinct solution to the same system of equations.

**Definition**: Consider a polyhedron $P$ defined by linear equality and inequality constraints, and let $x \in \mathbb{R}^n$. Then the vector $x$ is a basic solution if

1. all equality constraints are active;

2. of the constraints that are active at $x$, $n$ of them are linearly independent.
CHAPTER 2. THE GEOMETRY OF LINEAR PROGRAMMING

Definition: If $x$ is a basic solution that satisfies all of the constraints, we say that it is a basic feasible solution.

• Given the polyhedron $P$ shown in Figure 2.1, determine which of the following points are feasible, basic, or basic feasible solutions:
  A: basic feasible
  B: basic feasible
  C: basic feasible
  D: nonbasic infeasible
  E: nonbasic feasible

Q. What would happen if the equality constraint $x_1 + x_2 + x_3 = 1$ were to be replaced by the constraints $x_1 + x_2 + x_3 \leq 1$ and $x_1 + x_2 + x_3 \geq 1$?

A. Then the point $D$ would become a basic solution (but still infeasible).

Remark: Whether a point is a basic solution or not may depend on the way that a polyhedron is represented!

![Figure 2.2: Examples of basic solutions.](image)

• Given the polygon in Figure 2.2, which of the following points are feasible, basic, or basic feasible solutions?
  A: basic
  B: basic
  C: basic feasible
  D: basic feasible
  E: basic feasible
  F: basic feasible
  G: nonbasic infeasible
  H: nonbasic feasible
Remark: So far, we have introduced two geometric definitions (extreme point, vertex) and one algebraic condition (basic feasible solution). The following theorem shows that these three definitions are actually equivalent, so that these concepts can be used interchangeably.

Theorem 2.5 (Vertices): Let $P$ be a nonempty polyhedron and let $x \in P$. Then the following are equivalent:

(a) $x$ is a vertex;

(b) $x$ is an extreme point;

(c) $x$ is a basic feasible solution.

Proof: We represent $P$ as a set of constraints $a^i x \geq b_i$ and $a^i x = b_i$.

Vertex $\Rightarrow$ Extreme point:
Let $x \in P$ be a vertex. Then there exists $c \in \mathbb{R}^n$ such that $c^T x < c^T y$ for every point $y$ in $P$ not equal to $x$. If $y$ and $z$ are any two such points and $t \in [0,1]$, then $c^T x = c^T (tx + (1-t)z) < c^T (ty + (1-t)z)$. Hence $ty + (1-t)z \neq x$. That is, $x$ cannot be expressed as a convex combination of any two other points of $P$ and is therefore an extreme point.

Extreme point $\Rightarrow$ Basic feasible solution:
Let $x \in P$ and $I = \{i : a^i x = b_i\}$. If $x$ were not a basic feasible solution, there would not exist $n$ linearly independent vectors in the set $\{a_i : i \in I\}$, which would span a proper subset of $\mathbb{R}^n$. Choose a vector nonzero $d \in \mathbb{R}^n$ orthogonal to this set, so that $a^i d = 0$ for all $i \in I$. For $\epsilon > 0$ consider the points $y = x + \epsilon d$ and $z = x - \epsilon d$. We see that $a^i y = a^i x = b_i$ for all $i \in I$. Moreover, for $i \notin I$, we know that $a^i x > b_i$. If we choose $\epsilon$ so that $\epsilon |a^i d| < a^i x - b_i$ for all $i \notin I$, then $a^i y > b_i$ and $a^i z > b_i$ for all $i \notin I$, so that $y$ and $z$ also belong to $P$. Since $x = (y + z)/2$, we see then that $x$ could not be an extreme point of $P$.

Basic feasible solution $\Rightarrow$ Vertex:
Let $x$ be a basic feasible solution of $P$, $I = \{i : a^i x = b_i\}$, and $c = \sum_{i \in I} a_i$. Then

$$c^T x = \sum_{i \in I} a^i x = \sum_{i \in I} b_i.$$  

For any $y \in P$ we know that $a^i y \geq b_i$, so that

$$c^T y = \sum_{i \in I} a^i y \geq \sum_{i \in I} b_i = c^T x.$$  \hspace{1cm} (2.4)

That is, $x$ is the optimal solution to the problem of minimizing $c^T y$ over $P$. Equality in (2.4) holds iff $a^i y = b_i$ for all $i \in I$. But this system of equations has a unique
solution since $\mathbf{x}$ is a basic feasible solution, with $n$ linearly independent constraints active at $\mathbf{x}$. That is, equality in (2.4) holds only at the unique minimizer $\mathbf{x}$. This means that $\mathbf{x}$ is a vertex of $P$.

**Corollary 2.5.1:** Given a finite number of linear inequality constraints, there can only be a finite number of basic solutions.

Proof: Let $m$ linear inequality constraints be imposed on a vector $\mathbf{x} \in \mathbb{R}^n$. At a basic solution, $n$ linearly independent constraints must be active, uniquely defining a point in $\mathbb{R}^n$. Different basic solutions correspond to different sets of $n$ linearly independent active constraints chosen from the $m$ imposed constraints. Therefore, an upper bound to the number of basic solutions is $\binom{m}{n}$.

**Remark:** The number of basic feasible solutions, while finite, can be very large: the unit cube $\{\mathbf{x} \in \mathbb{R}^n : 0 \leq x_i \leq 1, i = 1, \ldots, n\}$ has $2n$ constraints but $2^n$ basic feasible solutions.

Q. What about a half space in the plane? How many basic solutions are there?

**Definition:** Two distinct basic solutions to a set of linear constraints in $\mathbb{R}^n$ are **adjacent** if there are $n-1$ linearly independent constraints active at both of them.

• In Figure 2.2, $D$ and $E$ are adjacent to $B$, also $A$ and $C$ are adjacent to $D$.

**Definition:** The line segment joining two adjacent basic feasible solutions is called an **edge**.

## 2.H Polyhedra in standard form

To find the basic solutions for a given polyhedron, it is convenient to express it in the standard form $\{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$, where without loss of generality $A$ is an $m \times n$ matrix with linearly independent rows (which requires $m \leq n$).

**Remark:** Every basic solution must satisfy the $m$ linear independent equality constraints $A\mathbf{x} = \mathbf{b}$; this provides us with $m$ active constraints. To obtain a total of $n$ active constraints, we need to choose $n - m$ of the $n$ decision variables to be zero such that the resulting set of $n$ active constraints is linearly independent. The following theorem gives us some insight as to how this can be accomplished.
Theorem 2.6: Consider the constraints \(Ax = b\) and \(x \geq 0\), where the \(m \times n\) matrix \(A\) has linearly independent rows. A vector \(x \in \mathbb{R}^n\) is a basic solution iff \(Ax = b\) and there exist indices \(B(1), \ldots, B(m)\) such that

i) the columns \(A_{B(1)}, \ldots, A_{B(m)}\) are linearly independent;

ii) if \(i \neq B(1), \ldots, B(m)\), then \(x_i = 0\).

Proof: Suppose that conditions (i) and (ii) hold at some \(x \in \mathbb{R}^n\). Then

\[
\sum_{i=1}^{m} A_{B(i)}x_{B(i)} = \sum_{i=1}^{n} A_ix_i = Ax = b.
\]

The linear independence of the columns \(A_{B(1)}, \ldots, A_{B(m)}\) implies that the solution \(x_{B(1)}, \ldots, x_{B(m)}\) of this system of \(m\) equations is unique. On setting the remaining \(n - m\) decision variables to zero, we obtain a unique solution \(x\) in \(\mathbb{R}^n\) determined by \(n\) linearly independent active constraints. Since all \(m\) equality constraints are active, we know that \(x\) is a basic solution.

Conversely, suppose that \(x\) is a basic solution. Denote the nonzero components of \(x\) by \(x_{B(1)}, \ldots, x_{B(k)}\). Since \(x\) is a basic solution, the system of equations formed by the active constraints \(\sum_{i=1}^{n} A_ix_i = b\) and \(x_i = 0, i \neq B(1), \ldots, B(k)\) has a unique solution. If the columns \(A_{B(1)}, \ldots, A_{B(k)}\) were linearly dependent, we could find scalars \(\lambda_i\) (not all zero) such that \(\sum_{i=1}^{k} A_{B(i)}\lambda_i = 0\), so that \(\sum_{i=1}^{k} A_{B(i)}(x_i + \lambda_i) = b\), contradicting the uniqueness of the solution. Thus the columns \(A_{B(1)}, \ldots, A_{B(k)}\) are linearly independent, which implies that \(k \leq m\). But since \(A\) has \(m\) linearly independent columns, it must have a total of \(m\) linearly independent columns. Hence there exist \(m - k\) additional columns \(A_{B(k+1)}, \ldots, A_{B(m)}\) of \(A\) such that the \(m\) columns \(A_{B(1)}, \ldots, A_{B(m)}\) are linearly independent. Since the decision variables corresponding to the other columns of \(A\) are zero, we see that conditions (i) and (ii) are both satisfied.

Remark: The previous theorem suggests the following procedure for finding all basic solutions.

\begin{itemize}
  \item 1. Choose \(m\) linearly independent columns \(A_{B(1)}, \ldots, A_{B(m)}\).
  \item 2. Let \(x_i = 0\) for all \(i \neq B(1), \ldots, B(m)\).
  \item 3. Solve the system of \(m\) equations \(Ax = b\) for the unknowns \(x_{B(1)}, \ldots, x_{B(m)}\).
\end{itemize}
If a basic solution constructed according to this procedure is **nonnegative**, then it is a basic **feasible** solution. Conversely, since every basic feasible solution is a basic solution, it can be obtained from this procedure.

**Definition:** If \( \mathbf{x} \) is a basic solution, the variables \( x_{B(1)}, \ldots, x_{B(m)} \) are called **basic variables**; the remaining variables are called nonbasic. The columns \( \mathbf{A}_{B(1)}, \ldots, \mathbf{A}_{B(m)} \) are called the **basic columns** and, since they are linearly independent, they form a basis of \( \mathbb{R}^m \). We will sometimes refer to bases being **distinct**: distinct bases involve different sets \( \{B(1), \ldots, B(m)\} \) of basic indices; if two bases involve the same set of indices in a different order, they will be viewed as equivalent.

By arranging the \( m \) basic columns next to each other, we obtain an \( m \times m \) matrix \( \mathbf{B} \) called a **basis matrix**, which is invertible. We also define a vector \( \mathbf{x}_B \) with the values of the basic variables:

\[
\mathbf{B} = [\mathbf{A}_{B(1)}, \ldots, \mathbf{A}_{B(m)}], \quad \mathbf{x}_B = \begin{bmatrix} x_{B(1)} \\ \vdots \\ x_{B(m)} \end{bmatrix}
\]

The basic variables \( x_{B(1)}, \ldots, x_{B(m)} \) are uniquely determined by solving the equation \( \mathbf{Bx}_B = \mathbf{b} \):

\[
\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}.
\]

![Figure 2.3: Determining basic solutions.](image)
Problem 2.1: Find the vertices of the polyhedron $P$ defined by

\[ x_1 - x_2 \leq 2, \]
\[ x_1 + 2x_2 \leq 5, \]
\[ x_1, x_2 \geq 0, \]

illustrated in Figure 2.3.

To put this problem in standard form, we need to add two slack variables $x_3$ and $x_4$, yielding a total of $n = 4$ decision variables satisfying $m = 2$ equality constraints:

\[
A = \begin{bmatrix} 1 & -1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ 5 \end{bmatrix},
\]

Now we consider all possible choices for the basis matrix $B$, corresponding to all possible ways of choosing two linearly independent column vectors from $A$.

For $B(1) = 1$ and $B(2) = 2$:

\[
B = \begin{bmatrix} 1 & -1 \\ 1 & 2 \end{bmatrix}, \quad x_B = \begin{bmatrix} 1 & -1 \\ 1 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 2/3 & 1/3 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}.
\]

This corresponds to the basic solution $x = (3,1,0,0)$. Since all elements of $x$ are non-negative, we see that this is actually a basic feasible solution, corresponding to the vertex (3,1) in Figure 2.3.

For $B(1) = 1$ and $B(2) = 3$:

\[
B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \quad x_B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} -3 \\ 5 \end{bmatrix}.
\]

This corresponds to the basic solution $x = (5,0,-3,0)$. Since the third element of $x$ is negative, we see that this basic solution, corresponding to the dot at (5,0) in Figure 2.3, is infeasible.

For $B(1) = 1$ and $B(2) = 4$:

\[
B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad x_B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}.
\]

This corresponds to the basic solution $x = (2,0,0,3)$. Since all elements of $x$ are non-negative, we see that this is actually a basic feasible solution, corresponding to the vertex (2,0) in Figure 2.3.

For $B(1) = 2$ and $B(2) = 3$:

\[
B = \begin{bmatrix} -1 & 1 \\ 2 & 0 \end{bmatrix}, \quad x_B = \begin{bmatrix} -1 & 1 \\ 2 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 5/2 \\ 9/2 \end{bmatrix}.
\]

This corresponds to the basic solution $x = (0,5/2,9/2,0)$. Since all elements of $x$ are non-negative, we see that this is actually a basic feasible solution, corresponding to the vertex (0,5/2) in Figure 2.3.
For $B(1) = 2$ and $B(2) = 4$:

\[
B = \begin{bmatrix}
-1 & 0 \\
2 & 1
\end{bmatrix}, \quad x_B = \begin{bmatrix}
-1 & 0 \\
2 & 1
\end{bmatrix}^{-1} \begin{bmatrix}
2 \\
5
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
-2 & -1
\end{bmatrix} \begin{bmatrix}
2 \\
5
\end{bmatrix} = \begin{bmatrix}
-2 \\
9
\end{bmatrix}.
\]

This corresponds to the basic solution $x = (0, -2, 0, 9)$. Since the second element of $x$ is negative, we see that this basic solution, corresponding to the dot at $(0, -2)$ in Figure 2.3, is infeasible.

For $B(1) = 3$ and $B(2) = 4$:

\[
B = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}, \quad x_B = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}^{-1} \begin{bmatrix}
2 \\
5
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
2 \\
5
\end{bmatrix} = \begin{bmatrix}
2 \\
5
\end{bmatrix}.
\]

This corresponds to the basic solution $x = (0, 0, 2, 5)$. Since all elements of $x$ are non-negative, we see that this is actually a basic feasible solution, corresponding to the vertex $(0, 0)$ in Figure 2.3.

We thus see that our procedure has found all $\binom{n}{m} = \binom{4}{2} = 6$ basic solutions, and determined that four of them are feasible, precisely as observed in Figure 2.3.

### 2.I Correspondence of bases and basic solutions:

Different basic solutions correspond to different bases. However, two different bases may lead to the same basic solutions, e.g., when $b = 0$.

**Definition:** Two basis matrices are *adjacent* if they share all but one basic column.

Adjacent basic solutions can always be obtained from two adjacent bases. Conversely, if two adjacent bases lead to distinct basic solutions, then the latter are adjacent.

- In Prob., we see that adjacent basis matrices produce adjacent basic solutions.

### 2.J Degeneracy

**Definition:** A basic solution $x \in \mathbb{R}^n$ is *degenerate* if more than $n$ of the constraints are active at $x$.

- Which of the basic solutions illustrated in Figure 2.4 are degenerate and which are feasible?
  
  $A$: degenerate basic feasible
  
  $B$: non-degenerate basic feasible
  
  $C$: degenerate basic feasible
  
  $D$: degenerate basic infeasible
  
  $E$: non-degenerate basic feasible
2.J. DEGENERACY

Figure 2.4: a) a polyhedron in $\mathbb{R}^3$. b) $P$ is a polyhedron in $\mathbb{R}^2$.

Remark: Consider the standard form polyhedron $P = \{ x \in \mathbb{R}^n : Ax = b, x \geq 0 \}$, where $A$ is an $m \times n$ matrix. Let $x$ be a basic solution. The vector $x$ is a degenerate basic solution if **more than** $n - m$ of the components of $x$ are zero.

Remark: Degeneracy is not a purely geometric property. Consider the polyhedron

$$ P = \left\{ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} : x_1 - x_2 = 0, x_1 + x_2 + 2x_3 = 2, x_1, x_2, x_3 \geq 0 \right\}. $$

Which of the following basic solutions are degenerate? Here $n = 3$ and $m = 2$.

$(1 \ 1 \ 0)^T$: nondegenerate (3 active constraints)
$(0 \ 0 \ 1)^T$: degenerate (4 active constraints)

Now consider the polyhedron represented by the same set but when the constraint $x_2 \geq 0$ is relaxed:

$$ P = \left\{ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} : x_1 - x_2 = 0, x_1 + x_2 + 2x_3 = 2, x_1, x_3 \geq 0 \right\}. $$

Now which of the following are degenerate? Here $n = 3$ and $m = 2$.

$(1 \ 1 \ 0)^T$: nondegenerate (3 active constraints)
$(0 \ 0 \ 1)^T$: nondegenerate (3 active constraints)

Remark: For standard form representations, if a basic solution is degenerate, then it is degenerate under every standard form representation.

Remark: We construct basic solutions by choosing $n$ linearly independent constraints to be satisfied with an equality. Degeneracy arises when additional constraints are also satisfied with equality at that basic solution.
2.K Existence of extreme points

**Definition**: A polyhedron $P \subset \mathbb{R}^n$ contains a line if there exists a vector $x \in P$ and a nonzero vector $d \in \mathbb{R}^n$ such that $x + td \in P$ for all scalars $t \in \mathbb{R}$.

**Remark**: A polyhedron in standard form is contained in the positive orthant and hence does not contain a line.

**Remark**: Recall that a half space has no extreme points.

**Theorem 2.7**: Suppose that the polyhedron $P = \{x \in \mathbb{R}^n : a_i^\top x \geq b_i, i = 1, \ldots, m\}$ is nonempty. Then the following are equivalent:

(a) $P$ has at least one extreme point.

(b) $P$ does not contain a line.

(c) The set $\{a_1, \ldots, a_m\}$ contains $n$ linearly independent vectors.

**Proof:**

(b) $\Rightarrow$ (a) Let $x$ be an element of $P$ and let $I = \{i : a_i^\top x = b_i\}$. If $n$ of the vectors $a_i, i \in I$, are linearly independent then $x$ is a basic feasible solution (extreme point). Otherwise, these vectors span a subspace of $\mathbb{R}^n$. Let $d \neq 0$ be a vector orthogonal to this subspace, so that $a_i^\top d = 0$ for every $i \in I$. Consider $y = x + td$ for $t \in \mathbb{R}$. For $i \in I$, we know that $a_i^\top y = a_i^\top x + ta_i^\top d = a_i^\top x = b_i$. That is, all constraints active at $x$ remain active on the line $L = \{x + td : t \in \mathbb{R}\}$. However, since $P$ contains no lines, it must have an intersection with $L$ at some point $x +Td$, where a new constraint, say $a_j^\top(x +Td) = b_j$ for some $j \notin I$, becomes active. Now $j \notin I$ implies that $a_j^\top x \neq b_j$, so $a_j^\top d \neq 0$ and hence is not a linear combination of $a_i, i \in I$. Thus, by moving from $x$ to $x +Td$, we have increased the number of active constraints by at least one. Using induction, we then see on repeating this argument that we will eventually reach a basic solution where $n$ linearly independent constraints are active. Since are movements are always confined within $P$, such a point is a basic feasible solution (extreme point).
(a) ⇒ (c) If $P$ has an extreme point $x$, then by definition there exists $n$ linearly independent active constraints at $x$. The corresponding constraint vectors $a_i$ are therefore linearly independent.

(c) ⇒ (b) Suppose $n$ of the vectors $a_i$ are linearly independent. Let us label them $a_1, \ldots, a_n$. If $P$ were to contain a line, $\{x + td : t \in \mathbb{R}\}$, where $d \neq 0$, then $a_i^\top (x + td) \geq b_i$ for all $i$ and all $t \in \mathbb{R}$. The only way that can happen is if $a_i^\top d = 0$ for all $i$. The linear independence of the vectors $a_i$ then implies that $d = 0$. This contradiction establishes that $P$ cannot contain a line.

**Corollary 2.7.1:** Every nonempty bounded polyhedron and every nonempty polyhedron in standard form has at least one basic feasible solution.

### 2.L Optimality of extreme points

**Theorem 2.8:** Consider the linear programming problem of minimizing $c^\top x$ over a polyhedron $P$. Suppose that $P$ has at least one extreme point and that there exists an optimal solution. Then there exists an optimal solution which is an extreme point of $P$.

*Proof:* Express $P = \{x \in \mathbb{R}^n : Ax \geq b\}$. Let $v$ be the minimal value of the cost $c^\top x$. Consider the polyhedron $Q = \{x \in \mathbb{R}^n : Ax \geq b, c^\top x = v\}$, the (nonempty) set of all optimal solutions. By Theorem 2.7, we see that $P$ contains no lines, and hence $Q \subset P$ also contains no lines and therefore has an extreme point, say $x^\ast$. If $x^\ast$ were not also an extreme point of $P$, then there would exist points $y, z \in P$ distinct from $x^\ast$ and some $t \in (0,1)$ such that $x^\ast = ty + (1-t)z$ and hence $v = c^\top x^\ast = tc^\top y + (1-t)c^\top z \geq tv + (1-t)v = v$ since $v$ is the optimal cost. Hence $c^\top y = v$ and $c^\top z = v$, so that $y$ and $z$ would both belong to $Q$. But this would contradict the fact that $x^\ast$ is an extreme point of $Q$. Thus the optimal solution $x^\ast$ is also an extreme point of $P$.

**Remark:** By keeping track of the costs in Theorem 2.7 as we move towards an extreme point, one can prove a strengthened version of Theorem 2.8, which shows that if the optimal cost is finite, the existence of an optimal solution is guaranteed:

**Theorem 2.9:** Consider the linear programming problem of minimizing $c^\top x$ over a polyhedron $P$. Suppose that $P$ has at least one extreme point. Then either the optimal cost is unbounded from below (equals $-\infty$) or there exists an extreme point which is optimal.

**Remark:** Since every linear programming problem can be transformed into an equivalent standard-form problem, and every nonempty polyhedron in standard form has at least one extreme point (Corollary 2.7.1), we can further simplify these results to:
Corollary 2.9.1: Consider the linear programming problem of minimizing $c^\top x$ over a nonempty polyhedron. Then either the optimal cost is unbounded from below (equals $-\infty$) or there exists an optimal solution.

Remark: The previous result does not hold for nonlinear cost functions. Consider the problem of minimizing $1/x$ subject to $x \geq 1$: the optimal cost is 0 but an optimal solution does not exist!
Chapter 3

The Simplex Method

Consider the standard-form problem

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

Let \( P \) be the corresponding feasible set and \( A \in \mathbb{R}^{m \times n} \) have \( m \) linearly independent rows. We will continue to denote the \( i \)th row of \( A \) as \( a_i \) and the \( j \)th column of \( A \) as \( A_j \).

Definition: Let \( x \) be an element of the polyhedron \( P \). A vector \( d \in \mathbb{R}^n \) is a feasible direction at \( x \) if there exists a positive scalar \( t \) for which \( x + td \in P \).

- Which of the directions in the following figure are feasible?
Remark: Suppose we move from a basic feasible solution \( x \in \mathbb{R}^n \) to \( x + td \) by selecting a nonbasic variable \( x_j \) (which is initially zero) and increasing it by \( t > 0 \). That is, we choose \( d_j = 1 \) and set \( d_i = 0 \) for all other nonbasic indices \( i \neq j \). The remaining (basic) components of \( d \) still need to be determined. Since we are only interested in feasible solutions, we require \( A(x + td) = b \). Thus,

\[
A(x + td) = b \quad \Rightarrow \quad Ad = 0 \Rightarrow \sum_{i=1}^{n} A_i d_i = \sum_{i=1}^{m} A_{B(i)} d_{B(i)} + A_j = Bd_B + A_j = 0,
\]

where

\[
d_B = -B^{-1} A_j
\]

contains the basic components \( \{d_{B(1)}, \ldots, d_{B(m)}\} \) of \( d \).

Definition: The direction vector \( d \) constructed above is called the \( j \)th basic direction.

So far, the equality constraints are guaranteed to be satisfied along this direction. How about the inequality constraints: \( x \geq 0 \)?

Q. Are they respected for nonbasic variables?

A. Yes, since the nonbasic variables are initially zero and only the \( j \)th one is adjusted, by a positive amount \( t \).

Remark: For the basic variables, we distinguish the following cases:

i) \( x \) is a nondegenerate basic feasible solution. Then \( x_B > 0 \), resulting in \( x_B + td_B > 0 \) for sufficiently small \( t \). Thus \( d \) is a feasible direction.

ii) \( x \) is a degenerate basic feasible solution. Then \( d \) is not always a feasible direction. It is possible that a basic variable \( x_{B(i)} \) is zero while \( d_{B(i)} \) is negative.

Let \( n = 5, n - m = 2 \). We can visualize the feasible set by standing on the two dimensional set defined by the constraint \( Ax = b \), where the edges of the feasible set are associated with the non-negativity constraints \( x_i \geq 0 \).

What kind of solution occurs at point \( E \)?
nondegenerate basic feasible solution

What are the basic and nonbasic variables at \( E \)?
nonbasic: \( x_1 \) and \( x_3 \) basic: \( x_2, x_4, x_5 \)

Consider the direction obtained by increasing \( x_1 \) while holding \( x_3 \) zero.
Is this a basic direction? Yes. Is it also feasible? Yes.
What kind of solution occurs at point $F$?
Degenerate basic feasible solution.

Give an example of basic and nonbasic variables at $F$.

nonbasic: $x_3$ and $x_5$

basic: $x_1$, $x_2$, $x_4$

Consider the direction obtained by increasing $x_3$ while keeping the other nonbasic variable $x_5$ at zero level ($d_3 = 1$, $d_5 = 0$).

Is this a basic direction? Yes. Is it also feasible? No.

**Remark**: We need to know the effect on the cost function of moving by an amount $t$ in the $j$th basic direction $d$:

$$c^\top(x + td) - c^\top x = tc^\top d$$

**Definition**: Let $c_B = (c_{B(1)}, \ldots, c_{B(m)})$. The rate of cost change $c^\top d$ along $d$ is

$$c^\top d = c_B^\top d_B + c_j = c_j - c_B^\top B^{-1}A_j.$$

Here $c_j$ represents the cost per unit increase in the variable $x_j$ and the term $c_B^\top B^{-1}A_j$ represents the cost of the compensating change in the basic variables required to satisfy the constraint $Ax = b$.

**Definition**: Let $x$ be a basic solution, $B$ be an associated basis matrix, and $c_B$ be the associated costs of the basic variables. For each $j$, we define the reduced cost $\bar{c}_j$ of the variable $x_j$:

$$\bar{c}_j = c_j - c_B^\top B^{-1}A_j.$$
Problem 3.1: Consider the linear programming problem

\[
\begin{aligned}
\text{minimize} & \quad c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4 \\
\text{subject to} & \quad x_1 + x_2 + x_3 + x_4 = 2 \\
& \quad 2x_1 + x_3 + 3x_3 + 4x_4 = 2 \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{aligned}
\]

Since \( A_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \) and \( A_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \) are linearly independent, so they form a basis of \( \mathbb{R}^2 \), with basis matrix

\[ B = \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix}. \]

Let \( b = (2, 2) \). We now determine the basic variable vector

\[ x_B = B^{-1} b = \frac{1}{2} \begin{bmatrix} 0 & -1 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \]

which corresponds to the nondegenerate basic feasible solution \( x = (1, 1, 0, 0) \). The basic direction corresponding to increasing the nonbasic variable \( x_3 \) has nonbasic components

\[
\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} d_{B(1)} \\ d_{B(2)} \end{bmatrix} = dB = -B^{-1}A_3 = -\frac{1}{2} \begin{bmatrix} 0 & -1 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} -3/2 \\ 1/2 \end{bmatrix}.
\]

Then the reduced cost of \( x_3 \) is

\[
c_3 = c_3 + c_B^T d_B = c_3 - c_B^T B^{-1} A_3 = c_3 + [c_1 \quad c_2] \begin{bmatrix} -3/2 \\ 1/2 \end{bmatrix} = -\frac{3}{2} c_1 + \frac{1}{2} c_2 + c_3.
\]

Remark: Let \( e_i \) denote the \( i \)th unit vector. If we were to adopt the same definition as for the non-basic variables, we would find

\[
c_{B(i)} = c_{B(i)} - c_B^T B^{-1} A_{B(i)} = c_{B(i)} - c_B^T e_i = c_{B(i)} - c_{B(i)} = 0.
\]

That is, the reduced cost of every basic variable is zero! Therefore, we can now calculate the entire vector of reduced costs.

Definition: The reduced cost vector is given by \( \bar{c}^T = c^T - c_B^T B^{-1} A \).

Is there a relationship between the reduced cost and an optimal solution?

**Theorem 3.1:** Consider a basic feasible solution \( x \) associated with a basic matrix \( B \), and let \( \bar{c} \) be the corresponding reduced cost vector. Then

i) if \( \bar{c} \geq 0 \), \( x \) is optimal.

ii) if \( x \) is optimal and non-degenerate, then \( \bar{c} \geq 0 \).
3.A. DEVELOPMENT OF THE SIMPLEX METHOD

Proof:

i) Given \( \bar{c} \geq 0 \), let \( y \) be an arbitrary feasible solution. The difference vector \( d = y - x \) satisfies \( Ad = Ax - Ay = b - b = 0 \). That is,

\[
Bd_B + \sum_{i \in N} A_i d_i = 0,
\]

where \( N \) is the set of nonbasic indices. Thus

\[
d_B = -\sum_{i \in N} B^{-1} A_i d_i.
\]

For every \( i \in N \) we know that the nonbasic variable \( x_i = 0 \) and since \( y_i \geq 0 \), we see that \( d_i \geq 0 \) for all \( i \in N \). The change in the cost by moving from \( x \) to \( y \) is therefore

\[
c^T d = c_B^T d_B + \sum_{i \in N} c_i d_i = \sum_{i \in N} (c_i - c_B^T B^{-1} A_i) d_i = \sum_{i \in N} c_i d_i \geq 0.
\]

Therefore \( x \) is an optimal basic feasible solution.

ii) Suppose that \( x \) is a nondegenerate basic feasible solution and that \( c_j < 0 \) for some \( j \). Since the reduced cost of a basic variable is always zero, \( x_j \) must be a nonbasic variable and \( c_j \) is the rate of cost change along the \( j \)th basic direction. Since \( x \) is nondegenerate, the \( j \)th basic direction is a feasible direction of cost decrease and therefore \( x \) is not optimal.

Remark: If \( x \) is a degenerate optimal basic feasible solution, it is still possible that \( \bar{c}_j < 0 \) for some nonbasic index \( j \).

Definition: A basis matrix \( B \) is optimal if both of the following conditions are satisfied:

1. feasibility: \( x_B \geq 0 \)
2. non-negativity of the reduced costs: \( \bar{c} \geq 0 \).

Remark: If an optimal basis is found, the corresponding basic solution is feasible, satisfies the optimality conditions, and is therefore optimal. On the other hand, in the degenerate case, having an optimal basic solution does not necessarily mean that the reduced costs are non-negative.

3.A Development of the simplex method

For now, assume that every basic feasible solution is non-degenerate. Suppose that we are at a basic feasible solution \( x \), for which we have computed the reduced costs \( c_j \) of the nonbasic variables. Then
• if all of them are nonnegative, according to Theorem 3.1, we have an optimal solution and we stop;

• if the reduced cost \( c_j \) of some nonbasic variable \( x_j \) is negative, the \( j \)th basic direction \( d \) is a feasible direction of cost decrease. This direction is obtained by letting \( d_j = 1 \), \( d_i = 0 \), for \( i \neq j \), \( B(1), \ldots, B(m) \), and \( d_B = -B^{-1}A_j \).

**Remark:** While moving along this direction \( d \), the nonbasic variable \( x_j \) becomes positive and all other nonbasic variables remain at zero. We describe this situation by saying that \( x_j \) (or \( A_j \)) enters or is brought into the basis.

**Q.** How far should we move along the direction \( d \)?

**A.** We want to move as far as possible in this direction, while staying inside the polyhedron.

**Remark:** Suppose we move from \( x \to x + td \). The maximum possible value of \( t \) is given by

\[
    t^* = \max \{ t \geq 0 : x + td \in P \}.
\]

Can we provide a formula for \( t^* \)? The only condition on \( t \) is to keep \( x + td \) feasible. Since \( Ad = 0 \), we know that \( A(x + td) = Ax + td = Ax = b \) is always satisfied. That is, the equality constraints are never violated. It remains to check whether the entries of \( x + td \) are nonnegative. There are two cases:

1. If \( d \geq 0 \), then \( x + td \geq 0 \) for all \( t \geq 0 \). The vector \( x + td \) never becomes infeasible, so \( t^* = \infty \).

2. If \( d_i < 0 \) for some \( i \), the constraint \( x_i + td_i \geq 0 \) becomes \( t \leq -x_i/d_i \). This condition must hold for every \( i \) such that \( d_i < 0 \). For a nonbasic variable \( x_i \), either \( d_i = 1 \) or \( d_i = 0 \). Thus, \( d_i \) is certainly nonnegative for nonbasic variables. We therefore only need to consider the basic components \( \{d_{B(i)} : i = 1, \ldots, m\} \) of \( d \):

\[
    t^* = \min_{i=1,\ldots,m: d_{B(i)} < 0} \left( \frac{x_{B(i)}}{d_{B(i)}} \right).
\]

Note that \( t^* \) is always nonnegative since \( x_{B(i)} \) is nonnegative for a basic feasible solution. Furthermore, if the basic feasible solution is nondegenerate, then \( x_{B(i)} > 0 \) implies that \( t^* > 0 \).
Remark: In general, when $t^*$ is chosen, and assuming it is finite, we move to the new feasible solution $y = x + t^*d$. Since $x_j = 0$ and $d_j = 1$, we have $y_j = t^* > 0$. Let \( \ell \) be a minimizing index:

\[
-\frac{x_B(\ell)}{d_B(\ell)} = \min_{\{i=1,\ldots,m:d_B(i)<0\}} \left( -\frac{x_B(i)}{d_B(i)} \right) = t^*.
\]

In particular, $d_B(\ell) < 0$, and $x_B(\ell) + t^*d_B(\ell) = 0$.

We observe that the $\ell$th basic variable has now become zero, whereas the nonbasic variable $x_j$ has become positive. This suggests that $A_j$ should replace $A_B(\ell)$ in the basis. That is, we replace the old basis matrix $B$ with

\[
\bar{B} = [A_B(1), \ldots, A_B(\ell-1), A_j, A_B(\ell+1), \ldots, A_B(m)].
\]

Equivalently, the basis \{\(B(1), \ldots, B(m)\)\} is replaced by \{\(\bar{B}(1), \ldots, \bar{B}(m)\)\} where

\[
\bar{B}(i) = \begin{cases} 
B(i) & \text{if } i \neq \ell, \\
A_j & \text{if } i = \ell.
\end{cases}
\]

Let us revisit Problem 3.1 and consider the basic feasible solution $x = (1, 1, 0, 0)$ for which we found the reduced cost $c_3$ of the nonbasic variable $x_3$ is $-3/2c_1 + 1/2c_2 + c_3$. Suppose that $c = (2, 0, 0, 0)$. Since $c_3 = -3$ is negative, we would want to move in the corresponding basic direction $d = (-3/2, 1/2, 1, 0)$. As $t$ increases, the only component of $x + td$ that decreases is the first one, since only $d_1 < 0$. The largest possible value of $t$ is $t^* = -x_1/d_1 = 2/3$, which takes us to the point $y = x + 2d/3 = (0, 4/3, 2/3, 0)$. Note that the columns $A_2 = (1, 0)$ and $A_3 = (1, 3)$ are linearly independent and they therefore form a new basis matrix

\[
\bar{B} = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix}.
\]

At this point, the variable $x_3$ has entered the basis and the variable $x_1$ has exited. Notice also that $y$ is the basic feasible solution corresponding to $\bar{B}$:

\[
\bar{B}^{-1}b = \frac{1}{3} \begin{bmatrix} 3 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 4/3 \\ 2/3 \end{bmatrix}.
\]

**Theorem 3.2:**

(i) The columns \{\(A_B(i): i = 1, \ldots, m\)\} are linearly independent, and hence, $\bar{B}$ is a basis matrix.

(ii) The vector $y = x + t^*d$ is a basic feasible solution associated with the basis matrix $\bar{B}$. 
CHAPTER 3. THE SIMPLEX METHOD

Proof: (i) If the vectors \( \{ A_B(i) : i = 1, \ldots, m \} \) were linearly dependent, then there would exist coefficients \( \lambda_1, \ldots, \lambda_m \), not all zero, such that

\[
\sum_{i=1}^{m} \lambda_i A_B(i) = 0.
\]

On multiplying this statement by \( B^{-1} \), we would then find that the vectors \( B^{-1} A_B(i) \) are also linearly dependent. Since \( B^{-1} B = 1 \), where 1 is the \( m \times m \) identity matrix and the \( i \)th column of \( B \) is \( A_B(i) \), we know that \( B^{-1} A_B(i) \) is just the \( i \)th unit vector. But then \( B^{-1} A_j \) must be linearly independent of the vectors \( \{ B^{-1} A_B(i) : i \neq \ell \} \) since their \( \ell \)th component is zero, while its \( \ell \)th component is the positive value \( -d_{B(\ell)} \). This is a contradiction.

(ii) We note that \( y \geq 0 \), \( Ay = b \), and \( y_i = 0 \) for \( i \neq B(1), \ldots, B(m) \). From (i), we know that the columns of \( B \) are linearly independent. Thus, \( y \) is a basic feasible solution corresponding to \( B \).

3.B An iteration of the simplex method

1. Start with a basis consisting of the basic columns \( A_B(1), \ldots, A_B(m) \), and an associated basic feasible solution \( x \).

2. Let \( u = -d_B = B^{-1} A_j \) where \( A_j \) is the column that enters the basis. Compute the reduced costs \( \bar{c}_j = c_j - c_B^T u \) for all nonbasic indices \( j \). If they are all nonnegative, the current basic feasible solution is optimal, and the algorithm terminates. Otherwise choose some \( j \) for which \( \bar{c}_j < 0 \).

3. If no component of \( u \) is positive, we have \( t^* = \infty \), the optimal cost is \( -\infty \) and the algorithm terminates.

4. If some component of \( u \) is positive, let

\[
t^* = \min_{i=1,\ldots,m:u_i>0} \frac{x_{B(i)}}{u_i}.
\]

5. Let \( \ell \) be such that \( t^* = x_{B(\ell)}/u_\ell \). Form a new basis by replacing \( A_{B(\ell)} \) with \( A_j \).

If \( y \) is the new basic feasible solution, the values of the new basic variables are \( y_j = t^* \) and \( y_{B(i)} = x_{B(i)} - t^* u_i \) for \( i \neq \ell \).

Theorem 3.3: Assume that the feasible set is nonempty and that every basic feasible solution is nondegenerate. Then the simplex method terminates after a finite number of iterations. At termination there are two possibilities:
1. We have an optimal basis $B$ and an associated basic feasible solution which is optimal.

2. We have found a vector $d$ satisfying $Ad = 0, d \geq 0$ and $c^T d < 0$, and the optimal cost is $-\infty$.

### 3.C The simplex method for degenerate problems

The following possibilities may be encountered:

a) If the current basic feasible solution $x$ is degenerate, $t^*$ can be zero. Then the new basic feasible solution $y$ is the same as $x$. This happens when $x_{B(l)} = 0$ and $d_{B(l)} < 0$. Nevertheless, we can still define the new basis $\bar{B}$, by replacing $A_{B(l)}$ with $A_j$, and Theorem 3.2 still holds.

b) Even if $t^*$ is positive, it may happen that more than one of the original basic variables becomes zero at the new point $x + t^*d$. Since only one of them exits the basis, the other zero values remain in the basis, so that the new basic feasible solution $y$ is also degenerate.

- In Figure 3.C we visualize the feasible set in standard form, with $n - m = 2$, by standing on the two-dimensional plane defined by $Ax = b$.

  What kind of point is $x$?

  Degenerate basic feasible solution.

  Assume that $x_4$ and $x_5$ are the nonbasic variables. What are the corresponding basic directions?

  $g$ and $f$.

  What are the corresponding value of $t^*$?

  0.

  If we perform a change of basis, with $x_4$ entering and $x_6$ exiting, what are the new nonbasic variables?

  $x_5$ and $x_6$.

  What are their corresponding basic directions?

  $h$ and $-g$.

  If we follow the direction $h$, can we reach a new basic feasible solution at a lower cost?

  Yes, we will arrive at the new basic solution $y$, which has a lower cost than $x$.

A sequence of basis changes may lead back to the initial basis, in which case the algorithm may loop indefinitely. This undesirable phenomenon is called *cycling*. 
3.D Implementation of the simplex method

A number of implementations of the simplex method are available. Which implementation is most efficient for a given problem depends very much on the structure of the matrix $A$, as well as the cost vector $c$.

3.D.1 Naive implementation

If we carry no auxiliary information from one iteration to another, then there are three major computations to be handled at each iteration (a linear solve, a matrix-vector multiply, followed by another linear solve):

$$p^\top = c_B^\top B^{-1} \Rightarrow Bp^\top = c_B^\top,$$
$$c_j = c_j - p^\top A_j, \quad j \in N,$$
$$Bu = A_j,$$

where $N$ is the set of nonbasic indices. However, depending on the implementation, one may not need to calculate $c_j$ for every $j \in N$. If one wants to always follow the direction with the most negative rate of cost change, one needs to compute the reduced costs for every nonbasic variable. The same situation holds if one wants to follow the direction which leads to the greatest cost reduction $-t^*c_j$. However, if one simply chooses the first variable encountered with a negative reduced cost, one need not compute the reduced costs for the remaining nonbasic variables. Because of this savings, one typically finds in practice that the latter choice is most efficient,
even though one it doesn’t necessarily follow the *path of steepest descent*. These implementation-dependent choices as to which variables enter and exit (at degenerate vertices) are known as *pivot rules*.

Once the entering variable \( x_j \) is known, we need to compute the displacement vector \( u = B^{-1}A_j \) that determines the direction of motion \( d \) and limiting parameter \( t^* \). The computational cost of the two linear solves is \( O(m^3) \). If we assume that \( n \gg m \), the computational cost of computing all non-basic reduced costs is \( O(nm) \). The overall computational cost of a naive implementation is thus \( O(m^3 + nm) \).

### 3.D.2 Revised simplex method

The expensive steps of the naive implementation are the two linear solves. Since the matrix \( B \) appears in both linear equations, it may seem reasonable to first compute \( B^{-1} \) and then perform the two matrix–vector multiplies \( c_B^T B^{-1} \) and \( B^{-1}A_j \). However, this alternative still requires \( O(m^3 + nm) \) operations for an entire iteration. Fortunately, there is a more efficient method for updating the matrix \( B^{-1} \) each time that we effect a change of basis, based on the previously calculated inverse. Recall that

\[
B = \begin{bmatrix} A_{B(1)}, \ldots, A_{B(m)} \end{bmatrix}
\]

and

\[
\bar{B} = \begin{bmatrix} A_{B(1)}, \ldots, A_{B(\ell-1)}, A_j, A_{B(\ell+1)}, \ldots, A_{B(m)} \end{bmatrix}.
\]

Note that

\[
B^{-1} \bar{B} = [e_1, \ldots, e_{\ell-1}, u, e_{\ell+1}, \ldots, e_m],
\]

since \( u = B^{-1}A_j \).

**Definition:** The operation of adding a constant multiple of one row of a matrix to another row (or even the same row) is called an *elementary row operation*.

**Remark:** An elementary row operation on an \( m \times n \) matrix \( A \) is equivalent to multiplication on the left by an \( m \times m \) matrix \( Q \).

- Let

\[
Q = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}
\]

The matrix \( QA \) is the result of adding twice the second row of \( A \) to the first:

\[
A = \begin{bmatrix} 7 & 10 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}.
\]

Let us now apply the following sequence of elementary row operations to \( B^{-1} \bar{B} \):
1. For each \( i \neq \ell \), add the \( \ell \)th row times \(-u_i/u_\ell\) to the \( i \)th row, recalling that \( u_\ell > 0 \). This reduces \( u_i \) to 0.

2. Divide the \( \ell \)th row by \( u_\ell \). This sets \( u_\ell \) to 1.

The above operations are equivalent to multiplication on the left by a matrix \( Q \). We have chosen this sequence specifically to reduce \( B^{-1}B \) to the \( m \times m \) identity matrix \( 1 \). That is,

\[
QB^{-1}B = 1.
\]

Since

\[
QB^{-1} = B^{-1}
\]

we now see that to compute \( B^{-1} \), we only need to apply the above sequence of elementary row operations to \( B^{-1} \)! This requires only \( O(m) \) arithmetic computations. This leads to the following implementation, known as the revised simplex method:

1. In a typical iteration, we start with a basis consisting of the basic columns \( A_{B(1)}, \ldots, A_{B(m)} \), an associated basic feasible solution \( x \), and the inverse \( B^{-1} \) of the basis matrix.

2. Compute the row vector \( p^\top = c_B^\top B^{-1} \) and the reduced costs \( \tilde{c}_j = c_j - p^\top A_j \). If they are all nonnegative, the current basic feasible solution is optimal, and the algorithm terminates; else, choose some \( j \) for which \( c_j < 0 \).

3. Compute \( u = B^{-1}A_j \). If no component of \( u \) is positive, the optimal cost is \(-\infty\), and the algorithm terminates.

4. If some component of \( u \) is positive, let

\[
t^* = \min_{i=1,\ldots,m;u_i>0} \frac{xB(i)}{u_i}.
\]

5. Let \( \ell \) be such that \( t^* = x_B(\ell)/u_\ell \). Form a new basis by replacing \( A_{B(\ell)} \) with \( A_j \). If \( y \) is the new basic feasible solution, the values of the new basic variables are \( y_j = t^* \) and \( y_B(i) = x_B(i) - t^* u_i \) for \( i \neq \ell \).

6. Form the \( m \times (m+1) \) matrix \([B^{-1}|u]\) . Add to each of its rows a multiple of the \( \ell \)th row to make the last column equal to the unit vector \( e_\ell \). The first \( m \) columns of the result is the matrix \( B^{-1} \).

### 3.D.3 Full tableau implementation

Instead of maintaining and updating the matrix \( B^{-1} \), let us maintain and update the \( m \times (n+1) \) matrix \( B^{-1}[b|A] \) with columns \( B^{-1}b, B^{-1}A_1, \ldots, B^{-1}A_n \). One advantage of doing this is that one does not need to need to allocate a separate column in the
tableau for the vector \( u = B^{-1}A_j \), which corresponds to the variable entering the basis. This special column is called the pivot column. If the \( j \)th basic variable exits the basis, the \( \ell \)th row of the tableau is called the pivot row. The element corresponding to both pivot row and column is the pivot element. Note that the pivot element \( u_\ell \) remains positive until the algorithm meets the termination condition.

We have also extended our tableau to include a zeroth column \( B^{-1}b \) containing the components of the basic variable \( x_B \). One can interpret this enlarged tableau by multiplying the standard-form constraint \( b = Ax \) by \( B^{-1} \) on each side:

\[
B^{-1}b = B^{-1}Ax
\]

The rows of the extended tableau tabulate the coefficients of this equality constraint. It is also convenient to add a zeroth row to represent the negative of the cost \(-c^T_B x_B\) in the zeroth column followed by the reduced costs \( \tilde{c}_j \) of each of the \( n \) variables in subsequent columns:

\[
\begin{array}{c|ccc}
-c^T_B x_B & \tilde{c}_1 & \ldots & \tilde{c}_n \\
B^{-1}b & B^{-1}A_1 & \ldots & B^{-1}A_n \\
\end{array}
\]

Remark: The reason for the minus sign in the expression \(-c^T_B x_B\) tabulated in the zeroth column and zeroth row is for consistency with the reduced costs \( \tilde{c}_j = c_j - c^T_B B^{-1}A_j \); this allows the same update rule to be used for all rows: a multiple of the pivot row is added to the zeroth row to set the reduced cost of the entering variable to zero, as must be the case for a basic variable. At the beginning of the iteration, the zeroth row is of the form

\[
[0|c^T] - p^T [b|A]
\]

where \( p^T = c^T_B B^{-1} \). Notice that \( p^T [b|A] \) is a linear combination of the columns of \([b|A]\). Since the pivot row \([B^{-1}b|B^{-1}A]\) is also a linear combination of the columns of \([b|A]\), adding a multiple of the pivot row will leave the zeroth row as \([0|c^T]\) minus some linear combination of the columns of \([b|A]\):

\[
[0|c^T] - \tilde{p}^T [b|A].
\]

Our update rule will zero the zeroth entry of the pivot column:

\[
c_j - \tilde{p}^T A_j = 0.
\]

Recall that \( B(\ell) = j \). For \( i \neq \ell \), we know that \( B^{-1}A_{B(i)} \) is the \( i \)th unit vector, the reduced costs of all the other basic variables will remain unchanged. That is,

\[
\tilde{c}_i = c^T_B - \tilde{p}^T B = 0,
\]
from which we see that \( p^t = c_B^t B^{-1} \), so that the updated zeroth row equals
\[
[0|c^t] - c_B^t B^{-1}[b|A],
\]
as desired.

Given a basis matrix \( B \) one initializes the full tableau with the basic components of the solution \( x_B = B^{-1} b \) and the matrix product \( B^{-1} A \). The negative cost \( -c_B^t x_B \) and reduced cost vector \( \bar{c}^T = c^T - c_B^T B^{-1} A \) are entered in the zeroth row of the tableau. Here is a single iteration of the full simplex tableau:

1. If the reduced costs are all nonnegative, the current basic feasible solution is optimal, and the algorithm terminates.

2. Otherwise, choose some \( j \) for which \( c_j < 0 \). Denote the \( j \)th column (pivot column) of the tableau by \( u \). If no component of \( u \) is positive, the optimal cost is \( -\infty \), and the algorithm terminates.

3. For each \( i \) for which \( u_i \) is positive, compute the ratio \( x_B(i)/u_i \). Let \( \ell \) be the index of a row that corresponds to the smallest ratio. The column \( A_B(\ell) \) exits the basis and the column \( A_j \) enters the basis.

4. Add to each row of the tableau a constant multiple of the \( \ell \)th row (the pivot row) so that \( u_\ell \) (the pivot element) becomes 1 and all other entries of the pivot column become 0.

**Problem 3.2:** Use the simplex method to solve the linear programming problem

\[
\text{minimize } -10x_1 - 12x_2 - 12x_3 \\
\text{subject to } \begin{align*}
x_1 + & \quad 2x_2 + & \quad 2x_3 & \leq 20, \\
2x_1 + & \quad x_2 + & \quad 2x_3 & \leq 20, \\
2x_1 + & \quad 2x_2 + & \quad x_3 & \leq 20, \\
x_1, x_2, x_3 & \geq 0.
\end{align*}
\]

First, we introduce slack variables \( x_4, x_5, x_6 \) to form the standard-form problem

\[
\text{minimize } -10x_1 - 12x_2 - 12x_3 \\
\text{subject to } \begin{align*}
x_1 + & \quad 2x_2 + & \quad 2x_3 + & \quad x_4 & = 20, \\
2x_1 + & \quad x_2 + & \quad 2x_3 + & \quad x_5 & = 20, \\
2x_1 + & \quad 2x_2 + & \quad x_3 & = 20, \\
x_1, x_2, x_3, x_4, x_5, x_6 & \geq 0.
\end{align*}
\]

We need a basic feasible solution to initialize the simplex method. An obvious basic solution is \((0, 0, 0, 20, 20, 20)\), with \( B(1) = 4, B(2) = 5, B(3) = 6 \), and the identity basis matrix. This corresponds to the vertex \( A = (0, 0, 0) \) of the original problem depicted in Figure 3.3.
Since $c_B = 0$, the zeroth row of the initial tableau contains the negative cost $-c_B^T x_B = 0$ followed by the elements of the reduced costs $\bar{c} = c$:

\[
\begin{array}{ccccccc}
0 & -10 & -12 & -12 & 0 & 0 & 0 \\
20 & 1 & 2 & 2 & 1 & 0 & 0 \\
20 & 2 & 1 & 2 & 0 & 1 & 0 \\
\end{array}
\]

The decision variables corresponding to each column are tabulated above the zeroth row. The basic variables are listed next to each of their values in the zeroth column. We are now ready to begin the simplex iteration.

Let us use the first column with a negative reduced cost, $x_1$ as our entering variable. We highlight the corresponding pivot column with an asterisk and let $u = (1, 2, 2)$. We then compute the ratios $x_B(i)/u(i)$ for each $i$ such that $u(i) > 0$. These ratios are $20/1 = 20$, $20/2 = 10$, $20/2 = 10$, respectively. For the exit variable, we choose the first variable ($x_5$) that achieves the minimum ratio (10). We highlight the pivot row with a dagger. We then apply elementary operations to the matrix to reduce the pivot column to (0, 0, 0, 1):

\[
\begin{array}{ccccccc}
100 & 0 & -7 & -2 & 0 & 5 & 0 \\
10 & 0 & 3/2 & 1 & 1 & -1/2 & 0 \\
10 & 1 & 1/2 & 1 & 0 & 1/2 & 0 \\
0 & 0 & 1 & -1 & 0 & -1 & 1 \\
\end{array}
\]

This corresponds to the degenerate vertex $D = (10, 0, 0)$ of the original problem depicted in Figure 3.3. The basic variables are now $x_4$, $x_1$, and $x_6$. In the tableau, the degeneracy is apparent from the value $x_6 = 0$. We now repeating the iteration, choosing $x_2$ as our entering variable and letting $u = (3/2, 1/2, 1)$. We compute $x_4/u(1) = 10/(3/2) = 20/3$, $x_1/u(2) = 10/(1/2) = 20$, $x_6/u(3) = 0/1 = 0$. We therefore choose $x_6$ as our exiting variable and row reduce to obtain (0, 0, 0, 1) in the pivot column. This row reduction accomplishes a change of basis but leaves us still at the point $D = (10, 0, 0)$:

\[
\begin{array}{ccccccc}
100 & 0 & 0 & -9 & 0 & -2 & 7 \\
10 & 0 & 0 & 5/2 & 1 & 1 & -3/2 \\
10 & 1 & 0 & 3/2 & 0 & 1 & -1/2 \\
0 & 0 & 1 & -1 & 0 & -1 & 1 \\
\end{array}
\]

Our basic variables are now $x_4$, $x_1$, and $x_2$. Next, we choose $x_3$ as our entering variable and let $u = (5/2, 3/2, -1)$. We compute $x_4/u(1) = 10/(5/2) = 4$ and $x_1/u(2) = 10/(3/2) = 20/3$. The exiting variable is thus $x_4$. Upon row reducing, we obtain a zeroth row containing no negative reduced costs:

\[
\begin{array}{ccccccc}
136 & 0 & 0 & 0 & 18/5 & 8/5 & 8/5 \\
4 & 0 & 0 & 1 & 2/5 & 2/5 & -3/5 \\
4 & 1 & 0 & 0 & -3/5 & 2/5 & 2/5 \\
4 & 0 & 1 & 0 & 2/5 & -3/5 & 2/5 \\
\end{array}
\]
We have thus reached the optimal solution $E = (4, 4, 4)$, for which the optimal cost is $-136$.

**Problem 3.3:** Use the simplex method to solve the standard-form linear programming problem

$$\begin{align*}
\text{minimize} & \quad x_1 + x_2 + x_3 \\
\text{subject to} & \quad 2x_1 - x_2 + x_3 = 6, \\
& \quad x_1 - x_2 - x_4 = 2, \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}$$

We need a basic feasible solution to initialize the tableau. The choice $B(1) = 1$ and $B(2) = 2$ corresponds to

$$B = \begin{bmatrix} 2 & -1 \\ 1 & -1 \end{bmatrix}, \quad x_B = \frac{1}{-1} \begin{bmatrix} -1 & 1 & 6 \\ -2 & 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix},$$

and

$$B^{-1} A = \frac{1}{-1} \begin{bmatrix} -1 & 1 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 2 & -1 & 1 & 0 \\ -1 & 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}.$$
Let \( c = (1, 1, 1, 0) \). We can use the last two columns of the above result to easily calculate the reduced costs of the nonbasic variables:

\[
\tilde{c}_3 = c_3 - c_B^T B^{-1} A_3 = 1 - [1 \ 1] \begin{bmatrix} 1 \\ 1 \end{bmatrix} = -1,
\]

\[
\tilde{c}_4 = c_4 - c_B^T B^{-1} A_4 = 0 - [1 \ 1] \begin{bmatrix} 1 \\ 2 \end{bmatrix} = -3.
\]

We enter these results, along with the negative cost at the initial basic feasible solution, \(-c_B^T x_B = -6\), in the tableau:

\[
\begin{array}{cccc}
   & x_1 & x_2 & x_3^* & x_4 \\
-6 & 0 & 0 & -1 & -3 \\
x_1 & 4 & 1 & 0 & 1 \\
x_2 & 2 & 0 & 1 & 1 \\
\end{array}
\]

Two iterations of the simplex method then bring us to the optimal solution \((3, 0, 0, 1)\) with optimal cost 3:

\[
\begin{array}{cccc}
   & x_1 & x_2 & x_3 & x_4^* \\
-4 & 0 & 1 & 0 & -1 \\
x_1 & 2 & 1 & -1 & 0 \\
x_2 & 2 & 0 & 1 & 1 \\
\end{array}
\]

\[
\begin{array}{cccc}
   & x_1 & x_2 & x_3 & x_4 \\
-3 & 0 & 3/2 & 1/2 & 0 \\
x_1 & 3 & 1 & -1/2 & 1/2 \\
x_4 & 1 & 0 & 1/2 & 1/2 \\
\end{array}
\]

**Remark:** The entering decision variable is always one with a negative cost. Often there is more than one choice for this entering variable. In the previous examples of the full simplex tableau, we always chose the entering variable to be the one with the smallest index. This is particularly advantageous in implementations of the revised simplex method, where the reduced costs are computed separately from the tableau. If one computes the reduced costs sequentially, starting with the lowest index, one can stop computing reduced costs as soon as the first negative value is encountered. Although the full simplex tableau is perhaps more elegant, in that the reduced cost computation is incorporated directly into the row operations, calculating the reduced costs in this iterative manner has the disadvantage that all reduced costs must be calculated (no matter what pivot rule is used), so that reduced costs of the other variables are made available for future iterations.

**Remark:** For both the revised and full tableau simplex methods, there is nevertheless an important advantage in always choosing the entering variable to be the variable with a negative cost that has the lowest index. If we also choose the exit variable to be the variable with lowest index that achieves the minimum value of ratio \( x_{B(i)}/u_i \), it has been shown that the simplex method never cycles and will always terminate. This widely used pivot rule is known as Bland’s rule.
Problem 3.4: Consider the linear programming problem

minimize \(-3x_1 + 80x_2 - 14x_3 + 24x_4 - 12x_7\)

subject to \(\frac{1}{4}x_1 - 8x_2 - x_3 + 9x_4 + x_5 = 0,\)

\(\frac{1}{2}x_1 - 12x_2 - \frac{1}{2}x_3 + 3x_4 + x_6 = 0,\)

\(x_3 + x_7 = 4,\)

\(x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0.\)

An obvious basic solution is \((0, 0, 0, 0, 0, 0, 4),\) with \(B(1) = 5, B(2) = 6, B(3) = 7,\) and the identity basis matrix. Since \(c_B = (0, 0, -12),\) the zeroth row of the initial tableau contains the negative cost \(-c_B^\top x_B = 48\) followed by the elements of the reduced cost vector \([-3, 80, -14, 24, 0, 0, -12] - [0, 0, -12]A:\)

<table>
<thead>
<tr>
<th></th>
<th>(x_1^*)</th>
<th>(x_2)</th>
<th>(x_3)</th>
<th>(x_4)</th>
<th>(x_5)</th>
<th>(x_6)</th>
<th>(x_7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_5^\dagger)</td>
<td>48</td>
<td>-3</td>
<td>80</td>
<td>-2</td>
<td>24</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(x_6)</td>
<td>0</td>
<td>1/4</td>
<td>-8</td>
<td>-1</td>
<td>9</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(x_7)</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

If we pivot according to Bland’s rule, the simplex method reaches the optimal solution in six iterations:

<table>
<thead>
<tr>
<th></th>
<th>(x_1)</th>
<th>(x_2^*)</th>
<th>(x_3)</th>
<th>(x_4)</th>
<th>(x_5)</th>
<th>(x_6)</th>
<th>(x_7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_1)</td>
<td>48</td>
<td>0</td>
<td>-16</td>
<td>-14</td>
<td>132</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>(x_6^\dagger)</td>
<td>0</td>
<td>1</td>
<td>-32</td>
<td>-4</td>
<td>36</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>(x_7)</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>3/2</td>
<td>-15</td>
<td>-2</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>(x_1)</th>
<th>(x_2)</th>
<th>(x_3^*)</th>
<th>(x_4)</th>
<th>(x_5)</th>
<th>(x_6)</th>
<th>(x_7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_1)</td>
<td>48</td>
<td>0</td>
<td>0</td>
<td>-8</td>
<td>72</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>(x_2)</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>8</td>
<td>-84</td>
<td>-12</td>
<td>8</td>
</tr>
<tr>
<td>(x_7)</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>3/8</td>
<td>-15/4</td>
<td>-1/2</td>
<td>1/4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>(x_1)</th>
<th>(x_2)</th>
<th>(x_3)</th>
<th>(x_4^*)</th>
<th>(x_5)</th>
<th>(x_6)</th>
<th>(x_7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_3)</td>
<td>48</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-12</td>
<td>-8</td>
<td>12</td>
</tr>
<tr>
<td>(x_2)</td>
<td>0</td>
<td>1/8</td>
<td>0</td>
<td>1</td>
<td>-12/2</td>
<td>-3/2</td>
<td>1</td>
</tr>
<tr>
<td>(x_7)</td>
<td>4</td>
<td>-1/8</td>
<td>0</td>
<td>0</td>
<td>21/2</td>
<td>3/2</td>
<td>-1</td>
</tr>
</tbody>
</table>
However, suppose we always select the entering variable with the most negative reduced cost. The first four iterations will be exactly as above, until we arrive at

After two further iterations using the most-negative reduced cost, we arrive back at the initial tableau:

Notice that at each iteration, we always remain at the degenerate vertex \((0, 0, 0, 0, 0, 0, 4)\), with nonoptimal cost \(-48\). Although a change of basis is performed at each iteration, eventually we return to the initial basis. With this pivot rule, the simplex method cycles forever, never terminating. This example of *cycling* emphasizes the importance of Bland’s rule in practical implementations of the simplex method.
3. E Finding an initial basic feasible solution

We now describe a systematic procedure for finding a basic feasible solution and associated basis matrix to initialize the simplex tableau. First, multiply by \(-1\) any equality constraints that have a negative right-hand side, to ensure that the standard-form problem

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

satisfies \(b \geq 0\).

Next, introduce a vector \(y \in \mathbb{R}^m\) of artificial variables and consider the auxiliary problem

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{m} y_i \\
\text{subject to} & \quad Ax + y = b \\
& \quad x \geq 0 \\
& \quad y \geq 0
\end{align*}
\]

**Remark:** If \(x\) is a feasible solution to the original problem, \((x, y = 0)\) yields a zero cost solution to the auxiliary problem. Hence, if the optimal cost of the auxiliary problem is nonzero, the original problem is *infeasible*.

**Remark:** On the other hand, a zero-cost basic feasible solution to the auxiliary problem satisfies \(y = 0\), so that \(x\) is a feasible solution to the original problem. If the associated basis matrix \(B\) contains only columns of \(A\), the columns in the tableau that correspond to the artificial variables \(y\) can simply be dropped.

**Remark:** The situation is more complicated if some of the artificial variables \(y\) are in the final basis \(B\). In this case, we must drive the artificial variables out of the basis. Denote the columns of \(A\) that belong to the basis to the auxiliary problem by \(A_{B(1)}, \ldots, A_{B(k)}\), where \(k < m\). Suppose that the \(\ell\)th basic variable is an artificial variable of the zero-cost optimal solution of the auxiliary problem, where \(\ell > k\).

If the \(\ell\)th row of \(B^{-1}A\) is zero, then the matrix \(A\) has linearly dependent rows. We can therefore remove that row and continue. Otherwise, there is some \(j\) such that the \(\ell\)th entry of \(B^{-1}A_j\) is nonzero. Since \(B^{-1}A_{B(i)} = e_i\) for \(i = 1, \ldots, k\), and \(k < \ell\) we know that the \(\ell\) entry of these vectors is zero. This means that \(B^{-1}A_j\) is not a linear combination of these vectors, so that \(A_j\) is linearly independent of \(A_{B(1)}, \ldots, A_{B(k)}\). We can therefore allow \(x_j\) to enter the basis, with \(x_\ell\) exiting. We can accomplish this by our usual row reduction. Although the pivot element (the \(\ell\)th entry of the \(j\)th column) could be negative, this doesn’t affect the mechanics of the row operations; it is only important that the pivot element be nonzero.
Problem 3.5: In Prob. 3.3, we solved the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad x_1 + x_2 + x_3 \\
\text{subject to} & \quad 2x_1 - x_2 + x_3 = 6, \\
& \quad x_1 - x_2 - x_4 = 2, \\
& \quad x_1, x_2, x_3, x_4 \geq 0,
\end{align*}
\]

starting from a given basis matrix and corresponding basic feasible solution. Note that the right-hand side vector \( b = (6, 2) \) is already non-negative. A systematic way to find an initial basis matrix and tableau is to introduce artificial variables \( x_5 \) and \( x_6 \) form the auxiliary problem

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

starting from the initial solution \((0, 0, 0, 0, 6, 2)\) and identity basis matrix:

\[
\begin{array}{ccccccc}
x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\
-8 & -3 & 2 & -1 & 1 & 0 & 0 \\
x_5 & = & 6 & 2 & -1 & 1 & 0 & 1 & 0 \\
x_6 & = & 2 & 1 & -1 & 0 & -1 & 0 & 1
\end{array}
\]

Two iterations of the simplex method then lead to an optimal zero-cost solution of the auxiliary problem:

\[
\begin{array}{ccccccc}
x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\
-2 & 0 & -1 & -1 & -2 & 0 & 3 \\
x_5 & = & 2 & 0 & 1 & 1 & 2 & 1 & -2 \\
x_1 & = & 2 & 1 & -1 & 0 & -1 & 0 & 1
\end{array}
\]

\[
\begin{array}{ccccccc}
x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\
0 & 0 & 0 & 0 & 1 & 1 \\
x_2 & = & 2 & 0 & 1 & 1 & 2 & 1 & -2 \\
x_1 & = & 4 & 1 & 0 & 1 & 1 & 1 & -1
\end{array}
\]

Since none of the final basic variables are artificial variables, we can simply drop the columns in the tableau associated with the artificial variables. Except for the zeroth row and the order of the basic variables (the order of the rows in the tableau is irrelevant), the resulting truncated tableau is identical to the initial solution we found in Prob 3.3. The tabulated columns of \( B^{-1}A \) can then be used to calculate the reduced costs in the zeroth row.
3.F The two-phase simplex method

The above considerations lead us to the so-called *two-phase simplex method*. In the first phase, the auxiliary problem is solved to find an initial tableau for the second phase. The second phase is the application of the simplex method on the original problem, starting from the basic feasible solution found in the first phase.

**Phase I:**

1. Multiply the constraint as needed by $-1$ so that $b \geq 0$.

2. If a basic feasible solution is not known, introduce artificial variables $y_1, y_2, \ldots, y_m$ and apply the simplex method to the auxiliary problem with cost $\sum_{i=1}^{m} y_i$.

3. If the optimal cost in the auxiliary problem is positive, the original problem is infeasible and the algorithm terminates.

4. If the optimal cost in the auxiliary problem is zero, a feasible solution to the original problem has been found. If no artificial variable is in the final basis, the artificial variables and the corresponding columns are eliminated, and a feasible basis for the original problem is available.

5. If the $\ell$th basic variable is an artificial one, examine the $\ell$th entry of the columns $B^{-1}A_j, j = 1, \ldots, n$. If all of these entries are zero, the $\ell$th row represents a redundant constraint and is eliminated. Otherwise, if the $\ell$th entry of the $j$th column is nonzero, apply a change of basis (with this entry serving as the pivot element): the $\ell$th basic variable exits and $x_j$ enters the basis. Repeat this procedure until all artificial variables are driven out of the basis.

**Phase II:**

1. Let the final basis and tableau obtained from Phase I be the initial basis and tableau for Phase II.

2. Compute the reduced costs of all variables for this initial basis, using the cost coefficients of the original problem.

3. Apply the simplex method to the original problem.

**Problem 3.6:** Use the two-phase simplex method to determine the initial tableau for the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad x_1 - x_4 \\
\text{subject to} & \quad x_1 + x_2 = 4, \\
& \quad x_3 = 2, \\
& \quad x_2 + x_4 = -1, \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]
Using the two-phase simplex method, the problem is solved as follows.

**Phase I:**

**Step 1.** Multiply every constraint with a negative right-hand side by $-1$, so that $b \geq 0$:

\[
\begin{align*}
\text{minimize} & \quad x_1 - x_4 \\
\text{subject to} & \quad x_1 + x_2 + x_3 = 4, \\
& \quad x_3 = 2, \\
& \quad x_2 - x_4 = 1, \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]

**Step 2.** In order to find a feasible solution, the artificial variables $x_5, \ldots, x_7$ are introduced to form the auxiliary problem

\[
\begin{align*}
\text{minimize} & \quad x_5 + x_6 + x_7 \\
\text{subject to} & \quad x_1 + x_2 + x_5 = 4, \\
& \quad x_3 + x_6 = 2, \\
& \quad x_2 - x_4 + x_7 = 1, \\
& \quad x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0.
\end{align*}
\]

We apply the simplex method, starting from the basic feasible solution $x = 0, y = b$, $B(1) = 5, B(2) = 6, B(3) = 7$, the identity basis matrix, $c_B = (1, 1, 1)$, and the nonbasic reduced costs $c_j = -c_B^t A_j$.

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_5$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$x_6$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$x_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
**Phase II:** Let \( c = (1, 1, 1, 0) \) be the cost vector of the original problem. The basic variables are \( x_1, x_2, \) and \( x_3, \) so that \( c_B = (1, 0, 0) \) and the negative cost is \( -c_B^T x_B = -3. \) The reduced costs of the basic variables is zero. We use the fourth column of the final tableau to calculate the nonbasic reduced cost

\[
c_4 = c_4 - c_B^T B^{-1} A_4 = -1 - [1 \ 0 \ 0] \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} = -2.
\]

Starting from this initial basic feasible solution, we apply the simplex method to compute the optimal solution to the original problem:

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

\[
\begin{array}{c|cccc}
\hline
& x_1 & x_2 & x_3 & x_4 \\
\hline
x_1 & 3 & 1 & 0 & 0 & 1 \\
x_3 & 2 & 0 & 0 & 1 & 0 \\
x_2 & 1 & 0 & 1 & 0 & -1 \\
\hline
\end{array}
\]

The optimal solution is \((0, 4, 2, 3),\) with optimal cost \(-3.\)
Index

active, 20
adjacent, 24, 28
affine, 16
artificial variables, 52
auxiliary problem, 52

basic columns, 26
basic direction, 34
basic feasible solution, 22
basic solution, 21
basic variables, 26
basis, 26
basis matrix, 26
binding, 20
Bland’s rule, 49
bounded, 14

concave, 15, 16
concave down, 15
concave up, 15
constraints, 5
contains a line, 30
convex, 15, 16
convex hull, 16
cost function, 10
cost vector, 10
cycling, 41, 51
decision variables, 5, 7, 10
degenerate, 28
decision variables, 5, 7, 10
degenerate, 28

edge, 24
elementary row operation, 43
Elimination of free variables, 12
Elimination of inequality constraints, 12
enters the basis, 40

extreme point, 20
feasible direction, 33
feasible set, 6, 7
feasible solution, 10
feasible solutions, 6
feasible vector, 10
finite number, 5
finitely, 5
free, 10
full simplex tableau, 46
global minimum, 16
graph, 15
half space, 14
hyperplane, 14
infeasible, 52
infinite number, 5
line segment, 16
linear, 9, 10
linear constraint, 10
linear cost function, 10
linear interpolation, 15
linear optimization, 5
linear programming, 5
local minimum, 16
nonbasic, 26
objective function, 6, 7, 10
optimal, 37
optimal cost, 10
optimal feasible solution, 7, 10
optimal set, 7
optimal solution, 7, 10
optimal value, 7
optimization problem, 5

parameter, 15
parametric form, 15
partial ordering, 10
path of steepest descent, 43
piecewise linear, 18
pivot column, 45
pivot element, 45
pivot row, 45
pivot rules, 43
polyhedron, 14
Procedure for constructing basic solutions, 25

rate of cost change, 35
reduced cost, 35
reduced cost vector, 36
revised simplex method, 44

slack variable, 12
standard form, 11

two-phase simplex method, 54
unrestricted, 10

vertex, 20

zero, 34
zeroth column, 45
zeroth row, 45