## Contents

1 Linear Programming .................................................. 6
   1.A Linear Programming Problems .................................. 11
   1.B Standard form .................................................. 12
   1.C Reduction to standard form .................................... 13

2 The Geometry of Linear Programming ............................... 15
   2.A Polyhedra ...................................................... 15
   2.B Review of convexity for functions $f : \mathbb{R} \to \mathbb{R}$ .... 16
   2.C Convexity for functions $f : \mathbb{R}^n \to \mathbb{R}$ ............... 17
   2.D The convex hull .............................................. 17
   2.E Piecewise linear convex objective functions ................. 19
   2.F Problems involving absolute values ............................ 20
   2.G Extreme points ............................................... 21
   2.H Polyhedra in standard form ................................... 26
   2.I Correspondence of bases and basic solutions .................. 30
   2.J Degeneracy .................................................... 30
   2.K Existence of extreme points .................................. 32
   2.L Optimality of extreme points .................................. 34

3 The Simplex Method .................................................... 35
   3.A Development of the simplex method ........................... 40
   3.B An iteration of the simplex method ............................ 42
   3.C The simplex method for degenerate problems ................. 43
   3.D Implementation of the simplex method ......................... 45
      3.D.1 Naive implementation ..................................... 45
      3.D.2 Revised simplex method ................................... 45
      3.D.3 Full tableau implementation ............................... 47
   3.E Finding an initial basic feasible solution ..................... 54
   3.F The two-phase simplex method ................................ 60
   3.G Application to transportation problems ....................... 62
## CONTENTS

### Duality
- 4.A Introduction .................................................. 66
- 4.B The dual problem ........................................... 69
- 4.C The duality theorem ....................................... 71
- 4.D Complementary slackness ............................... 73
- 4.E Farkas' lemma and linear inequalities ............... 75

### Sensitivity Analysis
- 5.A A new variable is added .................................... 77
- 5.B A new inequality constraint is added ................. 79
- 5.C The dual simplex method ............................... 80
- 5.D Changes in the target vector $b$ ......................... 82
- 5.E Changes in the cost vector $c$ ......................... 83

### Parametric Programming

Bibliography ..................................................... 86

Index ............................................................ 87
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 labour. 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 $f(B,C) = 600B + 500C$. 

[Bertsimas & Tsitsiklis 1997]
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 green triangle represents the feasible set, which is the collection of all feasible solutions \( (B, C) \) that satisfy the 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 $B = 0$ and $C = 20$. 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 $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$, over the feasible set $F \subset \mathbb{R}^n$ of solutions satisfying all of the constraints, can be written symbolically as

$$\max_{\mathbf{x} \in F} f(\mathbf{x})$$

or

$$\min_{\mathbf{x} \in F} f(\mathbf{x})$$. 


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.

**Remark:** In determining optimal solutions, one considers only points \(x\) that are feasible.

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

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

- For the problem
  \[
  \min_{x \geq -1, \quad x \leq 1, \quad y \leq 2} (x^2 + y^2),
  \]
  the feasible set is \(\{(x, y) : x \in [-1, 3], y \in (-\infty, 2]\}\), 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 to an optimization problem:

\[
\text{The problem is } \begin{cases} 
\text{infeasible (no feasible solutions)} \\
\text{feasible} \begin{cases} 
\text{no optimal solutions (unbounded)} \\
\text{optimal solutions} \begin{cases} 
\text{finitely many} \\
\text{infinitely many}
\end{cases}
\end{cases}
\end{cases}
\]

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 25% lamb, while the deluxe version should contain at least 50% lamb. The cost of rice is \$1/kg, whereas lamb costs \$2/kg. You have 300 kg of rice and 100 kg of lamb available. Formulate a linear programming problem that determines 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 \( \ell_e \) and \( \ell_d \) represent the corresponding weights of lamb. The objective function is 
\[
f(r_e, r_d, \ell_e, \ell_d) = 3(r_e + \ell_e) + 4(r_d + \ell_d) - (r_e + r_d) - 2(\ell_e + \ell_d).
\]
One wants to maximize this function subject to the constraints
\[
\begin{align*}
  r_e + r_d &\leq 300, \\
  \ell_e + \ell_d &\leq 100, \\
  \ell_e &\leq \frac{1}{4}(r_e + \ell_e), \\
  \ell_d &\geq \frac{1}{2}(r_d + \ell_d), \\
  r_e, \ell_e, r_d, \ell_d &\geq 0.
\end{align*}
\]

1.A Linear Programming Problems

Definition: A function \( f : \mathbb{R}^n \to \mathbb{R} \) 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
\[
f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)
\]
for all vectors \( x \) and \( y \) 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 $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ where $x_i \leq y_i$ for all $i$, we say $x \leq y$. Similarly, $x \geq y$ 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 $c^\top x = \sum_{i=1}^n c_i x_i$ over all vectors $x = (x_1, \ldots, x_n)$ in $\mathbb{R}^n$, subject to a set of linear equality and inequality constraints. The vector $c = (c_1, \ldots, c_n)$ is called a cost vector.

We recall that in a linear programming problem of the form
\[
\begin{align*}
\text{minimize} & \quad c^\top x \\
\text{subject to} & \quad a_i^\top x \leq b_i, \quad i \in M_1, \\
& \quad a_i^\top x \geq b_i, \quad i \in M_2, \\
& \quad a_i^\top x = b_i, \quad i \in M_3, \\
& \quad x_j \leq 0, \quad j \in N_1, \\
& \quad x_j \geq 0, \quad j \in N_2,
\end{align*}
\] (1.1)
where $M_1$, $M_2$, $M_3$, $N_1$, and $N_2$ are finite sets of integers, the variables $x = (x_1, \ldots, x_n)$ are called decision variables. When $x$ 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 $N_1 \cup N_2$, we say that $x_j$ is a free or unrestricted variable. The linear function $f(x) = c^\top x$ is called the objective function or cost function. A feasible solution $x^*$ that minimizes the objective function is called an optimal feasible solution or optimal solution. The value $c^\top x^*$ is the optimal cost.

Problem 1.2: By introducing an appropriate coefficient matrix $A$ and target vector $b$, 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.B 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 $A$, 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 nonnegative 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></th>
<th>food 1</th>
<th>$\cdots$</th>
<th>food $n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>nutrient 1</td>
<td>$a_{11}$</td>
<td>$\cdots$</td>
<td>$a_{1n}$</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\ddots$</td>
<td>$\vdots$</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 expressing the requirements of an ideal diet. The standard-form problem aims to mix nonnegative 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 $b$ instead represents the minimal requirements of an adequate diet, then the condition will become

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

1.C 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 nonnegative variables $x_j^+$ and $x_j^-$ (every real number can be as the difference of two nonnegative 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 nonnegative 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 nonnegative 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 a corresponding feasible solution \((x_1, x_2^+, x_2^-, x_3) = (1, 0, 1, 4)\) (not unique), with 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 (convex) 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 $a$. 
Remark: A polyhedron can thus be expressed as an intersection of half-spaces.

Definition: The \textit{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 \( F : \mathbb{R}^n \to \mathbb{R}^{n+1}, F(x) = (x, f(x)) \).

\section*{2.B Review of convexity for functions \( f : \mathbb{R} \to \mathbb{R} \)}

Definition: A function \( f : \mathbb{R} \to \mathbb{R} \) is \textit{convex} (sometimes called \textit{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 \textit{concave} (sometimes called \textit{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
\[
f(x) \leq f(p) + \frac{f(q) - f(p)}{q - p}(x - p) \quad \text{for all } x \in [p, q], \quad p, q \in I. \tag{2.1}
\]

This condition may be rewritten by re-expressing the \textit{linear interpolation} of \( f \) between \( p \) and \( q \) on the right-hand side of (2.1):
\[
f(x) \leq \left(\frac{q - x}{q - p}\right)f(p) + \left(\frac{x - p}{q - p}\right)f(q) \quad \text{for all } x \in [p, q], \quad p, q \in I. \tag{2.2}
\]

It is sometimes convenient to introduce the \textit{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 \textit{parametric form}:
\[
f(tp + (1 - t)q) \leq tf(p) + (1 - t)f(q) \quad \text{for all } t \in [0, 1], \quad p, q \in I. \tag{2.3}
\]
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 $q$ sufficiently near $p$.

**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 \( a^T(t x + (1 - t) y) \geq t b + (1 - t) b = b \). That is, \( a^T(t x + (1 - t) y) \in H \). Thus \( H \) is convex.

Theorem 2.2 (Properties of Convex Sets):

a) The intersection of convex sets is convex.

b) Every polyhedron \( \{ x \in \mathbb{R}^n : A x \geq b \in \mathbb{R}^m \} \) 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 \( x, y \in \bigcap_{i \in I} S_i \) then for each \( i \in I \) and \( t \in [0, 1] \), we know from the convexity of \( S_i \) that \( t x + (1 - t) y \in S_i \). Therefore \( t x + (1 - t) y \in \bigcap_{i \in I} S_i \). Thus \( \bigcap_{i \in I} S_i \) 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 convex 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 convex 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 \( t \in [0, 1] \). Then

\[
 t y + (1 - t) z = t \sum_{i=1}^{k} \alpha_i x_i + (1 - t) \sum_{i=1}^{k} \beta_i x_i = \sum_{i=1}^{k} [t \alpha_i + (1 - t) \beta_i] x_i.
\]
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 $f : \mathbb{R}^n \rightarrow \mathbb{R}$ 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 $M \geq c_i^T x + d_i, \quad i = 1, \ldots, m,$
\[
A x \geq b.
\]

**Theorem 2.3** (Piecewise convex functions): Let $f_1, \ldots, f_m : \mathbb{R}^n \rightarrow \mathbb{R}$ 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 handling piecewise linear convex objective functions, we can also handle piecewise affine constraints like
\[
\max_{i=1,\ldots,m} (e_i^\top x + d_i) \leq b
\]
by rewriting them as separate constraints \( e_i^\top x + d_i \leq b \), \( i = 1, \ldots, m \).

### 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 \( x = (x_1, \ldots, x_n) \) and the cost coefficients \( c_i \) are positive. 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 is an upper bound for both \( x_i \) and \(-x_i\) 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 nonnegative decision variables:
\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} c_i (x_i^+ + x_i^-) \\
\text{subject to} & \quad Ax^+ - Ax^- \geq b, \\
& \quad x^+, x^- \geq 0,
\end{align*}
\]
2.G. EXTREME POINTS

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 \( |x_i| = x_i^+ + x_i^- \).

**Remark:** A similar argument shows that one can apply this technique to any linear programming of the form

\[
\text{minimize } \sum_{i=1}^{n} c_i |x_i| + \sum_{i=1}^{n} d_i x_i \\
\text{subject to } \sum_{i=1}^{n} A_i |x_i| + \sum_{i=1}^{n} B_i x_i \leq b,
\]

where each \( c_i \) and all entries in each \( A_i \) are nonnegative.

**Problem 2.1:**

Reformulate the problem

\[
\text{minimize } x_1 + |x_2 - 1| \\
\text{subject to } |x_1 + 1| + |x_2| \leq 2.
\]

Let \( x_1 + 1 = x_1^+ - x_1^- \) and replace \( |x_1 + 1| \) by \( x_1^+ + x_1^- \). Let \( x_2 = x_2^+ - x_2^- \), replacing \( |x_2| \) by \( x_2^+ + x_2^- \). Finally, let \( x_2 - 1 = x_3^+ - x_3^- \), replacing \( |x_2 - 1| \) with \( x_3^+ + x_3^- \). Noting that we can ignore additive constants in the objective function, we obtain the equivalent linear programming problem

\[
\text{minimize } x_1^+ - x_1^- + x_3^+ + x_3^- \\
\text{subject to } x_1^+ + x_1^- + x_2^+ + x_2^- \leq 2, \\
x_2^+ - x_2^- - x_3^+ + x_3^- = 1, \\
x_1^+, x_1^-, x_2^+, x_2^-, x_3^+, x_3^- \geq 0.
\]

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 other points \( y, z \in P \) 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 $x$.

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

\[
\begin{align*}
    a_i^T x &\geq b_i & i &\in M_1, \\
    a_i^T x &\leq b_i & i &\in M_2, \\
    a_i^T x &= b_i & i &\in M_3.
\end{align*}
\]

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

Figure 2.1: The polyhedron $\{(x_1, x_2, x_3) : x_1 + x_2 + x_3 = 1, x_1, x_2, x_3 \geq 0\}$.

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

  - $A$: $x_1 + x_2 + x_3 = 1$, $x_2 = 0$, $x_3 = 0$
  - $B$: $x_1 + x_2 + x_3 = 1$, $x_1 = 0$, $x_3 = 0$
  - $C$: $x_1 + x_2 + x_3 = 1$, $x_1 = 0$, $x_2 = 0$
  - $D$: $x_1 = 0$, $x_2 = 0$, $x_3 = 0$
  - $E$: $x_1 + x_2 + x_3 = 1$, $x_3 = 0$
Theorem 2.4 (Active constraints): Let \( x \in \mathbb{R}^n \) and \( I = \{ i : a_i^\top x = b_i \} \) 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 \( a_i^\top y = b_i, i \in I \), has the unique solution \( y = x \).

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

\( \text{(b) } \iff \text{(c):} \) Suppose that \( x_1 \) and \( x_2 \) are solutions to the system of equations \( a_i^\top x = b_i, i \in I \). Then \( d = x_1 - x_2 \in \mathbb{R}^n \) satisfies \( a_i^\top d = 0 \) 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 \( a_i^\top x = b_i, i \in I \), 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. there are \( n \) linearly independent active constraints at \( x \).

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

Remark: A feasible solution \( x \) is basic if \( n \) linearly independent constraints are active at \( x \).
• Given the polyhedron $P$ shown in Figure 2.1, which of the points $A, B, C, D, E$ 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. 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!

• Given the polygon in Figure 2.2, which of the points $A, B, C, D, E, F, G, H$ 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

Figure 2.2: Examples of basic solutions.
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** (Characterization of 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: Represent $P$ as a set of constraints $a_i^T x \geq b_i$ and $a_i^T x = b_i$.

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 $c^T x = c^T (tx + (1-t)x) < c^T (ty + (1-t)z)$. Hence $x \neq ty + (1-t)z$. That is, $x$ cannot be expressed as a convex combination of any two other points of $P$; it is therefore an extreme point.

Extreme point $\Rightarrow$ Basic feasible solution:
Let $x \in P$ and $I = \{i : a_i^T x = b_i\}$. 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 therefore span a proper subset of $\mathbb{R}^n$. Choose a vector nonzero $d \in \mathbb{R}^n$ orthogonal to this set, so that $a_i^T d = 0$ for all $i \in I$. For $\epsilon > 0$ consider the points $y = x + \epsilon d$ and $z = x - \epsilon d$. We see that $a_i^T y = a_i^T z = b_i$ for all $i \in I$. Moreover, for $i \notin I$, we know that $a_i^T x > b_i$. If we choose $\epsilon$ so that $\epsilon |a_i^T d| < a_i^T x - b_i$ for all $i \notin I$, then $a_i^T y = a_i^T x + \epsilon a_i^T d > b_i$ and likewise $a_i^T z = a_i^T x - \epsilon a_i^T d > b_i$ for all $i \notin I$. Thus, $y$ and $z$ also belong to $P$. But then $x = (y + z)/2$ could not be an extreme point of $P$.
Basic feasible solution ⇒ Vertex:
Let \( x \) be a basic feasible solution of \( P \), \( I = \{ i : a_i^\top x = b_i \} \), and \( c = \sum_{i \in I} a_i \). Then
\[
c^\top x = \sum_{i \in I} a_i^\top x = \sum_{i \in I} b_i.
\]
For any \( y \in P \) we know that \( a_i^\top y \geq b_i \), so that
\[
c^\top y = \sum_{i \in I} a_i^\top y \geq \sum_{i \in I} b_i = c^\top x. \tag{2.4}
\]
That is, \( x \) is an optimal solution to the problem of minimizing \( c^\top y \) over \( P \). Equality in (2.4) holds iff \( a_i^\top y = b_i \) for all \( i \in I \). But this system of equations has a unique solution since \( x \) is a basic feasible solution, with \( n \) linearly independent constraints active at \( x \). That is, equality in (2.4) holds only at the unique minimizer \( x \). This means that \( x \) is a vertex of \( P \).

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

**Proof**: Suppose that \( m \geq n \) linear constraints are imposed on a basic solution \( x \in \mathbb{R}^n \). We know that \( n \) linearly independent constraints must be active at \( x \), 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 \( \{ x \in \mathbb{R}^n : 0 \leq x_i \leq 1, i = 1, \ldots, n \} \) 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 \( \{ x \in \mathbb{R}^n : Ax = b, x \geq 0 \} \), 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 $Ax = b$; 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) $x_i = 0$ for all $i \neq B(1), \ldots, B(m)$.

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

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

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. Since the remaining $n - m$ decision variables are zero, we see that the solution $x = (x_1, \ldots, x_n)$ in $\mathbb{R}^n$ satisfies $n$ linearly independent active constraints. Furthermore, all $m$ equality constraints are active. Thus, $x$ 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 $\sum_{i=1}^{n} A_{i}x_{i} = b$ 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 rows, 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.
CHAPTER 2. THE GEOMETRY OF LINEAR PROGRAMMING

Procedure for constructing basic solutions

1. Choose $m$ linearly independent columns $A_{B(1)}, \ldots, A_{B(m)}$.
2. Let $x_i = 0$ for all $i \neq B(1), \ldots, B(m)$.
3. Solve the system of $m$ equations $Ax = b$ for the unknowns $x_{B(1)}, \ldots, x_{B(m)}$.

If a basic solution constructed according to this procedure is nonnegative, then it is a basic feasible solution. Every basic feasible solution is a basic solution, and can thus be obtained from this procedure.

Definition: If $x$ 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 $A_{B(1)}, \ldots, A_{B(m)}$ 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 $B$ called a basis matrix, which is invertible. We also define a vector $x_B$ with the values of the basic variables:

$$B = [A_{B(1)}, \ldots, A_{B(m)}], \quad x_B = \begin{bmatrix} x_{B(1)} \\ \vdots \\ x_{B(m)} \end{bmatrix}$$

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

$$x_B = B^{-1}b.$$  

Problem 2.2: 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 $B(2) = 2$:

$$B = \begin{bmatrix} 1 & -1 \\ 1 & 2 \end{bmatrix}, \quad x_B = \begin{bmatrix} 1 & -1 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}.$$  

This corresponds to the basic solution $x = (3, 1, 0, 0)$. Since all elements of $x$ are nonnegative, 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 $B(2) = 3$:

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

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 $B(2) = 4$:

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

This corresponds to the basic solution $x = (2, 0, 0, 3)$. Since all elements of $x$ are nonnegative, 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 $B(2) = 3$:

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

This corresponds to the basic solution $x = (0, 5/2, 9/2, 0)$. Since all elements of $x$ are nonnegative, 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 $B(2) = 4$:

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

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 nonnegative, 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.

### 2.1 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, the latter are adjacent.

- In Prob. 2.2, we see that adjacent basis matrices produce adjacent basic solutions.

### 2.2 Degeneracy

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

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

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

**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:** Since the $n - m$ nonbasic variables of a basic solution must be zero, a basic solution is degenerate iff at least one of the basic variables is 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)\textsuperscript{T}: nondegenerate (3 active constraints)
(0 0 1)\textsuperscript{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)\textsuperscript{T}: nondegenerate (3 active constraints)
(0 0 1)\textsuperscript{T}: nondegenerate (3 active constraints)

Remark: For standard-form representations, if a basic solution is degenerate, 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 \( \mathbf{x} \in P \) and
a nonzero vector \( \mathbf{d} \in \mathbb{R}^n \) such that \( \mathbf{x} + t\mathbf{d} \in P \) 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: Notice for \( n > 1 \), that a half space in \( \mathbb{R}^n \) contains a line, but has no extreme
points. The following theorem states that these two properties are equivalent.

Theorem 2.7: Suppose that the polyhedron \( P = \{ \mathbf{x} \in \mathbb{R}^n : \mathbf{a}_i^\top \mathbf{x} \geq b_i, i = 1, \ldots, m \} \)
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 \( \{ \mathbf{a}_1, \ldots, \mathbf{a}_m \} \) contains \( n \) linearly independent vectors.
2.K. EXISTENCE OF EXTREME POINTS

Figure 2.5: (a) \( P \) contains a line; (b) \( Q \) does not contain a line.

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 proper 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, its boundary 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 \( a_j \) 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 these movements are always confined within \( P \), such a point is a basic feasible solution (extreme point).

\( (a) \Rightarrow (c) \) If \( P \) has an extreme point \( x \), then by definition there exists \( n \) linearly independent active constraints at \( x \), corresponding to \( n \) linearly independent active constraint vectors \( a_i \).

\( (c) \Rightarrow (b) \) 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.
CHAPTER 2. THE GEOMETRY OF LINEAR PROGRAMMING

2.L Optimality of extreme points

**Theorem 2.8**: Consider the linear programming problem of minimizing $c^T x$ 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 that 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 $c^T x$. Consider the polyhedron $Q = \{ x \in \mathbb{R}^n : Ax \geq b, c^T x = v \}$, 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 $x^*$. If $x^*$ were not also an extreme point of $P$, there would exist points $y, z \in P$ distinct from $x^*$ and some $t \in (0, 1)$ such that $x^* = ty + (1-t)z$ and hence $v = c^T x^* = tc^T y + (1-t)c^T z \geq tv + (1-t)v = v$ since $v$ is the optimal cost. Then $c^T y = c^T z = v$, so that $y$ and $z$ would both belong to $Q$. But this would contradict the fact that $x^*$ is an extreme point of $Q$. Thus the optimal solution $x^*$ 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 $c^T x$ 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 $P$ has an optimal extreme point.

**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 $c^T x$ 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!
Consider the standard-form problem

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

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 away 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,

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

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 \( x_B > 0 \), yielding \( x_B + td_B > 0 \) 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 and \( d_{B(i)} \) 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 nonnegativity constraints \( x_i \geq 0 \).

Q. What kind of solution occurs at point \( E \)?

nondegenerate basic feasible solution

Q. What are the basic and nonbasic variables at \( E \)?

nonbasic: \( x_1 \) and \( x_3 \) basic: \( x_2 \), \( x_4 \), \( x_5 \)

Q. Consider the direction obtained by increasing \( x_1 \) while holding \( x_3 \) zero. Is this a basic direction? Yes.
Q. Is it also feasible? Yes.

Q. What kind of solution occurs at point $F$? Degenerate basic feasible solution.

Q. Give examples of basic and nonbasic variables at $F$.
   - nonbasic: $x_3$ and $x_4$  basic: $x_1$, $x_2$, $x_5$
   - nonbasic: $x_3$ and $x_5$  basic: $x_1$, $x_2$, $x_4$
   - nonbasic: $x_4$ and $x_5$  basic: $x_1$, $x_2$, $x_3$

Q. Suppose the nonbasic variables are $x_3$ and $x_5$. Consider the direction obtained by increasing $x_3$ while holding the other nonbasic variable $x_5 = 0$ ($d_3 = 1$, $d_5 = 0$). Is this a basic direction? Yes.

Q. 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 $c_B = (c_{B(1)}, \ldots, c_{B(m)})$ be the basic cost vector. 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 change in 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 \( x_j \):

\[
\bar{c}_j = c_j - c_B^T B^{-1} A_j.
\]

**Problem 3.1:** Consider the linear programming problem

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

Since \( A_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \) and \( A_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \) are linearly independent, 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

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

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

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

Then the reduced cost of \( x_3 \) is

\[
\bar{c}_3 = c_3 + c_B^T d_B = c_3 + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \begin{bmatrix} -3/2 \\ 1/2 \end{bmatrix} = -\frac{3}{2} c_1 + \frac{1}{2} c_2 + c_3.
\]

**Definition:** Let \( e_i \) denote the \( i \)th unit vector.

**Remark:** What is the reduced cost of a basic variable? We find

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

That is, the reduced cost of every basic variable is zero, so we never need to compute the reduced cost of a basic variable.

**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 \( \mathbf{x} \) associated with a basis matrix \( \mathbf{B} \), and let \( \mathbf{c} \) be the corresponding reduced cost vector. Then

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

ii) if \( \mathbf{x} \) is optimal and nondegenerate, then \( \mathbf{c} \geq 0 \).

Proof:
Let \( \mathbf{y} \) be an arbitrary feasible solution. The difference vector \( \mathbf{d} = \mathbf{y} - \mathbf{x} \) satisfies \( A\mathbf{d} = Ax - Ay = b - b = 0 \). That is,

\[
B\mathbf{d}_B + \sum_{j \in N} A_jd_j = 0,
\]
where \( N \) is the set of nonbasic indices. Thus

\[
\mathbf{d}_B = -\sum_{j \in N} B^{-1}A_jd_j.
\]

For every \( j \in N \) we know that the nonbasic variable \( x_j = 0 \) and since \( y_j \geq 0 \), we see that \( d_j \geq 0 \) for all \( j \in N \). The change in the cost by moving from \( \mathbf{x} \) to \( \mathbf{y} \) is therefore

\[
\mathbf{c}^\top \mathbf{y} - \mathbf{c}^\top \mathbf{x} = \mathbf{c}^\top \mathbf{d} = \mathbf{c}_B^\top \mathbf{d}_B + \sum_{j \in N} c_jd_j = \sum_{j \in N} (c_j - \mathbf{c}_B^\top B^{-1}A_j)d_j = \sum_{j \in N} \hat{c}_jd_j.
\]

Thus, \( \hat{c}_j \) is the rate of cost change along the \( j \)th basic direction.

i) Given \( \mathbf{c} \geq 0 \), we then see that

\[
\mathbf{c}^\top \mathbf{y} - \mathbf{c}^\top \mathbf{x} = \sum_{j \in N} \hat{c}_j d_j \geq 0.
\]
That is, \( \mathbf{x} \) is an optimal basic feasible solution.

ii) Suppose that \( \mathbf{x} \) is a nondegenerate basic feasible solution and that \( \hat{c}_j < 0 \) for some \( j \). Since the reduced cost of a basic variable is always zero, \( x_j \) must be a nonbasic variable. Since \( \mathbf{x} \) is nondegenerate and \( \hat{c}_j < 0 \), the \( j \)th basic direction is a feasible direction of cost decrease and therefore \( \mathbf{x} \) is not optimal.

**Remark**: If \( \mathbf{x} \) is a degenerate optimal basic feasible solution, it is still possible that \( \hat{c}_j < 0 \) 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. nonnegativity of the reduced costs: $\bar{c} \geq 0$.

Remark: The basic solution corresponding to an optimal basis matrix is feasible and satisfies the optimality conditions. It is therefore an optimal solution. On the other hand, in the degenerate case, having an optimal basic solution does not necessarily mean that the reduced costs are nonnegative.

3.A Development of the simplex method

For now, assume that every basic feasible solution is nondegenerate. Suppose that we are at a basic feasible solution $x$, for which we have computed the reduced costs $\bar{c}_j$ 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 $\bar{c}_j$ 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: In moving along this direction $d$, the nonbasic variable $x_j$ becomes positive, while all other nonbasic variables are fixed 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 \( \mathbf{d} \geq \mathbf{0} \), then \( \mathbf{x} + t\mathbf{d} \geq \mathbf{0} \) for all \( t \geq 0 \). The vector \( \mathbf{x} + t\mathbf{d} \) 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 \( \mathbf{d} \):

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

Note that \( t^* \) is always nonnegative since \( x_B(i) \) is nonnegative for a basic feasible solution. Furthermore, at a nondenerate basic feasible solution \( x_B(i) > 0 \) and hence \( t^* > 0 \).

**Remark:** In general, once \( t^* \) is determined, assuming it is finite, we can move to the new feasible solution \( y = x + t^* \mathbf{d} \). Since \( x_j = 0 \) and \( d_j = 1 \), we have \( y_j = t^* > 0 \). Let \( \ell \) be a minimizing index:

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

In particular,

\[
d_B(\ell) < 0, \quad \text{and} \quad x_B(\ell) + t^*d_B(\ell) = 0.
\]

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 \( \mathbf{B} \) with

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

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

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

We say that \( x_B(\ell) \) (or \( A_B(\ell) \)) **exits the basis** and \( x_j \) (or \( A_j \)) **enters the basis**.

- Let us revisit Problem 3.1 and consider the basic feasible solution \( \mathbf{x} = (1, 1, 0, 0) \) for which we found the reduced cost of the nonbasic variable \( x_3 \) is \( \tilde{c}_3 = -3/2c_1 + 1/2c_2 + c_3 \). Suppose that \( \mathbf{c} = (2, 0, 0, 0) \). Then \( \tilde{c}_3 = -3 \) is negative, so we would want to move in the corresponding basic direction \( \mathbf{d} = (-3/2, 1/2, 1, 0) \). As \( t \) increases, the only component of \( \mathbf{x} + t\mathbf{d} \) 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 \( y = \mathbf{x} + 2\mathbf{d}/3 = (0, 4/3, 2/3, 0) \). At this point, the variable \( x_3 \) has entered the basis.
and the variable $x_1$ has exited. Note that the columns $A_2 = (1, 0)$ and $A_3 = (1, 3)$ are linearly independent and they therefore form a new basis matrix

$$B = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix}.$$  

Notice also that $y$ is the basic feasible solution corresponding to $B$:

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

**Theorem 3.2:**

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

(ii) The vector $y = x + t^*d$ is a basic feasible solution associated with the basis matrix $B$.

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 $B^{-1}$, the vectors $B^{-1}A_B(i)$ would be seen to be 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 $x$. 


3.C. THE SIMPLEX METHOD FOR DEGENERATE PROBLEMS

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

\[
t^* = \min \frac{x_B(i)}{u_i}.
\]

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 \( c^Td < 0 \), 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 \( x_{B(\ell)} = 0 \) and \( d_{B(\ell)} < 0 \). Nevertheless, we can still define the new basis \( \bar{B} \), by replacing \( A_{B(\ell)} \) 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 3.1 we visualize the feasible set in standard form, with \( n - m = 2 \), by standing on the two-dimensional plane defined by \( Ax = b \).
Q. What kind of point is \( x \)?

Degenerate basic feasible solution.

Q. Assume that \( x_4 \) and \( x_5 \) are the nonbasic variables. What are the corresponding basic directions?

\( g \) and \( f \).

Q. What are the corresponding values of \( t^* \)?

0 and 0.

Q. If we perform a change of basis, with \( x_4 \) entering and \( x_6 \) exiting, what are the new nonbasic variables?

\( x_5 \) and \( x_6 \).

Q. What are their corresponding basic directions?

\( h \) and \( -g \).

Q. 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 \( x \).

Figure 3.1:

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, up to $n - m$ dot products, followed by another linear solve):

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

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 $-t^* \bar{c}_j$. 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 $\mathcal{O}(m^3)$. If we assume that $n \gg m$, the computational cost of computing all nonbasic reduced costs is $\mathcal{O}(nm)$. The overall computational cost of a naive implementation is thus $\mathcal{O}(m^3 + nm)$.

3.D.2  Revised simplex method

Typically, 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 $\mathcal{O}(m^3 + nm)$ 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

$$B = \begin{bmatrix} A_{B(1)}, \ldots, A_{B(m)} \end{bmatrix}$$
CHAPTER 3. THE SIMPLEX METHOD

and

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

Note that

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

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 $B^{-1}\overline{B}$:

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. Note that this is equivalent to adding $-1/u_\ell$ times the $\ell$th row to itself.

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

$$QB^{-1}\overline{B} = 1.$$  

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 $\bar{c}_j = c_j - p^\top A_j$. If they are all nonnegative, the current basic feasible solution is optimal, and the algorithm terminates; else, choose some $j$ for which $\bar{c}_j < 0$.

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

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 $i \neq \ell$.

6. Form the $m \times (n+1)$ 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 for the vector $u = B^{-1}A_j$, corresponding to the variable entering the basis. This special column is called the pivot column. If the $\ell$th basic variable exits the basis, the $\ell$th row 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 terminates.

We now extend this matrix into a tableau by including a zeroth column $B^{-1}b$ containing the components of the basic variable $x_B$. One way to remember the form the tableau is to multiply 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 the negative of the cost $-c_B^\top x_B$ in the zeroth column followed by the reduced costs $\bar{c}_j$ of each of the $n$ variables in subsequent columns:

\[
\begin{array}{c|cccc}
-c_B^\top x_B & \bar{c}_1 & \ldots & \bar{c}_n \\
B^{-1}b & B^{-1}A_1 & \ldots & B^{-1}A_n \\
\end{array}
\]
**Remark:** As we now show, the reason for the minus sign in the expression $-c_B^T x_B$ tabulated in the zeroth column and zeroth row is for consistency with the reduced costs $c_j = c_j - c_B^T B^{-1} A_j$: 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 $x_j$ 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 $p^T = c_B^T B^{-1}$. Notice that $p^T [b | A]$ is a linear combination of the columns of $[b | A]$. Since $[B^{-1} b | B^{-1} A]$ is also a linear combination of the columns of $[b | A]$, adding a multiple of the pivot row to the zeroth row yields $[0 | c^T]$ minus some linear combination of the columns of $[b | A]$:
\[
[0 | c^T] - p^T [b | A].
\]
Our update rule is specifically chosen to annihilate the $j$th entry in the resulting zeroth row:
\[c_j - p^T A_j = 0.\]
Since $B^{-1} A_{B(i)} = e_i$, adding a multiple of the pivot row to the zeroth row will not change the reduced costs of the other basic variables $x_{B(i)} (i \neq \ell)$ from their previous values of 0. That is, the reduced cost of every one of the new basic variables is zero:
\[c_B^T = c_B^T - p^T B = 0.
\]
From this equation, 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.

The full simplex tableau algorithm can now be summarized. 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 $\tilde{c}^T = c^T - c_B^T B^{-1} A$ are entered in the zeroth row of the tableau. A single iteration of the **full simplex tableau** involves these four steps:

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 $\tilde{c}_j < 0$. 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. D. IMPLEMENTATION OF THE SIMPLEX METHOD

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 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

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

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

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

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 3.2. 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 \( \bar{c} = c \):

\[
\begin{array}{ccccccc}
\text{x}_1 & \text{x}_2 & \text{x}_3 & \text{x}_4 & \text{x}_5 & \text{x}_6 \\
0 & -10 & -12 & -12 & 0 & 0 & 0 \\
\hline
x_4 & = & 20 & 1 & 2 & 2 & 1 & 0 & 0 \\
x_5^1 & = & 20 & 2 & 1 & 2 & 0 & 1 & 0 \\
x_6 & = & 20 & 2 & 2 & 1 & 0 & 0 & 1 \\
\end{array}
\]

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 an dagger. We then
apply elementary operations to the matrix to reduce the pivot column to \((0, 0, 1, 0)\):

\[
\begin{array}{cccccc}
\text{x}\_1 & \text{x}\_2 & \text{x}\_3 & \text{x}\_4 & \text{x}\_5 & \text{x}\_6 \\
100 & 0 & -7 & -2 & 0 & 5 & 0 \\
x\_4 & = & 10 & 0 & 3/2 & 1 & 1 & -1/2 & 0 \\
x\_1 & = & 10 & 1 & 1/2 & 1 & 0 & 1/2 & 0 \\
x\_6\text{}\uparrow & = & 0 & 0 & 1 & -1 & 0 & -1 & 1 \\
\end{array}
\]

This corresponds to the degenerate vertex \(D = (10, 0, 0)\) of the original problem depicted in Figure 3.2. 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 \((0, 0, 1)\) in the pivot column. This row reduction accomplishes a change of basis but leaves us still at the point \(D = (10, 0, 0)\):

\[
\begin{array}{cccccc}
\text{x}\_1 & \text{x}\_2 & \text{x}\_3 & \text{x}\_4 & \text{x}\_5 & \text{x}\_6 \\
100 & 0 & 0 & -9 & 0 & -2 & 7 \\
x\_4\text{}\uparrow & = & 10 & 0 & 5/2 & 1 & 1 & -3/2 \\
x\_1 & = & 10 & 1 & 3/2 & 0 & 1 & -1/2 \\
x\_2 & = & 0 & 0 & 1 & -1 & 0 & -1 & 1 \\
\end{array}
\]

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:

\[
\begin{array}{cccccc}
\text{x}\_1 & \text{x}\_2 & \text{x}\_3 & \text{x}\_4 & \text{x}\_5 & \text{x}\_6 \\
136 & 0 & 0 & 0 & 18/5 & 8/5 & 8/5 \\
x\_3 & = & 4 & 0 & 0 & 1 & 2/5 & 2/5 & -3/5 \\
x\_1 = & 4 & 1 & 0 & 0 & -3/5 & 2/5 & 2/5 \\
x\_2 & = & 4 & 0 & 1 & 0 & 2/5 & -3/5 & 2/5 \\
\end{array}
\]

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

\[
B = \begin{bmatrix} 2 & -1 \\ 1 & -1 \end{bmatrix}, \quad x_B = \frac{1}{-1} \begin{bmatrix} -1 & 1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} 6 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix},
\]
3.D. IMPLEMENTATION OF THE SIMPLEX METHOD

Figure 3.2: Simplex iterations.

and

\[
B^{-1}A = \frac{1}{-1} \begin{bmatrix}
-1 & 1 \\
-1 & 2
\end{bmatrix}
\begin{bmatrix}
2 & -1 & 1 & 0 \\
1 & -1 & 0 & -1
\end{bmatrix}
= \begin{bmatrix}
1 & 0 & 1 & 1 \\
0 & 1 & 1 & 2
\end{bmatrix}.
\]

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:

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

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

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

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3^* )</th>
<th>( x_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-6)</td>
<td>(0)</td>
<td>(0)</td>
<td>(-1)</td>
</tr>
<tr>
<td>( x_1 )</td>
<td>(4)</td>
<td>(0)</td>
<td>(1)</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>(2)</td>
<td>(0)</td>
<td>(1)</td>
</tr>
</tbody>
</table>

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

\[
\begin{array}{c|cccc}
   & x_1 & x_2 & x_3^* & x_4^* \\
\hline
x_1 & -4 & 0 & 1 & 0 \\
x_2 & 2 & 1 & -1 & 0 \\
x_3 & 2 & 0 & 1 & 1 \\
\end{array}
\]
CHAPTER 3. THE SIMPLEX METHOD

Remark: The entering decision variable always has 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 with the lowest index that has a negative reduced cost. If we also choose the exiting variable to be the variable with the lowest index \( i \) that achieves the minimum ratio \( x_{B(i)}/u_i \), the simplex method never cycles and will always terminate after a finite number of iterations. This widely used pivot rule is known as Bland’s rule.

Problem 3.4: Consider the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad -3x_1 + 80x_2 - 14x_3 + 24x_4 - 12x_7 \\
\text{subject to} & \quad \frac{1}{4} x_1 - 8x_2 - x_3 + 9x_4 + x_5 = 0, \\
& \quad \frac{1}{2} x_1 - 12x_2 - \frac{1}{2} x_3 + 3x_4 + x_6 = 0, \\
& \quad x_3 + x_7 = 4, \\
& \quad x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0.
\end{align*}
\]

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 \(-c_B^T x_B = 48\) followed by the elements of the reduced cost vector \([-3, 80, -14, 24, 0, 0, -12] - [0, 0, -12] A):

\[
\begin{bmatrix}
48 & -3 & 80 & -2 & 24 & 0 & 0 & 0 \\
0 & 1/4 & -8 & -1 & 9 & 1 & 0 & 0 \\
0 & 1/2 & -12 & -1/2 & 3 & 0 & 1 & 0 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 
\end{bmatrix}
\]

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

\[
\begin{bmatrix}
48 & 0 & -16 & -14 & 132 & 12 & 0 & 0 \\
0 & 1 & -32 & -4 & 36 & 4 & 0 & 0 \\
0 & 0 & 4 & 3/2 & -15 & -2 & 1 & 0 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 
\end{bmatrix}
\]

\[
\begin{bmatrix}
48 & 0 & 0 & -8 & 72 & 4 & 4 & 0 \\
0 & 1 & 0 & 8 & -84 & -12 & 8 & 0 \\
0 & 0 & 1 & 3/8 & -15/4 & -1/2 & 1/4 & 0 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 
\end{bmatrix}
\]

\[
\begin{bmatrix}
48 & 1 & 0 & 0 & -12 & -8 & 12 & 0 \\
0 & 1/8 & 0 & 1 & -21/2 & -3/2 & 1 & 0 \\
0 & -3/64 & 1 & 0 & 3/16 & 1/16 & -1/8 & 0 \\
4 & -1/8 & 0 & 0 & 21/2 & 3/2 & -1 & 1 
\end{bmatrix}
\]

\[
\begin{bmatrix}
48 & -2 & 64 & 0 & 0 & -4 & 4 & 0 \\
0 & -5/2 & 56 & 1 & 0 & 2 & -6 & 0 \\
0 & -1/4 & 16/3 & 0 & 1 & 1/3 & -2/3 & 0 \\
4 & 5/2 & -56 & 0 & 0 & -2 & 6 & 1 
\end{bmatrix}
\]

\[
\begin{bmatrix}
256/5 & 0 & 96/5 & 0 & 0 & -28/5 & 44/5 & 4/5 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
2/5 & 0 & -4/15 & 0 & 1 & 2/15 & -1/15 & 1/10 \\
8/5 & 1 & -112/5 & 0 & 0 & -4/5 & 12/5 & 2/5 
\end{bmatrix}
\]

\[
\begin{bmatrix}
68 & 0 & 8 & 0 & 42 & 0 & 6 & 5 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
3 & 0 & -2 & 0 & 15/2 & 1 & -1/2 & 3/4 \\
4 & 1 & -24 & 0 & 6 & 0 & 0 & 2 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
3 & 0 & -2 & 0 & 15/2 & 1 & -1/2 & 3/4 \\
4 & 1 & -24 & 0 & 6 & 0 & 0 & 2 \
\end{bmatrix}
\]
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

\[
\begin{array}{cccccccc}
\text{x}_1 & \text{x}_2 & \text{x}_3 & \text{x}_4 & \text{x}_5 & \text{x}_6 & \text{x}_7 \\
48  & -2  & 64  & 0  & 0  & -4  & 4  & 0 \\
0  & -5/2 & 56  & 1  & 0  & 2  & -6 & 0 \\
0  & -1/4 & 16/3 & 0  & 1  & 1/3 & -2/3 & 0 \\
4  & 5/2  & -56  & 0  & 0  & -2  & 6  & 1 \\
\end{array}
\]

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

\[
\begin{array}{cccccccc}
\text{x}_1 & \text{x}_2 & \text{x}_3 & \text{x}_4 & \text{x}_5 & \text{x}_6 & \text{x}_7 \\
48  & -7  & 176 & 2  & 0  & 0  & -8  & 0 \\
0  & -5/4 & 28  & 1/2 & 0  & 1  & -3 & 0 \\
0  & 1/6  & -4  & -1/6 & 1  & 0  & 1/3 & 0 \\
4  & 0  & 0  & 1  & 0  & 0  & 0  & 1 \\
\end{array}
\]

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 \textit{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

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

satisfies \(b \geq 0\).
Next, introduce a vector \( y \in \mathbb{R}^m \) of artificial variables and consider the auxiliary problem

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

**Remark:** If \( x^* \) is a feasible solution to the original problem, \((x, y) = (x^*, 0)\) yields an optimal 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 then a feasible solution to the original problem. If the associated basis matrix \( B \) contains only columns of \( A \), the columns in the tableau corresponding to the artificial variables \( y \) can simply be dropped.

**Remark:** A basic feasible solution to the auxiliary problem is given by \((x, y) = (0, b)\), with an identity basis matrix. Since all elements of \( c_B \) are one, the reduced cost of a nonbasic variable \( x_j \) is just \(-\sum_{i=1}^{m} A_{i,j} \), noting that \( c_j = 0 \) in the auxiliary problem.

**Remark:** 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 nonnegative. 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

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

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

\[
\begin{array}{cccccccc}
& x_1^* & x_2 & x_3 & x_4 & x_5 & x_6 \\
-8 & -3 & 2 & -1 & 1 & 0 & 0 \\
x_5 = 6 & 2 & -1 & 1 & 0 & 1 & 0 \\
x_6 = 2 & 1 & -1 & 0 & -1 & 0 & 1
\end{array}
\]
Two iterations of the simplex method then lead to an optimal zero-cost solution of the auxiliary problem:

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

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.

**Remark:** One can sometimes reduce the number of required artificial variables by exploiting the structure of \(A\). If a column \(A_j\) is a positive multiple of a unit vector \(e_k\), we can use the corresponding decision variable \(x_j\) as a proxy basic variable and remove the artificial variable \(y_k\) from the constraints. However, since \(x_j\) is merely acting as a proxy for the basic variable \(x_k\), all \(m\) elements of \(c_B\) are still one, and so \(x_j\) has the same reduced cost \(-\sum_{i=1}^{m} A_{i,j}\) and \(c_Bx_B\) has the same cost as without this optimization.

**Remark:** In a standard-form problem with \(m\) slack (not surplus) variables, artificial variables can thus be completely eliminated: the auxiliary cost function reduces to zero (the optimal value), leaving one with an obvious basic feasible solution, as we have previously observed.

**Problem 3.5:** Let us optimize our search for an initial basic feasible solution for Prob 3.3 by exploiting the fact that \(A_3 = e_1\): introduce a single artificial variable \(x_5\) to form the auxiliary problem

\[
\begin{align*}
\text{minimize} & \quad x_5 \\
\text{subject to} & \quad 2x_1 - x_2 + x_3 = 6, \\
& \quad x_1 - x_2 - x_4 + x_5 = 2, \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\end{align*}
\]
3.E. FINDING AN INITIAL BASIC FEASIBLE SOLUTION

Since the artificial variable $x_5$ is no longer in our basis, we can drop the last column to obtain the same truncated tableau as before.

**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 for 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 $\ell$th 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 3.6:** Find a basic feasible solution for the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad -5x_1 - x_2 + 12x_3 \\
\text{subject to} & \quad 3x_1 + 2x_2 + 2x_3 = 10, \\
& \quad 5x_1 + 3x_2 + x_3 + x_4 = 16, \\
& \quad x_1 + x_2 + x_3 - 2x_5 = 5, \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\end{align*}
\]

**Step 1.** Multiply every constraint with a negative right-hand side by $-1$, so that $b \geq 0$. In this case, the constraints are already in this form.
Step 2. In order to find a basic feasible solution, the artificial variables $x_6$ and $x_7$ are introduced to form the auxiliary problem

\[
\begin{align*}
&\text{minimize} & 3x_1 + 2x_2 + 2x_3 + x_6 + x_7 \\
&\text{subject to} & 3x_1 + 3x_2 + x_3 + x_4 &= 16, \\
& & x_1 + x_2 + x_3 - 2x_5 + x_7 &= 5, \\
& & x_1, x_2, x_3, x_4, x_5, x_6, x_7 &\geq 0.
\end{align*}
\]

We then perform simplex iterations, starting from the initial tableau:

<table>
<thead>
<tr>
<th>$x_6$</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>-31</td>
<td>-9</td>
<td>-6</td>
<td>-4</td>
<td>-1</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

$\begin{array}{l}
x_6 = 10 \\
x_4^* = 16 \\
x_7 = 5
\end{array}$

<table>
<thead>
<tr>
<th>$x_6^*$</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>-11/5</td>
<td>0</td>
<td>-3/5</td>
<td>-11/5</td>
<td>4/5</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

$\begin{array}{l}
x_6 = 2/5 \\
x_4 = 16/5 \\
x_7 = 9/5
\end{array}$

<table>
<thead>
<tr>
<th>$x_2$</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>-1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>-1</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

$\begin{array}{l}
x_2 = 2 \\
x_4^* = 2 \\
x_7 = 1
\end{array}$

<table>
<thead>
<tr>
<th>$x_2^*$</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>0</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3/2</td>
<td>0</td>
</tr>
</tbody>
</table>

$\begin{array}{l}
x_2 = 5 \\
x_4 = 1 \\
x_7^* = 0
\end{array}$

After 3 iterations, we have found an optimal (zero-cost) solution of the auxiliary problem. However, we need to drive the variable $x_7$ out of the final basis, with $x_1$ entering, dropping the columns associated with the artificial variables:

<table>
<thead>
<tr>
<th>$x_2$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>-6</td>
</tr>
</tbody>
</table>

$\begin{array}{l}
x_2 = 5 \\
x_4 = 1 \\
x_1 = 0
\end{array}$

We have thus obtain a basic feasible solution $(0, 5, 0, 1, 0)$ and corresponding values of $B^{-1}A$. 
Problem 3.7: Find a basic feasible solution for the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad 2x_1 + 6x_2 + x_3 + x_4 \\
\text{subject to} & \quad x_1 + 2x_2 + x_4 = 6, \\
& \quad x_1 + 2x_2 + x_3 + x_4 = 7, \\
& \quad x_1 + 3x_2 - x_3 + 2x_4 = 7, \\
& \quad x_1 + x_2 + x_3 = 5, \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]

**Step 1.** Multiply every constraint with a negative right-hand side by \(-1\), so that \(b \geq 0\). In this case, there is no need to change the constraints.

**Step 2.** In order to find a basic feasible solution, the artificial variables \(x_5, \ldots, x_8\) are introduced to form the auxiliary problem

\[
\begin{align*}
\text{minimize} & \quad x_5 + x_6 + x_7 + x_8 \\
\text{subject to} & \quad x_1 + 2x_2 + x_4 + x_5 = 6, \\
& \quad x_1 + 2x_2 + x_3 + x_4 + x_6 = 7, \\
& \quad x_1 + 3x_2 - x_3 + 2x_4 + x_7 = 7, \\
& \quad x_1 + x_2 + x_3 + x_8 = 5, \\
& \quad x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8 \geq 0.
\end{align*}
\]

\[
\begin{array}{cccccccccc}
\hline
& x_1^* & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\
\hline
x_5 & -25 & -4 & -8 & -1 & -4 & 0 & 0 & 0 & 0 \\
x_6 & 6 & 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\
x_7 & 7 & 1 & 3 & -1 & 2 & 0 & 0 & 1 & 0 \\
x_8 & 5 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
\hline
\end{array}
\]

\[
\begin{array}{cccccccccc}
\hline
& x_1 & x_2^* & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\
\hline
x_5 & 1 & 0 & 1 & -1 & 1 & 1 & 0 & 0 & -1 \\
x_6 & 2 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & -1 \\
x_7 & 2 & 0 & 2 & -2 & 2 & 0 & 0 & 1 & -1 \\
x_1 & 5 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
\hline
\end{array}
\]

\[
\begin{array}{cccccccccc}
\hline
& x_1 & x_2 & x_3^* & x_4 & x_5 & x_6 & x_7 & x_8 \\
\hline
x_2 & -1 & 0 & 0 & -1 & 0 & 4 & 0 & 0 & 0 \\
x_6 & 1 & 0 & 1 & -1 & 1 & 1 & 0 & 0 & -1 \\
x_7 & 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\
x_1 & 4 & 1 & 0 & 2 & -1 & -1 & 0 & 0 & 2 \\
\hline
\end{array}
\]
We have arrived at an optimal solution of the auxiliary problem. However, we need to drive out the third variable in the basis, \( x_7 \), since it is an artificial variable. Since the third elements of \( B^{-1}A_1, \ldots, B^{-1}A_4 \) are all zero, we drop the third row, along with the columns associated with the artificial variables (since they do not appear in the final basis):

\[
\begin{array}{cccccccc}
& x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\
0 & 0 & 0 & 0 & 0 & 3 & 1 & 0 & 0 \\
x_2 &= 2 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & -1 \\
x_3 &= 1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\
x_7 &= 0 & 0 & 0 & 0 & 0 & -2 & 0 & 1 & 1 \\
x_1 &= 2 & 1 & 0 & 0 & -1 & 1 & -2 & 0 & 2 \\
\end{array}
\]

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 constraints as needed by \(-1\) so that \( b \geq 0 \).

2. If a basic feasible solution is not known, introduce nonnegative artificial variables \( y_1, \ldots, y_m \) 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 and the tabulated values of \( B^{-1}A \).

3. Apply the simplex method to the original problem.

**Problem 3.8:** Use the two-phase simplex method to determine the initial tableau for the linear programming problem

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

**Phase I:**

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

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

**Step 2.** In order to find a basic feasible solution, the artificial variable \( x_4 \) is introduced to form the auxiliary problem

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

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>-1</td>
<td>-2</td>
<td>1</td>
</tr>
<tr>
<td>( x_1 )</td>
<td>( x_2 )</td>
<td>( x_3 )</td>
<td>( x_4 )</td>
</tr>
<tr>
<td>-5</td>
<td>-1</td>
<td>-2</td>
<td>1</td>
</tr>
<tr>
<td>( x_4 = 4 )</td>
<td>( 1 )</td>
<td>( 1 )</td>
<td>0</td>
</tr>
<tr>
<td>( x_4 = 1 )</td>
<td>( 0 )</td>
<td>( 1 )</td>
<td>-1</td>
</tr>
</tbody>
</table>
CHAPTER 3. THE SIMPLEX METHOD

Phase II:

In the solution to the auxiliary problem, we may simply drop the artificial variable $x_4$ since it is not in the basis. The reduced costs of the basic variables $x_1$ and $x_2$ are zero. The reduced cost of the nonbasic variable $x_3$ is $c_3 - c_B^t B^{-1} A_3 = -1 - (1,0) \cdot (1, -1) = -2$.

The optimal solution is $(0, 4, 3)$, with optimal cost $-3$.

3.G Application to transportation problems

Consider the following transportation problem. We want to ship identical items from $M$ warehouses to $N$ stores. Warehouse $i$ has $a_i$ units available and store $j$ requires $b_j$ units. If the total supply $\sum_{i=1}^M a_i$ equals the total demand $\sum_{j=1}^N b_j$, and the cost to ship an item from warehouse $i$ to store $j$ is $c_{ij}$, how many items $x_{ij}$ should we ship from warehouse $i$ to store $j$ in order to minimize the overall transportation cost?

Problem 3.9: Given $M = 3$ warehouses and $N = 2$ stores, it is convenient to relabel $(x_{11}, x_{12}, x_{21}, x_{22}, x_{31}, x_{32})$ as $(x_1, x_2, x_3, x_4, x_5, x_6)$. Suppose the corresponding cost vector $(c_1, c_2, c_3, c_4, c_5, c_6)$ is $(1, 5, 1, 3, 1, 4)$ (in dollars), with $a_1 = 20$, $a_2 = 3$, $a_3 = 12$, $b_1 = 15$, $b_2 = 20$. How many units should each warehouse ship to each store to minimize the total transportation cost?
Since demand equals supply, each warehouse must ship all available units. The total outflow from warehouse \( i \) is \( \sum_{j=1}^{2} x_{ij} = a_i \) and the total inflow into store \( j \) is \( \sum_{i=1}^{3} x_{ij} = b_j \). This leads to the linear programming problem

\[
\begin{align*}
\text{minimize} \quad & x_1 + 5x_2 + x_3 + 3x_4 + x_5 + 4x_6 \\
\text{subject to} \quad & x_1 + x_2 = 20, \\
& x_3 + x_4 = 3, \\
& x_5 + x_6 = 12, \\
& x_1 + x_3 + x_5 = 15, \\
& x_2 + x_4 + x_6 = 20, \\
& x_1, x_2, x_3, x_4, x_5, x_6 \geq 0.
\end{align*}
\]

Using the two-phase simplex method, the problem can be solved as follows.

**Phase I:**

**Step 1.** Multiply every constraint with a negative right-hand side by \(-1\), so that \( b \geq 0 \).
In this case, the constraints are already in this form.

**Step 2.** In order to find a basic feasible solution, the artificial variables \( x_7, \ldots, x_{11} \) are introduced to form the auxiliary problem

\[
\begin{align*}
\text{minimize} \quad & x_7 + x_8 + x_9 + x_{10} + x_{11} \\
\text{subject to} \quad & x_1 + x_2 + x_3 + x_4 = 20, \\
& x_5 + x_6 + x_7 = 3, \\
& x_1 + x_3 + x_5 + x_8 = 12, \\
& x_2 + x_4 + x_6 + x_9 = 15, \\
& x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11} \geq 0.
\end{align*}
\]

\[
\begin{array}{cccccccccc}
& x_1^* & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\
-70 & -2 & -2 & -2 & -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 \\
x_7=20 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
x_8=3 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
x_9=12 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
x_{10}=15 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
x_{11}=20 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
\hline
\end{array}
\]

\[
\begin{array}{cccccccccc}
& x_1 & x_2^* & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\
-40 & 0 & -2 & 0 & -2 & 0 & -2 & 0 & 0 & 0 & 2 & 0 \\
x_7^+=5 & 0 & 1 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 \\
x_8=3 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
x_9=12 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
x_1=15 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
x_{11}=20 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
\hline
\end{array}
\]
Phase II:

\[
\begin{array}{cccccccccccc}
   & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\
   \hline
-115 & -2 & 0 & 0 & 0 & 0 & 0 & 1 \\
 x_2= & 20 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
x_3= & 3 & 1 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & -1 & 1 \\
x_4= & 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & -1 \\
x_5= & 12 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
x_{11}= & 15 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 1 & 1 \\
\end{array}
\]
The minimal transportation cost of $97 is realized by the optimal solution $(15, 5, 0, 3, 0, 12)$, where Warehouse 1 ships 15 units to Store 1 and 5 units to Store 2 and Warehouse 2 and 3 ship 3 and 12 units to Store 2, respectively.

**Remark:** The name *simplex method* derives from the following definitions, which extend the notion of a triangle in $\mathbb{R}^2$ dimensions or a tetrahedron in $\mathbb{R}^3$ to arbitrary dimensions:

**Definition:** A *$n$-simplex* is a polyhedron in $\mathbb{R}^n$ that is the convex hull of its $n + 1$ vertices $v_i \in \mathbb{R}^n$, $i = 0, \ldots, n$:

$$\left\{ t_0 v_0 + \ldots + t_n v_n : \sum_{i=0}^{n} t_i = 1, \quad t_i \geq 0, \quad i = 0 \ldots, n \right\}.$$ 

**Definition:** The *standard $n$-simplex* is the $n$-simplex obtained when the vectors $v_i$ are chosen to be the unit vectors $e_i$:

$$\left\{ (t_0, \ldots, t_n) \in \mathbb{R}^{n+1} : \sum_{i=0}^{n} t_i = 1, \quad t_i \geq 0, \quad i = 0 \ldots, n \right\}.$$
Chapter 4

Duality

4.A Introduction

Suppose that \((x_1, x_2, x_3)\) is a feasible solution to the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad 4x_1 + 2x_2 + x_3 \\
\text{subject to} & \quad x_1 - x_2 \geq 3, \\
& \quad 2x_1 + x_2 + x_3 \geq 4, \\
& \quad x_1, x_2, x_3 \geq 0.
\end{align*}
\]

If the problem has an optimal cost, we can find a lower bound for the optimal cost with the following procedure. Let \(y_1\) and \(y_2\) be nonnegative numbers. On summing the first constraint multiplied by \(y_1\) with the second constraint multiplied by \(y_2\), we obtain

\[
y_1(x_1 - x_2) + y_2(2x_1 + x_2 + x_3) \geq 3y_1 + 4y_2.
\]

On rearranging this result, we find

\[
x_1(y_1 + 2y_2) + x_2(-y_1 + y_2) + x_3y_2 \geq 3y_1 + 4y_2.
\]

If we enforce the constraints \(y_1 + 2y_2 \leq 4, -y_1 + y_2 \leq 2,\) and \(y_2 \leq 1,\) we obtain a lower bound to the optimal cost:

\[
4x_1 + 2x_2 + x_3 \geq x_1(y_1 + 2y_2) + x_2(-y_1 + y_2) + x_3y_2 \geq 3y_1 + 4y_2.
\]

The linear programming problem

\[
\begin{align*}
\text{maximize} & \quad 3y_1 + 4y_2 \\
\text{subject to} & \quad y_1 + 2y_2 \leq 4, \\
& \quad -y_1 + y_2 \leq 2, \\
& \quad y_2 \leq 1, \\
& \quad y_1, y_2 \geq 0
\end{align*}
\]

that determines the largest possible value for our lower bound to the optimal cost is known as the dual problem to the original linear programming problem.
Remark: Duality theory can be also motivated as an outgrowth of the Lagrange multiplier method.

- Consider the optimization problem

\[
\begin{align*}
\text{minimize} & \quad x^2 + y^2 \\
\text{subject to} & \quad x + y = 1
\end{align*}
\]

Instead of enforcing the hard constraint \( x + y = 1 \), we allow it to be violated and associate a constant Lagrange multiplier, or price, \( p \) to the amount \( 1 - x - y \) by which it is violated. Instead of minimizing the cost subject to the hard constraint \( x + y = 1 \), we minimize the Lagrangian

\[
L(x, y, p) = x^2 + y^2 + p(1 - x - y)
\]

over all \( x \) and \( y \), without any further constraints:

\[
0 = \frac{\partial L}{\partial x} = 2x - p,
\]

\[
0 = \frac{\partial L}{\partial y} = 2y - p.
\]

The Lagrangian takes on its minimum value at the critical point \((x, y) = (p/2, p/2)\), which depends on \( p \). If we now enforce the constraint \( x + y = 1 \), we see that \( p = 1 \) and \( x = y = 1/2 \). At this value of \( p \), the presence or absence of the hard constraint does not affect the optimal cost: the optimal solution of the unconstrained problem is thus also the optimal solution of the original constrained problem.

In a similar manner, we can associate a price variable with each constraint of a linear programming problem and search for prices under which the presence or absence of the constraints does not affect the optimal cost. The right prices can be found by solving a new linear programming problem that is the dual of the original problem. Consider the standard-form problem

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

which we will call the primal problem, and assume that an optimal solution \( x^* \) exists. Let us introduce a relaxed problem in which the constraint \( Ax = b \) is replaced by a penalty \( p^\top (b - Ax) \), where \( p \) is a price vector with the same length as \( b \). This results in the following problem

\[
\begin{align*}
\text{minimize} & \quad c^\top x + p^\top (b - Ax) \\
\text{subject to} & \quad x \geq 0.
\end{align*}
\]
Let \( g(p) \) be the optimal cost for the relaxed problem. We expect \( g(p) \) to be no larger than the optimal primal cost \( c^\top x^* \):

\[
g(p) = \min_{x \geq 0} [c^\top x + p^\top (b - Ax)] \leq c^\top x^* + p^\top (b - Ax^*) = c^\top x^*,
\]

noting that \( Ax^* = b \). Thus, for every price vector \( p \), the value of \( g(p) \) provides a lower bound to the optimal cost \( c^\top x^* \). The tightest possible such lower bound can be found by solving the problem

\[
\begin{align*}
\text{maximize} & \quad g(p) \\
\text{subject to} & \quad \text{no constraints},
\end{align*}
\]

which is known as the dual problem. One might expect that the tightest lower bound will match the optimal value of the primal problem. Indeed, the main result in duality theory asserts that the optimal value \( g(p) \) of the dual problem equals the optimal value \( c^\top x^* \) of the primal problem. In other words, when the price vector \( p \) is chosen to optimize the dual problem, the option of violating the constraints \( Ax = b \) is of no value. Note that

\[
g(p) = \min_{x \geq 0} [c^\top x + p^\top (b - Ax)] = p^\top b + \min_{x \geq 0} (c^\top - p^\top A)x,
\]

where

\[
\min_{x \geq 0} (c^\top - p^\top A)x = \begin{cases} 0 & \text{if } c^\top - p^\top A \geq 0^\top, \\ -\infty & \text{otherwise}. \end{cases}
\]

In maximizing \( g(p) \), we only need to consider those values of \( p \) for which \( g(p) \) is not \(-\infty\). Hence, the dual problem is equivalent to the LPP

\[
\begin{align*}
\text{maximize} & \quad p^\top b \\
\text{subject to} & \quad p^\top A \leq c^\top.
\end{align*}
\]

**Remark:** If instead of the equality constraint \( Ax = b \), the primal problem had inequality constraints in the form \( Ax \geq b \), they could be replaced by \( Ax - s = b \) and \( s \geq 0 \), with cost function \( c^\top x + 0^\top s \):

\[
[A|-1] \begin{bmatrix} x \\ s \end{bmatrix} = b,
\]

leading to the dual constraints

\[
p^\top [A|-1] \preceq [c^\top |0^\top] ;
\]

that is,

\[
p^\top A \leq c^\top, \quad p \geq 0.
\]
4.B. THE DUAL PROBLEM

Remark: Alternatively, if the vector $\mathbf{x}$ were a free variable, we would use the fact that

$$\min_{\mathbf{x}} (\mathbf{c}^\top - \mathbf{p}^\top \mathbf{A}) \mathbf{x} = \begin{cases} 0 & \text{if } \mathbf{c}^\top - \mathbf{p}^\top \mathbf{A} = 0^\top, \\ -\infty & \text{otherwise,} \end{cases}$$

leading to the constraint $\mathbf{p}^\top \mathbf{A} = \mathbf{c}^\top$ in the dual problem.

4.B The dual problem

Let $\mathbf{A}$ be a matrix with rows $\mathbf{a}_i^\top$ and columns $\mathbf{A}_j$. The objective function $\mathbf{p}^\top \mathbf{b}$ in the dual problem can be equivalently written as $\mathbf{b}^\top \mathbf{p}$. Moreover, by taking a transpose, dual row constraints like $\mathbf{p}^\top \mathbf{A} = \mathbf{c}^\top$ (and inequality versions thereof) can be as rewritten as column constraints like $\mathbf{A}^\top \mathbf{p} = \mathbf{c}$. One then notices that the transformation between the primal and the dual problems can be accomplished by swapping $\mathbf{x}$ and $\mathbf{p}$, swapping $\mathbf{b}$ and $\mathbf{c}$ and taking the transpose of $\mathbf{A}$. For a primal minimization problem with the structure shown on the left, the corresponding dual maximization problem is listed on the right:

**minimize** \ $\mathbf{c}^\top \mathbf{x}$ 
subject to 
\begin{align*}
\mathbf{a}_i^\top \mathbf{x} & \leq b_i, \quad i \in \mathcal{M}_1, \\
\mathbf{a}_i^\top \mathbf{x} & \geq 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, \\
x_j & \text{free}, \quad j \in \mathcal{N}_3,
\end{align*}

**maximize** \ $\mathbf{b}^\top \mathbf{p}$ 
subject to 
\begin{align*}
p_i & \leq 0, \quad i \in \mathcal{M}_1, \\
p_i & \geq 0, \quad i \in \mathcal{M}_2, \\
p_i & \text{free}, \quad i \in \mathcal{M}_3, \\
\mathbf{A}_j^\top \mathbf{p} & \geq c_j, \quad j \in \mathcal{N}_1, \\
\mathbf{A}_j^\top \mathbf{p} & \leq c_j, \quad j \in \mathcal{N}_2, \\
\mathbf{A}_j^\top \mathbf{p} & = c_j, \quad j \in \mathcal{N}_3.
\end{align*}

Remark: Each (inequality or equality) constraint in the primal problem is associated with a (sign-constrained or free) variable in the dual problem, and vice-versa:

<table>
<thead>
<tr>
<th>constraints</th>
<th>minimize</th>
<th>maximize</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\leq b_i$</td>
<td>$\leq 0$</td>
<td>variables</td>
</tr>
<tr>
<td>$\geq b_i$</td>
<td>$\geq 0$</td>
<td></td>
</tr>
<tr>
<td>$= b_i$</td>
<td>free</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>variables</th>
<th>$\leq c_j$</th>
<th>$\geq c_j$</th>
<th>constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\leq 0$</td>
<td>$\geq 0$</td>
<td>free</td>
<td></td>
</tr>
<tr>
<td>$\geq 0$</td>
<td>$\leq c_j$</td>
<td>$= c_j$</td>
<td></td>
</tr>
<tr>
<td>free</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• For example, the dual to the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad x_1 + 2x_2 + 3x_3 \\
\text{subject to} & \quad -x_1 + 3x_2 = 5, \\
& \quad 2x_1 - x_2 + 3x_3 \geq 6, \\
& \quad x_3 \leq 4, \\
& \quad x_1 \geq 0, \\
& \quad x_2 \leq 0.
\end{align*}
\]

is

\[
\begin{align*}
\text{maximize} & \quad 5p_1 + 6p_2 + 4p_3 \\
\text{subject to} & \quad -p_1 + 2p_2 \leq 1, \\
& \quad 3p_1 - p_2 \geq 2, \\
& \quad 3p_2 + p_3 = 3, \\
& \quad p_2 \geq 0, \\
& \quad p_3 \leq 0.
\end{align*}
\]

• If we start with the equivalent minimization problem of the dual in the above example, and rename each \( p_i \) to \( x_i \) we obtain the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad -5x_1 - 6x_2 - 4x_3 \\
\text{subject to} & \quad x_1 - 2x_2 \geq -1, \\
& \quad -3x_1 + x_2 \leq -2, \\
& \quad -3x_2 - x_3 = -3, \\
& \quad x_2 \geq 0, \\
& \quad x_3 \leq 0.
\end{align*}
\]

We can then formulate the dual to the above minimization problem:

\[
\begin{align*}
\text{maximize} & \quad -p_1 - 2p_2 - 3p_3 \\
\text{subject to} & \quad p_1 - 3p_2 = -5, \\
& \quad -2p_1 + p_2 - 3p_3 \leq -6, \\
& \quad -p_3 \geq -4, \\
& \quad p_1 \geq 0, \\
& \quad p_2 \leq 0.
\end{align*}
\]

Notice that we have arrived back at the equivalent maximization version of the primal problem!

**Remark:** The following theorem expresses the fact that the behaviour we observed in the previous example, that the dual to the dual is equivalent to the primal problem, is general.

**Theorem 4.1:** If we transform the dual into an equivalent minimization problem and then form its dual, we obtain a problem equivalent to the original problem.
Remark: The following result is also straightforward to show.

**Theorem 4.2:** Suppose that a linear programming problem $\Pi_1$ is transformed to another linear programming problem $\Pi_2$ by a sequence of transformations of the following types:

1. Replace each free variable with the difference of two nonnegative variables.
2. Replace an inequality constraint by an equality constraint involving a nonnegative slack variable.
3. If some row of the matrix $A$ in a feasible standard-form problem is a linear combination of the other rows, eliminate the corresponding equality constraint.

Then the duals of $\Pi_1$ and $\Pi_2$ are equivalent in the sense that they are either both infeasible or have the same optimal cost.

### 4.C The duality theorem

**Remark:** Notice from (4.1) that if $x$ and $p$ are feasible solutions to the primal and dual problem, respectively, $u_i = p_i (a_i x - b_i) \geq 0$ for $i = 1, \ldots, m$ and $v_j = (c_j - A_j^T p)x_j \geq 0$ for $j = 1, \ldots, n$. Furthermore,

$$0 \leq \sum_{i=1}^{m} u_i + \sum_{j=1}^{n} v_j = p^T (Ax - b) + (c - A^T p)^T x = p^T (Ax - b) + (c^T - p^T A)x = -p^T b + c^T x.$$

This inequality is expressed in the following theorem.

**Theorem 4.3** (Weak duality): If $x$ is a feasible solution to the primal problem and $p$ is a feasible solution to the dual problem, then

$$p^T b \leq c^T x.$$  

**Corollary 4.3.1:**

(a) If the optimal cost in the primal problem is $-\infty$, the dual problem is infeasible.

(b) If the optimal cost in the dual problem is $\infty$, the primal problem is infeasible.

**Proof:**

(a) If the optimal cost in the primal problem is $-\infty$, and the dual problem had a feasible solution $p$, then by Theorem 4.3 we know that $p^T b \leq c^T x$ for every feasible solution $x$ of the primal problem. On taking the minimum over all feasible solutions $x$ to the primal problem, we find that $p^T b \leq -\infty$, which is impossible for finite vectors $p$ and $b$. The dual problem is thus infeasible.

(b) Exercise.
Corollary 4.3.2: Let \( x \) and \( p \) be feasible solutions to the primal and dual problems, respectively, and suppose that \( p^\top b = c^\top x \). Then, \( x \) and \( p \) are optimal solutions.

Proof: For every feasible primal solution \( y \) and feasible dual solution \( q \), Theorem 4.3 guarantees that \( q^\top b \leq c^\top x = p^\top b \leq c^\top y \), from which we see that \( x \) and \( p \) are optimal.

Theorem 4.4 (Strong duality): If a linear programming problem has an optimal solution, so does its dual, and the respective optimal costs are equal.

Proof: Given a linear programming problem \( \Pi' \) with an optimal solution, transform the linear programming into an equivalent standard-form problem \( \Pi \) with the same optimal cost, such that the equality constraint matrix \( A \) has linearly independent rows. Suppose that \( \Pi \) has the form

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

Using Bland’s rule, the simplex method terminates with an optimal solution \( x \), optimal basis \( B \), and reduced cost vector

\[ c^\top = c^\top - p^\top A \geq 0, \]

where \( p^\top = c_B^\top B^{-1} \) and \( c_B \) is the basic cost vector. Then \( p^\top A \leq c^\top \), so that \( p \) is a feasible solution to the dual problem

\[
\begin{align*}
\text{maximize} & \quad p^\top b \\
\text{subject to} & \quad p^\top A \leq c^\top.
\end{align*}
\]

Furthermore

\[ p^\top b = c_B^\top B^{-1} b = c_B^\top x_B = c^\top x, \]

where \( x_B = B^{-1} b \) is the vector of basic variables. Corollary 4.3.2 then tells us that \( p \) is the optimal solution to the dual, with optimal cost \( p^\top b \) equal to the optimal primal cost \( c^\top x \). That is, the dual of \( \Pi \) has the same optimal cost as \( \Pi \). Theorem 4.2 guarantees that the duals of \( \Pi \) and \( \Pi' \) have identical optimal costs, both equal to the optimal primal cost of \( \Pi \), which has the same optimal cost as the original linear programming problem \( \Pi' \).

Remark: It is possible for both the primal and dual problem to be infeasible. The infeasible primal problem

\[
\begin{align*}
\text{minimize} & \quad x_1 + 2x_2 \\
\text{subject to} & \quad x_1 + x_2 = 1 \\
& \quad 2x_1 + 2x_2 = 3
\end{align*}
\]
has dual

maximize \( p_1 + 3p_2 \)
subject to \( p_1 + 2p_2 = 1 \)
\( p_1 + 2p_2 = 2. \)

**Remark:** Recall that there are three possible outcomes when solving a linear programming problem:

1. There exists an optimal solution;
2. The problem is *unbounded*: the optimal cost is \(-\infty (\infty)\) for minimization (maximization) problems;
3. The problem is infeasible.

This leads to nine combinations of outcomes for the primal and dual problems. Theorem 4.4 guarantees that if one problem has an optimal solution, so does the other. Moreover, Corollary 4.3.1 establishes that if one problem is unbounded, the other problem is infeasible. This allows us to complete the following table describing the various possible outcomes for a primal problem and its dual. We observe that of the nine outcomes, only four are possible.

<table>
<thead>
<tr>
<th>Primal \ Dual</th>
<th>Finite optimum</th>
<th>Unbounded</th>
<th>Infeasible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finite optimum</td>
<td>Possible</td>
<td>Impossible</td>
<td>Impossible</td>
</tr>
<tr>
<td>Unbounded</td>
<td>Impossible</td>
<td>Impossible</td>
<td>Possible</td>
</tr>
<tr>
<td>Infeasible</td>
<td>Impossible</td>
<td>Possible</td>
<td>Possible</td>
</tr>
</tbody>
</table>

### 4.D Complementary slackness

An important relationship between the primal and dual optimal solutions is provided by the *complementary slackness* conditions:

**Theorem 4.5** (Complementary slackness): Let \( \mathbf{x} \) and \( \mathbf{p} \) be feasible solutions to the primal and dual problem, respectively. Then \( \mathbf{x} \) and \( \mathbf{p} \) are optimal solutions if and only if

\[
p_i(a_i \mathbf{x} - b_i) = 0, \quad \text{for all } i = 1, \ldots, m
\]

and

\[
(c_j - A_j^\top \mathbf{p})x_j = 0, \quad \text{for all } j = 1, \ldots, n.
\]
CHAPTER 4. DUALITY

Proof: We previously noted for feasible solutions $x$ and $p$ that $u_i = p_i(a_i x - b_i) \geq 0$ for each $i = 1, \ldots, m$ and $v_j = (c_j - A_j^T p)x_j \geq 0$ for $j = 1, \ldots, n$. Moreover, we found

$$0 \leq \sum_{i=1}^m u_i + \sum_{j=1}^n v_j = -p^T b + c^T x.$$

If $x$ and $p$ are optimal then Theorem 4.4 implies that the right-hand side is zero, from which the desired conditions immediately follow. The converse result follows on applying Corollary 4.3.2.

Remark: Complementary slackness means that $x$ and $p$ are optimal iff for each $i$ either $a_i x = b_i$ or $p_i = 0$ and for each $j$ either $A_j^T p = c_j$ or $x_j = 0$. That is, in each case there must be an active constraint in one of the two domains.

Problem 4.1: Determine if $x = (2/5, 1/5, 0)$ is an optimal solution to the problem

\[
\begin{align*}
\text{minimize} & \quad -3x_1 - 6x_2 - 2x_3 \\
\text{subject to} & \quad -3x_1 - 4x_2 - x_3 \geq -2, \\
& \quad -x_1 - 3x_2 - 2x_3 \geq -1, \\
& \quad x_1, x_2, x_3 \geq 0.
\end{align*}
\]

The dual problem is

\[
\begin{align*}
\text{maximize} & \quad -2p_1 - p_2 \\
\text{subject to} & \quad -3p_1 - p_2 \leq -3, \\
& \quad -4p_1 - 3p_2 \leq -6, \\
& \quad -p_1 - 2p_2 \leq -2, \\
& \quad p_1, p_2 \geq 0.
\end{align*}
\]

For the solution $x = (2/5, 1/5, 0)$, since

\[
\begin{align*}
-3x_1 - 4x_2 - x_3 &= -2, \\
-x_1 - 3x_2 - 2x_3 &= -1,
\end{align*}
\]

we see that $x$ is a feasible solution, but because these two constraints are active, we obtain no information on $p_1$ and $p_2$. Since $x_1 > 0$ and $x_2 > 0$ but $x_3 = 0$, optimality also requires

\[
\begin{align*}
-3p_1 - p_2 &= -3, \\
-4p_1 - 3p_2 &= -6,
\end{align*}
\]

which has the unique solution $p_1 = 3/5$, $p_2 = 6/5$. Since $-p_1 - 2p_2 = -3 \leq -2$, we see that this represents a feasible solution to the dual problem. Since both $x$ and $p$ are feasible solutions that satisfy all of the complementary slackness conditions, we conclude that these are optimal solutions. Note that the optimal cost for both problems is $-12/5$. 
Problem 4.2: Determine if \( x = (0, 2) \) is an optimal solution to the problem

\[
\begin{align*}
\text{minimize} & \quad 3x_1 + x_2 & \quad \text{maximize} & \quad 2p_1 + 10p_2 \\
\text{subject to} & \quad 2x_1 + x_2 \geq 2, & \quad \text{subject to} & \quad 2p_1 + 3p_2 \leq 3, \\
& \quad 3x_1 + 5x_2 \geq 10, & \quad & \quad p_1 + 5p_2 \leq 1, \\
& \quad x_1, x_2 \geq 0. & \quad & \quad p_1, p_2 \geq 0.
\end{align*}
\]

At the given \( x \), we see that the first two constraints are active. Since \( x_2 \neq 0 \), complementary slackness requires that \( p_1 + 5p_2 = 1 \), which means that \( p_1 = 1 - 5p_2 \). For \( p_2 \geq 0 \), the first constraint is automatically satisfied since \( 2(1 - 5p_2) + 3p_2 = 2 - 7p_2 \leq 2 < 3 \). Furthermore, \( p_1 \geq 0 \) requires that \( p_2 \leq 1/5 \). Any solution of the form \((1 - 5p_2, p_2)\) where \( p_2 \in [0, 1/5] \) is a feasible solution of the dual satisfying the complementary slackness conditions and is thus an optimal solution, with optimal cost 2. Likewise, \( x = (0, 2) \) is an optimal solution to the primal problem, also with optimal cost 2.

4.E Farkas’ lemma and linear inequalities

Consider the set of standard-form constraints \( Ax = b, x \geq 0 \). If there exists some \( p \) such that \( p^T A \geq 0^T \) and \( p^T b < 0 \), then \( x \geq 0 \Rightarrow p^T Ax \geq 0 \), from which we see that \( Ax \neq b \). We have thus found a certificate of in feasibility for this linear programming problem. This is expressed in the following theorem.

Theorem 4.6 (Farkas’ lemma): Let \( A \) be an \( m \times n \) matrix and let \( b \in \mathbb{R}^m \). Exactly one of the following alternatives holds:

(a) There exists some \( x \geq 0 \) such that \( Ax = b \).

(b) There exists some vector \( p \) such that \( p^T A \geq 0^T \) and \( p^T b < 0 \).

Proof. One direction is easy. If there exists some \( x \geq 0 \) satisfying \( Ax = b \) and if \( p^T A \geq 0^T \), then \( p^T b = p^T Ax \geq 0 \), so the second alternative cannot hold.

Let us now assume that there exists no vector \( x \geq 0 \) satisfying \( Ax = b \). Consider the pair of problems

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

and note that the first is the dual of the second. We are given that the maximization problem is infeasible, which implies that the minimization problem is either unbounded (the optimal cost is \( -\infty \)) or infeasible. Since \( p = 0 \) is a feasible solution to the minimization problem, it follows that the minimization problem is unbounded. In particular, there exists some feasible \( p \) for which \( p^T A \geq 0^T \) and the cost is negative: \( p^T b < 0 \).
Problem 4.3: Use Farka’s lemma to show that the following system of inequalities is inconsistent.

\[
\begin{align*}
\begin{gathered}
 x_1 + x_2 + x_3 &= 1, \\
x_1 - x_2 + 2x_3 &= -2, \\
x_1, x_2, x_3 &\geq 0.
\end{gathered}
\end{align*}
\]

Since \( p_1 = p_2 = 1 \) satisfies the dual constraints

\[
\begin{align*}
 p_1 + p_2 &\geq 0, \\
p_1 - p_2 &\geq 0, \\
p_1 + 2p_2 &\geq 0,
\end{align*}
\]

but \( p_1 - 2p_2 = -1 < 0 \), we see by Farkas’ lemma that the given equations are inconsistent.

Problem 4.4: Use a generalization of Farkas’ lemma to determine whether the following set of inequalities is consistent.

\[
\begin{align*}
 x_1 &\leq 2, \\
x_2 &\leq -3, \\
-x_1 - 2x_2 &\leq 5,
\end{align*}
\]

Consider the dual problems

\[
\begin{align*}
 \text{maximize} & \quad 0^\top x \\
 \text{subject to} & \quad x_1 \leq 2, \\
 & \quad x_2 \leq -3, \\
 & \quad -x_1 - 2x_2 \leq 5
\end{align*}
\]

\[
\begin{align*}
 \text{minimize} & \quad 2p_1 - 3p_2 + 5p_3 \\
 \text{subject to} & \quad p_1 - p_3 = 0, \\
 & \quad p_2 - 2p_3 = 0, \\
 & \quad p_1, p_2, p_3 \geq 0.
\end{align*}
\]

A feasible solution of the minimization problem requires \( p_1 = p_3 \) and \( p_2 = 2p_3 \), so that the cost \( p^\top b \) reduces to \( 2p_3 - 3(2p_3) + 5p_3 = p_3 \geq 0 \). Thus the minimization problem is not unbounded. The solution \( p_1 = p_2 = p_3 = 0 \) achieves this minimal cost of zero, so the problem is not infeasible. The maximization problem is therefore feasible; that is, the original inequalities are consistent.
Chapter 5

Sensitivity Analysis

In this chapter, we consider the standard-form problem

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

We assume that we already have an optimal basis \( B \) and the associated optimal solution \( x^* \). We want to see what happens when some entry of \( A, b, \) or \( c \) is changed. We first look under which conditions \( B \) remains optimal, i.e., the following two conditions hold:

1. feasibility: \( x_B = B^{-1}b \geq 0 \);
2. optimality: \( c^\top = c^\top - c_B^\top B^{-1}A \geq 0^\top \).

5. A A new variable is added

\[
\begin{align*}
\text{minimize} \quad & c^\top x + c_{n+1}x_{n+1} \\
\text{subject to} \quad & Ax + A_{n+1}x_{n+1} = b, \\
& x \geq 0.
\end{align*}
\]

Then

\[
\begin{bmatrix}
   x \\
   x_{n+1}
\end{bmatrix}
= 
\begin{bmatrix}
   x \\
   0
\end{bmatrix}
\]

is a basic feasible solution with the same basis matrix \( B \). This will be an optimal solution if and only if the additional optimal condition

\[
c_{n+1} = c_{n+1} - c_B^\top B^{-1}A_{n+1} \geq 0
\]

holds. If not, then add this variable to the final simplex tableau for the original problem using this reduced cost and continue simplex iterations from there.
Problem 5.1: Use the simplex method to solve the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad -5x_1 - x_2 + 12x_3 \\
\text{subject to} & \quad 3x_1 + 2x_2 + x_3 + x_4 = 10, \\
& \quad 5x_1 + 3x_2 + 3x_4 = 16, \\
& \quad x_1, x_2, x_3, x_4 \geq 0
\end{align*}
\]

Two iterations of the simplex method (Phase II) lead us to the optimal solution \((2, 2, 0, 0)\), with optimal cost \(-12\):

\[
\begin{array}{c|cccc}
& x_1 & x_2 & x_3 & x_4 \\
x_3 & 10 & 3 & 2 & 1 & 0 \\
x_4 & 16 & 5 & 3 & 0 & 1 \\
\end{array}
\]

\[
\begin{array}{c|cccc}
& x_1 & x_2 & x_3 & x_4 \\
x_3 & \frac{56}{5} & 0 & -\frac{2}{5} & 0 & 1 & \frac{41}{5} \\
x_1 & \frac{2}{5} & 1 & \frac{1}{5} & 1 & -\frac{3}{5} \\
\end{array}
\]

\[
\begin{array}{c|cccc}
& x_1 & x_2 & x_3 & x_4 \\
x_2 & 12 & 0 & 0 & 2 & 7 \\
x_1 & 2 & 1 & 0 & -3 & 2 \\
\end{array}
\]

Problem 5.2: Use the optimal solution of Problem 5.1 to construct an initial simplex tableau for the extended problem

\[
\begin{align*}
\text{minimize} & \quad -5x_1 - x_2 + 12x_3 - x_5 \\
\text{subject to} & \quad 3x_1 + 2x_2 + x_3 + x_4 + x_5 = 10, \\
& \quad 5x_1 + 3x_2 + x_4 + x_5 = 16, \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0,
\end{align*}
\]

in which a new variable \(x_5\) has been added.

Recalling that the simplex tableau records \(B^{-1}A\) at each iteration, and noting the final two columns of \(A\) form the identity matrix, we can simply read off

\[
B^{-1} = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}
\]

from the final two columns of our tableau. Let us now introduce the extra variable \(x_5\) into our tableau, entering

\[
B^{-1}A_5 = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}
\]
5.B. A NEW INEQUALITY CONSTRAINT IS ADDED

in the final column. We also need the reduced cost

\[
\bar{c}_5 = c_5 - c^T_B B^{-1} A = -1 - [\begin{array}{cc} -1 & -5 \end{array}] \begin{array}{c} 2 \\ -1 \end{array} = -4.
\]

Since \(\bar{c}_5\) is negative, we see that \((2, 2, 0, 0, 0)\) is not an optimal solution to the extended problem. We therefore perform an additional simplex iteration:

\[
\begin{array}{c|cccccc}
& x_1 & x_2 & x_3 & x_4 & x_5^* \\
x_2 & 12 & 0 & 0 & 2 & 7 & -4 \\
x_1 & 2 & 1 & 0 & -3 & 2 & -1 \\
\end{array}
\]

\[
\begin{array}{c|cccc}
& x_1 & x_2 & x_3 & x_4 & x_5 \\
x_5 & 16 & 0 & 2 & 12 & 1 & 0 \\
x_1 & 3 & 1 & 1/2 & -1/2 & 1/2 & 0 \\
\end{array}
\]

to obtain the optimal solution \((3, 0, 0, 0, 1)\) with optimal cost \(-16\).

5.B  A new inequality constraint is added

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad A x = b, \\
& \quad a^T_{m+1} x \geq b_{m+1}, \\
& \quad x \geq 0.
\end{align*}
\]

If the optimal solution \(x^*\) to the original problem satisfies the inequality constraint, it is also an optimal solution to the new problem.

Otherwise, \(a^T_{m+1} x^* < b_{m+1}\) and we need to analyze the new problem by introducing a slack variable \(x_{n+1}\):

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad A x = b, \\
& \quad a^T_{m+1} x - x_{n+1} = b_{m+1}, \\
& \quad x \geq 0, \\
& \quad x_{n+1} \geq 0.
\end{align*}
\]

Let \(\bar{x} = (x, x_{n+1})\) and \(\bar{b} = (b, b_{m+1})\). Then \(\bar{A} \bar{x} = \bar{b}\), where

\[
\bar{A} = \begin{bmatrix} A & 0 \\ a^T_{m+1} & -1 \end{bmatrix}.
\]

Consider the basis

\[
\bar{B} = \begin{bmatrix} B & 0 \\ a^T & -1 \end{bmatrix}
\]
where \( e \) contains those elements of \( \mathbf{a}_{m+1} \) associated with the original basic columns. The associated basic solution can be calculated as

\[
\begin{bmatrix}
\mathbf{x}^* \\
\mathbf{a}_{m+1}^\top \mathbf{x}^* - b_{m+1}
\end{bmatrix}.
\]

This solution is infeasible since \( \mathbf{a}_{m+1}^\top \mathbf{x}^* < b_{m+1} \). However, since

\[
\mathbf{B}^{-1} = \begin{bmatrix}
\mathbf{B}^{-1} & 0 \\
\mathbf{a}^\top \mathbf{B}^{-1} & -1
\end{bmatrix}
\]

we can precompute

\[
\mathbf{B}^{-1} \mathbf{A} = \begin{bmatrix}
\mathbf{B}^{-1} & 0 \\
\mathbf{a}^\top \mathbf{B}^{-1} & -1
\end{bmatrix} \begin{bmatrix}
\mathbf{A} & 0 \\
\mathbf{a}^\top_{m+1} & -1
\end{bmatrix} = \begin{bmatrix}
\mathbf{B}^{-1} \mathbf{A} & 0 \\
\mathbf{a}^\top \mathbf{B}^{-1} \mathbf{A} - \mathbf{a}^\top_{m+1} & 1
\end{bmatrix}
\]

to find the new reduced cost vector

\[
[\mathbf{c}^\top - \mathbf{c}^\top_{\mathbf{B}}] \begin{bmatrix}
\mathbf{B}^{-1} \mathbf{A} & 0
\end{bmatrix} = [\mathbf{c}^\top - \mathbf{c}^\top_{\mathbf{B}}] \begin{bmatrix}
\mathbf{B}^{-1} \mathbf{A} & 0
\end{bmatrix} \geq 0^\top.
\]

As we saw in the proof of Theorem 4.4, the condition \( \mathbf{c} \geq 0 \) for optimality of the primal solution is equivalent to the condition \( \mathbf{p}^\top \mathbf{A} \leq \mathbf{c}^\top \) for feasibility of the dual problem. That is, \( \mathbf{B} \) is a feasible basis for the dual problem. We can apply the so-called *dual simplex method*, described below, using the above values of the reduced costs and \( \mathbf{B}^{-1} \mathbf{A} \) to construct the initial simplex tableau.

### 5.C The dual simplex method

The *dual simplex method* can be used when one is given an initial tableau that has an infeasible basic solution, with basis matrix \( \mathbf{B} \), but all reduced costs are nonnegative. Here is a single iteration of the dual simplex algorithm:

1. Examine the components of the vector \( \mathbf{B}^{-1} \mathbf{b} \) in the zeroth column of the tableau. If they are all nonnegative, we have an optimal basic feasible solution and the algorithm terminates; otherwise, choose the smallest index \( \ell \) such that \( x_{B(\ell)} < 0 \).

2. The \( \ell \)th row of the tableau is the pivot row, with elements \( x_{B(\ell)}, v_1, \ldots, v_n \). If \( v_j \geq 0 \) for all \( j \), the optimal cost of the dual problem is \( \infty \) and the algorithm terminates.

3. Let \( j \) be the smallest index that minimizes \( \{ \mathbf{c}_j / -v_j : v_j < 0 \} \). The column \( \mathbf{A}_{B(\ell)} \) will exit the basis and the column \( \mathbf{A}_j \) will take its place.

4. Add to each row of the tableau a multiple of the \( \ell \)th row (the pivot row) so that \( v_j \) (the pivot element) becomes 1 and all other entries of the pivot column become 0.
For example, the basic solution \((0, 0, 0, 2, -1)\) given by the following tableau is infeasible:

\[
\begin{array}{|c|ccccc|}
\hline
 & x_1 & x_2^* & x_3 & x_4 & x_5 \\
\hline
x_4 & 0 & 2 & 6 & 10 & 0 \\
x_5 & -1 & -2 & 4 & 1 & 1 \\
\end{array}
\]

However, since the reduced costs in the zeroth row are all nonnegative, we can apply the dual simplex method to obtain a basic feasible solution of the primal problem. After one iteration, we obtain the optimal solution \((0, 1/2, 0, 0, 0)\), with optimal cost 3:

\[
\begin{array}{|c|ccccc|}
\hline
 & x_1 & x_2 & x_3 & x_4 & x_5 \\
\hline
x_4 & -3 & 14 & 0 & 1 & 0 \\
x_2 & 0 & 6 & 0 & -5 & 1 \\
\end{array}
\]

Problem 5.3: Solve Prob. 5.1 with the added constraint \(x_1 + x_2 \geq 5\).

With the added constraint linear programming problem can be written in standard form as

\[
\begin{align*}
\text{minimize} & \quad -5x_1 - x_2 + 12x_3 \\
\text{subject to} & \quad 3x_1 + 2x_2 + x_3 = 10, \\
& \quad 5x_1 + 3x_2 + x_4 = 16, \\
& \quad x_1 + x_2 - x_5 = 5, \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\end{align*}
\]

Let \(a_{m+1} = (1, 1, 0, 0, 0)\) and \(b_{m+1} = 5\). In the final tableau of the original problem, \(x_2\) and \(x_1\) are the basic variables, so \(B = \{A_2 A_1\}\). Thus \(a = (1, 1)\). We first compute

\[
a^T B^{-1} A - a_m^T = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 5 & -3 \\ 1 & 0 & -3 & 2 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 2 & -1 \\ -1 & 0 & 0 \end{bmatrix}.
\]

Finally for the given basic solution \((2, 2, 0, 0)\) of the original problem, we see that the new slack variable \(x_5\) must be \(-1\) in order to satisfy the added constraint. Since \(x_5 < 0\), the basic solution to the extended problem is infeasible. We then need to apply the dual simplex method to the tableau

\[
\begin{array}{|c|ccccc|}
\hline
 & x_1 & x_2 & x_3 & x_4^* & x_5 \\
\hline
x_2 & 12 & 0 & 0 & 2 & 7 \\
x_1 & 2 & 0 & 1 & 5 & -3 \\
x_5 & -1 & 0 & 0 & 2 & -1 \\
\end{array}
\]

Since \(v_4\) is the only negative value in the row for \(x_5\), the corresponding variable \(x_4\) enters and \(x_5\) exits. One iteration of the dual simplex method then brings us to the optimal
solution \((0, 5, 0, 1, 0)\) with optimal cost \(-5\):

\[
\begin{array}{c|ccccc}
\hline
 & x_1 & x_2 & x_3 & x_4 & x_5 \\
\hline
x_2 & = 5 & 0 & 0 & 16 & 0 & \frac{7}{0} \\
x_1 & 0 & 1 & 0 & 1 & 0 & 2 \\
x_4 & 1 & 0 & 0 & -2 & 1 & -1 \\
\hline
\end{array}
\]

5.D Changes in the target vector \(b\)

Assume that \(b\) is changed to \(b + \delta e_k\) for some \(k \in \{1, \ldots, m\}\). We want to determine the range of values of \(\delta\) under which the current basis remains optimal. Since the optimality condition \(\bar{c} \geq 0\) is unchanged, we only need to check the feasibility condition

\[x_B = B^{-1}(b + \delta e_k) = x_B + \delta B^{-1}e_k \geq 0.\]

Let \(g = B^{-1}e_k = (\beta_{1k}, \ldots, \beta_{mk})\) be the \(k\)th column of \(B^{-1}\). Feasibility will be maintained if

\[x_B + \delta g \geq 0,
\]

that is, if

\[
\max_{i; \beta_{ik} > 0} \left(-\frac{x_B(i)}{\beta_{ik}}\right) \leq \delta \leq \min_{i; \beta_{ik} < 0} \left(-\frac{x_B(i)}{\beta_{ik}}\right)
\]

Then the optimal cost is given by

\[c_B^T \bar{x}_B = c_B^T B^{-1}(b + \delta e_k) = p^T b + \delta p_k,
\]

where \(p^T = c_B^T B^{-1}\) is the optimal solution of the dual problem. If \(\delta\) is outside the above range, the current solution satisfies the optimality (or dual feasibility) conditions but is prime infeasible. Then one can apply the dual simplex algorithm, starting with the current basis.

Problem 5.4: Consider the optimal tableau in Problem 5.1. If we add \(\delta\) to \(b_1\), for which range of \(\delta\) does the solution remain optimal? Within this range, what is the rate of change of the optimal cost per unit \(\delta\)?

We only need check that our basic solution remains feasible:

\[
\bar{x}_B = B^{-1}(b + \delta e_1) = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix} \begin{bmatrix} 10 + \delta \\ 16 \end{bmatrix} = \begin{bmatrix} 2 + 5\delta \\ -3 \end{bmatrix}.
\]

For both components of \(\bar{x}_B\) to remain nonnegative, we require \(\delta \in [-2/5, 2/3]\). The rate of change of the optimal cost per unit \(\delta\) is

\[c_B^T B^{-1} e_1 = [-1 & -5] \begin{bmatrix} 5 \\ -3 \end{bmatrix} = 10.
\]

So if we choose \(\delta = -2/5\), our total cost would change by \(-2/5 \times 10 = -4\) (a cost reduction of 4).
5.E Changes in the cost vector $c$

Assume that $c_j$ changes to $c_j + \delta$ for some index $j$. Prime feasibility is not affected. We therefore focus on optimality:

$$c^\top - c_B^\top B^{-1}A \geq 0^\top \iff c_B^\top B^{-1}A \leq c^\top.$$ 

Case 1: If $x_j$ is a nonbasic variable, $c_B$ does not change, and the only inequality that is affected is the one for the reduced cost of $x_j$:

$$c_B^\top B^{-1}A_j \leq c_j + \delta.$$ 

Thus, if $\delta \geq c_B^\top B^{-1}A_j - c_j = -\bar{c}_j$, the current basis remains optimal; otherwise, we can apply the primal simplex method starting from the current basic feasible solution.

Case 2: If $x_j$ is the $\ell$th basic variable, i.e. $j = B(\ell)$, then $c_B$ becomes $c_B + \delta e_\ell$. The optimality conditions for the new problem are then

$$0 \leq c_i - (c_B + \delta e_\ell)^\top B^{-1}A_i = \bar{c}_i - \delta q_i \quad \text{for all } i \neq j,$$

where $q$ is the $\ell$th row of $B^{-1}A$ from the simplex tableau, noting for $i = j$ that the reduced cost of the basic variable $x_j$ remains zero. Optimality thus requires

$$q_i\delta \leq c_i \quad \text{for all } i \neq j.$$

**Problem 5.5:** For $j = 1, \ldots, 4$, determine the range of change $\delta_j$ of $c_j$ under which the final basis in Problem 5.1 remains optimal.

We note that $c = (0, 0, 2, 7)$, with the nonbasic variables being $x_3$ and $x_4$. So we require

$$\delta_3 \geq -2;$$
$$\delta_4 \geq -7.$$

For the basic variable $x_1$, we see that $q = (1, 0, -3, 2)$. We require that $q_i\delta_1 \leq \bar{c}_i$ for all $i \neq 1$. That is,

$$0\delta_1 \leq 0;$$
$$-3\delta_1 \leq 2;$$
$$2\delta_1 \leq 7.$$

The solution will thus remain optimal as long as $\delta_1 \in [-2/3, 7/2]$.

For the basic variable $x_2$, we see that $q = (0, 1, 5, -3)$. We require that $q_i\delta_2 \leq \bar{c}_i$ for all $i \neq 2$. That is,

$$0\delta_2 \leq 0;$$
$$5\delta_2 \leq 2;$$
$$-3\delta_2 \leq 7.$$

The solution will thus remain optimal as long as $\delta_2 \in [-3/7, 2/5]$. 
Chapter 6

Parametric Programming

Sometimes a linear programming problem contains unknown parameters. The simplex method can nevertheless still be used, if algebraic (symbolic) computations are substituted for numerical calculations. Various cases corresponding to different parameter regimes may arise. For example, consider the linear programming problem

\[
\begin{align*}
\text{minimize} & \quad (-3 + 2t)x_1 + (3 - t)x_2 + x_3 \\
\text{subject to} & \quad x_1 + 2x_2 - 3x_3 \leq 5, \\
& \quad 2x_1 + x_2 - 4x_3 \leq 7, \\
& \quad x_1, x_2, x_3 \geq 0.
\end{align*}
\]

We introduce slack variables \(x_4\) and \(x_5\) to put the problem into standard form:

\[
\begin{align*}
\text{minimize} & \quad (-3 + 2t)x_1 + (3 - t)x_2 + x_3 \\
\text{subject to} & \quad x_1 + 2x_2 - 3x_3 + x_4 = 5, \\
& \quad 2x_1 + x_2 - 4x_3 + x_5 = 7, \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\end{align*}
\]

An initial simplex tableau is then

<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>
</tr>
</thead>
<tbody>
<tr>
<td>(x_4)</td>
<td>0</td>
<td>-3 + 2t</td>
<td>3 - t</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(x_5)</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>-3</td>
<td>1</td>
</tr>
</tbody>
</table>

Our analysis then bifurcates into various cases.

Case 1: \(t \in [3/2, 3]\): Since \(-3 + 2t \geq 0\) and \(3 - t \geq 0\), the simplex method terminates with the optimal solution \(x^* = (0, 0, 0, 5, 7)\) and optimal cost 0.

Case 2: \(t > 3\). We see that \(x_2\) enters and \(x_4\) exits, yielding

\[
\begin{align*}
\text{Case 2: } t > 3. & \quad \text{We see that } x_2 \text{ enters and } x_4 \text{ exits, yielding}
\end{align*}
\]

<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>
</tr>
</thead>
<tbody>
<tr>
<td>(x_2)</td>
<td>(5t - 15)/2</td>
<td>(5t - 9)/2</td>
<td>0</td>
<td>(11 - 3t)/2</td>
<td>(t - 3)/2</td>
</tr>
<tr>
<td>(x_5)</td>
<td>9/2</td>
<td>3/2</td>
<td>0</td>
<td>-5/2</td>
<td>-1/2</td>
</tr>
</tbody>
</table>
Case 2.1: $t \in (3, 11/3]$: The simplex method terminates with the optimal solution $x^* = (0, 5/2, 0, 0, 9/2)$ and optimal cost $(15 - 5t)/2$.

Case 2.2: $t > 11/3$: The problem is unbounded, with optimal cost $-\infty$.

Case 3: $t < 3/2$. Then $x_1$ enters and $x_5$ exits, yielding

\[
\begin{array}{c|ccccc}
& x_1 & x_2 & x_3 & x_4 & x_5 \\
\hline
x_4 & 3/2 & 0 & 3/2 & -1 & 1 & -1/2 \\
x_1 & 7/2 & 1 & 1/2 & -2 & 0 & 1/2 \\
\end{array}
\]

Case 3.1: $t \in [5/4, 3/2]$: The simplex method terminates with the optimal solution $x^* = (7/2, 0, 0, 3/2, 0)$ and optimal cost is $7t - 21/2$.

Case 3.2: $t < 5/4$: The problem is unbounded, with optimal cost $-\infty$. 
Bibliography

Index

active, 22
adjacent, 26, 30
affine, 17
artificial variables, 55
auxiliary problem, 55

basic columns, 28
basic cost vector, 37
basic direction, 36
basic feasible solution, 23
basic solution, 23
basic variables, 28
basis, 28
basis matrix, 28
Bland’s rule, 52
bounded, 15

certificate of in feasibility, 75
coefficient matrix, 12
complementary slackness, 73
concave, 16, 17
concave down, 16
concave up, 16
constraints, 7
contains a line, 32
convex, 16, 17
convex hull, 17
cost function, 12
cost vector, 12
cycling, 44, 54
decision variables, 6, 8, 12
degenerate, 30
dual, 67–69
dual problem, 66
dual simplex method, 80
deedge, 26
elementary row operation, 46
Elimination of free variables, 13
Elimination of inequality constraints, 14
enters the basis, 41, 43
exits the basis, 41
extreme point, 21
feasible direction, 35
feasible set, 7, 8
feasible solution, 12
feasible solutions, 7
feasible vector, 12
finite number, 6
finitely, 6
free, 12
full simplex tableau, 48
global minimum, 17
graph, 16
half space, 15
hyperplane, 15
infeasible, 55
infinite number, 6
line segment, 17
linear, 11
linear constraint, 11
linear cost function, 12
linear interpolation, 16
linear optimization, 6
linear programming, 6
local minimum, 17
nonbasic, 28
objective function, 7, 8, 12
optimal, 40
optimal cost, 12
optimal feasible solution, 9, 12
optimal set, 9
optimal solution, 9, 12
optimal value, 9
optimization problem, 6

parameter, 16
parametric form, 16
partial ordering, 12
path of steepest descent, 45
piecewise linear, 19
pivot column, 47
pivot element, 47
pivot row, 47
pivot rules, 45
polyhedron, 15
primal, 69
primal problem, 67
Procedure for constructing basic solutions, 28
proxy basic variable, 56
rate of cost change, 37
reduced cost, 38
reduced cost vector, 38
revised simplex method, 46

simplex, 65
simplex method, 65
slack variable, 14
standard n-simplex, 65
standard form, 12

tableau, 47
target vector, 12
two-phase simplex method, 60

unbounded, 73
unrestricted, 12

vertex, 22