THE ZERO-PERTURBATION SIMPLEX METHOD ACCORDING TO PIVOT RULE

BY

MISS PANTHIRA JAMRUNROJ

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE (MATHEMATICS)
DEPARTMENT OF MATHEMATICS AND STATISTICS
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2017

THE ZERO-PERTURBATION SIMPLEX METHOD ACCORDING TO PIVOT RULE

BY

MISS PANTHIRA JAMRUNROJ

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE (MATHEMATICS)
DEPARTMENT OF MATHEMATICS AND STATISTICS
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2017

THAMMASAT UNIVERSITY

FACULTY OF SCIENCE AND TECHNOLOGY

THESIS

BY

MISS PANTHIRA JAMRUNROJ

ENTITLED

THE ZERO-PERTURBATION SIMPLEX METHOD

ACCORDING TO PIVOT RULE

was approved as partial fulfillment of the requirements for

the degree of Master of Science (Mathematics)

on July 19 , 2018

Chairman ______________________________

(Assistant Professor Krung Sinapiromsaran, Ph.D.)

Member and Advisor ______________________________

(Aua-aree Boonperm, Ph.D.)

Member ______________________________

(Assistant Professor Supachara Kongnuan, Ph.D.)

Member ______________________________

(Assistant Professor Wutiphol Sintunavarat, Ph.D.)

Dean ______________________________

(Associate Professor Somchai Chakhatrakan, Ph.D.)

Thesis Title	THE ZERO-PERTURBATION SIMPLEX METHOD ACCORDING TO PIVOT RULE
Author	Miss Panthira Jamrunroj
Degree	Master of Science (Mathematics)
Department/Faculty/University	Mathematics and Statistics Faculty of Science and Technology Thammasat University
Thesis Advisor	Aua-aree Boonperm, Ph.D.
Academic Year	2017

ABSTRACT

The simplex method is used to solve a linear programming problem which starts at an initial basic feasible solution corresponding to an initial basis which is chosen first. If the chosen initial basis gives a primal or dual feasible solution, then the simplex method or the dual simplex method can start, respectively. Otherwise, artificial variables are introduced to start the simplex method requiring more variables and needs more computation. The perturbation simplex method was presented when an initial basis gives primal and dual infeasible solutions without using artificial variables. It starts by perturbing the reduced costs of the objective function for the dual feasible, then the dual simplex method is used. However, an appropriate perturbation value is not mentioned which it effects the choice of an entering variable. In this thesis, we propose the zero-perturbation value for a selected entering variable. We start by choosing an entering variable according to some pivot rule, then its reduced cost is set to zero. Next, a leaving variable is chosen using the maximum ratio. From the computational results, we found that determining the zero-perturbation value to the entering variable according to some chosen pivot rule causes the number of iterations. Therefore, the efficiency of the zero-perturbation simplex method depends on the pivot rule.

Keywords: Perturbation Simplex Method, Artificial Variable, Pivot Rule

ACKNOWLEDGEMENTS

I would like to sincerely thank my thesis advisor, Dr. Aua-aree Boonperm, who not only gives me inspiration to start this thesis and her invaluable help throughout the academic years, but also gave me opportunities to experience the real-world life living abroad. I would not have achieved this far and this thesis would not have been completed without all the support that I have always received from her.

Next, I would like to thank the thesis committee consisting of Assistant Professor Dr. Krung Sinapiromsaran, Assistant Professor Dr. Supachara Kongnuan and Assistant Professor Dr. Wutiphol Sintunavarat for having their precious time to suggest ideas and feedbacks involving in this thesis.

In addition, I would like to acknowledge the Research Professional Development Project under the Science Achievement Scholarship of Thailand (SAST) for supporting throughout my studies.

Finally, I most gratefully acknowledge my parents and my friends for all their support throughout the period of this research.

Miss Panthira Jamrunroj

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Linear programming problems

In the real-world problems, there are many problems seeking for their minimum costs or maximum profits. They can be expressed as the functions of some certain variables under some limitations. The mathematical programming algorithm is the procedure for searching the best solution from available solutions for the objective function of the problem. Then, it can be defined as the following model.

$$\begin{aligned} \max \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g_i(\mathbf{x}) \leqslant 0, \quad i = 1, ..., m, \end{aligned} \tag{1.1.1}$$

where $f(\mathbf{x})$ is the objective function,
$\mathbf{x}$ is a vector of decision variables,
$g_i(\mathbf{x})$ is the function of i^{th} constraint,
m is the number of constraints.

From (1.1.1), we would like to find the value of vector $\mathbf{x}$ such that $f(\mathbf{x})$ attains the maximum value subject to all constraints $g_i(\mathbf{x}) \leqslant 0$, $\forall\, i \in \{1, ..., m\}$. If the decision variable vector $\mathbf{x}$ satisfies all constraints, then $\mathbf{x}$ is called a **feasible solution**. The set of all feasible solutions is called the **feasible region**. If the feasible region of the problem is empty, then we say that the problem is **infeasible**.

One important case of the mathematical programming is a linear programming which consists of a linear objective function under linear equality or inequality constraints and the sign restriction on each variable. For each variable x_j, the sign restriction can either say: $x_j \geqslant 0$, $x_j \leqslant 0$, or x_j unrestricted where $j = 1, ..., n$. Then, we can write the maximization model of a general linear programming problem as follows:

$$\begin{array}{lccccccccc}
\max & c_1x_1 & + & c_2x_2 & + & \cdots & + & c_nx_n & & \\
\text{s.t.} & a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & \leqslant & b_1 \\
 & a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & \leqslant & b_2 \\
 & \vdots & & \vdots & & & & \vdots & & \vdots \\
 & a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & \leqslant & b_m \\
 & & & & & & & x_1,x_2,\dots,x_n & \geqslant & 0.
\end{array} \tag{1.1.2}$$

From (1.1.2), we let

$$\mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \text{ and } \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

Then, we can rewrite the linear programming model in the matrix form as follows:

$$\begin{array}{lccc}
\max & \mathbf{c}^T\mathbf{x} & & \\
\text{s.t.} & \mathbf{Ax} & \leqslant & \mathbf{b}, \\
 & \mathbf{x} & \geqslant & \mathbf{0},
\end{array} \tag{1.1.3}$$

where **c** is a column vector of coefficients of the objective function,

x is a column vector of decision variables,

b is a column vector of parameters which is called right-hand-side vector,

A is a coefficient matrix of constraints.

The solution to a linear programming problem depends on the objective function and constraints. There are three possibilities of a solution for a linear programming problem as follows:

1. A linear programming problem has **the optimal solution** which maximizes or minimizes the objective function. For some cases, it has multiple optimal solutions, but there is only one optimal objective value.

2. A linear programming problem is **unbounded** when it has a feasible solution made infinitely large of the objective value for the maximization problem or made infinitely small of the objective value for the minimization problem.

3. If a linear programming problem has no solution, then it is **infeasible**.

There are plenty of methods which can solve a linear programming problem such as the graphical method, the interior point method [1], the simplex method [2] and the criss-cross algorithm [16]. For the graphical method, we start from drawing a graph corresponding to all constraints. So, it is suitable for solving only two or three dimensional linear programming problem. The simplex method and the interior point method have been popularly used to solve multidimensional linear programming problems. The interior point method was presented by Karmarkar [1] in 1984. This method is effective for solving a large problem with the sparse matrix and it guarantees that the number of iterations is less than one hundred (see in [15]) while the simplex method is effective for solving small and medium problems. However, the simplex method is also appropriate for post-optimality analysis even though Klee and Minty [3] had shown the worst case running time. As a result, many researchers are interested in improving the performance of the simplex method.

1.2 Literature reviews

The simplex method was published by Dantzig [2], in 1947. It starts from a basic feasible solution. If the origin point is feasible, then the simplex algorithm starts at this point. Otherwise, artificial variables will be added to obtain the initial basic feasible solution, so the number of variables is increased. Consequently, the following question arises: can we solve such problems without using artificial variables? By this question, many researchers have been attempting to improve the simplex algorithm by avoiding the use of artificial variables.

In 1997, Arsham [7, 8] proposed the algorithm without using artificial variables.

This algorithm composes of two phases; phase 1 searches for a basic feasible solution while phase 2 starts the simplex method at a basic feasible solution from phase 1. However, in 1998, Enge and Hunh [9] presented a counterexample that Arsham's algorithm notified the infeasibility of a feasible linear programming problem.

In 2000, Pan [10] proposed an algorithm for solving a linear programming problem without using artificial variables. It starts by choosing an initial basis. If the basis exhibits primal and dual infeasible solutions, then the reduced cost of the objective function in the primal problem will be perturbed by a positive value to gain the dual feasibility, and the dual simplex method will be performed. However, the appropriate perturbation value was not mentioned.

In 2006, Arsham [11, 13] still proposed a new algorithm for solving a general linear programming problem without introducing artificial variables. This algorithm starts at the origin point with all nonnegative right-hand-side values when there exists at least one $\leqslant$ constraint, and all violated $\geqslant$ constraints are relaxed. After the optimal solution of the relaxed problem is found, the $\geqslant$ constraints are restored to the relaxed problem, and the dual simplex method is performed. However, if the problem does not have the $\leqslant$ constraint, then the original positive costs are changed to zero for dual feasibility, and the dual simplex method is used. This step is similar to the perturbation simplex method. Since this algorithm perturbs all reduced costs to zero, the degeneracy occurs for the dual simplex method causing more iterations.

In that year, Corley et al. [12] presented the algorithm which is similar to Arsham's algorithm. Their algorithm starts by solving the sequence of the relaxed linear programming problems until the optimal solution of the original problem was found. The relaxed problem consists of the original objective function subject to a single constraint that makes the minimum cosine angle with the gradient vector of the objective function. For each iteration, the relaxed constraint which has the smallest cosine angle with the gradient vector of the objective function among those constraints will be added and the dual simplex method is used. However, the limitation of this algorithm is that all parameters of the linear programming problem must be positive.

Later, in 2014, Boonperm and Sinapiromsaran [15] presented the artificial-free simplex algorithm based on the non-acute constraint relaxation. This algorithm starts by separating constraints into two groups; the group of acute constraints and the group of non-acute constraints. Then, the relaxed problem is constructed by relaxing the non-acute constraints which is called the non-acute constraint relaxation problem. They proved that this relaxed problem is always feasible. So, artificial variables are not used. After the relaxed problem is solved by the simplex method, the non-acute constraints will be restored to find the optimal solution. However, for some cases, if the relaxed problem is unbounded, then a constraint in the group of non-acute constraints is restored into the problem one by one. If the solution satisfies all constraints, then it is unbounded. Otherwise, the current basis gives the primal and dual infeasible solutions, and the perturbation simplex method is used to solve the problem by perturbing all negative reduced costs to 10^{-6}. From their computational results, they concluded that the algorithm is slow when the relaxed problem is unbounded.

According to the above researches, the perturbation simplex method is used to solve a linear programming problem when a basis gives primal and dual infeasible solutions. However, a perturbation value of each work was proposed differently. So, the central question in this thesis is how can we examine the suitable value for the perturbation simplex method. Since a perturbation value affects the choice of an entering variable when the dual simplex method is used, an appropriate value for the perturbation simplex method is examined. For choosing an entering variable of the dual simplex method, it is done by the minimum ratio test, and the smallest value of the minimum ratio is zero. Therefore, if we know that the nonbasic variable is suitable for entering a basis, then we will set its reduced cost to zero while other negative reduced costs of nonbasic variables are set to some positive constants. So, the suitable nonbasic variable is always chosen to be an entering variable.

Consequently, in this thesis, we propose the zero perturbation value for the perturbation simplex method. Our algorithm starts by choosing an entering variable according to the pivot rule. Then, its reduced cost is set to zero while the others are set to some positive values. Next, a leaving variable is chosen by using the maximum ratio for maximum increased value of the chosen entering variable. The aim of the thesis is to design the algorithm for solving a linear programming problem without using the artificial variables, and it can reduce the computation for the simplex method.

1.3 Overviews

In this thesis, we design an algorithm for solving linear programming problems without using the artificial variables. Before the proposed algorithm is described, we acquaint the necessary geometric properties for comprehending the theory of a linear programming problem such as a convex set, a polyhedral set, and a basic feasible solution in Chapter 2. In addition, we present some current solution methods which are used to solve linear programming problems such as the graphical method, the simplex method and the perturbation simplex method. Moreover, we present the pivot rules which are the criterions to choose a nonbasic variable for entering a basis, and a basic variable leaves a basis.

In Chapter 3, we introduce the proposed method called the **zero-perturbation simplex method** which improves Pan's algorithm [10]. Furthermore, the illustrative examples are presented.

Next, in Chapter 4, the efficiency of our algorithm by comparing the number of iterations between our algorithm, the simplex method and the perturbation simplex method is shown, and the results are discussed. Finally, we will conclude the findings and present the future works.

CHAPTER 2

SOLUTION METHODS

In this chapter, we will give some important definitions and theorems behind linear programming problems including the current and well-known methods for solving linear programming problems. However, before the algorithm was designed, the format of linear programming would be described. So, we first start by introducing the formats of a linear programming problem.

2.1 Formats of a linear programming problem

Two general formats of a linear programming problem are the **standard form** and the **canonical form**. For the simplex algorithm, it is designed to deal with the standard form which is defined as follows:

Table of Standard Form

Maximization problem	Minimization problem
$\begin{array}{ll} \max & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0} \end{array}$	$\begin{array}{ll} \min & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0} \end{array}$

Table of Canonical Form

Maximization problem	Minimization problem
$\begin{array}{ll} \max & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} \leqslant \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0} \end{array}$	$\begin{array}{ll} \min & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} \geqslant \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0} \end{array}$

Generally, an algorithm is designed to solve the problem in the specific form. Hence, we need the ways to change the problem according to the required format of the algorithm. So, we give the following techniques to convert it.

Manipulation of a linear programming problem

A linear programming problem can be in many forms, and some algorithms deal with the specified form. Then, we must manipulate the problem from one form to suit the required form. The details of each manipulation are as follows:

1. Convert a minimization problem to maximization problem and vice versa:

$$\begin{aligned} \min \mathbf{c}^T\mathbf{x} \quad &\equiv \quad -[\ \max -\mathbf{c}^T\mathbf{x}], \\ \max \mathbf{c}^T\mathbf{x} \quad &\equiv \quad -[\ \min -\mathbf{c}^T\mathbf{x}]. \end{aligned}$$

2. Transform inequality constraints to equality constraints:

   (a) We can transform an inequality constraint $\sum\limits_{j=1}^{n} a_{ij}x_j \leqslant b_i$ for $i = 1, \ldots, m$ to an equality constraint by adding a **slack variable** (s_i):

   $$\sum_{j=1}^{n} a_{ij}x_j + s_i = b_i, \ \ s_i \geqslant 0.$$

   (b) We can transform an inequality constraint $\sum\limits_{j=1}^{n} a_{ij}x_j \geqslant b_i$ for $i = 1, \ldots, m$ to an equality constraint by subtraction a **surplus variable** (s_i):

   $$\sum_{j=1}^{n} a_{ij}x_j - s_i = b_i, \ \ s_i \geqslant 0.$$

3. Transform an equality constraint $\sum\limits_{j=1}^{n} a_{ij}x_j = b_i$ for $i = 1, \ldots, m$ into two inequality constraints:

$$\sum_{j=1}^{n} a_{ij}x_j \leqslant b_i \quad \text{and} \quad \sum_{j=1}^{n} a_{ij}x_j \geqslant b_i.$$

4. If the variable x_j can be negative, positive or zero, it is called **unrestricted** in sign, then it is separated into two new variables as:

$$x_j = x_j^+ - x_j^-, \quad x_j^+, x_j^- \geqslant 0.$$

Now, we have the techniques to change a linear programming problem to the standard form which is appropriate to start the simplex method. Before the simplex method is explained, we have to know about the important definitions and theorems of a linear programming problem.

2.2 Theoretical backgrounds

The following definitions and theorems are useful for the proposed method and the simplex method.

2.2.1 Geometry of a linear programming problem

Definition 2.2.1 (Convex Set)**.** The set $\mathbf{X} \subseteq \mathbb{R}^n$ is called **convex** if and only if for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbf{X}$, we have $\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2 \in \mathbf{X}$ for all $\lambda \in [0,1]$.

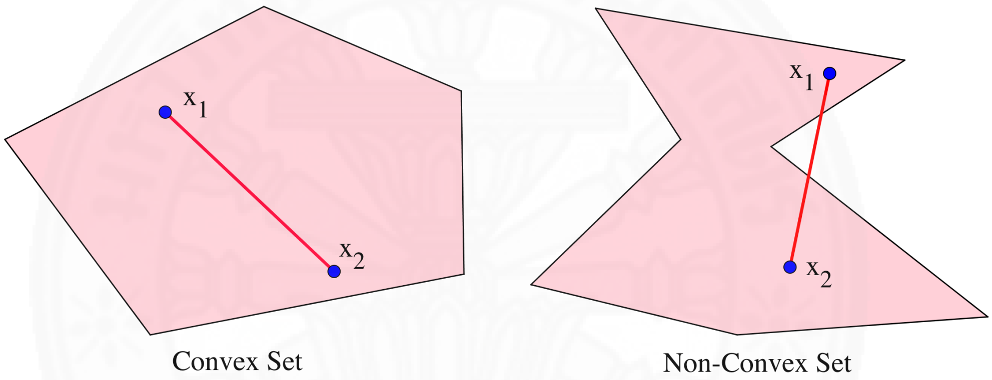

Figure 2.1: Example of convex and non-convex sets

For a linear programming problem, the convex set is important since its feasible set is the convex set called *polyhedral set* which is the intersection of half-spaces. Then, we begin with introducing a *hyperplane* and *half-space*.

Definition 2.2.2 (Hyperplane)**.** Let $\mathbf{a} \in \mathbb{R}^n$ be a constant vector in the n-dimensional space $\mathbb{R}^n$, and let $b \in \mathbb{R}$ be a constant scalar. The set of points

$$\mathbf{H} = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{a}^T \mathbf{x} = b\}$$

is called a **hyperplane** in the n-dimensional space $\mathbb{R}^n$. A vector $\mathbf{a}$ is called the normal gradient vector of hyperplane $\mathbf{H}$.

Definition 2.2.3 (Half-Space)**.** Let $\mathbf{a} \in \mathbb{R}^n$ be a constant vector in the n-dimensional space and let $b \in \mathbb{R}$ be a constant scalar. The set of points

$$\mathbf{H}_l = \{\mathbf{x} | \mathbf{a}^T \mathbf{x} \leqslant b\},$$

$$\mathbf{H}_u = \{\mathbf{x} | \mathbf{a}^T \mathbf{x} \geqslant b\}$$

are called the **half-spaces** defined by the hyperplane $\mathbf{a}^T \mathbf{x} = b$.

Definition 2.2.4 (Polyhedral Set)**.** Let $\mathbf{a}_1, ..., \mathbf{a}_m \in \mathbb{R}^n$ be constant vectors and let $b_1, ..., b_m \in \mathbb{R}$ be constants. The set

$$\mathbf{P} = \bigcap_{i=1}^{m} \mathbf{H}_i,$$

where

$$\mathbf{H}_i = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} \leqslant b_i\},$$

is called a **polyhedral set**.

Therefore, a polyhedral set is a set of the intersection of finite half-spaces. If it is nonempty and bounded, then it is called a **polytope**. Every polytope has at least one vertex on its corner, called an *extreme point*, which is defined below.

Definition 2.2.5 (Extreme Point of a Convex Set)**.** Let $\mathbf{C} \subseteq \mathbb{R}^n$ be a convex set. A point $\mathbf{x}_0 \in \mathbf{C}$ is called an **extreme point** of $\mathbf{C}$ if there are *no points* $\mathbf{x}_1$ and $\mathbf{x}_2$ ($\mathbf{x}_1 \neq \mathbf{x}_0$ or $\mathbf{x}_2 \neq \mathbf{x}_0$) in $\mathbf{C}$ such that $\mathbf{x}_0 = \lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2$ for some $\lambda \in (0, 1)$.

Example 2.2.6. Consider the following inequalities

$$\begin{aligned} 3x_1 - x_2 &\geqslant 5, \\ -2x_1 + 3x_2 &\leqslant 6, \\ -2x_1 + 9x_2 &\geqslant 5, \\ 3x_1 + 5x_2 &\geqslant 48, \\ x_1, x_2 &\geqslant 0. \end{aligned}$$

The inequalities are half-spaces, and the set of the intersection forms the polyhedral set which is the shadowed region in Figure 2.2.

Moreover, we can see that $\mathbf{x}_1 = [2\ 1]^T$, $\mathbf{x}_2 = [3\ 4]^T$, $\mathbf{x}_3 = [6\ 6]^T$, and $\mathbf{x}_4 = [11\ 3]^T$ are the corners of the polytope. They are all the extreme points. ■

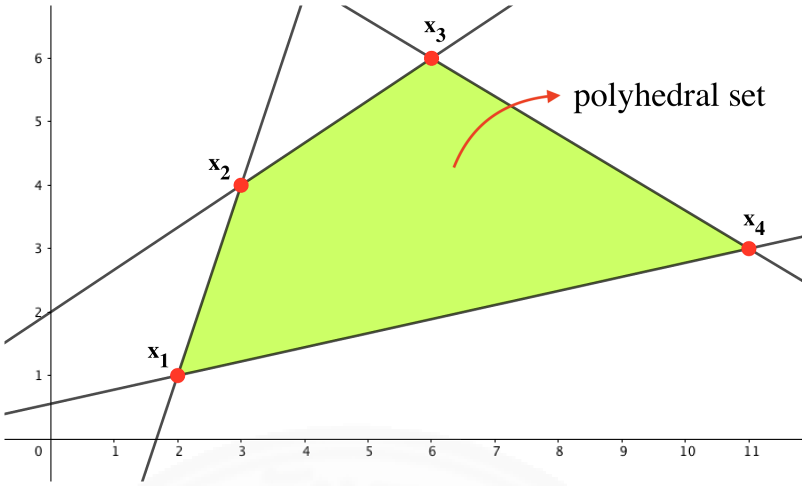

Figure 2.2: Polyhedral set of the example 2.2.6

Definition 2.2.7. Let $\mathbf{P} \subseteq \mathbb{R}^n$ be a polyhedral set defined by

$$\mathbf{P} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} \leqslant \mathbf{b}\}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$. A point $\mathbf{x}_0 \in \mathbf{P}$ is an extreme point of $\mathbf{P}$ if and only if $\mathbf{x}_0$ lies on some n linearly independent hyperplanes from the set defining $\mathbf{P}$.

Since all extreme points are the corners of the feasible region, they are possible solutions of such problem. One important property of the extreme point can be stated as the following theorem.

Theorem 2.2.8. If the feasible region is bounded, then there exists an optimal solution that is an optimal extreme point.

From Theorem 2.2.8, we know that there will be the optimal solution which is the extreme point. Thus, the simple technique that can use to find the optimal solution is determining from all the extreme points. Then, their objective values are compared. By this computing, the number of solved system for all possible extreme points is $C_{m,n}$ which wastes the time for searching all of them. So, many researchers have tried to design the algorithms that is not necessary to consider all extreme points for solving the problem. The simplex method is one of these method which starts at some extreme point. The details will be explained.

Next, we will describe the important theorem which can identify the solution of linear programming problems. Since this theorem mentions about the dual problem, the dual problem will be explained in the next section.

2.2.2 Duality

A linear programming problem always has an associated problem which is called the **dual problem**, and the original linear programming problem is called the **primal problem**. The primal and dual problems have the same set of parameters. One important property of duality is that we can identify the solution of the original problem by solving its dual. The dual problem is defined below.

Definition 2.2.9 (The Dual Problem)**.** Consider the primal problem P in the canonical form:

$$\begin{array}{llrcl} \mathrm{P}: & \max & \mathbf{c}^T\mathbf{x} & & \\ & \text{s.t.} & \mathbf{Ax} & \leqslant & \mathbf{b}, \\ & & \mathbf{x} & \geqslant & \mathbf{0}, \end{array} \tag{2.2.1}$$

and its dual problem D can be written as the following:

$$\begin{array}{llrcl} \mathrm{D}: & \min & \mathbf{b}^T\mathbf{w} & & \\ & \text{s.t.} & \mathbf{A}^T\mathbf{w} & \leqslant & \mathbf{c}. \\ & & \mathbf{w} & \geqslant & \mathbf{0}, \end{array} \tag{2.2.2}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{c},\mathbf{x} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{w} \in \mathbb{R}^m$ which is called the **dual variable**.

Since the simplex method deals with the standard form, we can use the manipulation of a linear programming problem to convert the standard form to the canonical form as follows.

Consider the following primal problem P in the standard form:

$$\begin{array}{llrcl} \mathrm{P}: & \max & \mathbf{c}^T\mathbf{x} & & \\ & \text{s.t.} & \mathbf{Ax} & = & \mathbf{b}, \\ & & \mathbf{x} & \geqslant & \mathbf{0}. \end{array} \tag{2.2.3}$$

Then, the problem (2.2.3) is converted to the canonical form as the following:

$$\text{P}: \quad \begin{aligned} \max \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leqslant \mathbf{b}, \\ & -\mathbf{A}\mathbf{x} \leqslant -\mathbf{b}, \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned} \tag{2.2.4}$$

and its dual problem D can be written as follows:

$$\text{D}: \quad \begin{aligned} \min \quad & \mathbf{b}^T\mathbf{w}_1 - \mathbf{b}^T\mathbf{w}_2 \\ \text{s.t.} \quad & \mathbf{A}^T\mathbf{w}_1 - \mathbf{A}^T\mathbf{w}_2 \geqslant \mathbf{c}, \\ & \mathbf{w}_1, \mathbf{w}_2 \geqslant \mathbf{0}. \end{aligned} \tag{2.2.5}$$

Let $\mathbf{w} = \mathbf{w}_1 - \mathbf{w}_2$ where $\mathbf{w}_1,\ \mathbf{w}_2 \geqslant \mathbf{0}$. Then, the dual problem (2.2.5) can be rewritten as the following:

$$\text{D}: \quad \begin{aligned} \min \quad & \mathbf{b}^T\mathbf{w} \\ \text{s.t.} \quad & \mathbf{A}^T\mathbf{w} \geqslant \mathbf{c}, \end{aligned} \tag{2.2.6}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{c}, \mathbf{x} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{w} \in \mathbb{R}^m$.

Example 2.2.10. Consider the following linear programming problem in the standard form:

$$\begin{aligned} \max \quad & z = 13x_1 + 9x_2 \\ \text{s.t.} \quad & 3x_1 + 5x_2 + s_1 = 120 \\ & 2x_1 + 2x_2 + s_2 = 160 \\ & x_1 + s_3 = 35 \\ & x_1, x_2, s_1, s_2, s_3 \geqslant 0. \end{aligned} \tag{2.2.7}$$

Since the problem (2.2.7) is in the standard form, the dual problem is written as follows:

$$\begin{aligned} \min \quad & 120w_1 + 160w_2 + 35w_3 \\ \text{s.t.} \quad & 3w_1 + 2w_2 + w_3 \geqslant 13 \\ & 5w_1 + 2w_2 \geqslant 9 \\ & w_1, w_2, w_3 \ \text{unrestricted}. \end{aligned} \tag{2.2.8}$$

Consider the following primal problem P in the canonical form:

$$
\begin{array}{llrcl}
\mathrm{P}: & \max & \mathbf{c}^T\mathbf{x} & & \\
 & \text{s.t.} & \mathbf{A}\mathbf{x} & = & \mathbf{b}, \\
 & & \mathbf{x} & \geqslant & \mathbf{0}.
\end{array}
\tag{2.2.9}
$$

and its dual problem D is as follows:

$$
\begin{array}{llrcl}
\mathrm{D}: & \min & \mathbf{b}^T\mathbf{w} & & \\
 & \text{s.t.} & \mathbf{A}^T\mathbf{w} & \leqslant & \mathbf{c}. \\
 & & \mathbf{w} & \geqslant & \mathbf{0},
\end{array}
\tag{2.2.10}
$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{c}, \mathbf{x} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{w} \in \mathbb{R}^m$.

Then, we can summarize the relationships between the primal and dual problems as below.

Lemma 2.2.11. Let $\mathbf{x}_0$ and $\mathbf{w}_0$ be feasible solutions of the primal problem P and the dual problem D, respectively. Then,

$$\mathbf{c}^T\mathbf{x}_0 \leqslant \mathbf{b}^T\mathbf{w}_0.$$

This is called the **weak duality property**.

Corollary 2.2.12. If $\mathbf{x}_0$ and $\mathbf{w}_0$ are feasible solutions of the primal problem P and the dual problem D, respectively such that $\mathbf{c}^T\mathbf{x}_0 = \mathbf{b}^T\mathbf{w}_0$, then $\mathbf{x}_0$ and $\mathbf{w}_0$ are the optimal solutions of the primal problem P and the dual problem D, respectively.

Corollary 2.2.13. If the primal problem P is unbounded, then the dual problem D is infeasible. Similarly, if the dual problem D is unbounded then the primal problem P is infeasible.

Corollary 2.2.14. If the primal problem P is infeasible, then the dual problem D is either infeasible or unbounded. If the dual problem D is infeasible, then the primal problem P is either infeasible or unbounded.

Lemma 2.2.15. If a linear programming problem has the optimal solution, then both the primal and dual problems have the optimal solutions (there exist the optimal solutions $\mathbf{x}^*$ and $\mathbf{w}^*$, respectively) and

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{w}^*.$$

This is called the **strong duality property**

Moreover, if we have the optimal solution of the primal or the dual problem, then we can always find the optimal solutions of the associated problem by the following theorem called **Karush-Kuhn-Tucker Optimality Conditions**.

Theorem 2.2.16 (The Karush-Kuhn-Tucker Optimality Conditions (KKT conditions))**.** Consider a linear programming problem:

$$\begin{array}{llrcl} \mathrm{P}: & \max & \mathbf{c}^T\mathbf{x} & & \\ & \text{s.t.} & \mathbf{A}\mathbf{x} & = & \mathbf{b}, \\ & & \mathbf{x} & \geqslant & \mathbf{0}, \end{array} \tag{2.2.11}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{c} \in \mathbb{R}^n$. Then, $\mathbf{x}^* \in \mathbb{R}^n$ is an optimal solution to the problem P if and only if there exists a vector $\mathbf{w}^* \in \mathbb{R}^m$ so that:

1. (Primal Feasibility)

$$\mathbf{A}\mathbf{x}^* = \mathbf{b}$$
$$\mathbf{x}^* \geqslant \mathbf{0},$$

2. (Dual Feasibility)

$$\mathbf{A}^T\mathbf{w}^* \geqslant \mathbf{c}$$
$$\mathbf{w}^* \text{ unrestricted},$$

3. (Complementary Slackness)

$$\mathbf{w}^{*T}(\mathbf{A}\mathbf{x}^* - \mathbf{b}) = 0$$
$$(\mathbf{c}^T - \mathbf{w}^{*T}\mathbf{A})\mathbf{x}^* = 0.$$

By the KKT conditions, in condition 1 means that $\mathbf{x}^*$ must be the feasible solution of the primal problem, condition 2 indicates that $\mathbf{w}^*$ must be the feasible solution of the dual problem, and condition 3 requires that the objective values of the primal and the dual problems are equal. Thus, if we have the optimal solution to the dual problem $\mathbf{w}^*$ then we can guarantee that the optimal solution of the primal problem exists by KKT conditions.

2.3 Solution methods

After we know about the theory behind the linear programming problem, we will present the methods for solving linear programming problems. First, we begin with the graphical method.

2.3.1 Graphical method

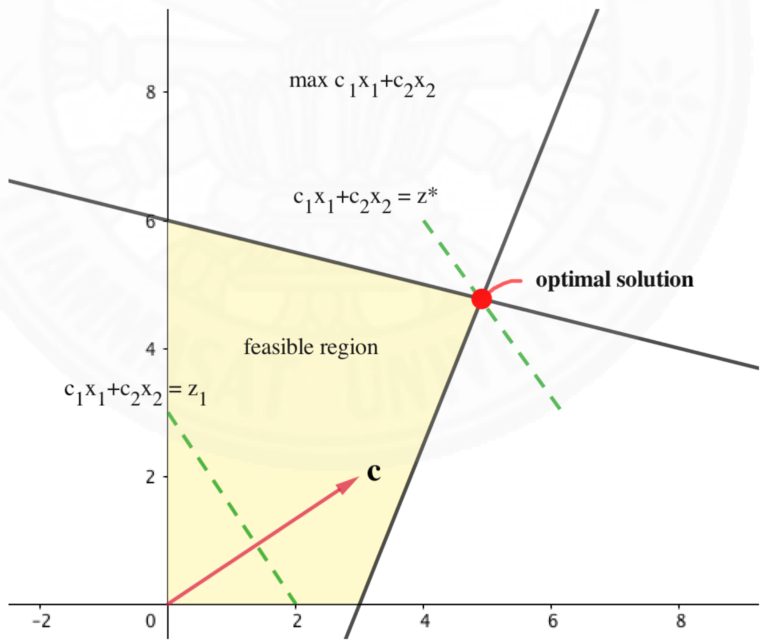

Figure 2.3: Example of the graphical method

The graphical method is used to solve a small linear programming problem which can plot a graph of the feasible region, and its steps are as follows:

1. Construct a graph and plot all constraint lines.

2. Determine the valid side of each constraint line and specify the feasible region.

3. Plot the objective function line which is orthogonal with the coefficient of objective function vector to determine the direction of improvement.

4. Find the most attractive corner. The most attractive corner is the last point in the feasible region touched by a line that is parallel to the objective function line.

5. Determine the optimal solution by algebraically calculating coordinates of the most attractive corner.

6. Determine the value of the objective function for the optimal solution.

Example 2.3.1. Consider a linear programming problem:

$$\begin{aligned} \min \quad & -3x_1 - x_2 \\ \text{s.t.} \quad & -3x_1 + x_2 \leqslant 3 \\ & 2x_1 - 4x_2 \leqslant 4 \\ & x_1 + x_2 \leqslant 5 \\ & x_1, x_2 \geqslant 0. \end{aligned} \tag{2.3.1}$$

From this example, we can plot a graph as Figure 2.4, and the shadowed area is the feasible region. Then, we can see that the optimal solution is $(x_1^*, x_2^*) = (4, 1)$ with the objective value $z^* = -5$. ■

Although a linear programming problem which has more than 3 dimensions cannot be solved by the graphical method because it is difficult to plot a graph, it can be solved by the simplex method which is a popular method for solving a linear programming problem. Additionally, we would like to improve the simplex method. So, the simplex method will be described in details.

2.3.2 The simplex method

Before we give the details of the simplex method, we first give the definition of a *basic feasible solution* that plays the key role in the simplex method.

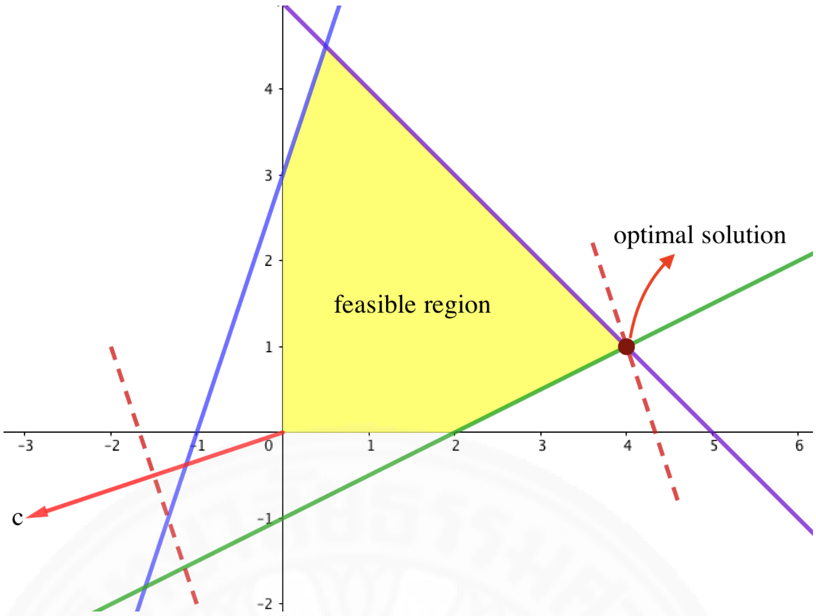

Figure 2.4: Feasible region of the linear programming problem in Example 2.3.1.

Definition 2.3.2. Consider a linear programming problem in the standard form:

$$\begin{array}{llrcl} \mathrm{P}: & & \max & \mathbf{c}^T\mathbf{x} & \\ & & \text{s.t.} \quad \mathbf{Ax} & = & \mathbf{b}, \\ & & \mathbf{x} & \geqslant & \mathbf{0}, \end{array} \tag{2.3.2}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x}$, $\mathbf{c} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$.

Let $\mathbf{A} = [\mathbf{B} \;\; \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m\times m}$ which $\mathbf{B}$ is a nonsingular matrix called a **basic matrix** or **basis** and $\mathbf{N} \in \mathbb{R}^{m\times(n-m)}$ is called a **nonbasic matrix**. Then, the vector of decision variables $\mathbf{x}$ is separated into 2 parts which are the basic variable vector $\mathbf{x}_B \in \mathbb{R}^m$ and the nonbasic variable vector $\mathbf{x}_N \in \mathbb{R}^{n-m}$.

Let $\mathbf{x}_N = \mathbf{0}$. Then, the solution

$$\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b},$$

is called a **basic solution** of the system $\mathbf{Ax} = \mathbf{b}$.

If $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} \geqslant \mathbf{0}$ and $\mathbf{x}_N = \mathbf{0}$, then the solution $[\mathbf{x}_B \; \mathbf{x}_N]^T$ is called a **basic feasible solution**.

From the previous section, we know that an extreme point can be the optimal solution to the problem from Theorem 2.2.8. Thus, the relationship between the extreme point and the basic feasible solution is given in to the following theorem.

Theorem 2.3.3. Let P= $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{Ax} = \mathbf{b},\ \mathbf{x} \geqslant \mathbf{0}\}$ where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, then $\mathbf{x}$ is an extreme point of P if and only if $\mathbf{x}$ is a basic feasible solution of P.

Since a basic feasible solution is an extreme point which can be the optimal solution, there exists a basic feasible solution which is optimal. Thus, the simplex method will visit a basic feasible solution representing the associated extreme point to search for the optimal solution. It starts with a basic feasible solution and moves to the adjacent basic feasible solution which gives the better objective value until the optimal solution is found. For each iteration, it chooses a nonbasic variable to enter the set of basic variables called an **entering variable**. Next, a basic variable is chosen to leave the set of basic variables called a **leaving variable**. After that, the optimality will be verified until the optimal solution is found. The details of the simplex method are explained below.

Before solving a linear programming problem by the simplex method, the problem must be in the standard form as follows:

$$\begin{array}{lrcl} \max & \mathbf{c}^T\mathbf{x} & & \\ \text{s.t.} & \mathbf{Ax} & = & \mathbf{b}, \\ & \mathbf{x} & \geqslant & \mathbf{0}, \end{array} \tag{2.3.3}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$.

Next, the simplex tableau can be constructed by considering the problem (2.3.3). First, we start with introducing a new variable z and let $\mathbf{A} = [\mathbf{B}\ \ \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m \times m}$, $\mathbf{N} \in \mathbb{R}^{m \times (n-m)}$, and $\mathbf{x} = [\mathbf{x}_B\ \ \mathbf{x}_N]^T \geqslant \mathbf{0}$ where $\mathbf{x}_B \in \mathbb{R}^m$, $\mathbf{x}_N \in \mathbb{R}^{n-m}$. Then, the problem (2.3.3) can be rewritten as follows:

$$\begin{array}{lrcl} \max & z = \mathbf{c}_B^T\mathbf{x}_B + \mathbf{c}_N^T\mathbf{x}_N & & \\ \text{s.t.} & \mathbf{Bx}_B + \mathbf{Nx}_N & = & \mathbf{b}, \\ & \mathbf{x}_B, \mathbf{x}_N & \geqslant & \mathbf{0}. \end{array} \tag{2.3.4}$$

From the constraints of problem (2.3.4), we multiply $\mathbf{B}^{-1}$:

$$\mathbf{x}_B + \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N = \mathbf{B}^{-1}\mathbf{b}. \tag{2.3.5}$$

Next, we can multiply this Equation (2.3.5) by $\mathbf{c}_B^T$:

$$\mathbf{c}_B^T\mathbf{x}_B + \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N = \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b}. \tag{2.3.6}$$

If we add this Equation (2.3.6) to the equation $z - \mathbf{c}_B^T\mathbf{x}_B - \mathbf{c}_N^T\mathbf{x}_N = 0$, then we have:

$$z + \mathbf{0}^T\mathbf{x}_B + \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N - \mathbf{c}_N^T\mathbf{x}_N = \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b}, \tag{2.3.7}$$

where $\mathbf{0}$ is the zero vector of the appropriate size. This Equation (2.3.7) can be written as:

$$z + \mathbf{0}^T\mathbf{x}_B + (\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T)\mathbf{x}_N = \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b} \tag{2.3.8}$$

Now, we can rewrite the problem (2.3.3) in the following form:

$$\begin{aligned} \max \quad & z = \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b} - (\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T)\mathbf{x}_N \\ \text{s.t.} \quad & \mathbf{I}\mathbf{x}_B + \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N \;=\; \mathbf{B}^{-1}\mathbf{b}, \\ & \mathbf{x}_B, \mathbf{x}_N \;\geqslant\; \mathbf{0}. \end{aligned} \tag{2.3.9}$$

From the problem (2.3.9), we can see that if we need to increase the value z then we can consider the vector $\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T$. If $\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T \not\geqslant \mathbf{0}$, then the value of some nonbasic variables can be increased. Otherwise, if we increase the value of some nonbasic variable, then the value z may be decreased. So, the optimal solution is found when $\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T \geqslant \mathbf{0}$. Then, we can represent the problem (2.3.9) as the following tableau.

	z	$\mathbf{x}_B$	$\mathbf{x}_N$	RHS	
z	1	$\mathbf{0}$	$\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_\mathbf{N}^T$	$\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b}$	Row 0
$\mathbf{x}_B$	0	$\mathbf{I}$	$\mathbf{B}^{-1}\mathbf{N}$	$\mathbf{B}^{-1}\mathbf{b}$	Row 1 through m

It is called the **initial tableau**.

Let $\mathbf{y}_j = \mathbf{B}^{-1}\mathbf{A}_{:j}$ where $\mathbf{A}_{:j}$ is a column vector j^{th} of $\mathbf{A}$, $z_j = \mathbf{c}_B^T\mathbf{y}_j$ and $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b}$ where $j \in I_N$, and $z_j - c_j$ is called the **reduced cost**. Then, we illustrate the steps of the simplex method as below.

Main Steps of the Simplex Method

1. Let $\mathbf{K} = \{j \in I_N \mid z_j - c_j < 0\}$.

   (a) If $\mathbf{K} = \varnothing$, then stop. The current solution is optimal.

   (b) Otherwise, choose $k \in \mathbf{K}$ and go to step 2.

2. Choose a leaving variable by examining $\mathbf{y}_k$ where $\mathbf{y}_k = \mathbf{B}^{-1}\mathbf{A}_{\cdot k}$.

   (a) If $\mathbf{y}_k \leqslant 0$, then the problem is unbounded.

   (b) Otherwise, determine the index r as follows:

$$\frac{\overline{b}_r}{y_{rk}} = \min_{1 \leqslant i \leqslant m} \left\{ \frac{\overline{b}_i}{y_{ik}} : y_{ik} > 0 \right\}.$$

   This is called the **minimum ratio test**.

3. Update the tableau by pivoting at y_{rk} called **pivot element**. Update the basic and nonbasic variables where x_k enters the basis and x_{B_r} leaves the basis, and repeat the main steps.

Remark 2.3.4. For the minimization problem, we can solve by changing only step 1, that is, let $\mathbf{K} = \{j \in I_N \mid z_j - c_j > 0\}$

Next, we will show the illustrative example which is solved by the simplex method.

Example 2.3.5. Consider the following linear programming problem:

$$\begin{array}{lrcl} \max & z = 2x_1 + 6x_2 - x_3 & & \\ \text{s.t.} & 3x_1 + x_2 - x_3 + s_1 \qquad & = & 120 \\ & x_1 + 2x_2 + 2x_3 \qquad + s_2 & = & 160 \\ & x_1, x_2, x_3, s_1, s_2 & \geqslant & 0. \end{array} \tag{2.3.10}$$

From the problem (2.3.10), we get

$$\mathbf{A} = \begin{bmatrix} 3 & 1 & -1 & 1 & 0 \\ 1 & 2 & 2 & 0 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 120 \\ 160 \end{bmatrix}, \quad \mathbf{c}^T = \begin{bmatrix} 2 & 6 & -1 & 0 & 0 \end{bmatrix} \text{ and } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ s_1 \\ s_2 \end{bmatrix}.$$

It is easy to choose an initial basic feasible solution for starting the simplex method because we have a set of slack variables s_1 and s_2. If we choose these variables to be a basic feasible solution, then we have a basic matrix as an identity matrix which is a simple way to choose a basic feasible solution. Then, we can construct the initial simplex tableau with this basic feasible solution as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	-2	-6	1	0	0	0
s_1	0	3	1	-1	1	0	120
s_2	0	1	2	2	0	1	160

From the above tableau, we have $\mathbf{K} = \{1\ ,2\}$. Since $\mathbf{K} \neq \emptyset$, we can choose $k \in \mathbf{K}$, that is, either x_1 or x_2 can be chosen an entering variable. Suppose that x_1 is chosen as the entering variable. Then, the column vector $\mathbf{y}_1$ is examined. Since there exists $y_{i2} > 0$ for some $i = 1,2$, the minimum ratio test is used to determine a leaving variable x_{B_r} as follows:

$$x_{B_r} = \frac{\overline{b}_r}{y_{r1}} = \min_{1 \leqslant i \leqslant 2} \left\{ \frac{\overline{b}_1}{y_{11}} = \frac{120}{3} = 40, \frac{\overline{b}_2}{y_{21}} = \frac{160}{1} = 160 \right\}.$$

The minimum ratio is 40 corresponding to $r = 1$, that is, s_1 is chosen as the leaving variable. So, the tableau is updated by pivoting at y_{11} as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	0	-16/3	1/3	2/3	0	80
x_1	0	1	1/3	-1/3	1/3	0	40
s_2	0	0	5/3	5/3	-1/3	1	120

Next, the main steps are repeated until the optimal solution is found. Now, the set $\mathbf{K} = \{2\}$ which is nonempty set. Then, x_2 is chosen to be the entering variable, and the column vector $\mathbf{y}_2$ is examined. Since there exists $y_{i2} > 0$ for some $i = 1,2$, the minimum ratio test is used to determine a leaving variable as the following:

$$x_{B_r} = \frac{\bar{b}_r}{y_{r2}} = \min_{1 \leqslant i \leqslant 2} \left\{ \frac{\bar{b}_1}{y_{12}} = \frac{40}{1/3} = 120, \frac{\bar{b}_2}{y_{22}} = \frac{120}{5/3} = 72 \right\}.$$

Then, s_2 is chosen to be the leaving variables. So, the updated tableau becomes:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	0	0	39/5	-2/5	16/5	464
x_1	0	1	0	-4/5	2/5	-1/5	16
x_2	0	0	1	7/5	-1/5	3/5	72

From this tableau, the set $\mathbf{K} = \{4\}$ which is a nonempty set. Then, s_1 is chosen to be the entering variable, and there exists only $y_{14} > 0$. So, x_1 is chosen to be the leaving variable. The tableau is updated as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	1	0	7	0	3	480
s_1	0	5/2	0	-2	1	-1/2	40
x_2	0	1/2	1	1	0	1/2	80

From the above tableau, it has all nonnegative reduced costs. Then, the optimal solution is found, and it is $(x_1^*, x_2^*, x_3^*, s_1^*, s_2^*) = (0, 80, 0, 40, 0)$ with $z^* = 480$. The total number of iterations is three. ■

From Example 2.3.5, we can see that x_1 and x_2 can enter the basis. The next example is shown that the variable x_2 is chosen to be an entering variable first.

Example 2.3.6. From the Example 2.3.5, the initial simplex tableau is constructed as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS	ratio
z	1	-2	-6	1	0	0	0	
s_1	0	3	1	-1	1	0	120	$120/1 = 120$
s_2	0	1	2	2	0	1	160	$160/2 = 80$

Suppose that x_2 is chosen as the entering variable, and s_2 is chosen as a leaving variable by the minimum ratio. Thus, the tableau is updated as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	1	0	7	0	3	480
s_1	0	5/2	0	-2	1	-1/2	40
x_2	0	1/2	1	1	0	1/2	80

Now, the current tableau has all nonnegative reduced costs. Therefore, we found the optimal solution which is $(x_1^*, x_2^*, x_3^*, s_1^*, s_2^*) = (0, 80, 0, 40, 0)$ with $z^* = 480$ from this tableau. The total number of iterations is one. ■

From Example 2.3.5 and Example 2.3.6, the orders of choosing the entering variables are different. We can see that the total numbers of iterations of these examples are also different. Thus, the order of choosing an entering variable affects to the number of iterations. So, many researchers proposed the criterion for choosing an entering variable called the *pivot rule*.

2.3.3 Pivot rules

The operation for moving a basic feasible solution to an adjacent basic feasible solution is called the **pivot rule**. It starts by choosing a nonzero pivot element in a nonbasic column. If the row contains this element, then it is divided by itself to change this element to 1. Then, the other entries in the column are updated to be 0. The variable corresponding to the pivot column enters the set of basic variables called the **entering variable**, and the variable being replaced leaves the set of basic variables called the **leaving variable**. In addition, the set of basic variables is changed one by one element. The criterion for choosing the nonbasic variable to enter a basis is important since it affects the number of iterations.

There are many pivot rules for the simplex method. In this thesis, we focus on only three rules, that are the Dantzig rule, the cosine rule and the largest-distance pivot rule for choosing an entering variable. The details of each pivot rule are described below.

► Dantzig Rule

Dantzig rule is a classic pivot rule for choosing an entering variable because it considers only the reduced costs. For the maximization linear programming problems, we choose a nonbasic variable which has the minimum negative reduced cost as an entering variable. By this choice, the maximum rate for increasing the value of a nonbasic variable is considered. In addition, the classical simplex method and the original perturbation simplex method use this pivot rule to choose an entering variable. The example of the simplex method with the Dantzig rule can see in the Example 2.3.6.

► Cosine Rule

The cosine rule proposed by Yeh and Corley [16] which is used to modify the choice of the entering variable. It uses the primal-dual relationship by considering primal variables which associate to dual constraints. Since many researches [2, 3, 4, 10] stated that the intersection between the constraints which are closest in angle to the objective function gives the optimal solution or the nearby optimal solution, the angles between the gradient vectors of each dual constraint and the gradient vector of the dual objective function are considered. From Figure 2.5, we can see that the gradient vectors of constraints 2 and 3 are closest in angle to the gradient vector of the objective function. Then, we choose these constraints to consider the intersection between them. We observe that this intersection is the optimal solution of the dual problem. Thus, if we choose the primal variables corresponding to these dual constraints enter the basis then the optimal solution of the primal is found at this basis.

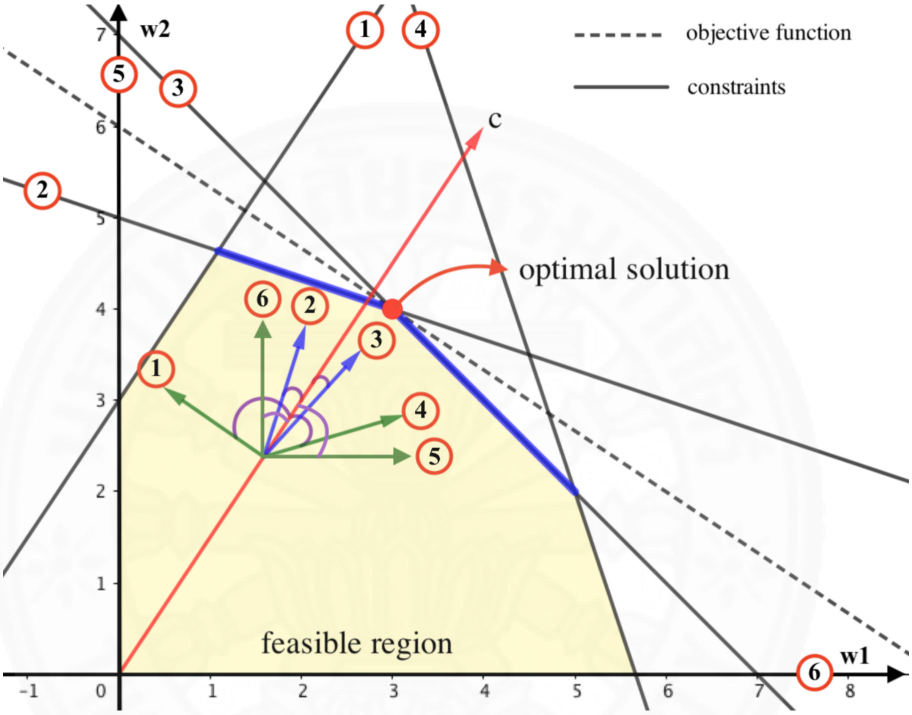

Figure 2.5: Graph of the dual problem

Consider the following linear programming problem:

$$
\begin{aligned}
\mathrm{P}: \qquad & \max \quad \mathbf{c}^T\mathbf{x} \\
& \text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \\
& \qquad\quad \mathbf{x} \geqslant \mathbf{0}.
\end{aligned}
\tag{2.3.11}
$$

Then, the dual problem can be written as follows:

$$
\begin{aligned}
\mathrm{D}: \qquad & \min \quad \mathbf{b}^T\mathbf{w} \\
& \text{s.t.} \quad \mathbf{A}^T\mathbf{w} \geqslant \mathbf{c}, \\
& \qquad\quad \mathbf{w} \quad \text{unrestricted} .
\end{aligned}
\tag{2.3.12}
$$

For each angle, we can compute as the following formula:

$$\theta_j = \arccos \frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\| \cdot \|\mathbf{b}\|},$$

where θ_j is the angle between the gradient vector of the dual constraint j and the gradient vector of the dual objective function, and $\|\cdot\|$ is the Euclidean norm. After θ_j for all $j = 1, \ldots, n$ are computed, the variable x_j associated with the dual constraint j is chosen to enter the basis. Their computational results showed that this rule is quite effective. Next, we will show the example for the selection of an entering variable by the cosine rule, and a leaving variable by the minimum ratio. Furthermore, the θ_j for all $j = 1, \ldots, n$ are calculated only once at first and are used until the optimal solution is found.

Then, the steps of the simplex method according to the cosine rule are stated as follows:

1. Let $\mathbf{K} = \{j \in I_N \mid z_j - c_j < 0\}$.
   (a) If $\mathbf{K} = \varnothing$, then stop. The current solution is optimal.
   (b) Otherwise, compute θ_j for all $j = 1, \ldots, n$ as the following:
   $$\theta_j = \arccos \frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\| \cdot \|\mathbf{b}\|},$$
   and go to step 2.
2. Choose $k = \arg\min\{\theta_j \mid j \in I_N\}$.
3. Choose a leaving variable by examining $\mathbf{y}_k$ where $\mathbf{y}_k = \mathbf{B}^{-1}\mathbf{A}_{\cdot k}$.
   (a) If $\mathbf{y}_k \leqslant 0$, then the problem is unbounded.
   (b) Otherwise, determine the index r as follows:
   $$\frac{\overline{b}_r}{y_{rk}} = \min_{1 \leqslant i \leqslant m} \left\{ \frac{\overline{b}_i}{y_{ik}} : y_{ik} > 0 \right\}.$$
   This is called the minimum ratio test.

4. Update the tableau by pivoting at y_{rk} called pivot element. Update the basic and nonbasic variables where x_k enters the basis and x_{B_r} leaves the basis. Set $\mathbf{K} = \{j \in I_N \mid z_j - c_j < 0\}$.

   (a) If $\mathbf{K} = \varnothing$, then stop. The current solution is optimal.

   (b) Otherwise, go to step 2.

Example 2.3.7. Consider the following linear programming problem:

$$\begin{aligned} \max \quad & z = 2x_1 + 6x_2 - x_3 \\ \text{s.t.} \quad & 3x_1 + x_2 - x_3 + s_1 \quad\quad = 120 \\ & x_1 + 2x_2 + 2x_3 \quad\quad + s_2 = 160 \\ & \quad\quad x_1, x_2, x_3, s_1, s_2 \geqslant 0. \end{aligned} \tag{2.3.13}$$

From this example, we construct the simplex tableau with a basic feasible solution $\mathbf{x}_B = [s_1 \;\; s_2]^T$ as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	-2	-6	1	0	0	0
s_1	0	3	1	-1	1	0	120
s_2	0	1	2	2	0	1	160

Then, we have $\mathbf{K} = \{1,\ 2\}$ which is a nonempty set. Next, θ_j for all $j = 1, \ldots, n$ are computed as follows:

Table 2.3.1: angle of each variable

	x_1	x_2	x_3	s_1	s_2
θ_j (degree)	34.69	10.31	63.44	53.13	36.87

Since the dual problem of this problem can be plotted, the angles between the gradient vector of each dual constraint and the gradient vector of the dual objective function is exhibited as the following Figure 2.6:

Next, the entering variable x_k is chosen by $k = \arg\min\{\theta_1,\ \theta_2,\ \theta_3\}$. From table 2.3.1, we have θ_2 is the minimum value, that is, $k = 2$. Then, x_2 is chosen to be the

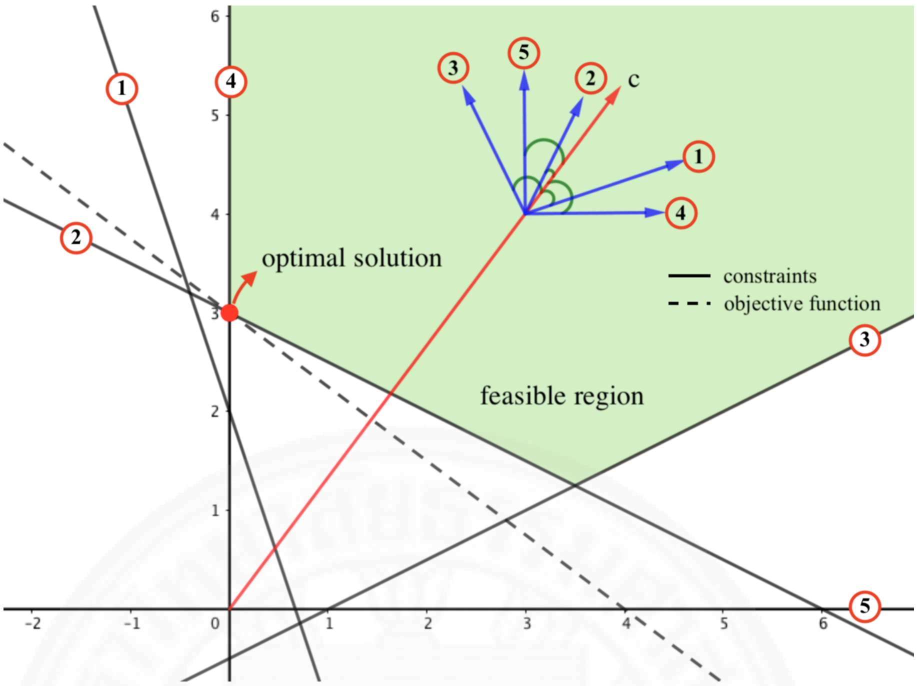

Figure 2.6: Graph of the Example 2.3.7

entering variable. Since there exists $y_{i2} > 0$ for some $i = 1, 2$, the minimum ratio test is used to determine a leaving variable as follows:

$$x_{B_r} = \frac{\bar{b}_r}{y_{r2}} = \min_{1 \leqslant i \leqslant 2} \left\{ \frac{\bar{b}_1}{y_{12}} = \frac{120}{1} = 120, \frac{\bar{b}_2}{y_{22}} = \frac{160}{2} = 80 \right\}.$$

The minimum ratio is 80 corresponding to $r = 2$. Then, s_2 is chosen to be the leaving variable. The updated tableau is as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	1	0	7	0	3	480
s_1	0	5/2	0	-2	1	-1/2	40
x_2	0	1/2	1	1	0	1/2	80

A basis of this tableau gives the primal and dual feasible solutions. Therefore, the optimal solution is found at this tableau. ■

► Largest-Distance Pivot Rule

The largest-distance pivot rule is the pivot rule based on the normalized reduced cost. The formula of this rule is given by:

$$\beta_j = \frac{z_j - c_j}{\|\mathbf{A}_{\cdot j}\|},$$

where $j \in I_N$ and $\|\cdot\|$ is the Euclidean norm.

The variable x_j for $j \in I_N$ which has the minimum value of β_j is chosen as an entering variable in each iteration.

For this pivot rule, the distance between the current basis and violated constraints in the dual problem is considered, and a primal variable corresponding to the dual constraints which has the largest distance is chosen as an entering variable.

The steps of the simplex method according to the largest-distance pivot rule are stated as the following:

1. Let $\mathbf{K} = \{j \in I_N \mid z_j - c_j < 0\}$.

   (a) If $\mathbf{K} = \varnothing$, then stop. The current solution is optimal.

   (b) Otherwise, compute β_j for $j \in I_N$ as follows:

   $$\beta_j = \frac{z_j - c_j}{\|\mathbf{A}_{\cdot j}\|}.$$

   Choose $k = \arg\min\{\beta_j | j \in I_N\}$ and go to step 2.

2. Choose a leaving variable by examining $\mathbf{y}_k$ where $\mathbf{y}_k = \mathbf{B}^{-1}\mathbf{A}_{\cdot k}$.

   (a) If $\mathbf{y}_k \leqslant 0$, then the problem is unbounded.

   (b) Otherwise, determine the index r as follows:

   $$\frac{\overline{b}_r}{y_{rk}} = \min_{1 \leqslant i \leqslant m} \left\{ \frac{\overline{b}_i}{y_{ik}} : y_{ik} > 0 \right\}.$$

   This is called the minimum ratio test.

3. Update the tableau by pivoting at y_{rk} called pivot element. Update the basic and nonbasic variables where x_k enters the basis and x_{B_r} leaves the basis, and go to step 1.

For more understanding, we will demonstrate this rule by the following example.

Example 2.3.8. Consider the following linear programming problem:

$$\begin{array}{lrcl} \max & z = 2x_1 + 6x_2 - x_3 & & \\ \text{s.t.} & 3x_1 + x_2 - x_3 + s_1 \qquad & = & 120 \\ & x_1 + 2x_2 + 2x_3 \qquad + s_2 & = & 160 \\ & x_1, x_2, x_3, s_1, s_2 & \geqslant & 0. \end{array} \tag{2.3.14}$$

From this example, we construct the simplex tableau with a basic feasible solution $\mathbf{x}_B = [s_1 \;\; s_2]^T$ as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	-2	-6	1	0	0	0
s_1	0	3	1	-1	1	0	120
s_2	0	1	2	2	0	1	160

Then, the set $\mathbf{K} = \{1,2\}$ which is a nonempty set, and β_j for $j \in I_N = \{1,2,3\}$ is calculated as the following table:

	x_1	x_2	x_3
β_j	-0.63	-2.68	0.45

Next, the entering variable x_k is chosen by $k = \arg\min\{\beta_1,\ \beta_2,\ \beta_3\}$. Since β_2 is the minimum value, $k = 2$. So, x_2 is chosen to be the entering variable, and there exists $y_{i2} > 0$ for some $i = 1,2$. The minimum ratio test is used to determine a leaving variable as follows:

$$x_{B_r} = \frac{\bar{b}_r}{y_{r2}} = \min_{1 \leqslant i \leqslant 2} \left\{ \frac{\bar{b}_1}{y_{12}} = \frac{120}{1} = 120,\ \frac{\bar{b}_2}{y_{22}} = \frac{160}{2} = 80 \right\}.$$

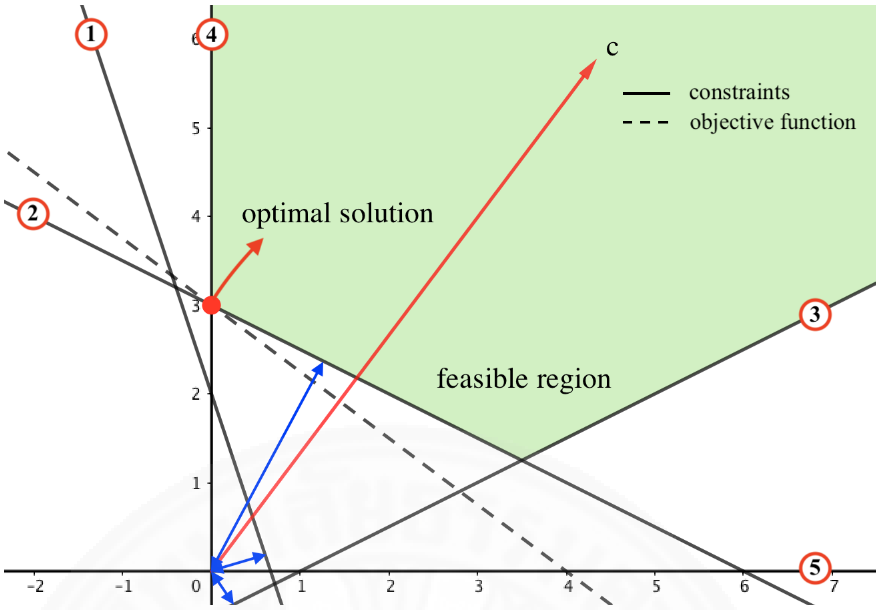

Figure 2.7: Graph of the Example 2.3.8

The minimum ratio is 80 that corresponds to $r = 2$. Thus, s_2 is chosen as the leaving variable. The updated tableau is as follows:

	z	x_1	x_2	x_3	s_1	s_2	RHS
z	1	1	0	7	0	3	480
s_1	0	5/2	0	-2	1	-1/2	40
x_2	0	1/2	1	1	0	1/2	80

From this tableau, the basis gives the primal and dual feasible solutions, that is, we gain the optimal solution at this tableau. ■

Since the dual problem of this example can be visualized a graph, the graph is plotted as Figure 2.7:

From Figure 2.7, we can see that the current basis is associated with the origin point, and the violated constraints are the constraints 1 and 2. Then, the violated constraint which is the largest distance from the origin point is the constraint 2. Thus, we choose the primal variable x_2 which corresponds to this constraint as an entering variable, and the optimal solution is found.

2.3.4 Artificial variables

From the simplex method, it starts at a basic feasible solution. A simple basis to get a basic feasible solution is an identity matrix when the right-hand-side vector is nonnegative. However, some problem is difficult to choose an initial basis to start the simplex method. Then, new variables are introduced for constructing the identity basis to start the simplex method. The new variables are called **artificial variables**.

Consider the following linear programming problem in the standard form:

$$\begin{aligned} \mathrm{P}: \qquad & \max \quad \mathbf{c}^T\mathbf{x} \\ & \text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \\ & \qquad\quad \mathbf{x} \geqslant \mathbf{0}, \end{aligned} \tag{2.3.15}$$

where $\mathbf{b} \geqslant \mathbf{0}$.

Suppose that each constraint $\mathbf{A}_{i:}\mathbf{x} = b_i$ where $\mathbf{A}_{i:}$ is the row i^{th} of matrix $\mathbf{A}$ associates with an artificial variable x_{a_i}. We can add the artificial variable x_{a_i} to the i^{th} constraint as follows:

$$\mathbf{A}_{i:}\mathbf{x} + x_{a_i} = b_i, \tag{2.3.16}$$

Since $b_i \geqslant 0$, we will require $x_{a_i} \geqslant 0$. If $x_{a_i} = 0$, then this is simply the original constraint. Thus, if we can find values for the legitimate decision variables $\mathbf{x}$ and $x_{a_i} = 0$, then constraint i is satisfied. If all artificial variables are zero, then the modified constraints described by Equation (2.3.16) are satisfied, and we have identified an initial basic feasible solution.

Theorem 2.3.9. Let $\mathbf{x}^*$, $\mathbf{x}_a^*$ be an optimal feasible solution to the problem. The problem is feasible if and only if $\mathbf{x}_a^* = \mathbf{0}$ and $\mathbf{A}\mathbf{x}^* = \mathbf{b}$, $\mathbf{x}^* \geqslant \mathbf{0}$.

After artificial variables are added into the problem, there are two popular methods for solving the problem that are the two-phase method and the big-M method.

2.3.5 Two-phase simplex method

The two-phase simplex method is used to solve a linear programming problem when artificial variables are introduced, and it composes of two phases:

Phase I finds a basic feasible solution,

Phase II starts the simplex method at the basic feasible solution from Phase I.

Obviously, we would like to make all artificial variables zero because we want to identify an initial basic feasible solution. So, we write a new linear programming problem as follows:

$$
\begin{aligned}
\mathrm{P}_1: \qquad \min \quad & \mathbf{1}^T\mathbf{x}_a \\
\text{s.t.} \quad \mathbf{A}\mathbf{x}+\mathbf{I}_m\mathbf{x}_a &= \mathbf{b}, \\
\mathbf{x}, \mathbf{x}_a &\geqslant \mathbf{0},
\end{aligned}
\tag{2.3.17}
$$

where $\mathbf{1}$, $\mathbf{x}_a \in \mathbb{R}^m$ and $\mathbf{I}_m \in \mathbb{R}^{m\times m}$ is an identity matrix.

Phase I is a linear programming problem. After we solve the problem (2.3.17), there are two possible solutions for this phase. If $\mathbf{x}_a^* \neq \mathbf{0}$ appears in the solution of Phase I, then the original problem is infeasible. Otherwise, that is $\mathbf{x}_a^* = 0$, there are two possibilities:

1. If $\mathbf{x}_a^* = \mathbf{0}$ and is out of a basis, then the current basis at optimality of Phase I has only vector $\mathbf{x}$. Thus, we can specify a basic feasible solution $\mathbf{x} = [\mathbf{x}_B \ \mathbf{x}_N]^T$. Thereupon, we can start Phase II with using this basic feasible solution.

2. If $\mathbf{x}_a^* = \mathbf{0}$ and within a basis at least one, then we begin Phase II with the optimal tableau from Phase I which contains artificial variables. Moreover, we can guarantee that no artificial variable will be assigned as an entering variable.

Now, we can solve a linear programming problem containing artificial variables by using the two-phase method as the following steps.

The Steps of the Two-Phase Simplex Method

1. Convert maximization (or minimization) problem into the standard form:

$$\begin{aligned} \max \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned} \tag{2.3.18}$$

   with $\mathbf{b} \geqslant \mathbf{0}$.

2. Introduce artificial variables $\mathbf{x}_a$ and solve the **Phase I** problem:

$$\begin{aligned} \min \quad & \mathbf{1}^T\mathbf{x}_a \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} + \mathbf{I}_m\mathbf{x}_a = \mathbf{b}, \\ & \mathbf{x}, \mathbf{x}_a \geqslant \mathbf{0}, \end{aligned} \tag{2.3.19}$$

   where $\mathbf{1}$, $\mathbf{x}_a \in \mathbb{R}^m$ and $\mathbf{I}_m \in \mathbb{R}^{m\times m}$ is an identity matrix.

3. If $\mathbf{x}_a^* = \mathbf{0}$, then an initial basic feasible solution has been indicated. This solution can be converted into a basic feasible solution and go to step 4. Otherwise, there is no solution to the problem (2.3.18).

4. Start the simplex algorithm to solve the original problem by using the basic feasible solution without assigning artificial variables to enter the basis which is identified in step 3.

Example 2.3.10. Consider a linear programming problem:

$$\begin{aligned} \max \quad & z = -x_1 - 2x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 \geqslant 12 \\ & 2x_1 + 3x_2 \geqslant 20 \\ & x_1, x_2 \geqslant 0. \end{aligned} \tag{2.3.20}$$

Convert the problem to the standard form by substraction two surplus variables:

$$\begin{aligned} \max \quad & z = -x_1 - 2x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 - s_1 \qquad = 12 \\ & 2x_1 + 3x_2 \qquad - s_2 = 20 \\ & x_1, x_2, s_1, s_2 \geqslant 0. \end{aligned} \tag{2.3.21}$$

We see that it is difficult to choose an initial basic feasible solution. Obviously, we cannot set $x_1 = x_2 = 0$ because surplus variables $s_1 = -12$ and $s_2 = -20$ which is not feasible. Then, we will add two artificial variables (x_{a_1} and x_{a_2}) into the problem and construct the new problem as follows:

$$\begin{aligned} \min \quad & x_{a_1} + x_{a_2} \\ \text{s.t.} \quad & x_1 + 2x_2 - s_1 \qquad + x_{a_1} \qquad = 12 \\ & 2x_1 + 3x_2 \qquad - s_2 \qquad + x_{a_2} = 20 \\ & x_1, x_2, s_1, s_2, x_{a_1}, x_{a_2} \geqslant 0. \end{aligned} \tag{2.3.22}$$

Then, the problem (2.3.22) is solved by the simplex method. First, we choose $[x_{a_1} \ \ x_{a_2}]^T$ to be an initial basic feasible solution. Therefore, we have

$$\mathbf{x}_B = \begin{bmatrix} x_{a1} \\ x_{a2} \end{bmatrix}, \mathbf{x}_N = \begin{bmatrix} x_1 \\ x_2 \\ s_1 \\ s_2 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{N} = \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 3 & 0 & -1 \end{bmatrix},$$

$$\mathbf{b} = \begin{bmatrix} 12 \\ 20 \end{bmatrix}, \mathbf{c}_B^T = \begin{bmatrix} 1 & 1 \end{bmatrix}, \text{ and } \quad \mathbf{c}_N^T = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}.$$

Compute
$$\begin{aligned} \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 3 & 0 & -1 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 3 & 5 & -1 & -1 \end{bmatrix}, \end{aligned}$$

$$\mathbf{B}^{-1}\mathbf{N} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 3 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 3 & 0 & -1 \end{bmatrix},$$

$$\mathbf{B}^{-1}\mathbf{b} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 12 \\ 20 \end{bmatrix} = \begin{bmatrix} 12 \\ 20 \end{bmatrix},$$

and $$\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b} = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 12 \\ 20 \end{bmatrix} = 32.$$

Then, the initial tableau can be written as follows:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS
z	1	3	5	-1	-1	0	0	32
x_{a_1}	0	1	2	-1	0	1	0	12
x_{a_2}	0	2	3	0	-1	0	1	20

In this case, we can choose either x_1 or x_2 as an entering variable. Suppose that we choose x_1 as the entering variable and compute the minimum ratio test:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS	ratio
z	1	3	5	-1	-1	0	0	32	
x_{a_1}	0	1	2	-1	0	1	0	12	$12/1 = 12$
x_{a_2}	0	2	3	0	-1	0	1	20	$20/2 = 10$

Then, x_{a_2} is chosen to leave the basis. Thus, the updated tableau is as follows:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS
z	1	0	1/2	-1	1/2	0	-3/2	2
x_{a_1}	0	0	1/2	-1	1/2	1	-1/2	2
x_1	0	1	3/2	0	-1/2	0	1/2	10

Next, x_2 is chosen as the entering variable and compute the minimum ratio test:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS	ratio
z	1	0	1/2	-1	1/2	0	-3/2	2	
x_{a_1}	0	0	1/2	-1	1/2	1	-1/2	2	$2/(1/2) = 4$
x_1	0	1	3/2	0	-1/2	0	1/2	10	$10/(3/2) = 20/3$

From the minimum ratio test, we have x_{a_1} is the leaving variable and the updated tableau becomes:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS
z	1	0	0	0	0	-1	-1	0
x_2	0	0	1	-2	1	2	-1	4
x_1	0	1	0	3	-2	-3	2	4

From the above tableau, both artificial variables have been eliminated from the basis and we have indicated the initial basic feasible solution to the original problem: $x_1 = 4, x_2 = 4, s_1 = 0$ and $s_2 = 0$.

Then, we can use this initial basic feasible solution to start the simplex method, and the simplex tableau becomes:

	z	x_1	x_2	s_1	s_2	RHS
z	1	0	0	1	0	-12
x_2	0	0	1	-2	1	4
x_1	0	1	0	3	-2	4

From this tableau, it is the optimal tableau, and the optimal solution to the original problem is $(x_1^*, x_2^*, s_1^*, s_2^*) = (4, 4, 0, 0)$ with $z^* = -12$. ■

This is a simple example for solving a problem by the two-phase method. Furthermore, we have another one popular method for solving a linear programming problem with artificial variables that is the **big-M method**.

2.3.6 Big-M method

The big-M method is used to solve a linear programming problem when artificial variables are added like the two-phase simplex method. But, it is not necessary to separate into two phases.

For the big-M method, after adding artificial variables, problem P (2.3.15) will be modified as follows:

$$
\begin{aligned}
\mathrm{P}_M: \qquad \max \quad & \mathbf{c}^T\mathbf{x} - M\mathbf{1}^T\mathbf{x}_a \\
\text{s.t.} \quad \mathbf{A}\mathbf{x} + \mathbf{I}_m\mathbf{x}_a &= \mathbf{b}, \\
\mathbf{x},\ \mathbf{x}_a &\geqslant \mathbf{0},
\end{aligned} \tag{2.3.23}
$$

where M is a large positive constant, and $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{b}$, $\mathbf{1}$, $\mathbf{x}_a \in \mathbb{R}^m$ and $\mathbf{I}_m \in \mathbb{R}^{m\times m}$ is an identity matrix.

Remark 2.3.11. In the case of a minimization problem, the objective function in the big-M method becomes:

$$\min \quad \mathbf{c}^T\mathbf{x} + M\mathbf{1}^T\mathbf{x}_a, \tag{2.3.24}$$

where M is a large positive constant, $\mathbf{x}$, $\mathbf{c} \in \mathbb{R}^n$, and $\mathbf{1}$, $\mathbf{x}_a \in \mathbb{R}^m$.

The problem P_M can be solved by the simplex method when the initial basis is the identity. So, we can conclude the solution after solving P_M as follows:

- If the problem P_M attains the optimal solution, then
  - ▶ if $\mathbf{x}_a^* = \mathbf{0}$, then the optimal solution of the problem P is found.
  - ▶ if $\mathbf{x}_a^* \neq \mathbf{0}$, then the problem P is infeasible.
- If the problem P_M is unbounded, then
  - ▶ if $\mathbf{x}_a^* = \mathbf{0}$, then the problem P is unbounded.
  - ▶ if $\mathbf{x}_a^* \neq \mathbf{0}$, then the problem P is infeasible.

For more understanding, the example for solving by the big-M method is illustrated.

Example 2.3.12. Consider a linear programming problem:

$$
\begin{aligned}
\min \quad & z = x_1 + 2x_2 \\
\text{s.t.} \quad & x_1 + 2x_2 \geqslant 12 \\
& 2x_1 + 3x_2 \geqslant 20 \\
& x_1, x_2 \geqslant 0.
\end{aligned} \tag{2.3.25}
$$

Convert the problem to the standard form by subtraction two surplus variables:

$$\begin{aligned} \min \quad & z = x_1 + 2x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 - s_1 \qquad\quad = 12 \\ & 2x_1 + 3x_2 \qquad - s_2 = 20 \\ & \qquad\qquad x_1, x_2, s_1, s_2 \geqslant 0. \end{aligned} \tag{2.3.26}$$

Since this problem is the minimization problem, a new problem for the big-M method becomes:

$$\begin{aligned} \min \quad & z = x_1 + 2x_2 + M_1 x_{a_1} + M_2 x_{a_2} \\ \text{s.t.} \quad & x_1 + 2x_2 - s_1 \qquad + x_{a_1} \qquad = 12 \\ & 2x_1 + 3x_2 \qquad - s_2 \qquad + x_{a_2} = 20 \\ & \qquad\qquad x_1, x_2, s_1, s_2, x_{a_1}, x_{a_2} \geqslant 0, \end{aligned} \tag{2.3.27}$$

where M is a large positive number.

Let x_{a_1} and x_{a_2} be an initial basic feasible solution, then we construct the initial tableau as follows:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS	ratio
z	1	$3M$-1	$5M$-2	-M	-M	0	0	$32M$	
x_{a_1}	0	1	2	-1	0	1	0	12	$12/1 = 12$
x_{a_2}	0	2	3	0	-1	0	1	20	$20/2 = 10$

From the above tableau, x_1 and x_2 can enter a basis. Suppose that we choose x_1 as the entering variable and x_{a_2} as the leaving variable by minimum ratio test. Then, the updated tableau becomes:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS	ratio
z	1	0	(M-1)/2	-M	(M-1)/2	0	-(3M-1)/2	2M+10	
x_{a_1}	0	0	1/2	-1	1/2	1	-1/2	2	$2/0.5 = 4$
x_1	0	1	3/2	0	-1/2	0	1/2	10	$10/1.5 = 20/3$

Next, we choose x_2 as the entering variable and x_{a_1} as the leaving variable. We can update the tableau as follows:

	z	x_1	x_2	s_1	s_2	x_{a_1}	x_{a_2}	RHS
z	1	0	0	-1	0	$-M$	$-M$	12
x_2	0	0	1	-2	1	2	-1	4
x_1	0	1	0	3	-2	-3	2	4

From the above tableau, it is the optimal tableau and the optimal solution to the original problem is $(x_1^*, x_2^*, s_1^*, s_2^*) = (4,4,0,0)$ with $z^* = 12$. ■

From the previous examples, the simplex method can start when artificial variables are added first. Nevertheless, if we have the basis which gives the dual feasible solution, then we do not need to add artificial variables into the problem. In addition, we can use the method called the **dual simplex method** to solve this problem.

2.3.7 The dual simplex method

The dual simplex method is used when an initial basis gives the dual feasible solution while the primal solution is infeasible.

Now, we have the dual feasible solution. From the KKT conditions, we need to find the primal feasible solution.

Consider the following linear programming problem:

$$\begin{aligned} \mathrm{P}: \qquad & \max \quad \mathbf{c}^T\mathbf{x} \\ & \text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \qquad\quad \mathbf{x} \geqslant \mathbf{0}, \end{aligned} \tag{2.3.28}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$.

Let $\mathbf{A} = [\mathbf{B}\ \ \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m\times m}$ is a nonsingular matrix and $\mathbf{N} \in \mathbb{R}^{m\times(n-m)}$, and $\mathbf{x} = [\mathbf{x}_B\ \ \mathbf{x}_N]^T$ where $\mathbf{x}_B \in \mathbb{R}^m$, $\mathbf{x}_N \in \mathbb{R}^{n-m}$. The initial simplex tableau can be constructed as the following tableau.

	z	$\mathbf{x}_B$	$\mathbf{x}_N$	RHS	
z	1	$\mathbf{0}$	$\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_\mathbf{N}^T$	$\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b}$	Row 0
$\mathbf{x}_B$	0	$\mathbf{I}$	$\mathbf{B}^{-1}\mathbf{N}$	$\mathbf{B}^{-1}\mathbf{b}$	Row 1 through m

Consider the dual problem of the problem P as follows:

$$\text{D}: \qquad \begin{aligned} \min \quad & \mathbf{b}^T\mathbf{w} \\ \text{s.t.} \quad & \mathbf{A}^T\mathbf{w} \geqslant \mathbf{c}, \\ & \mathbf{w} \text{ unrestricted }. \end{aligned} \tag{2.3.29}$$

From the initial simplex tableau, we can consider $\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_N^T$. Let $\mathbf{w}^T = \mathbf{c}_B^T\mathbf{B}^{-1}$, $z_j - c_j = \mathbf{w}^T\mathbf{A}_{:j} - c_j$ for $j \in I_N$. If $z_j - c_j \geqslant 0$ for all $j \in I_N$, then we get $\mathbf{w}^T\mathbf{A}_{:j} - c_j \geqslant 0$ that is $\mathbf{w}^T\mathbf{A}_{:j} \geqslant c_j$ for all $j \in I_N$. It implies that the dual solution is feasible.

Next, the steps of the dual algorithm can be summarized below.

<u>The Steps of the Dual Simplex Algorithm</u>

1. Choose an initial basis $\mathbf{B}$ which $\mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{A}_{.j} - c_j \geqslant 0$ for all $j \in I_N$, where I_N is the index set of nonbasic variables to construct the initial simplex tableau.

2. If $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b} \geqslant \mathbf{0}$, then the optimal solution is found and stop.

   Otherwise, the dual problem is feasible (since $z_j - c_j \geqslant 0$). Go to step 3.

3. Choose a leaving variable $\mathbf{x}_{B_r}$ by selecting the pivot row r where

   $$\overline{b}_r = \min\,\{\overline{b}_i < 0 \mid i = 1, ..., m\}.$$

4. Examine $y_{rj} < 0$ for $j \in I_N$ to choose the entering variable x_k by using the following minimum ratio test:

   $$\frac{z_k - c_k}{|y_{rk}|} = \min\left\{\frac{z_j - c_j}{|y_{rj}|} : j \in I_N, y_{rj} < 0\right\}.$$

5. If no an entering variable can be selected, then the dual problem is unbounded and the primal problem is infeasible and stop. Otherwise, pivot on element y_{rk}, x_k enters and x_{B_r} leaves the basis, and return to step 2.

Example 2.3.13. Consider the following linear programming problem:

$$\begin{aligned} \max \quad & z = -x_1 - x_2 \\ \text{s.t.} \quad & 2x_1 + x_2 - s_1 \quad\quad = 4 \\ & x_1 + 2x_2 \quad\quad - s_2 = 2 \\ & x_1, x_2, s_1, s_2 \geqslant 0. \end{aligned} \tag{2.3.30}$$

We can construct the initial simplex tableau for the dual simplex algorithm as follows:

	z	x_1	x_2	s_1	s_2	RHS
z	1	1	1	0	0	0
s_1	0	-2	-1	1	0	-4
s_2	0	-1	-2	0	1	-2

From the initial tableau, the dual solution is feasible while the primal solution is infeasible. So, we can use the dual simplex method. First, we can choose a leaving variable which is either s_1 or s_2. Then, the most negative value is chosen as the leaving variable, say s_2. Then, an entering variable is chosen by computing:

$$\frac{z_1 - c_1}{|\bar{a}_{21}|} = \frac{1}{|-1|} \quad \text{and} \quad \frac{z_2 - c_2}{|\bar{a}_{22}|} = \frac{1}{|-2|}.$$

Clearly, x_2 is the entering variable. After we update the tableau, we get the following tableau.

	z	x_1	x_2	s_1	s_2	RHS
z	1	1/2	0	0	1/2	-1
s_1	0	-3/2	0	1	-1/2	-3
x_2	0	1/2	1	0	-1/2	1

From this tableau, the dual problem is still feasible while the primal problem is infeasible. Then, we choose s_1 as the leaving variable. From the minimum ratio test, we will choose x_1 to enter the basis, and the final tableau is as follows:

	z	x_1	x_2	s_1	s_2	RHS
z	1	0	0	1/3	1/3	-2
x_1	0	1	0	-2/3	1/3	2
x_2	0	0	1	1/3	-2/3	0

This tableau is optimal since we have achieved primal and dual feasibilities and the complementary slackness. ■

2.3.8 The perturbation simplex method

From the dual simplex method, it is used to solve a problem without using artificial variable when an initial basis gives the dual feasible solution while the primal solution is infeasible. For some problem, an initial basis gives both the primal and dual infeasible solutions, so the dual simplex method cannot used to solve this case. However, Pan [10] proposed the method to solve this case without using artificial variables called the **perturbation simplex method**, and the details of this method are described below.

Consider a linear programming problem:

$$\begin{array}{llrcl} \mathrm{P}: & \max & \mathbf{c}^T\mathbf{x} & & \\ & \text{s.t.} & \mathbf{A}\mathbf{x} & = & \mathbf{b}, \\ & & \mathbf{x} & \geqslant & \mathbf{0}. \end{array} \tag{2.3.31}$$

When an initial basis $\mathbf{B}$ gives the primal and dual infeasible solutions, that is, there are some negative reduced cost and $\bar{b}_i < 0$ for $i = 1, \ldots, m$. Then, let

$$d = \min\{\bar{b}_i \mid i = 1, \ldots, m\} \text{ and } p = \min\{z_j - c_j \mid j = 1, \ldots, n\}.$$

So, the perturbation simplex method is used when both $d < 0$ and $p < 0$. Then, the main steps of this method can be stated as follows:

The Steps of the Perturbation Simplex Method

1. Perturb the negative reduced costs to a single positive constant.
2. Perform the dual simplex method until the primal feasible solution is found.
3. Restore the original objective function and perform the simplex method to find the optimal solution.

Example 2.3.14. Consider a linear programming problem:

$$\begin{aligned} \min \quad & z = -4x_1 - x_2 \\ \text{s.t.} \quad & -x_1 + 4x_2 + x_3 = -4 \\ & -2x_1 - 5x_2 + x_4 = -18 \\ & 2x_1 - x_2 + x_5 = 22 \\ & x_1, x_2, x_3, x_4, x_5 \geqslant 0. \end{aligned} \tag{2.3.32}$$

First, we choose $I_B = \{3,4,5\}$ as the index set of the initial basis and construct the initial simplex tableau.

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	-4	-1	0	0	0	0
x_3	0	-1	4	1	0	0	-4
x_4	0	-2	-5	0	1	0	-18
x_5	0	2	-1	0	0	1	22

We see that $d = -18$ and $p = -4$. So, the initial basis of this tableau gives the primal and dual infeasible solutions. First, the negative reduced costs of x_1 and x_2 are set to a single positive constant. Suppose that this constant is 1, then the new tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	1	1	0	0	0	0
x_3	0	-1	4	1	0	0	-4
x_4	0	-2	-5	0	1	0	-18
x_5	0	2	-1	0	0	1	22

Then, the dual simplex method can be performed. Suppose that we choose x_4 as the leaving variable. Then, we see that x_1 and x_2 can enter the basis because $y_{21} < 0$ and $y_{22} < 0$. If we use the Dantzig rule to choose an entering variable, it depends on only coefficients of a leaving variable. Then, x_2 is chosen as the entering variable. After the tableau is updated , we get the following tableau.

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0.6	0	0	0.2	0	-3.6
x_3	0	-2.6	0	1	0.8	0	-18.4
x_2	0	0.4	1	0	-0.2	0	3.6
x_5	0	2.4	0	0	-0.2	1	25.6

From this tableau, the basis does not give the primal feasible solution then the dual simplex method is used to continue solving the problem. Then, x_3 and x_1 are chosen to be the leaving variable and the entering variable, respectively. So, the updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	0	0.23	0.39	0	-7.85
x_1	0	1	0	-0.39	-0.31	0	7.08
x_2	0	0	1	0.15	-0.08	0	0.77
x_5	0	0	0	0.92	0.54	1	8.62

From the above tableau, we can see that the basis gives both the primal and dual feasible solutions. Then, the coefficients of the original objective function are restored into the tableau. The updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	0	-1.39	-1.31	0	29.08
x_1	0	1	0	-0.39	-0.31	0	7.08
x_2	0	0	1	0.15	-0.08	0	0.77
x_5	0	0	0	0.92	0.54	1	8.62

Afterward, the simplex method is performed in 3 iterations. Then, the optimal tableau attains:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	0	0.86	0	2.43	50
x_1	0	1	0	0.14	0	0.57	12
x_2	0	0	1	0.29	0	0.14	2
x_5	0	0	0	1.71	1	1.86	16

Therefore, the optimal solution of the original problem is found in 5 iterations. It is $(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (12, 2, 0, 0, 16)$ with $z^* = 50$. ■

Since this method perturbs the negative reduced costs to a single positive constant, the entering variable will depend on only coefficients of a leaving variable. For some cases, choosing the inappropriate entering variable will cause more iterations and computation. So, in this thesis, we propose the appropriate selection for the perturbation simplex method according to the effective pivot rule.

CHAPTER 3

THE ZERO-PERTURBATION SIMPLEX METHOD

In this chapter, we will present the algorithm that is used to solve a linear programming problem without using artificial variable. This algorithm is improved from Pan's algorithm [10] by perturbing the reduced cost of the chosen entering variable to zero called the *zero-perturbation value* and choosing a leaving variable by the *maximum ratio*. Then, we begin with the zero-perturbation value.

3.1 The zero-perturbation value

Consider the following linear programming problem:

$$\begin{aligned} \max \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned} \tag{3.1.1}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, and $\mathbf{b} \in \mathbb{R}^m$.

Let $\mathbf{A} = [\mathbf{A}_{:1}, \mathbf{A}_{:2}, \ldots, \mathbf{A}_{:n}] = [\mathbf{B} \;\; \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m\times m}$ which is a nonsingular matrix, $\mathbf{N} \in \mathbb{R}^{m\times(n-m)}$, and $\mathbf{x} = [\mathbf{x}_B \;\; \mathbf{x}_N]^T$ where $\mathbf{x}_B \in \mathbb{R}^m$, $\mathbf{x}_N \in \mathbb{R}^{n-m}$. Let I_B and I_N be the index sets of basic variables and nonbasic variables, respectively. First, the problem (3.1.1) can be written as follows:

$$\begin{aligned} \max \quad & z = \mathbf{c}_B^T\mathbf{x}_B + \mathbf{c}_N^T\mathbf{x}_N \\ \text{s.t.} \quad & \mathbf{B}\mathbf{x}_B + \mathbf{N}\mathbf{x}_N = \mathbf{b}, \\ & \mathbf{x}_B,\, \mathbf{x}_N \geqslant \mathbf{0}. \end{aligned} \tag{3.1.2}$$

Let $\mathbf{y}_j = \mathbf{B}^{-1}\mathbf{A}_{:j}$, $z_j = \mathbf{c}_B^T\mathbf{y}_j$ where $j \in I_N$, $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b}$ and $z_0 = \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b}$. Then, the problem (3.1.2) can be rewritten as follows:

$$\begin{aligned} \max \quad & z + \sum_{j \in I_N} (z_j - c_j) x_j = z_0 \\ \text{s.t.} \quad & \mathbf{x}_B + \sum_{j \in I_N} (\mathbf{y}_j) x_j = \bar{\mathbf{b}}, \\ & \mathbf{x}_B \geqslant \mathbf{0},\ x_j \geqslant 0,\ j \in I_N. \end{aligned} \tag{3.1.3}$$

Next, we will consider the existence of a basic feasible solution for the problem (3.1.3) by letting

$$\mathbf{P} = \{i \mid \bar{b}_i < 0,\ i = 1, ..., m\}, \tag{3.1.4}$$

$$\mathbf{D} = \{j \mid z_j - c_j < 0,\ j \in I_N\}. \tag{3.1.5}$$

From the dual simplex section, we show that the dual solution is feasible when $z_j - c_j \geqslant 0$ for all $j \in I_N$. Therefore, both the dual and primal problems of (3.1.3) have basic feasible solutions when **D** and **P** are empty. Assume that neither the primal nor the dual problems of (3.1.3) have basic feasible solutions, then the simplex method and the dual simplex method cannot start. However, the primal-perturbation simplex method can start for solving the problem without using artificial variable. This algorithm starts by perturbing the negative reduced costs of the objective function for the dual feasibility as the following value.

Let

$$z'_j - c'_j = \begin{cases} \delta_j, & j \in \mathbf{D}, \\ z_j - c_j, & j \in I_N/\mathbf{D} \end{cases} \tag{3.1.6}$$

where δ_j is a positive constant.

Then, the dual simplex method is performed until the primal feasible solution will be found. Since this method perturbs the negative reduced costs to a single positive constant, the choosing entering variable can be chosen randomly among all negatives of z row. For some cases, if the entering variable is inappropriate then it causes more iterations. So, if we know the suitable nonbasic variable, we should choose this variable to enter the basis in order to gain the speed, and the number of iterations might be decreased. Therefore, the algorithm starts by selecting an entering variable according to the chosen pivot rule: say x_k.

Next, we will revise the Equation (3.1.6) as follows:

$$z'_j - c'_j = \begin{cases} 0, & j = k, \\ \delta_j, & j \in \mathbf{D}/\{k\}, \\ z_j - c_j, & j \in I_N/\mathbf{D}, \end{cases} \tag{3.1.7}$$

where δ_j is a positive constant.

Obviously, the dual problem of (3.1.3) has a basic feasible solution. Then, we can use the dual simplex method to solve it, and no need to use any artificial variable. Since we improve the perturbation simplex method by perturbing the chosen reduced cost to zero, this procedure is called the **zero-perturbation simplex method**.

3.2 The maximum ratio

For the dual simplex method, it can choose any leaving variable which its $\overline{b}_i < 0$ for some $i = 1, \ldots, m$. Since the problem (3.1.1) requires to increase the objective value, the increased maximum value of the chosen entering variable x_k is examined. This maximum value of x_k can be considered from the ratio between $\overline{b}_i < 0$ and $y_{ik} < 0$ as follows:

$$\frac{\overline{b}_r}{y_{rk}} = \max_{1 \leqslant i \leqslant m} \left\{ \frac{\overline{b}_i}{y_{ik}} : b_i,\, y_{ik} < 0 \right\}, \tag{3.2.1}$$

This is called the **maximum ratio**, and the leaving variable x_{B_r} is chosen by this ratio.

After we have the entering variable x_k and the leaving variable x_{B_r}, the simplex tableau is pivoted at y_{rk}. This complete a single iteration. Next, these steps are repeated until $\overline{b}_i \geqslant 0$ for all $i = 1, \ldots, m$. After that, the optimal solutions for the problem (3.1.1) can be recovered by restoring the original coefficients of the objective function. Since such substitution does not affect to the right-hand-side values, the primal problem remains feasible. Thus, if $z_j - c_j \geqslant 0$ for all $j \in I_N$ then the optimal solution is already found. Otherwise, we continue solving with the simplex method.

In summary, before the algorithm starts, the basis B is chosen. Then, the simplex tableau can be constructed as the follows:

$$\begin{array}{c|c|c|c|c|} & z & \mathbf{x}_B & \mathbf{x}_N & \text{RHS} \\ \hline z & 1 & 0 & \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{N}-\mathbf{c}_\mathbf{N}^T & \mathbf{c}_B^T\mathbf{B}^{-1}\mathbf{b} \\ \hline \mathbf{x}_B & 0 & \mathbf{I} & \mathbf{B}^{-1}\mathbf{N} & \overline{\mathbf{b}} \\ \hline \end{array} \tag{3.2.2}$$

The initial simplex tableau can indicate that a basis gives the primal or dual infeasible solution, that is, there exist some negative reduced cost or $\overline{b}_i < 0$ for some $i = 1, ..., m$.

Let p be the minimum value of the right-hand-side values corresponding to basic variable $\mathbf{x}_B$:

$$p = \min_{i\in\{1,...,m\}} \{\overline{b}_i\}, \tag{3.2.3}$$

and d be the minimum value of the reduced costs:

$$d = \min_{j\in\{1,...,n\}} \{z_j - c_j\}. \tag{3.2.4}$$

Then, we notice the problem can be solved by considering values of p and d as the following four cases:

Table 3.2.1: all possible cases for the basis

cases	p	d	Primal Problem	Dual Problem	Method
1	$\geqslant 0$	$\geqslant 0$	feasible	feasible	Optimal solution found
2	$\geqslant 0$	< 0	feasible	infeasible	Simplex method
3	< 0	$\geqslant 0$	infeasible	feasible	Dual simplex method
4	< 0	< 0	infeasible	infeasible	Zero-perturbation simplex method

From Table 3.2.1, in case 1 means that the initial basis gives the primal and dual feasible solutions. Thus, the optimal solution is found. In case 2, $p \geqslant 0$ and $d < 0$ indicate that an initial basis gives the primal feasible solution while the dual solution is infeasible. Then, the simplex method is performed. Next case, an initial basis gives the dual feasible solution, but the primal solution is infeasible. So, we continue solving with the dual simplex method. In the last case, both d and p are less than zero, that is, an initial basis gives both the primal and dual infeasible solutions. Then, the zero-perturbation simplex method is used to solve the problem without using artificial variables. This method starts by selecting the effective pivot rule to choose an entering variable and set its reduced cost as zero while other negative reduced costs are set to a single positive value. By this setting, we ensure that the appropriate nonbasic variable will always enter the basis since zero is the possible minimum value of the minimum ratio. In addition, a leaving variable will be chosen by the maximum ratio to increase the value of the chosen entering variable as possible. Then, the steps are repeated until the primal is feasible. After that, the original coefficients of the objective function are restored, and the simplex method is performed until the optimal solution will be found. All steps of this algorithm will be summarized in the next section.

3.3 The steps of algorithm

Before our algorithm starts, the linear programming problem must be set to the maximization problem in the standard form. Then, the main steps of our algorithm can be summarized as follows:

Step 1: Choose the initial basis $\mathbf{B}$ and construct the initial simplex tableau.

Step 2: Set $p = \min_{i \in \{1,\ldots,m\}} \{\bar{b}_i\}$ and $d = \min_{j \in \{1,\ldots,n\}} \{z_j - c_j\}$.

Case 1: If $p \geqslant 0$ and $d \geqslant 0$, then the optimal solution is found and stop the algorithm.

Case 2: If $p \geqslant 0$ and $d < 0$, then perform the simplex method until the optimal solution is obtained.

Case 3: If $p < 0$ and $d \geqslant 0$, then perform the dual simplex method until the optimal solution is found.

Case 4: If $p < 0$ and $d < 0$, then perform the zero-perturbation simplex method.

Sub-algorithm of the zero-perturbation simplex method

4.1: Let $I'_N = I_N$ and $n = |I'_N|$.

4.2: Choose an entering variable according to the pivot rule: say x_k.

4.3: Examine $\mathbf{y}_k = \mathbf{B}^{-1}\mathbf{A}_{:k}$.

- If $y_{ik} < 0$ for $i = 1, \ldots, m$, then choose a leaving variable by the maximum ratio is given by:

$$\frac{\bar{b}_r}{y_{rk}} = \max_{1 \leqslant i \leqslant m} \left\{ \frac{\bar{b}_i}{y_{ik}} : b_i, y_{ik} < 0 \right\}$$

and go to step 4.4.

- Otherwise,

  - if $n \neq 0$, then set $n = n - 1$, $I'_N = I'_N - \{k\}$ and go to step 4.2.

  - Otherwise, the problem is infeasible and stop the algorithm.

4.4: Update the simplex tableau with pivot element y_{rk}.

4.5: If there exists $\bar{b}_i < 0$ for $i = 1, \ldots, m$, then set $I'_N = I_N$, $n = |I_N|$ and go to step 4.2. Otherwise, go to step 4.6.

4.6: Restore the original coefficient of the objective function by computing $z_j - c_j = \mathbf{w}^T\mathbf{A}_{:j} - c_j$ for all j which is perturbed. Perform the simplex method until the optimal solution is found and stop the algorithm.

3.4 Illustrative examples

The following two examples are shown the step by step of the proposed algorithm with the cosine rule and the largest-distance pivot rule, respectively.

Example 3.4.1. Consider a linear programming problem:

$$
\begin{array}{llcr}
\max & z = 4x_1 + x_2 & & \\
\text{s.t.} & -x_1 + 4x_2 \quad +x_3 & = & -4 \\
 & -2x_1 - 5x_2 \qquad +x_4 & = & -18 \\
 & 2x_1 - x_2 \qquad\qquad +x_5 & = & 22 \\
 & \qquad x_1,\ x_2,\ x_3,\ x_4,\ x_5 & \geqslant & 0
\end{array} \tag{3.4.1}
$$

First, we choose $I_B = \{3,4,5\}$ to be an index of the initial basis and construct the initial simplex tableau as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	-4	-1	0	0	0	0
x_3	0	-1	4	1	0	0	-4
x_4	0	-2	-5	0	1	0	-18
x_5	0	2	-1	0	0	1	22

From this tableau, we have $\mathbf{P} = \{1,\ 2\}$ and $\mathbf{D} = \{1,\ 2\}$ which are nonempty sets, and $p = -18 < 0$ and $d = -4 < 0$. It means that the initial basis gives both the primal and dual infeasible solutions. Thus, we can use the zero-perturbation simplex method to solve this problem. In this example, we use the cosine rule to choose an entering variable. First, θ_j for all $j = 1, \ldots, n$ are extracted as the following table.

	x_1	x_2	x_3	x_4	x_5
θ_j	12.84	73.74	98.05	129.05	39.65

Since $I_N = \{1,2\}$, we set $I'_N = I_N = \{1,2\}$ and $n = |I'_N| = 2$. From the above table, we can see that x_1 has the minimum value of θ_j for $j \in I_N$, so it is chosen to be the entering variable, and set $z_1 - c_1 = 0$. Moreover, there exists $z_2 - c_2 = -1 < 0$ in the initial tableau, so we set $z_2 - c_2 = 1$. After that, the maximum ratio is computed for choosing a leaving variable which is as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS	ratio
z	1	0	1	0	0	0	0	
x_3	0	-1	4	1	0	0	-4	$(-4)/(-1) = 4$
x_4	0	-2	-5	0	1	0	-18	$(-18)/(-2) = 9$
x_5	0	2	-1	0	0	1	22	-

From this tableau, the maximum ratio is in x_4's row. Then, we choose x_4 as the leaving variable and update the tableau as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	1	0	0	0	0
x_3	0	0	6.5	1	-0.5	0	5
x_1	0	1	2.5	0	-0.5	0	9
x_5	0	0	-6	0	1	1	4

From table 4, the basis which has an index set $I_B = \{3,\ 1,\ 5\}$ gives the primal feasible solution. Then, the original coefficients of the objective function are restored into the simplex tableau, and this tableau becomes:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	9	0	-2	0	36
x_3	0	0	6.5	1	-0.5	0	5
x_1	0	1	2.5	0	-0.5	0	9
x_5	0	0	-6	0	1	1	4

We see that the reduced costs of the current tableau have a negative value. Then, the simplex method is used by choosing x_4 as the entering variable and x_5 as the leaving variable. So, the new tableau becomes:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	-3	0	0	2	44
x_3	0	0	3.5	1	0	0.5	7
x_1	0	1	-0.5	0	0	0.5	11
x_4	0	0	-6	0	1	1	4

Next, x_2 is chosen as the entering variable and x_3 as the leaving variable. The tableau is updated as follows:

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	1	0	0	0.9	0	2.4	50
x_2	0	0	1	0.3	0	0.1	2
x_1	0	1	0	1	0	0.6	12
x_4	0	0	0	1.7	1	1.9	16

From this tableau, all values of the zero row and right-hand-side values are non-negative value, that is, the current basis gives the primal and dual feasible solutions. So, it is the optimal tableau, and the optimal solution is $(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (12, 0, 0, 16, 0)$ with $z^* = 50$. ■

From the above example, we also use the original perturbation simplex method and the simplex method to solve this problem. Then, the results of these method are shown in the following table.

Table 3.4.1: Comparison of the number of iterations between the zero-perturbation simplex method, the perturbation simplex method and the simplex method.

Method		the size of matrix operation	the number of iterations	the total number of iterations
The zero-perturbation simplex method		4×6	3	3
The perturbation simplex method		4×6	5	5
The two-phase simplex method	Phase I	4×8	2	
	Phase II	4×6	3	5

From this table, we can see that the total number of iterations for the zero-perturbation simplex method is 3 which is less than other methods. In addition, the size of matrix operation of the zero-perturbation simplex method is smaller than the phase I of the simplex method. Thus, we can reduce the number of iterations.

Next, the zero-perturbation simplex method according to the largest-distance pivot rule is demonstrated as the following example.

Example 3.4.2. Consider a linear programming problem:

$$\begin{aligned} \max \quad & z = 2x_1 + 3x_2 - 12x_3 - 12x_4 - 40x_5 - 41.25x_6 - 45.6x_7 \\ \text{s.t.} \quad & 2x_1 + x_2 + 4x_3 - x_4 - 8x_5 - 7.7x_6 - 7.6x_7 + x_8 = -2 \\ & x_1 + 3x_2 - 3x_3 - 2x_4 - 5x_5 - 5.5x_6 - 6x_7 + x_9 = -2 \\ & x_1,\ x_2,\ x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9 \geqslant 0 \end{aligned}$$

First, we choose $I_B = \{8,\ 9\}$ to be an index set of initial basis and construct the initial simplex tableau as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS
z	1	-2	-3	12	12	40	41.25	45.6	0	0	0
x_8	0	2	1	4	-1	-8	-7.7	-7.6	1	0	-2
x_9	0	1	3	-3	-2	-5	-5.5	-6	0	1	-2

Since $I_N = \{1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7\}$, we set $I_N' = I_N = \{1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7\}$ and $n = |I_N| = 7$. From this tableau, we have $\mathbf{P} = \{1,\ 2\}$ and $\mathbf{D} = \{1,\ 2\}$ which are nonempty sets, and $p = -2 < 0$ and $d = -3 < 0$. That is, the initial basis gives the primal and dual infeasible solutions. Thus, the zero-perturbation method is used to solve this problem. For this example, we use the largest-distance pivot rule to choose an entering variable. Then, the values of β_j for $j \in I_N$ are computed as the following table.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
β_j	-0.89	-0.95	2.4	5.37	4.24	4.36	4.71

In this table, the minimum value of β_j is $j = 2$ that is x_2 will be chosen to be the entering variable, but its column vector $\mathbf{y}_2 \geqslant \mathbf{0}$. Thus, it cannot enter the basis. Since $n = 7 \neq 0$, we can choose the new entering variable and set $n = n - 1 = 6$, $I_N' = \{1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7\} - \{2\} = \{1,\ 3,\ 4,\ 5,\ 6,\ 7\}$. The next nonbasic variable that has the minimum value is x_1, but the column vector $\mathbf{y}_1$ is also greater than or equal to zero. It cannot enter a basis, and we can choose the new variable since $n = 6 \neq 0$. Next, we set $n = 6-1 = 5$, $I'_N = \{1, 3, 4, 5, 6, 7\} - \{1\} = \{3, 4, 5, 6, 7\}$. After we consider the next minimum value of β_j for $j \in I_N$, we get β_3 is the minimum value. So, x_3 is chosen as the entering variable since there exists $y_{23} < 0$. Then, we set its reduced cost to zero while other negative reduced costs are set as positive constants. After that, we will choose a leaving variable by the maximum ratio test as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS	ratio
z	1	2	3	0	12	40	41.25	45.6	0	0	0	
x_8	0	2	1	4	-1	-8	-7.7	-7.6	1	0	-2	-
x_9	0	1	3	-3	-2	-5	-5.5	-6	0	1	-2	$(-2)/(-3) = 0.67$

From this tableau, we can see that there is only x_9 can leave the basis. Then, it is chosen to be the leaving variable, and the updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS
z	1	2	3	0	12	40	41.25	45.6	0	0	0
x_8	0	3.33	5	0	-3.67	-14.67	-15.03	-15.6	1	1.33	-4.67
x_3	0	-0.33	-1	1	0.67	1.67	1.83	2	0	-0.33	0.67

From the current tableau, there exists $\bar{b}_1 < 0$ then we set $I'_N = I_N = \{1, 2, 4, 5, 6, 7, 9\}$, $n = 7$ and choose the next entering variable by recalculating the β_j, $j \in I_N$ as the following table:

	x_1	x_2	x_4	x_5	x_6	x_7	x_9
β_j	0.6	1.77	1.07	1.35	1.27	1.37	2.91

In this table, x_1 has the minimum value of β_j, but its column vector which $\bar{b}_i < 0$ is nonnegative. Thus, it cannot enter the basis. Then, we will determine the new entering variable by the next minimum value of β_j, and we get x_4 which corresponds to this value. Since there exists $y_{14} < 0$, x_4 is chosen to be the entering variable. In addition, x_8 is the only basic variable which can leave the basis. So, the new tableau becomes:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS
z	1	2	3	0	0	40	41.25	45.6	0	0	0
x_4	0	-0.91	-1.36	0	1	4	4.1	4.25	1.09	5.45	1.27
x_3	0	0.27	-0.09	1	0	-1	-0.9	-0.84	0.18	-0.09	-0.18

Since the above tableau exists $\overline{b}_2 < 0$, we set $I'_N = I_N$, $n = 7$ and compute new β_j for $j \in I_N$. Then, we get x_6 as the entering variable and x_3 is the leaving variable. So, the updated tableau is as the follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS
z	1	2	3	0	0	40	0	45.6	0	0	0
x_4	0	0.33	-1.78	4.56	1	-0.56	0	0.44	0.56	-0.78	0.44
x_6	0	-0.3	0.1	-1.11	0	1.11	1	0.93	-0.2	0.1	0.2

From the above tableau, we can see that $\overline{\mathbf{b}} \geqslant \mathbf{0}$ then the basis gives the primal feasible solution. Then, the original objective function is restored, and the simplex method is performed until the optimal solution will be found. After the problem in this example was solved, the total number of iterations is 3. If we solve this problem by the two-phase simplex method and the original perturbation simplex method, then the results are concluded in the following table.

Table 3.4.2: Comparison of the number of iterations between the zero-perturbation simplex method, the perturbation simplex method and the simplex method.

Method		the size of matrix operation	the number of iteration	the total number of iteration
The zero-perturbation simplex method		3×10	3	3
The perturbation simplex		3×10	3	3
The two-phase simplex	Phase I	3×12	2	
	Phase II	3×10	4	6

From this table, we can see that the total number of iterations of the zero-perturbation simplex method is 3 which is less than or equal to the total number of the other methods. In addition, the size of matrix operation of the zero-perturbation simplex method is smaller than the phase I of the simplex method. Thus, we can reduce the computation.

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

In chapter 3, we designed the algorithm for solving linear programming problems without using artificial variable. In this chapter, we would like to test the efficiency of this algorithm. Then, we implemented the zero-perturbation simplex method, the original perturbation simplex algorithm and the two-phase simplex algorithm by using the Matlab program. In addition, the randomly generated problems were tested, and the average number of iterations of each algorithm were compared.

4.1 The tested problems

Consider the linear programming problem:

$$\begin{array}{lrcl} \max & \mathbf{c}^T\mathbf{x} & & \\ \text{s.t.} & \mathbf{A}\mathbf{x} & \leqslant & \mathbf{b}, \\ & \mathbf{x} & \geqslant & \mathbf{0}, \end{array} \tag{4.1.1}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, and $\mathbf{b} \in \mathbb{R}^m$.

For testing the efficiency of our algorithm, we randomly generated 100 tested problems of each size of the coefficient matrix $\mathbf{A}$ which are $m \times n = 10\times 10,\ 10\times 20,\ 10\times 30,\ 20\times 20,\ 20\times 30,\ 20\times 50,\ 30\times 10,\ 30\times 20,\ 30\times 30,$ and 40×40 according to this limitations:

1. a vector $\mathbf{c}$ with range of values between $c_j \in [-9,9]$ where $j = 1,\ldots,n$,
2. an element of the coefficient matrix with range of values between $a_{ij} \in [-9,9]$ where $i = 1,\ldots,m$ and $j = 1,\ldots,n$,
3. a solution vector $\mathbf{x}^*$ with range of values between $x_j^* \geqslant \mathbf{0}$ where $j = 1,\ldots,n$,
4. a vector $\mathbf{b}$ computed by $\mathbf{b} = \mathbf{A}\mathbf{x}^*$.

In addition, our tested problems have the number of artificial variables around 50 percent. The problem (4.1.1) is converted in the standard form by adding slacks variables which are chosen to be the initial basis. We used the Matlab program in version R2016b, 64-bit (maci64) for implementation the algorithm. These tests were run in MacBook Pro 8 GB 1867 MHz. We implemented the zero-perturbation simplex method according to three pivot rules that are the cosine rule, the largest-distance pivot rule and the Dantzig rule.

For the cosine rule, we modified the formula by computing $\alpha_j = \frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\|}$, for all $j = 1,...,n$ instead of $\theta_j = \arccos \frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\| \cdot \|\mathbf{b}\|}$, to choose an entering variable which has the maximum value first. Since the inverse function of cosine is a decreasing function, we can consider the maximum value of $\frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\| \cdot \|\mathbf{b}\|}$ where it is between -1 and 1. In addition, since $\|\mathbf{b}\|$ is a constant for $j = 1,...,n$, we can reduce the computation by considering the maximum value of the formula as follows:

$$\alpha_j = \frac{\mathbf{A}_{\cdot j}^T \cdot \mathbf{b}}{\|\mathbf{A}_{\cdot j}\|},$$

where $j = 1,...,n$.

Furthermore, we determined the reduced cost of the chosen entering variable; say x_k to zero and other negative reduced costs to $\delta_j = 1$, for all $j \in I_N - \{k\}$.

4.2 Computational results

According to the designed problems, we will show the comparison of the average number of iterations, the standard deviations of iterations, the ratio of the average number of iterations for the zero-perturbation simplex method according to three pivot rules to the average number of iterations for the simplex method and the original perturbation simplex method, and bar charts of the average number of iterations with standard deviations are presented.

First, the average number of iterations for the zero-perturbation simplex method according to three pivot rules, the original perturbation simplex method and the simplex method are shown in the following table.

Table 4.1: The average number of iterations for $m \times n$ with standard deviations.

Size	The average number of iterations with standard deviations									
	ZC		ZL		ZD		OP		CS	
	A	B	A	B	A	B	A	B	A	B
10x10	14.72	3.62	**7.41**	2.67	7.61	2.61	11.17	3.31	13.64	3.33
10x20	18.52	5.84	**6.41**	3.89	7.55	3.53	13.68	4.01	15.42	3.16
10x30	16.80	5.11	**5.85**	3.69	6.77	3.24	13.09	3.79	14.69	2.71
20x20	39.01	8.55	**23.07**	7.13	23.15	7.55	29.30	6.91	37.55	8.13
20x30	58.93	15.11	**30.80**	11.37	31.35	10.54	36.27	10.34	42.88	10.10
20x50	42.46	10.88	17.48	6.63	**16.56**	7.87	31.83	7.99	39.88	6.83
30x10	16.86	3.58	**8.27**	3.68	8.32	3.90	17.50	3.86	37.08	8.68
30x20	58.86	8.49	28.19	9.96	**26.58**	9.21	41.36	8.36	58.64	9.64
30x30	92.49	15.42	42.48	12.49	**41.76**	12.60	56.47	14.68	75.87	13.27
40x40	198.98	67.77	**69.74**	18.61	70.54	16.86	82.56	23.62	116.26	19.78

where A is the average number of iterations,

B is the standard deviation,

ZC is the zero-perturbation simplex method according to the cosine rule,

ZL is the zero-perturbation simplex method according to the largest-distance pivot rule,

ZD is the zero-perturbation simplex method according to the Dantzig rule,

OP is the original perturbation simplex method,

CS is the two-phase simplex method.

From Table 4.1, we can see that most of the minimum average number of iterations appear in the column of the zero-perturbation simplex method according to the largest-distance pivot rule while most of the maximum average number of iterations appear in the column of the zero-perturbation simplex method according to the cosine rule. Then, we can notice that the proposed method according to the largest-distance pivot rule is more effective than the other methods.

After that, the bar charts of the average number of iterations for all methods are plotted with standard deviations as Figure 4.1-4.4.

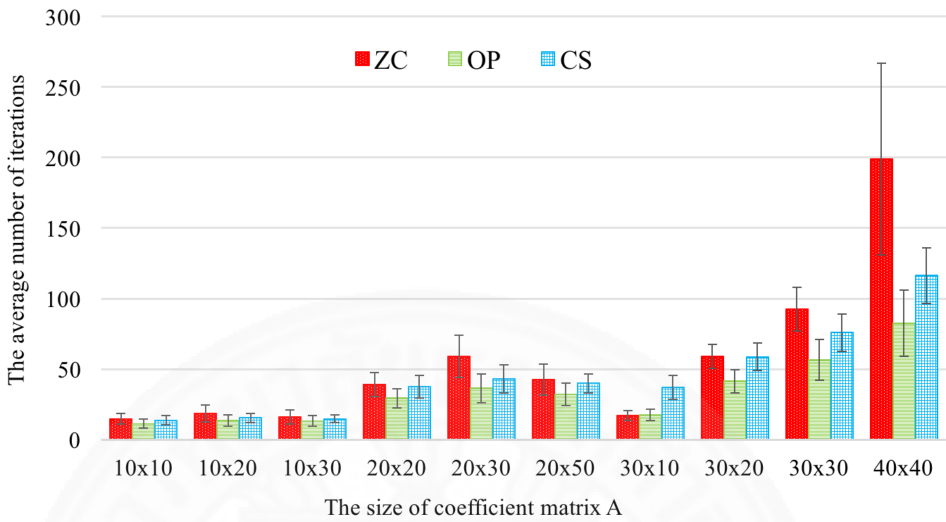

Figure 4.1: The average number of iterations for the zero-perturbation simplex method with the cosine rule, the original perturbation simplex method and the simplex method.

From Figure 4.1, we can see that the small tested problems have the low standard deviations. This implies that the average number of iterations is a good representation of the number of iterations. While the medium-sized tested problems have the high standard deviations, that is, the numbers of iterations may vary from the average. In this chart, most bars of the zero-perturbation simplex method are higher than other methods. It means that there are large variation using the cosine rule for these tested problems.

From Figure 4.2 and 4.3, we can see that all bars of the zero-perturbation simplex method according to the largest-distance pivot rule and the Dantzig rule are lower than the bars of other methods. That is, the average number of iterations for the zero-perturbation simplex method according to these pivot rules is less than other methods. Then, our method can reduce the number of iterations for these tested problems. For the small tested problem, we see that the standard deviations is low. It implies that the number of iterations for the small tested problems is not vary from the average. Then,

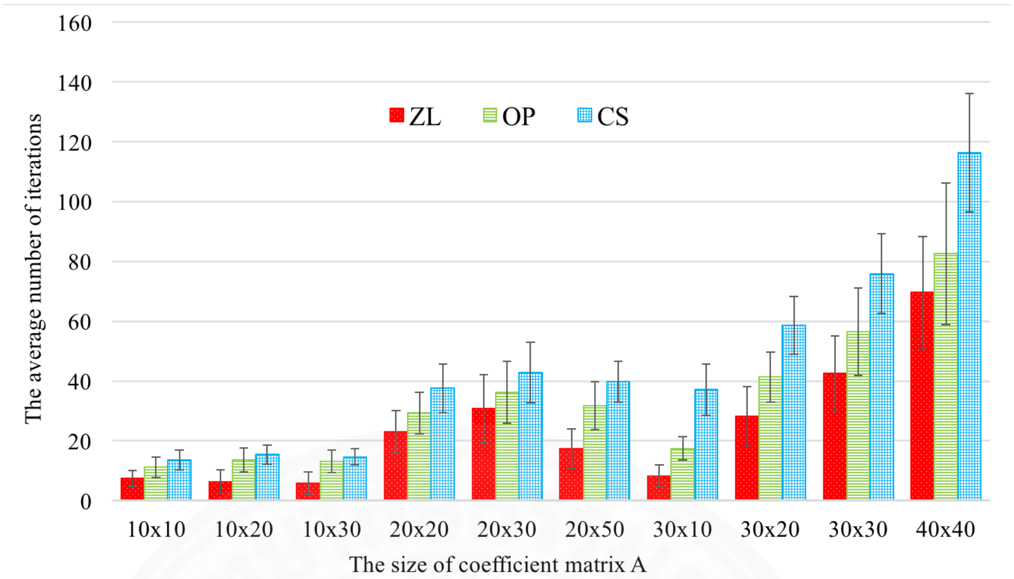

Figure 4.2: The average number of iterations for the zero-perturbation simplex method with the largest-distance pivot rule, the original perturbation simplex method and the simplex method.

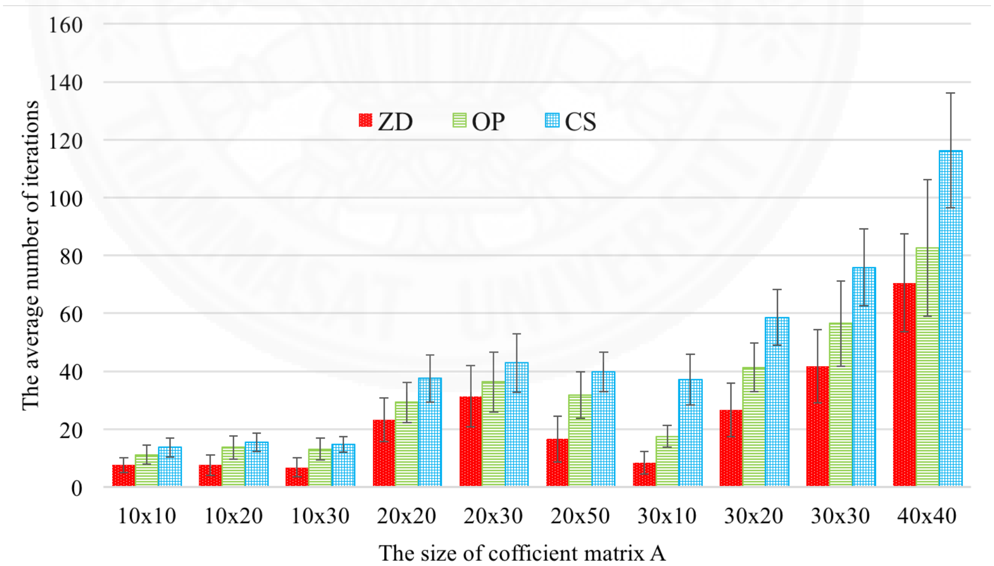

Figure 4.3: The average number of iterations for the zero-perturbation simplex method with the Dantzig rule, the original perturbation simplex method and the simplex method.

the average number of iterations for the small problems is a good representation of the number of iterations while the average number of iterations for the medium-size tested problems is an inadequate representation of the number of iterations.

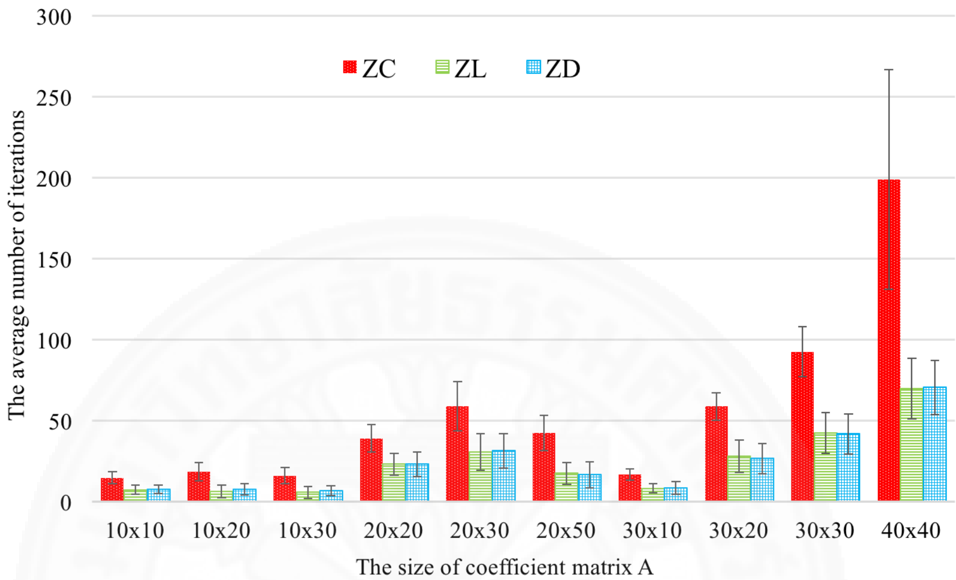

Figure 4.4: The average number of iterations for the zero-perturbation simplex method with three pivot rules.

From Figures 4.4, it shows the average number of iterations for the zero-perturbation simplex method according to three pivot rules. We can see that all bars of this method according to cosine rule are taller than other pivot rules, and most bars of this method according to the largest-distance pivot rule are shorter than other pivot rules. It means that our method according to the largest-distance pivot rule is effective than other pivot rules for these tested problems.

Table 4.3: The ratio of the average number of iterations.

Two-phase Simplex Method				Original Perturbation Simplex Method			
Problems	ZC/CS	ZL/CS	ZD/CS	Problems	ZC/OP	ZL/OP	ZD/OP
10x10	1.0792	0.5433	0.5579	10x10	1.3178	0.6634	0.6813
10x20	1.2010	0.4157	0.4896	10x20	1.3538	0.4686	0.5519
10x30	1.1436	0.3982	0.4609	10x30	1.2834	0.4469	0.5172
20x20	1.0389	0.6144	0.6165	20x20	1.3314	0.7874	0.7901
20x30	1.3743	0.7183	0.7311	20x30	1.6248	0.8492	0.8644
20x50	1.0647	0.4383	0.4152	20x50	1.3340	0.5492	0.5203
30x10	0.4472	0.2194	0.2207	30x10	0.9525	0.4672	0.4701
30x20	1.0038	0.4807	0.4533	30x20	1.4231	0.6816	0.6426
30x30	1.2191	0.5599	0.5504	30x30	1.6379	0.7523	0.7395
40x40	1.7115	0.5999	0.6067	40x40	2.4101	0.8447	0.8544
Average of total	1.2322	0.5297	0.5310	Total	1.6724	0.7189	0.7204

From Table 4.3, we can see that the ratios of the zero-perturbation simplex methods according to the largest-distance pivot rule and the Dantzig rule to the simplex method and the original perturbation simplex method are less than one. It means that the average number of iterations of such methods are less than the average number of iterations of the simplex method and the original perturbation simplex method. However, both ratios of the zero-perturbation simplex method with the cosine rule to the simplex method and the original perturbation simplex method are larger than one, that is, their average number of iterations are greater than the average number of iterations of the simplex method and the original perturbation simplex method.

From the average number of total iterations, we can see that our method according to the largest-distance pivot rule has more effective than the others.

4.3 Discussion

From the computational results, the average number of iterations for the zero-perturbation simplex method according to the cosine rule of all tested problems is greater than that of other pivot rules. Since the cosine rule considers the closest angle of constraints to the objective function in the dual problem, if such problem has a redundant constraint that is closest in angle to the objective function, then the primal entering variable corresponding to this redundant constraint is chosen first. Since the primal variable associated with the redundant constraint in the dual problem which should not be the nonbasic variable is chosen to enter the basis, the optimal solution is found slowly. This may be the reason why the number of iterations for the proposed method according to the cosine rule is greater than other pivot rules.

For Dantzig rule and the largest-distance pivot rule, the average numbers of iterations for the proposed method according to these pivot rules are similar since these pivot rules focus on the reduced costs. Dantzig rule considers only the reduced costs while the largest-distance pivot rule considers the normalized reduced costs. This might be the reason why the average numbers of iterations for these pivot rules are similar.

Next, we give the illustrative example for the reason why the cosine rule has low efficiency than the largest-distance pivot rule and Dantzig rule for the problem which its dual has redundant constraints.

Example 4.3.1. Consider the following linear programming problem.

$$\begin{array}{llcl} \max & -2x_1-3x_2+2x_3+9x_4+8x_5-9x_6 & & \\ \text{s.t.} & 6x_1-5x_2+9x_3-9x_4-8x_5-2x_6 \quad +x_7 & = & -68 \\ & 9x_1-x_2+2x_3+9x_4+9x_5+7x_6 \qquad\quad +x_8 & = & 200. \end{array} \tag{4.3.1}$$

First, we solve this problem with the zero-perturbation simplex method according to the cosine rule. We choose $I_B = \{7,\ 8\}$ as the index set of basic variables and construct the initial simplex tableau as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	-2	-9	-8	9	0	0	0
x_7	0	6	-5	9	-9	-8	-2	1	0	-68
x_8	0	9	-1	2	9	9	7	0	1	200

Next, the α_j for $j = 1, ..., n$ is computed as the following table.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
α_j	128.69	27.46	-22.99	189.51	194.66	210.99	-68	200

Then, x_6 has the maximum value of α_j, so it is chosen to be the entering variable. After that, x_7 is chosen to be the leaving variable by the maximum ratio. The updated tableau becomes:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	2	9	8	0	0	0	0
x_6	0	-3	2.5	-4.5	4.5	4	1	-0.5	0	34
x_8	0	30	-18.5	33.5	-22.5	-19	0	3.5	1	-38

Next, x_5 and x_8 are chosen to be the entering and leaving variables, respectively. The updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	2	9	0	0	0	0	0
x_6	0	3.32	-1.39	2.55	-0.24	0	1	0.24	0.21	26
x_5	0	-1.58	0.97	-1.76	1.18	1	0	-0.18	-0.05	2

From this tableau, the primal feasible is found. Then, the original coefficients of the objective function are restored into this tableau, and the simplex method is performed. Afterward, the sequence of the simplex path to the optimal solution is: $I_B = \{1, 5\}$, $I_B = \{3, 5\}$ and $I_B = \{3, 4\}$, respectively. Then, the dual problem of this example can be plotted as Figure 4.5. In addition, we can see the movement of the sequence of basis that corresponds to the dual constraints also in Figure 4.5, that is, the basis starts at the origin point and moves to point A, B, C, D and the optimal point, respectively.

From Figure 4.5, we can see that three dash lines are redundant constraints. If we can remove them, for the cosine rule, we will use only 2 iterations to solve this problem.

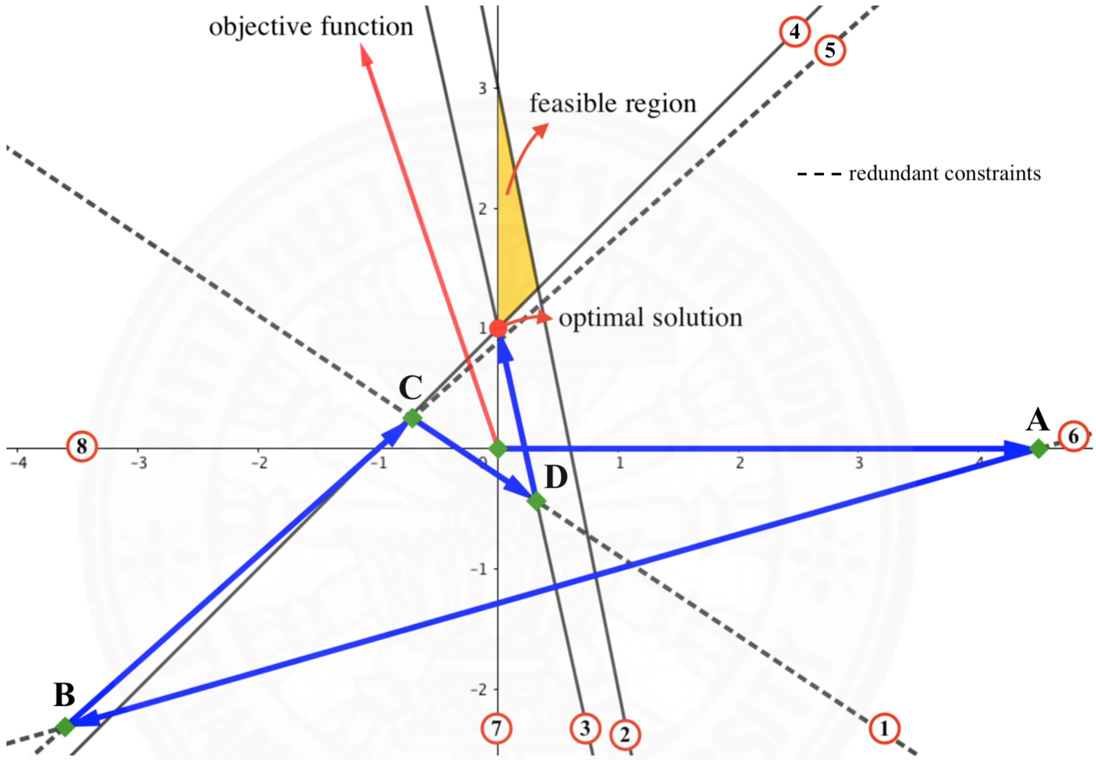

Figure 4.5: The sequence of basis of the zero-perturbation simplex method according to the cosine rule.

For the largest-distance pivot rule, we also choose $I_B = \{7, 8\}$ as the index set of basic variables and construct the initial simplex tableau as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	-2	-9	-8	9	0	0	0
x_7	0	6	-5	9	-9	-8	-2	1	0	-68
x_8	0	9	-1	2	9	9	7	0	1	200

Then, we compute the β_j for $j \in I_N$ as the following table.

	x_1	x_2	x_3	x_4	x_5	x_6
β_j	0.18	0.59	-0.22	-0.71	-0.66	1.24

From the above table, x_4 has the minimum value of β_j so it is chosen to be the entering variable. After the minimum ratio is computed, x_7 is chosen to be the leaving variable. Thus, the updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	2	0	8	9	0	0	0
x_4	0	-0.67	0.56	-1	1	0.89	0.22	-0.11	0	7.56
x_8	0	15	-6	11	0	1	5	1	1	132

From this tableau, the basis gives the primal feasible solution. Then, the original coefficients of the objective function are restored, and the simplex method is performed. Afterward, x_3 and x_8 are chosen to be the entering and leaving variables, respectively. The updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	11	2	3	0	1	16	0	1	200
x_4	0	0.7	0.01	0	1	0.98	0.68	-0.02	0.09	19.56
x_3	0	1.36	-0.55	0	0	0.09	0.455	0.09	0.09	12

From this tableau, we can see that the basis gives the primal and dual feasible solutions. So, it is the optimal tableau, and the optimal solution is $(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*, x_6^*) = (0, 0, 12, 19.56, 0, 0)$ with $z^* = 200$. In addition, we can see the movement of sequence of all basis corresponding to the dual constraints as Figure 4.6, that is, the initial basis starts at origin point and moves to point A and the optimal solution.

From Figure 4.6, we can see that the redundant constraints do not affect the number of iterations. However, the weak point of the largest-distance pivot rule is the new computation β_j for all $j \in I_N$ in each iteration while the cosine rule computes at once in the first iteration.

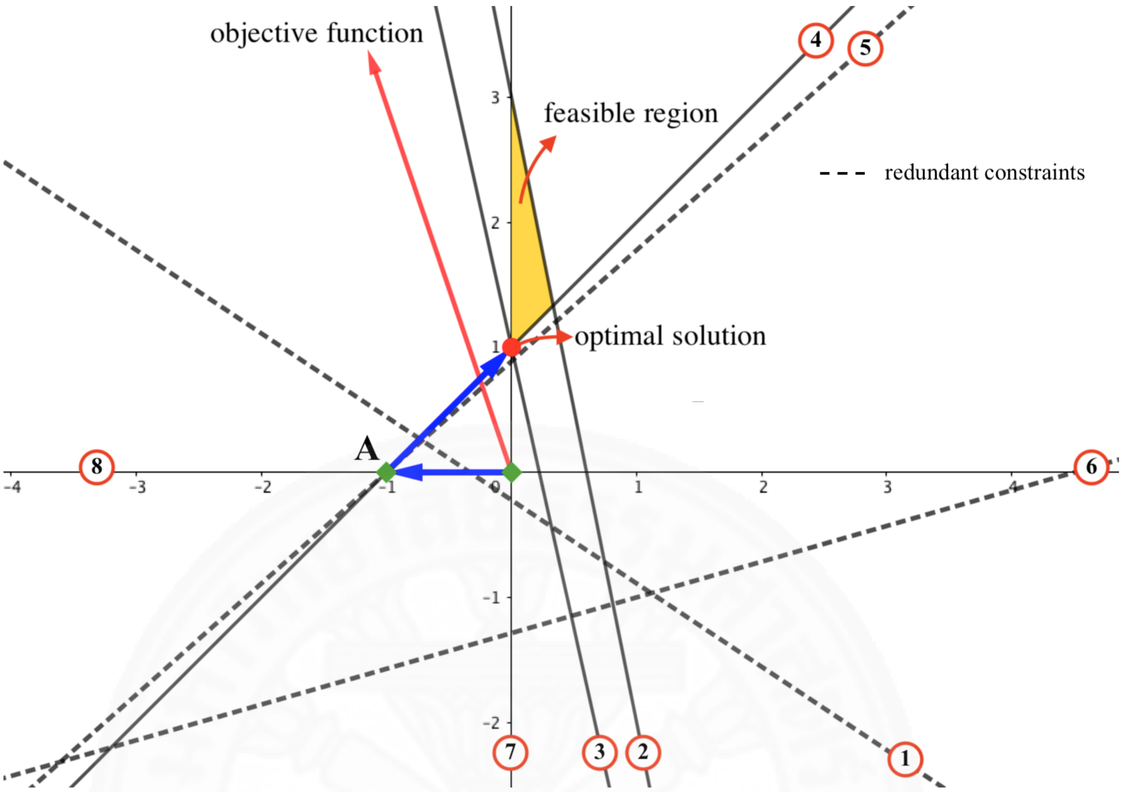

Figure 4.6: The sequence of basis the zero-perturbation simplex method according to the largest-distance rule.

For the Dantzig rule, we also choose $I_B = \{7,\ 8\}$ as the index set of basic variables and the initial simplex tableau is constructed as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	-2	-9	-8	9	0	0	0
x_7	0	6	-5	9	-9	-8	-2	1	0	-68
x_8	0	9	-1	2	9	9	7	0	1	200

Dantzig rule consider the minimum reduced cost to enter the basis. Then, x_4 is chosen to be an entering variable first, and x_7 is chosen to be a leaving variable by the minimum ratio test. The tableau is updated as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	2	3	2	0	8	9	0	0	0
x_4	0	-0.67	0.56	-1	1	0.89	0.22	-0.11	0	7.56
x_8	0	15	-6	11	0	1	5	1	1	132

From the above tableau, the basis gives the primal feasible solution. Then, the original coefficients of the objective function are restored, and the simplex method is performed. After that, x_3 and x_8 are chosen to be the entering and leaving variables, respectively. The updated tableau is as follows:

	z	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	RHS
z	1	11	2	3	0	1	16	0	1	200
x_4	0	0.7	0.01	0	1	0.98	0.68	-0.02	0.09	19.56
x_3	0	1.36	-0.55	0	0	0.09	0.455	0.09	0.09	12

From this tableau, we can see that the basis gives the primal and dual feasible solutions. So, it is the optimal tableau, and the optimal solution is $(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*, x_6^*) = (0,\ 0,\ 12,\ 19.56,\ 0,\ 0)$ with $z^* = 200$. In addition, we can see that the movement of sequence of all basis is similar to the largest-distance pivot rule. Then, we can see that the redundant constraints in the dual problem do not also affect the number of iterations. Therefore, the average numbers of iterations for the proposed method according to the largest-distance pivot rule and Dantzig rule are similar.

CHAPTER 5

CONCLUSIONS

In this thesis, we present the algorithm which improves Pan's algorithm [10] for solving a linear programming problem without using artificial variable. It will be used to solve a problem when an initial basis gives the primal and dual infeasible solutions called the zero-perturbation simplex method. The proposed method starts by choosing an entering variable according to a pivot rule, then its reduced cost is set to zero while the other negative reduced costs are set to a single positive value. Next, a leaving variable is chosen by the maximum ratio.

The zero-perturbation simplex method is an artificial-free method. Thus, the size of matrix operations of this method is smaller than the simplex method. Moreover, the proposed method does not need to update the row zero. From computational results, we found that the efficiency of the zero-perturbation simplex method depends on the pivot rule. Especially, the proposed method according to the largest-distance pivot rule and Dantzig rule can reduce the number of iterations.

For the future works, we would like to find the condition for eliminating the redundant constraints since the cosine rule is an effective pivot rule when a problem has no redundant constraint which is closest in angle to the objective function. In addition, we would like to design the effective pivot rule for the simplex method which it affects to choose an entering variable. Therefore, if we have the effective pivot rule, then we will find the optimal solution faster, and the number of iterations can be reduced.

REFERENCES

[1] N. Karmarkar, A new polynomial-time algorithm for linear programming, Combinatorica, 4(1984), 373-395.

[2] G.B. Dantzig, Linear Programming and Extensions, Princeton Univ. Press, Princeton, NJ, 6(1963).

[3] V. Klee, G. Minty, How good is the simplex algorithm, O. Shisha(Ed.), Inequalities-III, Academic Press, NY, (1972), 159-175.

[4] P.M.J. Harris, Pivot selection methods of the devex lp code, Mathematical Programming, 4(1975), 30-57.

[5] D. Goldfarb, J.K. Reid, A practicable stepest-edge simplex algorithm, Mathematical Programming, 12(1977), 361-371.

[6] R. G. Bland, Finite pivoting rules for simplex method. Mathematics of Operations Research, 2(1977), 103-107.

[7] H. Arsham, An artificial-free simplex-type algorithm for general LP models, Mathematical and Computer Modelling, 25(1997), 107-123.

[8] H. Arsham, Initialization of the simplex algorithm: an artificial-free approach, SIAM Review, 39(1997), 736-744.

[9] A. Enge, P. Huhn, A counterexample to H. Arsham Initialization of the simplex algorithm: an artificial-free approach, SIAM Review, 40(1998), (online).

[10] P.Q. Pan, Primal perturbation simplex algorithms for linear programming, Journal of Computational Mathematics, 18(2000), 587-596.

[11] H. Arsham, Big-M free solution algorithm for general linear programs, Indian Journal of Pure and Applied Mathematics, 32(2006), 37-52.

[12] H.W. Corley, J. Rosenberger, W.C. Yeh, T.K. Sung, The cosine simplex algorithm, International Journal of Advanced Manufacturing Technology, 27(2006), 1047-1050.

[13] H. Arsham, A computationally stable solution algorithm for linear programs, Applied Mathematics and Computation, 188(2007), 1549-1561.

[14] P.Q. Pan, A largest-distance pivot rule for the simplex algorithm, European Journal of Operational Research, 187(2008), 393-402.

[15] A. Boonperm, K. Sinapiromsaran, Artificial-free simplex algorithm based on the non-acute constraint relaxation, Applied Mathematics and Computation, 234(2014), 385-401.

[16] Y. Ma, L. Zhang, P.Q. Pan, Criss-cross algorithm based on the most-obtuse-angle rule and deficient basis, Applied Mathematics and Computation, 268(2015), 439-449.

[17] M.S. Bazaraa, J.J. Jarvis, H.D. Sherali, Linear Programming and Network Flows, 3rd, John Wiley & Sons, New York, 1990.

[18] C. Griffin, Linear Programming: Penn State Math 484 Lecture Notes Ver. 1.8.3, United States, 2009-2014.

[19] W.C. Yeh and H.W. Corley, The simple direct cosine simplex algorithm, Applied Mathematics and Computation, 214(2009), 178-186.

BIOGRAPHY

Name	Miss Panthira Jamrunroj
Date of Birth	October 21, 1993
Educational Attainment	Academic Year 2015: Bachelor of Science (Mathematics), Thammasat University, Thailand (Second-Class Honors)
	Academic Year 2017: Master of Science (Mathematics), Thammasat University, Thailand
Scholarships	2012-2017: Research Professional Development Project under the Science Achievement Scholarship of Thailand (SAST)

Publications

1. P. Jamrunroj, A. Boonperm, The zero-perturbation value for an entering variable based on cosine rule in the perturbation simplex method, Proceeding of Operation Research Network of Thailand (OR-NET 2018), (2018), 280-287.

2. P. Jamrunroj, A. Boonperm, Near-optimal relaxation for solving 2-dimensional linear programming problems, Proceeding of 10th International Conference on Advances in Sciences, Engineering and Technology (ICASET-2018), (2018), 21-26.

Oral Presentation

1. P. Jamrunroj, A. Boonperm, The zero-perturbation value for an entering variable based on cosine rule in the perturbation simplex method, in Operation Research Network of Thailand (OR-NET), 23-24 April 2018, Chonburi, Thailand.

Poster Presentation

1. P. Jamrunroj, A. Boonperm, Near-optimal relaxation for solving 2-dimensional linear programming problems, in 10th International Conference on Advances in Sciences, Engineering and Technology (ICASET-2018), 20-21 June 2018, Paris, France.