AN ARTIFICIAL-VARIABLE-FREE SIMPLEX METHOD BASED ON NEGATIVE RELAXATION OF DUAL PROBLEM

BY

MISS CHANISARA PRAYONGHOM

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

AN ARTIFICIAL-VARIABLE-FREE SIMPLEX METHOD BASED ON NEGATIVE RELAXATION OF DUAL PROBLEM

BY

MISS CHANISARA PRAYONGHOM

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 CHANISARA PRAYONGHOM

ENTITLED

AN ARTIFICIAL-VARIABLE-FREE SIMPLEX METHOD

BASED ON NEGATIVE RELAXATION OF DUAL PROBLEM

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 Wutiphol Sintunavarat, Ph.D.)

Member

(Adoon Pansuwan, Ph.D.)

Dean

(Associate Professor Somchai Chakhatrakan, Ph.D.)

Thesis Title	AN ARTIFICIAL-VARIABLE-FREE SIMPLEX METHOD BASED ON NEGATIVE RELAXATION OF DUAL PROBLEM
Author	Miss Chanisara Prayonghom
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 starts at a basic feasible solution and moves along the edge of a feasible region to the adjacent basic feasible solution until the optimal solution is found. For some problems, if a basic feasible solution could not be found easily, then artificial variables will be introduced for finding a basic feasible solution. Thus, the problem has more variables causing more computational time. In this thesis, we present the improvement of the simplex method for solving a linear programming problem without using artificial variables. By choosing an initial basis, if it gives a dual infeasible solution, then primal variables which cause its dual infeasible are relaxed. Then, the dual simplex method can be performed for finding the primal feasible solution. After the primal feasible solution is found, the relaxed variables will be restored and the simplex method starts. From the computational results, the average number of iterations solving by our method is less than the average number of iterations solving by the two-phase simplex method. Moreover, the CPU time for our method is also faster than the two-phase simplex method. So, our algorithm can reduce the computational time.

Keywords: Artificial variable, Dual simplex method, Relaxed problem

ACKNOWLEDGEMENTS

I would like to thanks to my thesis advisor, Dr. Aua-aree Boonperm for her invaluable help and constant encouragement throughout the course of this research.

In addition, I am grateful for Assistant Professor Dr. Krung Sinapiromsaran, Assistant Professor Dr. Wutiphol Sintunavarat, and Dr. Adoon Pansuwan for suggestions and all their help.

Moreover, I would like to thank for whole mater degree scholarship and supporting the fund for presentation in ICASET-18 in France from Faculty of Science and Technology, Thammasat University. I wish to thank the department of Mathematics and Statistics for supporting the fund for presentation in OR-NET 2018, Thailand.

Chanisara Prayonghom

TABLE OF CONTENTS

Page

ABSTRACT (i)

ACKNOWLEDGEMENTS (ii)

CHAPTER 1 INTRODUCTION 1

1.1 Linear programming problems 1

1.2 Literature reviews 3

1.3 Overviews 5

CHAPTER 2 PRELIMINARIES 6

2.1 Theoretical backgrounds 6

2.1.1 Convex sets and convex functions 6

2.1.2 Polyhedral sets 7

2.1.3 Extreme points and extreme directions 8

2.1.4 Representation of polyhedral sets 9

2.1.5 Optimality 9

2.2 The Simplex method 10

2.2.1 Basic feasible solutions (BFS) 10

2.2.2 Algebra of the simplex method 11

2.2.3 The simplex method in tableau format 13

2.2.4 The initial basic feasible solution 16

2.2.5 Two-phase method 18

2.2.6 The Big-M method 23

2.2.7 Duality 26

2.2.8 The Karush-Kuhn-Tucker (KKT) optimality conditions 31

2.3 The dual simplex method 32

CHAPTER 3 THE PROPOSED METHOD 36
3.1 Negative relaxation of dual problem 36
3.2 Negative relaxation of dual problem algorithm 40

CHAPTER 4 EXPERIMENTAL RESULTS 50

CHAPTER 5 CONCLUSIONS 54

REFERENCES 55

BIOGRAPHY 57

CHAPTER 1

INTRODUCTION

1.1 Linear programming problems

For some real-world problems such as assignment problems, transportation problems, and inventory problems can be formulated as a linear programming model which containing three important components.

- Decision variables: physical quantities controlled by the decision maker and represented by mathematical symbols.
- Objective function: the criterion for evaluating the solution. It is a linear combination of the decision variables .
- Constraints: a set of linear equality or inequality functions that represent physical, economic, technological, legal, ethical, or other restrictions on what numerical values assigned to the decision variables.

Therefore, the linear programming model can be written as follows:

$$\begin{aligned} &\text{minimize} \quad \text{or} \quad \text{maximize} \quad z = \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ &\text{subject to} \qquad \mathbf{A}\mathbf{x} = \mathbf{b}, \\ &\qquad\qquad\qquad\quad \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{1.1}$$

where $\mathbf{c} \in \mathbb{R}^n$ is a column vector of coefficients of the objective function,

$\mathbf{x} \in \mathbb{R}^n$ is a column vector of decision variables,

$\mathbf{b} \in \mathbb{R}^m$ is a column vector of parameters called the right-hand-side vector,

$\mathbf{A} \in \mathbb{R}^{m \times n}$ is a coefficient matrix of constraints.

For the model (1.1), let $X = \{\mathbf{x} | \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$. A vector $\mathbf{x}'$ is said to be a **feasible solution** or a **feasible point** if it satisfies all constraints, that is $\mathbf{x}' \in X$. A set of all feasible points is called **the feasible region**. After we construct the model for a linear programming problem, there are three possible outcomes that are summarized below.

- **Optimal solution**: an optimal solution to a linear programming problem is a feasible solution with the largest objective function value for maximization problem. A linear programming problem may have multiple optimal solutions, but it has only one optimal objective value.
- **Unbounded optimal solution**: A linear programming problem is unbounded if the objective value can be increased along the feasible direction for maximization problems or be decreased along the feasible direction for minimization problems.
- **Empty feasible region or infeasible**: A linear programming problem is infeasible if it has no feasible solutions, i.e., the feasible region is empty.

For getting three outcomes above, there are two popular methods for solving a linear programming problem; the interior point method presented by Karmarkar [1] and the simplex method presented by Dantzig [2]. The interior point method is suitable for solving large problems with sparse matrix. On the other hand, the simplex method is not suitable for large problems because the running time on the simplex method is exponential, see example from Klee and Minty [3] in 1972. However, a small-medium size problem and dense matrix, the simplex method is more efficient than the interior point method for solving a linear programming problem. Therefore, the simplex method is still in-used and improved. Many researchers have attempted to improve it in several ways such as improving pivoting rule, elimination of the redundant constraint, solving without using artificial variables.

Solving without using artificial variables is interesting because the problem size does not increased. The simplex method starts at a vertex in the feasible region and move along the edge of the feasible region to the adjacent vertex in a feasible region until the optimal solution is found. If the simplex method can not start at a vertex in the feasible region, artificial variables are introduced as Figure 1.1.

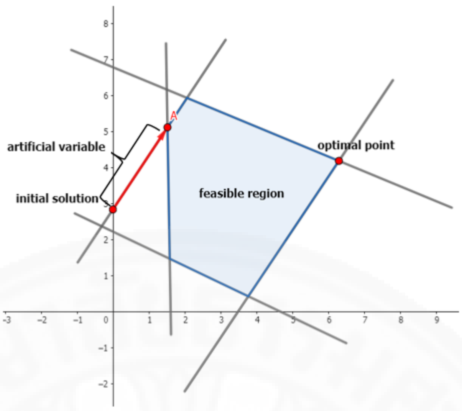


Figure 1.1: The artificial variable is introduced

From Figure 1.1, the initial solution is out of the feasible region. Then, an artificial variable is introduced for assuming this initial point feasible. Therefore, the simplex method can start and moves to the adjacent vertex which is in the feasible region.

However, adding artificial variables increases the number of variables and takes longer time to solve. Consequently, many reseachers have tried to improve the simplex method without using artificial variables.

1.2 Literature reviews

In 1997, Arsham [4] presented the simplex method without using artificial variables. In the first step, the basic feasible variable set (BVS) is determined to be the empty set. Then, the nonbasic varible is chosen to be the basic variable one by one until the BVS is full. After the problem has the complete BVS, the simplex method is performed. However, this method has the mistake as shown by Enge and Huhn [5] in 1998.

In 2000, Pan [6] proposed the simplex method by avoiding artificial variables.

The algorithm starts when the initial basis gives primal and dual infeasible solutions by adjusting negative reduced costs to a single positive value. Then, the dual solution is feasible and the dual simplex method is performed. After the optimal solution is found in this step, the original reduced costs are restored and the simplex method is performed.

Later, in 2006, Corley et.al [7] constructed the relaxed problem for improving the simplex method without using artificial variables. The cosine criterion is used for choosing the suitable constraints to construct the relaxed problem. Then, it is solved by the simplex method. After the optimal solution of the relaxed problem is found, the relaxed constraints will be restored into the current tableau, and the dual simplex method will be performed until the optimal solution is found. However, this algorithm can solve only the problem which has all positive coefficients.

In 2007, Arsham [8] still improved the algorithm without using the artificial variables by constructing the relaxed problem. The algorithm starts by relaxing greater-than or equal to constraints to avoid the use of the artificial variable. After the optimal solution of the relaxed problem is found, the relaxed constraints are restored, and the simplex method is performed. However, if the problem contains only greater-than or equal to constraints the perturbation simplex method will be used.

Later, in 2014, Boonperm and Sinapiromsaran [9] proposed the non-acute constraint relaxation technique that improves the simplex method without using artificial variables, and it can reduce the start-up time to solve the initial relaxation problem. The algorithm starts by relaxing the non-acute constraints which it can guarantee that the relaxed problem is always feasible. So, the relaxed problem can be solved without using artificial variables. After the optimal solution of the relaxed problem is found, the relaxed constraints are restored, and the dual simplex method is used to solve it. However, this algorithm is slow when the relaxed problem is unbounded.

Due to the above researches, to construct the relaxed problem without using artificial variables is interested for improving the simplex method. Since the relaxed problem can reduce variables or constriants, the computation can be reduced. Moreover, if the optimal solution is found in the relaxed problem, then the computational time can be reduced extremely. From the research of Arsham, the relaxed problem in the primal problem is constructed but the algorithm starts at only the origin point which is far from the optimal solution for some problems. Therefore, the research question is that can we construct the relaxed problem without using artificial variables by considering the dual problem? Since the constraint in dual is associated to the variable in primal, if the constraint in dual are relaxed, then the variable in primal will be relaxed instead. Therefore, solving without using artificial variables and relaxing some variables in primal are the reduction of the problem size. Hence, in this thesis, we present the improvement of the simplex method for a linear programming problem without using the artificial variables by relaxing some variables. First, we choose an initial basis. If the initial basis gives primal and dual infeasible solutions, then variables which cause its dual infeasible are relaxed. Then, the dual simplex method can be performed. By this relaxing, we can aviod the use of artificial variables. Therefore, the aim of this thesis is to reduce the computation time for solving linear programming problems.

1.3 Overviews

Our main purpose is to propose the algorithm for solving linear programming problems by the simplex method without using artificial variables. Next, we are going to decribe the contents of the thesis.

In chapter 2, definitions and theorems which are relevant to our algorithm are presented. In addition, methods for solving a linear programming problems that are the simplex method, the dual simplex method are decribed.

In chapter 3, the main idea and the step of our algorithm are presented.

In chapter 4, the efficiency of our algorithm is shown by testing with randomly generated problems, and the computational time is presented.

In the last chapter, we conclude and discuss the findings.

CHAPTER 2

PRELIMINARIES

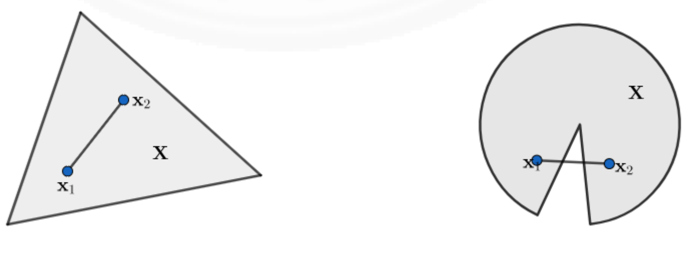
In this chapter, we will divide the contents into two important parts that are the theoretical part and methods for solving a linear programming problem. In the theoretical part, the theory behind linear programming problems including the convex set, the polyhedral set, extreme point, the representation of polyhedral set and optimality are decribed. While, the well-known methods for solving a linear programming problem that are the simplex method and the dual simplex method are described later.

2.1 Theoretical backgrounds

In this section, definitions and theorems which are the foundations for solving a linear programming problem are explained.

2.1.1 Convex sets and convex functions

Definition 2.1.1. The set $X \subseteq \mathbb{R}^n$ is said to be **convex** if and only if for all $\mathbf{x}_1, \mathbf{x}_2 \in X$, we have $\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2 \in X$ for all $\lambda \in [0,1]$.



convex set non-convex set

Figure 2.1: Example of convex and non-convex sets

Definition 2.1.2. Let $\mathbf{x}_1,...,\mathbf{x}_m \in \mathbb{R}^m$ and $\lambda_i \in [0,1]$ for all $i = 1,2,...,m$ with $\lambda_1 + ... + \lambda_m = 1$. A vecter

$$\mathbf{x} = \textstyle\sum_{i=1}^{m} \lambda_i \mathbf{x}_i = \lambda_1 \mathbf{x}_1 + ... + \lambda_m \mathbf{x}_m$$

is called a **convex combination** of $\mathbf{x}_1,...,\mathbf{x}_m$. If $\lambda_i \in (0,1)$ for all $i = 1,...,m$, then it is called a **strict convex combination**.

Definition 2.1.3. A function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ is called a **convex function** if it satisfies:

$$f(\lambda x_1 + (1-\lambda)\mathbf{x}_2) \leq \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2)$$

for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$ and for all $\lambda \in [0,1]$.

2.1.2 Polyhedral sets

Definition 2.1.4. Let $\mathbf{a} \in \mathbb{R}^n$ be a constant vector and let $b \in \mathbb{R}$ be a constant scalar. The set of points

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

is called a **hyperpane** in n-dimensional space.

Definition 2.1.5. Let $\mathbf{a} \in \mathbb{R}^n$ be a constant vector and let $b \in \mathbb{R}$ be a constant scalar. The set of points

$$H_l = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{a}^{\mathbf{T}}\mathbf{x} \leq b\},$$
$$H_u = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{a}^{\mathbf{T}}\mathbf{x} \geq b\}$$

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

Definition 2.1.6. 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. Consider the set of half-spaces:

$$H_i = \{\mathbf{x} \in \mathbb{R}^n | a_i^{\mathbf{T}}\mathbf{x} \leq b_i\}, i = 1,2,...,m.$$

Then, the set

$$P = \bigcap_{i=1}^{m} H_i$$

is called a **polyhedral set**.

Definition 2.1.7. Let $\mathbf{x}_0 \in \mathbb{R}^n$ be a point and let $\mathbf{d} \in \mathbb{R}^n$ be a vector called the **direction**. Then the **ray** with vertex $\mathbf{x}_0$ and direction $\mathbf{d}$ is the collection of points

$$\{\mathbf{x} | \mathbf{x} = \mathbf{x}_0 + \lambda \mathbf{d}, \lambda \geq 0\}.$$

Theorem 2.1.1. Suppose that $P \subseteq \mathbb{R}^n$ is a polyhedral set defined by

$$P = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}.$$

If $\mathbf{d}$ is a direction of P, then the following holds:

$$\mathbf{A}\mathbf{d} \leq 0, \mathbf{d} \geq 0, \mathbf{d} \neq 0.$$

2.1.3 Extreme points and extreme directions

Definition 2.1.8. Let $C \in \mathbb{R}^n$ be the convex set. A point $\mathbf{x}_0 \in C$ is called **an extreme point** of C if there are no points $\mathbf{x}_1$ and $\mathbf{x}_2$ so that

$$\mathbf{x}_0 = \lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2 \text{ for some } \lambda \in (0, 1).$$

Definition 2.1.9. Let $P \subseteq \mathbb{R}^n$ be a polyhedral set and suppose P is defined as:

$$P = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{A}\mathbf{x} \ \leq \mathbf{b}\}$$

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

Definition 2.1.10. Let $C \subseteq \mathbb{R}^n$ be a convex set. Then a direction $\mathbf{d}$ of C is called an **extreme direction** if there are no two other directions $\mathbf{d}_1$ and $\mathbf{d}_2$ of C and scalars $\lambda_1, \lambda_2 > 0$ so that $\mathbf{d} = \lambda_1 \mathbf{d}_1 + \lambda_2 \mathbf{d}_2$.

Theorem 2.1.2. A direction $\mathbf{d} \in D$ is an extreme direction of D if and only if $\mathbf{d}$ is an extreme point of D when D is taken as a polyhedral set.

2.1.4 Representation of polyhedral sets

The representation of a polyhedral set is described in terms of extreme points and extreme directions.

Theorem 2.1.3. Let P be a nonempty unbounded polyhedral set defined by:

$$P = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{Ax} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}.$$

Suppose that P has extreme points $\mathbf{x}_1, ..., \mathbf{x}_k$ and extreme directions $\mathbf{d}_1, ..., \mathbf{d}_l$. If $\mathbf{x} \in P$, then there exists constants $\lambda_1, ..., \lambda_k$ and $\mu_1, ..., \mu_l$ such that:

$$\begin{aligned} \mathbf{x} &= \textstyle\sum_{i=1}^{k} \lambda_i \mathbf{x}_i + \sum_{j=1}^{l} \mu_j \mathbf{d}_j, \\ \textstyle\sum_{i=1}^{k} \lambda_i &= 1, \\ \lambda_i &\geq 0, i = 1, 2, ..., k, \\ \mu_j &\geq 0, j = 1, 2, ..., l. \end{aligned}$$

Note: Every feasible point can be written as a linear combination of the extreme points and extreme directions.

2.1.5 Optimality

Consider the following linear programming problem:

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \text{subject to} \quad & \mathbf{Ax} \leq \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{2.1}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\text{rank}(\mathbf{A}) = m$.

Theorem 2.1.4. If the problem (2.1) has an optimal solution, then the problem (2.1) has an optimal extreme point.

Corollary 2.1.5. Problem (2.1) has a nonempty feasible set and a finite-value optimal solution if and only if $\mathbf{c^T d}_i \leq \mathbf{0}$ for all $i = 1, 2, ..., l$ when $\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_l$ are the extreme directions of the polyhedral set. Otherwise, the optimal solution value is unbounded.

Corollary 2.1.6. Problem (2.1) has alternative optimal solutions if there are at least two extreme points $\mathbf{x}_p$ and $\mathbf{x}_q$ so that $\mathbf{c^T x}_p = \mathbf{c^T x}_q$ and so that $\mathbf{x}_p$ is the extreme point solution to the linear programming problem.

2.2 The Simplex method

The simplex method is a method that starts from a basic feasible solution (BFS) of a linear programming problem expressed in the tableau form to another BFS, in such a way as to continually increase (or decrease) the value of the objective function until the optimal point is found.

2.2.1 Basic feasible solutions (BFS)

The basic feasible solution is defined as the following definition.

Definition 2.2.1. Consider the system

$$\mathbf{Ax} = \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0},$$

where $\mathbf{A}$ is an $m \times n$ matrix, $n > m$, $\mathbf{b}$ is an m-vector.

Suppose that $\text{rank}(\mathbf{A}, \mathbf{b}) = \text{rank}(\mathbf{A}) = m$. After possibly rearranging the columns of $\mathbf{A}$, let $\mathbf{A} = [\mathbf{B,N}]$ where $\mathbf{B}$ is an $m \times m$ invertible matrix and $\mathbf{N}$ is an $m \times (n-m)$ matrix.

The solution $\mathbf{x} = \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix}$ to the equation $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{x_B} = \mathbf{B}^{-1}\mathbf{b}$ and $\mathbf{x_N} = 0$ is called **a basic solution** of the system.

- If $\mathbf{x_B} = \mathbf{B}^{-1}\mathbf{b} \geq 0$, then $\mathbf{x}$ is called a **basic feasible solution (BFS)** of the system.

- Here $\mathbf{B}$ is called the **basic matrix** (or simply the basis) and $\mathbf{N}$ is called the **non-basic matrix**.
- The components of $\mathbf{x_B}$ are called **basic variables** and the components of $\mathbf{x_N}$ are called **nonbasic variables**.
- If $\mathbf{x_B} > 0$, then $\mathbf{x}$ is called a **nondegenerate basic feasible soultion**, and if at least one component of $\mathbf{x_B}$ is zero, then $\mathbf{x}$ is called a **degenerate basic feasible solution**.

Theorem 2.2.1. Let P=$\{\mathbf{x} \in \mathbb{R}^n : \mathbf{Ax} = \mathbf{b}\}$. Then $\mathbf{x}$ is an extreme point of P if and only if $\mathbf{x}$ is a basic feasible solution of P.

Since the extreme point is associated a basic feasible solution, the simplex method starts by choosing a basic feasible solution only.

2.2.2 Algebra of the simplex method

Consider the following linear programming problem:

$$\begin{aligned} \text{maximize} \quad & z = \mathbf{c^T x} \\ \text{subject to} \quad & \mathbf{Ax} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned}$$

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

Let $\mathbf{A} = [\mathbf{A}_{:1}, \mathbf{A}_{:2}, ..., \mathbf{A}_{:n}]$ and $\mathbf{A} = [\mathbf{B}, \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m\times m}$ is a nonsigular matrix, $\mathbf{N} \in \mathbb{R}^{m\times(n-m)}$, $\mathbf{I_B}$ is an index set of basic variables and $\mathbf{I_N}$ is an index set of nonbasic variables.

Suppose that $\mathbf{x} = \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix} = \begin{bmatrix} \mathbf{B}^{-1}\mathbf{b} \\ 0 \end{bmatrix}$ is a basic feasible solution. So the objective value is $z_0 = \mathbf{c^T x} = [\mathbf{c_B^T}, \mathbf{c_N^T}] \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix} = \mathbf{c_B^T x_B} + \mathbf{c_N^T x_N} = \mathbf{c_B^T B}^{-1}\mathbf{b}$.

Consider the constraints

$$\mathbf{Ax} = [\mathbf{B}, \mathbf{N}] \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix} = \mathbf{b}$$

$$\begin{aligned} \mathbf{Bx_B} + \mathbf{Nx_N} &= \mathbf{b} \\ \mathbf{x_B} + \mathbf{B}^{-1}\mathbf{Nx_N} &= \mathbf{B}^{-1}\mathbf{b} \\ \mathbf{x_B} &= \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{Nx_N}. \end{aligned}$$

Consider the objective function,

$$z = \mathbf{c^T x} = [\mathbf{c_B^T}, \mathbf{c_N^T}] \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix} = \mathbf{c_B^T x_B} + \mathbf{c_N^T x_N}.$$

Substitute $\mathbf{x_B} = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{Nx_N}$ in to the objective function, we get

$$\begin{aligned} z &= \mathbf{c_B^T}(\mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{Nx_N}) + \mathbf{c_N^T x_N} \\ &= \mathbf{c_B^T B}^{-1}\mathbf{b} - \mathbf{c_B^T B}^{-1}\mathbf{Nx_N} + \mathbf{c_N^T x_N} \\ &= \mathbf{c_B^T B}^{-1}\mathbf{b} - (\mathbf{c_B^T B}^{-1}\mathbf{Nx_N} - \mathbf{c_N^T x_N}) \\ &= \mathbf{c_B^T B}^{-1}\mathbf{b} - (\mathbf{c_B^T B}^{-1}\mathbf{N} - \mathbf{c_N^T})\mathbf{x_N} \\ &= \mathbf{c_B^T B}^{-1}\mathbf{b} \ - \sum_{j \in \mathbf{I_N}} (\mathbf{c_B^T B}^{-1}\mathbf{A}_{:j} - c_j)x_j. \end{aligned}$$

Let $\mathbf{y}_j = \mathbf{B}^{-1}\mathbf{A}_{:j}, z_j = \mathbf{c_B^T y}_j$ where $j \in \mathbf{I_N}$, $z_0 = \mathbf{c_B^T B}^{-1}\mathbf{b}$, and $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b}$. So, the linear programming problem can be written as follows:

$$\begin{aligned} \text{maximize} \quad & z = z_0 - \sum_{j \in \mathbf{I_N}} (z_j - c_j)x_j \\ \text{subject to} \quad & \mathbf{x_B} + \sum_{j \in \mathbf{I_N}} (\mathbf{y}_j)x_j = \overline{\mathbf{b}}, \\ & x_j \geq 0, j \in \mathbf{I_N}, \mathbf{x_B} \geq 0. \end{aligned} \tag{2.2}$$

Consider the value $z_j - c_j, j \in \mathbf{I_N}$.

- If $z_j - c_j \geq 0$ for all $j \in \mathbf{I_N}$, then the objective value decreases when x_j increases. So x_j should not be increased, that is the current basic feasible solution is **the optimal solution**.

- If $z_k - c_k < 0$ for some $k \in \mathbf{I_N}$, then the objective value increases when x_k increases. So x_k should be increased.

Consider $z = z_0 - (z_k - c_k)x_k$, how far to increase the entering basic variable x_k before stopping.

Consider the k^{th} column of the constraints,

$$(\mathbf{y}_k)x_k + \mathbf{x_B} = \overline{\mathbf{b}},$$

where $\overline{\mathbf{b}}^{\mathbf{T}} = [\overline{b}_1, \overline{b}_2, ..., \overline{b}_m]$.

- If $\mathbf{y}_k \leq 0$, then $x_{\mathbf{B}_i}$ increases when x_k increases. So x_k can be increased to infinity because $x_{\mathbf{B}_i}$ is still nonnegative. Therefore, the problem is unbounded.

- If $y_{ik} > 0$ for some $i = 1, 2, ..., m$, then $x_{\mathbf{B}_i}$ decreases when x_k increases. So x_k can be increased for $x_{\mathbf{B}_i} = \overline{b}_i - y_{ik}x_k \geq 0$, that is $x_k = \frac{\overline{b}_i}{y_{ik}}$ for $y_{ik} > 0$.

For feasibility, all variables must be nonnegative, x_k should be as follows:

$$x_k = \frac{\overline{b}_i}{y_{ik}} = \underset{1 \leq i \leq m}{\text{minimum}} \{ \frac{\overline{b}_i}{y_{ik}} : y_{ik} > 0 \}.$$

This is referred to as the **minimum ratio test**.

When we increase $x_k \leq \frac{\overline{b}_i}{y_{ik}}$, $x_{\mathbf{B}_i}$ decreases to zero. So x_k will enter to be a basic variable called an **entering variable** and $x_{\mathbf{B}_r}$ will leave to be a nonbasic variable called a **leaving variable**.

2.2.3 The simplex method in tableau format

Since the simplex method deals with the matrix in each iteration, for convenience, it will be represented in the tableau format.

Consider the following linear programming problem:

$$\text{maximize} \quad z = \mathbf{c}^{\mathbf{T}}\mathbf{x}$$

$$\text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0},$$

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

Let $\mathbf{A} = [\mathbf{B}, \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m\times m}$is a nonsigular matrix, $\mathbf{N} \in \mathbb{R}^{m\times(n-m)}$ and suppose that $\mathbf{x} = \begin{bmatrix} \mathbf{x_B} \\ \mathbf{x_N} \end{bmatrix} = \begin{bmatrix} \mathbf{B}^{-1}\mathbf{b} \\ 0 \end{bmatrix}$ is a basic feasible solution. Then, the linear programming problem can be rewritten as follows:

$$\begin{aligned} \text{maximize} \quad & z + \sum_{j\in\mathbf{I_N}} (z_j - c_j)x_j = z_0 \\ \text{subject to} \quad & \mathbf{I}\mathbf{x_B} + \sum_{j\in\mathbf{I_N}} (\mathbf{y}_j)x_j = \overline{\mathbf{b}}_i, \\ & \mathbf{x_B} \geq 0, x_j \geq 0, j \in \mathbf{I_N}. \end{aligned}$$

Fill all coefficients into the following tableau:

	z	$\mathbf{x_B}$	$\mathbf{x_N}$	RHS
	1	0	$z_j - c_j = \mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{A}_{:j} - c_j, \forall j \in \mathbf{I_N}$	$z_0 = \mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{b}$
$\mathbf{x_B}$	0	$\mathbf{I}$	$\mathbf{B}^{-1}\mathbf{N}$	$\mathbf{B}^{-1}\mathbf{b}$

This tableau is called **the initial tableau** for basis $\mathbf{B}$. The simplex algorithm can be summarized as follows:

The simplex method (maximization problem)

Initialization Step

Find an initial basic feasible solution with basis $\mathbf{B}$ and compute the initial tableau.

Main Step

1. Let $z_k - c_k = \underset{j\in\mathbf{I_N}}{\text{minimum}}\{z_j - c_j\}$ (**Dantzig Pivot rule**)

(a) If $z_k - c_k \geq 0$, then stop; the current solution is optimal.

(b) Otherwise, $z_k - c_k < 0$, examine $\mathbf{y}_k$ in step 2.

2. Examine $\mathbf{y}_k$ (coefficient in the kth column)

(a) If $\mathbf{y}_k \leq 0$, then stop; the optimal objective value is unbounded.

(b) Otherwise, that is $\mathbf{y}_k \nleq 0$, determine the index r as follows:

$$\frac{\bar{b}_r}{y_{rk}} = \underset{1 \leq i \leq m}{\text{minimum}} \{ \frac{\bar{b}_i}{y_{ik}} : y_{ik} > 0 \}.$$

3. Update the tableau by pivoting at y_{rk}. Update the basic and nonbasic variables where x_k enters the basis and $x_{\mathbf{B}_r}$ leaves the basis, and repeat the main step.

Remark 2.2.2. For minimization problem, we change only in step 1 as follows:

1. Let $z_k - c_k = \underset{j \in \mathbf{I_N}}{\text{maximum}} \{ z_j - c_j \}$ (**Dantzig Pivot rule**)

(a) If $z_k - c_k \leq 0$, then stop; the current solution is optimal.

(b) Otherwise, $z_k - c_k > 0$, examine $\mathbf{y}_k$ in step 2.

Example 2.2.3. Consider the following linear programming problem:

$$\begin{array}{lrcrcrcrcrcrcr}
\text{maximize} & x_1 & + & 5x_2 & - & 7x_3 & & & & & & & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & + & x_4 & & & & & = & 4 \\
& 2x_1 & - & x_2 & + & x_3 & & & + & x_5 & & & = & 5 \\
& -5x_1 & - & 2x_2 & + & 4x_3 & & & & & + & x_6 & = & 10 \\
& x_1, & & x_2, & & x_3, & & x_4, & & x_5, & & x_6 & \geq & 0.
\end{array}$$

From the above problem, we get

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & -1 & 1 & 0 & 1 & 0 \\ -5 & -2 & 4 & 0 & 0 & 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 4 \\ 5 \\ 10 \end{bmatrix}, \mathbf{c}^{\mathbf{T}} = \begin{bmatrix} 1 & 5 & -7 & 0 & 0 & 0 \end{bmatrix}.$$

If we choose $\mathbf{B} = \mathbf{I} = \mathbf{B}^{-1}$, then $\mathbf{x_B} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix}$, and $\mathbf{B}^{-1}\mathbf{N} = \begin{bmatrix} 1 & 1 & 1 \\ 2 & -1 & 1 \\ -5 & -2 & 4 \end{bmatrix}$.

Then, $\mathbf{c_B^T B^{-1} N} - \mathbf{c_N^T} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \mathbf{B^{-1} N} - \begin{bmatrix} 1 & 5 & -7 \end{bmatrix} = \begin{bmatrix} -1 & -5 & 7 \end{bmatrix}$

$$z_0 = \mathbf{c_B^T B^{-1} b} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 10 \end{bmatrix} = 0.$$

Then, the initial tableau can be written as follows:

		x_1	x_2	x_3	x_4	x_5	x_6	RHS
	1	-1	-5	7	0	0	0	0
x_4	0	1	1	1	1	0	0	4
x_5	0	2	-1	0	0	1	0	5
x_6	0	-5	2	0	0	0	1	10

By Dantzig pivot rule, the entering variable is x_2 and the leaving variable is x_4. After the tableau is updated, we get

		x_1	x_2	x_3	x_4	x_5	x_6	RHS
	1	4	0	12	5	0	0	20
x_2	0	1	1	1	1	0	0	4
x_5	0	3	0	2	1	1	0	9
x_6	0	-3	0	6	2	0	1	18

Since the $z_j - c_j \geq 0$ for all $j \in \mathbf{I_N}$, the optimal solution is found at $(x_1^*, x_2^*, x_3^*) = (0,4,0)$ with $z^* = 20$.

■

2.2.4 The initial basic feasible solution

The simplex method starts with a basic feasible solution and moves to an improved basic feasible solution until the optimal solution is reached, or unboundedness of the objective function is verified. However, in order to initialize the simplex medthod, a basis $\mathbf{B}$ with $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$ must be available. We will show that the simplex method can always be initiated with a very simple basis, namely, the identity.

Consider the following constraints:

$$\mathbf{Ax} \leq \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0},$$

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

By adding slack vector $\mathbf{s}$, the constraints can be put in the following standard form:

$$\mathbf{Ax} + \mathbf{Is} = \mathbf{b},$$
$$\mathbf{x}, \mathbf{s} \geq \mathbf{0}.$$

The new constraint matrix is $[\mathbf{A}, \mathbf{I}]$. If we let $\mathbf{B} = \mathbf{I}$ and $\mathbf{N} = \mathbf{A}$, then $\mathbf{s} = \mathbf{Ib} = \mathbf{b} \geq 0, \mathbf{x} = \mathbf{0}$.

Then, the initial basic feasible solution is $(\mathbf{x}, \mathbf{s})^{\mathbf{T}} = (\mathbf{0}, \mathbf{b})^{\mathbf{T}} \geq \mathbf{0}$, that is $\mathbf{x} = \mathbf{0}$ is a feasible point, and the simplex method can start.

For some cases, the initial basis can not be found easily such as the problem has the following constraints.

$$\mathbf{Ax} \geq \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0},$$

where $\mathbf{b} \nleq \mathbf{0}$. The standard form can be written by subtracting the surplus vector $\mathbf{s}$ as follows:

$$\mathbf{Ax} - \mathbf{Is} = \mathbf{b},$$
$$\mathbf{x}, \mathbf{s} \geq \mathbf{0}.$$

The new constraint matrix is $[\mathbf{A,\text{-}I}]$ that is difficult to pick a basis $\mathbf{B}$ with $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$. If we let $\mathbf{B} = \mathbf{\text{-}I}$ and $\mathbf{N} = \mathbf{A}$, then $\mathbf{s} = \mathbf{\text{-}Ib} = \mathbf{\text{-}b} \ngeq \mathbf{0}$.

Then, the $(\mathbf{x},\mathbf{s})^{\mathbf{T}} = (\mathbf{0},\mathbf{\text{-}b})^{\mathbf{T}}$ is not a basic feasible solution , that is $\mathbf{x} = \mathbf{0}$ is not a feasible point, and the simplex method can not start.

For this case, we cannot pick a basis $\mathbf{B}$ from the standard form. Therefore, we will introduce **artificial variables** to the problem to get a starting basic feasible solution as follows:

$$\mathbf{Ax} + \mathbf{Ix}_a = \mathbf{b},$$

$$\mathbf{x},\mathbf{x}_a \geq \mathbf{0},$$

where $\mathbf{x}_a$ is a vector of artificial variables. The new constraint matrix is $[\mathbf{A},\mathbf{I}]$. If we let $\mathbf{B} = \mathbf{I}$ and $\mathbf{N} = \mathbf{A}$, then $\mathbf{x}_a = \mathbf{Ib} = \mathbf{b} \geq \mathbf{0}, \mathbf{x} = 0$.

Then, the initial basic feasible solution is $(\mathbf{x},\mathbf{x}_a)^{\mathbf{T}} = (\mathbf{0},\mathbf{b})^{\mathbf{T}} \geq \mathbf{0}$, that is $\mathbf{x} = \mathbf{0}$ is a feasible point, and the simplex method can be performed.

The artificial variables are only a tool for getting the simplex method started. However, we must guarantee that these variables will eventually drop to zero. The two well-known techniques for eliminating artificial variables are **the two-phase method** and **the Big-M method**.

2.2.5 Two-phase method

The two-phase method is a method to find an initial basic feasible solution of the linear programming problem. The algorithm is separated into two phases.

Phase I: Solve the following a linear programming problem with the starting basic feasible solution $\mathbf{x} = \mathbf{0}$ and $\mathbf{x}_a = \mathbf{b}$:

$$\text{minimize} \quad x_0 = \mathbf{1}^{\mathbf{T}}\mathbf{x}_a$$

$$\text{subject to} \quad \mathbf{A}\mathbf{x} + \mathbf{x}_a = \mathbf{b}, \tag{2.3}$$

$$\mathbf{x}, \mathbf{x}_a \geq \mathbf{0}.$$

At optimality,

- If $\mathbf{x}_a^* \neq \mathbf{0}$, then the problem has no solution; the orginal problem is infeasible.

-If $\mathbf{x}_a^* = \mathbf{0}$, there are two occurrences: $\mathbf{x}_a^* = \mathbf{0}$ is out of the basis and $\mathbf{x}_a^* = \mathbf{0}$ is not out of the basis.

Case 1: $\mathbf{x}_a^* = \mathbf{0}$ is out of the basis.

We have identified a basic solution $\mathbf{x} = [\mathbf{x_B}, \mathbf{x_N}]^{\mathbf{T}}$ where $\mathbf{x_B}$ is the non-zero basic elements (in $\mathbf{x}$) and $\mathbf{x_N}$ is the remainder of the elements (not in $\mathbf{x}_a^* = \mathbf{0}$). We can then begin Phase II using this basic feasible solution.

Case 2: $\mathbf{x}_a^* = \mathbf{0}$ is not out of the basis.

We have identified a degenerate solution to the phase I problem. Therefore, this basis move to phase II and assign the cofficients of the artificial variables to zero.

Phase II: Solve the following linear programming problem by the simplex method at the starting basic feasible solution $\mathbf{x_B} = \mathbf{B}^{-1}\mathbf{b}$ and $\mathbf{x_N} = 0$ from Phase I.

$$\text{maximize} \quad z = \mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{x_B} + \mathbf{c}_{\mathbf{N}}^{\mathbf{T}}\mathbf{x_N}$$

$$\text{subject to } \mathbf{x_B} + \mathbf{B}^{-1}\mathbf{N}\mathbf{x_N} = \mathbf{b},$$

$$\mathbf{x_B}, \mathbf{x_N} \geq 0.$$

Example 2.2.4. Consider the following linear programming problem:

$$\begin{array}{llrcrcrcrcrcrcr}
\text{maximize} & z = & x_1 & + & 2x_2 & & & & & & & & & & \\
\text{subject to} & & x_1 & - & 2x_2 & + & s_1 & & & & & & & = & 4 \\
 & & x_1 & + & 2x_2 & & & + & s_2 & & & & & = & 5 \\
 & & -4x_1 & + & 3x_2 & & & & & + & s_3 & & & = & 6 \\
 & & x_1 & + & x_2 & & & & & & & - & s_4 & = & 1 \\
 & & x_1, & & x_2, & & s_1, & & s_2, & & s_3, & & s_4 & \geq & 0.
\end{array}$$

From this problem, we get $\mathbf{A} = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 \\ -4 & 3 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} 4 \\ 5 \\ 6 \\ 1 \end{bmatrix}$, $\mathbf{c}= \begin{bmatrix} 1 & 2 \end{bmatrix}$.

We see that the initial basis is difficult to choose. So, we will add the artificial variable to the constraint 4, we get

$$\begin{array}{llrcrcrcrcrcrcrcr}
\text{maximize} & z = & x_1 & + & 2x_2 & & & & & & & & & & & & \\
\text{subject to} & & x_1 & - & 2x_2 & + & s_1 & & & & & & & & & = & 4 \\
 & & x_1 & + & 2x_2 & & & + & s_2 & & & & & & & = & 5 \\
 & & -4x_1 & + & 3x_2 & & & & & + & s_3 & & & & & = & 6 \\
 & & x_1 & + & x_2 & & & & & & & - & s_4 & + & x_{a_1} & = & 1 \\
 & & x_1, & & x_2, & & s_1, & & s_2, & & s_3, & & s_4, & & x_{a_1} & \geq & 0.
\end{array}$$

Phase I: we will solve the following linear programming problem:

$$\begin{array}{llrcrcrcrcrcrcrcr}
\text{minimize} & z_1 = & x_{a_1} & & & & & & & & & & & & & & \\
\text{subject to} & & x_1 & - & 2x_2 & + & s_1 & & & & & & & & & = & 4 \\
 & & x_1 & + & 2x_2 & & & + & s_2 & & & & & & & = & 5 \\
 & & -4x_1 & + & 3x_2 & & & & & + & s_3 & & & & & = & 6 \\
 & & x_1 & + & x_2 & & & & & & & - & s_4 & + & x_{a_1} & = & 1 \\
 & & x_1, & & x_2, & & s_1, & & s_2, & & s_3, & & s_4, & & x_{a_1} & \geq & 0.
\end{array}$$

We choose $\mathbf{x_B^T} = \begin{bmatrix} s_1 & s_2 & s_3 & x_{a_1} \end{bmatrix}$ to be an initial basic feasible solution. Then, the initial tableau can be written as below:

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	1	2	0	0	0	−1	0	1
s_1	0	1	−2	1	0	0	0	0	4
s_2	0	1	2	0	1	0	0	0	5
s_3	0	−4	3	0	0	1	0	0	6
x_{a_1}	0	1	1	0	0	0	−1	1	1

For the minimization problem, x_2 will be chosen to be the entering variable, and x_{a_1} leaves the basis. After the tableau is updated, we get the following tableau.

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	1	0	0	0	0	−2	0	2
s_1	0	3	0	1	0	0	−2	0	6
s_2	0	−1	0	0	1	0	2	0	3
s_3	0	−7	0	0	0	1	3	0	3
x_2	0	1	1	0	0	0	−1	1	1

After steps of the simplex algorithm are repeated, we get

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	1.33	0	0	0	−0.33	0	−1	−2
s_1	0	−1.67	0	1	0	0.67	0	0	8
s_2	0	3.67	0	0	1	−0.67	0	0	1
s_4	0	−2.33	0	0	0	0.33	1	−1	1
x_2	0	−1.33	1	0	0	0.33	0	0	2

Phase I ends. Since $x^*_{a_1} = 0$, the basic feasible solution is found. Then, we can go to Phase II. After we compute the tableau for this basic feasible solution, we get the following tableau.

		x_1	x_2	s_1	s_2	s_3	s_4	RHS
	1	-3.67	0	0	0	0.67	0	2
s_1	0	-1.67	0	1	0	0.67	0	8
s_2	0	3.67	0	0	1	-0.67	0	1
s_4	0	-2.33	0	0	0	0.33	1	1
x_2	0	-1.33	1	0	0	0.33	0	2

Next, the entering variable is x_1 and the leaving variable is s_2, and the updated tableau is as follows:

		x_1	x_2	s_1	s_2	s_3	s_4	RHS
	1	0	0	0	1	0	0	5
s_1	0	0	0	1	0.5	0.33	0	8.5
x_1	0	1	0	0	0.25	-0.17	0	0.25
s_4	0	0	0	0	0.67	0	1	1.63
x_2	0	0	1	0	0.33	0	0	2.38

Then, we found that the optimal solution is $(x_1^*, x_2^*) = (0.25, 2.38)$ with $z^* = 5$.

■

2.2.6 The Big-M method

The Big-M method is a technique for dealing with artificial variables by assigning a very large coefficient for these artificial variables in the original objective fuction.

Consider the following linear programming problem:

$$\text{P: maximize} \quad z = \mathbf{c}^{\mathbf{T}}\mathbf{x}$$

$$\text{subject to } \mathbf{A}\mathbf{x} = \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0}.$$

If no convenient basis is known, we can introduce the artificial vector $\mathbf{x}_a$, that leads to the following system:

$$\mathbf{A}\mathbf{x} + \mathbf{x}_a = \mathbf{b},$$
$$\mathbf{x}, \mathbf{x}_a \geq \mathbf{0}.$$

The starting basic feasible solution is given by $\mathbf{x}_a = \mathbf{b}$. In order to the undesirability of a nonzero artificial vector, the objective function is modified such that a large penalty is assigned for such solution. More specifically, consider the folllowing problem:

$$\text{P(Big-M): maximize} \quad z_{Big-M} = \mathbf{c}^{\mathbf{T}}\mathbf{x} - M\mathbf{1}^{\mathbf{T}}\mathbf{x}_a$$

$$\text{subject to} \qquad \mathbf{A}\mathbf{x} + \mathbf{x}_a = \mathbf{b},$$
$$\mathbf{x}, \mathbf{x}_a \geq \mathbf{0},$$

where M is a very large positive number.

After solving it by the simplex method, one of the following two cases may occur:

(i) We found the optimal solution of P(Big-M).

- The artificial variables are all equal to zeroes. In this case, the original problem is feasible and the optimal solution is found.

- Some artificial variables are positive. In this case, the original problem is infeasible.

(ii) We found that the problem P(Big-M) has an unbounded solution. If all artificial variables are zero, then the original problem is unbounded. Otherwise, the original problem is infeasible.

Example 2.2.5. Consider the linear programming problem in Example 2.2.4. We will solve the following problem.

$$\begin{array}{llrcrccccrl}
\text{P(Big-M):} & \text{maximize } z_1 = & x_1 & + & 2x_2 & & & & & -Mx_{a_1} & \\
& \text{subject to} & x_1 & - & 2x_2 & +s_1 & & & & & = 4 \\
& & x_1 & + & 2x_2 & & +s_2 & & & & = 5 \\
& & -4x_1 & + & 3x_2 & & & +s_3 & & & = 6 \\
& & x_1 & + & x_2 & & & & -s_4 & +x_{a_1} & = 1 \\
& & x_1, & & x_2, & s_1, & s_2, & s_3, & s_4, & x_{a_1} & \geq 0.
\end{array}$$

When we choose $\mathbf{x}_\mathbf{B}^\mathbf{T} = \begin{bmatrix} s_1 & s_2 & s_3 & x_{a_1} \end{bmatrix}$ as the basic feasible solution, the initial simplex tableau for Big-M method can be written as follows:

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	$-1-M$	$-2-M$	0	0	0	M	0	$-M$
s_1	0	1	-2	1	0	0	0	0	4
s_2	0	1	2	0	1	0	0	0	5
s_3	0	-4	3	0	0	1	0	0	6
x_{a_1}	0	1	1	0	0	0	-1	1	1

From the initial tableau, the entering variable is x_2 and the leaving variable is x_{a_1}. After the tableau is updated, we get the following tableau.

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	1	0	0	0	0	-2	$2+M$	2
s_1	0	3	0	1	0	0	-2	2	6
s_2	0	-1	0	0	1	0	2	-2	3
s_3	0	-7	0	0	0	1	3	-3	3
x_2	0	1	1	0	0	0	-1	1	1

After the steps of the simplex algorithm are repeated, we get the following two tableaux.

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	-3.67	0	0	0	0.67	0	M	4
s_1	0	-1.67	0	1	0	0.67	0	0	8
s_2	0	3.67	0	0	1	-0.67	0	0	1
s_4	0	-2.33	0	0	0	0.33	1	-1	1
x_2	0	-1.33	1	0	0	0.33	0	0	2

		x_1	x_2	s_1	s_2	s_3	s_4	x_{a_1}	RHS
	1	0	0	0	1	0	0	M	5
s_1	0	0	0	1	0.5	0.33	0	0	8.5
x_1	0	1	0	0	0.25	-0.17	0	0	0.25
s_4	0	0	0	0	0.67	0	1	-1	1.63
x_2	0	0	1	0	0.33	0	0	0	2.38

From the last tableau, we found that the optimal solution is $(x_1^*, x_2^*) = (0.25, 2.38)$ with $z^* = 5$ since $x_a^* = 0$.

■

From Example 2.2.4 and Example 2.2.5, the number of iterations of both the two-phase method and the Big-M method are three iterations. However, the large number M is computed in each iteration for the Big-M method while the two-phase is not.

2.2.7 Duality

For each linear programming problem, there is another associated linear programming problem called the dual, and the original linear programming problem is called primal. The dual linear programming problem possesses many important properties relative to the original primal linear programming problem. The variables in the primal problem are equivalent to the constraints in the dual problem. Before we will describe about the dual problem, we will introduce two important representations of linear programming problems that are the canonical form and the standard form which are defined as the following table.

TABLE 2.1: Standard form and canonical form

	Minimization problem	Maximization problem
Standard form	minimize $\mathbf{c}^\mathbf{T}\mathbf{x}$ subject to $\mathbf{Ax} = \mathbf{b}$ $\mathbf{x} \geq \mathbf{0}$	maximize $\mathbf{c}^\mathbf{T}\mathbf{x}$ subject to $\mathbf{Ax} = \mathbf{b}$ $\mathbf{x} \geq \mathbf{0}$
Canornical form	minimize $\mathbf{c}^\mathbf{T}\mathbf{x}$ subject to $\mathbf{Ax} \geq \mathbf{b}$ $\mathbf{x} \geq \mathbf{0}$	maximize $\mathbf{c}^\mathbf{T}\mathbf{x}$ subject to $\mathbf{Ax} \leq \mathbf{b}$ $\mathbf{x} \geq \mathbf{0}$

Since many linear programming problems in term of maximization or minimization and variables may be nonnegative, unrestricted in sign or bounded which may not match the standard or the canonical form and some algorithms deal with a specific form, of linear programming problem, a problem must be manipulated to fit the required form.

For maximization and minimization problems, we can convert a maximization problem to a minimization problem and conversely as follows:

$$\textbf{Max} \quad \mathbf{c^T x} \equiv -[\textbf{Min} \quad -\mathbf{c^T x}],$$
$$\textbf{Min} \quad \mathbf{c^T x} \equiv -[\textbf{Max} \quad -\mathbf{c^T x}].$$

For inequality constraints, consider a constraint given by $\mathbf{Ax} \leq \mathbf{b}$. This constraint can be transformed to an equality constraint as follows:

$$\mathbf{Ax} + \mathbf{IS} = \mathbf{b},$$
$$\mathbf{S} \geq \mathbf{0},$$

where **S** is called a slack variable vector.

While a constraint given by $\mathbf{Ax} \geq \mathbf{b}$ can be converted to an equality constraint as follows:

$$\mathbf{Ax} - \mathbf{IS} = \mathbf{b},$$
$$\mathbf{S} \geq \mathbf{0},$$

where **S** is called a surplus variable vector.

For equality constraints, if a constraint has an equation form, i.e., $\mathbf{Ax} = \mathbf{b}$, it can be transformed into inequality constraints as follows:

$$\mathbf{Ax} \geq \mathbf{b} \quad \text{and} \quad \mathbf{Ax} \leq \mathbf{b}.$$

For nonnegativity of the variables, if the variables x_j can be positive, zero or negative, called **unrestricted** in sign, then it can be converted to two new nonnegative variables as follows:

$$x_j = x_j^+ - x_j^-,$$
$$x_j^+, x_j^- \geq 0.$$

For variables bounds, if $x_j \geq l_j$ or $x_j \leq u_j$, then it can be converted to the new nonnegative variable as follows:

$$x'_j = x_j - l_j \geq 0 \text{ for } x_j \geq l_j \text{ and}$$

$$x'_j = u_j - x_j \geq 0 \text{ for } x_j \leq u_j.$$

Next, we are back to the dual problem. The definition of the dual problem is stated as the following definition.

Definition 2.2.2. Suppose that the primal problem is given in the (canonical) form:

$$\begin{aligned} \text{P: maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

Then, the dual linear programming problem is defined by:

$$\begin{aligned} \text{D: minimize} \quad & \mathbf{b}^{\mathbf{T}}\mathbf{w} \\ \text{subject to} \quad & \mathbf{A}^{\mathbf{T}}\mathbf{w} \geq \mathbf{c}, \\ & \mathbf{w} \geq \mathbf{0}. \end{aligned}$$

From the manipulation, we can write the dual problem for a standard form as below.

Suppose that the primal linear programming problem is given in the (standard) form:

$$\begin{aligned} \text{P: maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

Then, this problem can be transformed to the canonical form as follows:

$$\begin{aligned} \text{P: maximize} \quad & \mathbf{c^T x} \\ \text{subject to} \quad & \mathbf{Ax} \leq \mathbf{b}, \\ & -\mathbf{Ax} \leq -\mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

Therefore, the dual linear programming problem is written by:

$$\begin{aligned} \text{D: minimize} \quad & \mathbf{b^T w_1} - \mathbf{b^T w_2} \\ \text{subject to} \quad & \mathbf{A^T w_1} - \mathbf{A^T w_2} \geq \mathbf{c}. \\ & \mathbf{w_1}, \mathbf{w_2} \geq \mathbf{0}. \end{aligned}$$

If we let $\mathbf{w} = \mathbf{w_1} - \mathbf{w_2}$ where $\mathbf{w_1}, \mathbf{w_2} \geq 0$, then we get the dual problem of the standard form as follows:

$$\begin{aligned} \text{D: minimize} \quad & \mathbf{b^T w} \\ \text{subject to} \quad & \mathbf{A^T w} \geq \mathbf{c}. \end{aligned}$$

Example 2.2.6. Consider the following linear programming problem:

$$\begin{array}{lccccccc} \text{maximize} & x_1 & + & 5x_2 & - & 7x_3 & & \\ \text{subject to} & x_1 & + & x_2 & + & x_3 & \leq & 4 \\ & 2x_1 & - & x_2 & + & x_3 & \leq & 5 \\ & -5x_1 & - & 2x_2 & + & 4x_3 & \leq & 10 \\ & x_1, & & x_2, & & x_3 & \geq & 0. \end{array}$$

We can write the dual problem for this problem as follows:

$$\begin{array}{lccccccc} \text{minimize} & 4w_1 & + & 5w_2 & + & 10w_3 & & \\ \text{subject to} & w_1 & + & 2w_2 & - & 5w_3 & \geq & 1 \\ & w_1 & - & w_2 & - & 2w_3 & \geq & 5 \\ & w_1 & + & w_2 & + & 4w_3 & \geq & -7 \\ & w_1, & & w_2, & & w_3 & \geq & 0. \end{array}$$

Example 2.2.7. Consider the following linear programming problem:

$$
\begin{array}{lrcrcrcl}
\text{maximize} & x_1 & + & 5x_2 & - & 7x_3 & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & = & 4 \\
 & 2x_1 & - & x_2 & + & x_3 & = & 5 \\
 & -5x_1 & - & 2x_2 & + & 4x_3 & = & 10 \\
 & x_1, & & x_2, & & x_3 & \geq & 0.
\end{array}
$$

The dual problem for this problem can be written as follows:

$$
\begin{array}{lrcrcrcl}
\text{minimize} & 4w_1 & + & 5w_2 & + & 10w_3 & & \\
\text{subject to} & w_1 & + & 2w_2 & - & 5w_3 & \geq & 1 \\
 & w_1 & - & w_2 & - & 2w_3 & \geq & 5 \\
 & w_1 & + & w_2 & + & 4w_3 & \geq & -7.
\end{array}
$$

■

Consider the primal problem in the canonical form:

$$
\begin{array}{ll}
\text{P: maximize} & \mathbf{c^T x} \\
\text{subject to} & \mathbf{Ax} \leq \mathbf{b}, \\
 & \mathbf{x} \geq \mathbf{0}.
\end{array}
$$

Then, the dual linear programming problem is defined by:

$$
\begin{array}{ll}
\text{D: minimize} & \mathbf{b^T w} \\
\text{subject to} & \mathbf{A^T w} \geq \mathbf{c}, \\
 & \mathbf{w} \geq \mathbf{0}.
\end{array}
$$

Lemma 2.2.8 (Weak Duality Property)**.** If $\mathbf{x}_0$ and $\mathbf{w}_0$ be feasible solutions to problem P and problem D respectively, then $\mathbf{c^T x}_0 \leq \mathbf{b^T w}_0$.

Corollary 2.2.9. If $\mathbf{x}_0$ and $\mathbf{w}_0$ are feasible solutions to the primal and dual problems, respectively, such that $\mathbf{c^T x}_0 = \mathbf{b^T w}_0$, then $\mathbf{x}_0$ and $\mathbf{w}_0$ are optimal solutions to their respective problem.

Corollary 2.2.10. If the problem P is unbounded, then problem D is infeasible. Likewise, If problem D is unbounded, then problem P is infeasible.

Corollary 2.2.11. If the problem P is infeasible, then problem D is either unbounded or infeasible. If problem D is infeasible, then problem P is unbounded or infeasible.

Lemma 2.2.12 (Strong Duality Property)**.** If one problem possesses an optimal solution, then both problems possess optimal solutions and two optimal objective values are equal.

Theorem 2.2.13 (Fundamental Theorem of Duality)**.** Consider problem P and problem D. Then, exactly one of the following statements is true:

1. Both problem P and problem D possess optimal solutions $\mathbf{x}^*$ and $\mathbf{w}^*$ respectively and $\mathbf{c^T x}^* = \mathbf{w}^{*\mathbf{T}}\mathbf{b}$.
2. Problem P is unbounded and problem D is infeasible.
3. Problem D is unbounded and problem P is infeasible.
4. Both problems are infeasible.

2.2.8 The Karush-Kuhn-Tucker (KKT) optimality conditions

Consider a primal linear programming problem:

$$\begin{aligned} \text{P: maximize} \quad & z = \mathbf{c^T x} \\ \text{subject to } & \mathbf{Ax} \leq \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

Then, the dual linear programming problem is defined by:

$$\text{D: minimize} \quad z = \mathbf{b^T w}$$

$$\text{subject to } \mathbf{A^T w} \geq \mathbf{c},$$
$$\mathbf{w} \geq \mathbf{0}.$$

Theorem 2.2.14 (The Karush-Kuhn-Tucker (KKT) Optimality Conditions)**.** The optimal conditions for a linear programming problem state that a necessary and sufficient condition for $\mathbf{x}^*$ to be an optimal point to problem (P) is that there exists a vector $\mathbf{w}^*$ such that

1. $\mathbf{Ax}^* \leq \mathbf{b}, \mathbf{x}^* \geq 0$ (Primal Feasibility),
2. $\mathbf{A^T w}^* \geq \mathbf{c}, \mathbf{w}^* \geq 0$ (Dual Feasibility),
3. $\mathbf{w}^{*\mathbf{T}}(\mathbf{Ax}^* - \mathbf{b}) = \mathbf{0}$ and $(\mathbf{c^T} - \mathbf{A^T w}^*)\mathbf{x}^* = \mathbf{0}$ (Complementary Slackness).

From KKT conditions, condition 1 indicates that $\mathbf{x}^*$ must be a feasible point for the primal problem while condition 2 indicates that $\mathbf{w}^*$ must be a feasible point for the dual problem.

Condition 3 is checking $\mathbf{c^T x}^* = \mathbf{w}^{*\mathbf{T}}\mathbf{b}$, that is the strong duality property is verified.

2.3 The dual simplex method

Consider the following linear programming problem:

$$\text{maximize} \quad \mathbf{c^T x}$$
$$\text{subject to } \mathbf{Ax} = \mathbf{b},$$
$$\mathbf{x} \geq \mathbf{0}.$$

In a certain instance, it is difficult to find a basic solution that is feasible (that is $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$) to a linear programming problem without adding artificial variables. In this same instance, it might be possible to find a starting basis which is not neccessary feasible, but its dual is feasible (that is, all $z_j - c_j \geq 0$ for a maximization problem).

Consider $\mathbf{c_B^T B^{-1} N} - \mathbf{c_N^T}$, that is $\mathbf{w^T A}_{:j} - c_j$ for all $j = 1, 2, ..., n$ where $\mathbf{w^T} = \mathbf{c_B^T B^{-1}}$. If $\mathbf{c_B^T B^{-1} N} - \mathbf{c_N^T} \geq \mathbf{0}$, then $\mathbf{w^T A}_{:j} - c_j \geq 0$ such that $\mathbf{w^T A}_{:j} \geq c_j$ for all $j = 1, 2, ..., n$. Therefore, $\mathbf{w^T}$ is the feasible solution for the dual problem.

In such case, it is useful to develop a variant of the simplex method that would produce the series of the simplex tableau that maintain dual feasibility and complementary slackness and strive toward primal feasibility.

The dual simplex method (maximization problem)

Initialization Step:

Find a basis $\mathbf{B}$ of the primal problem such that $z_j - c_j \geq 0$ for all $j = 1, 2, ..., n$.

Let $\mathbf{y}_j = \mathbf{B}^{-1}\mathbf{A}_{:j}$ where $j = 1, ..., n$.

Main Step:

1. If $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b} \geq 0$, then stop; the current solution is optimal. Otherwise, select a pivot row r with $\overline{b}_r < 0$; say

$$\overline{b}_r = \underset{1 \leq i \leq m}{\text{minimum}} \{\overline{b}_i\}.$$

2. If $y_{ij} \geq 0$ for $j = 1, 2, ..., n$, then stop; the dual is unbounded and the primal is infeasible.

Otherwise, select the pivot column k by the following minimum ratio test:

$$\frac{z_k - c_k}{|y_{rk}|} = \underset{1 \leq j \leq n}{\text{minimum}} \{\frac{z_j - c_j}{|y_{rj}|} : y_{rj} < 0\}$$

3. Pivot at y_{rk} and return to Step 1.

Example 2.3.1. Consider the following linear programming problem:

$$\begin{array}{rrcrcrcr}
\text{maximize} & -x_1 & - & 3x_2 & - & 2x_3 & & \\
\text{subject to} & x_1 & - & 2x_2 & + & x_3 & \leq & -2 \\
& -3x_1 & + & 3x_2 & - & 2x_3 & \leq & -3 \\
& x_1, & & x_2, & & x_3 & \geq & 0.
\end{array}$$

First, the problem must be converted to the standard form as follows:

$$
\begin{array}{lrcrcrcrcrcr}
\text{maximize} & -x_1 & - & 3x_2 & - & 2x_3 & & & & & & \\
\text{subject to} & x_1 & - & 2x_2 & + & x_3 & + & x_4 & & & = & -2 \\
 & -3x_1 & + & 3x_2 & - & 2x_3 & & & + & x_5 & = & -3 \\
 & x_1, & & x_2, & & x_3, & & x_4, & & x_5 & \geq & 0.
\end{array}
$$

When we choose $\mathbf{I_B} = \{4, 5\}$, the initial tableau can be written as below.

	z	x_1	x_2	x_3	x_4	x_5	RHS
	1	0	4	1.33	0	0.33	-1
x_4	0	1	-2	1	1	0	-2
x_5	0	-3	3	-2	0	1	-3

We can see that the dual is feasible while the primal is not. So, we can use the dual simplex method by choosing x_5 as the leaving variable and x_1 as the entering variable. Then, the tableau is updated as follows:

Iteration 1:

	z	x_1	x_2	x_3	x_4	x_5	RHS
	1	0	4	1.33	0	0.33	-1
x_4	0	0	-1	0.33	1	0.33	-3
x_1	0	1	-1	0.67	0	-0.33	1

Next, x_4 is the leaving variable and x_2 is the entering variable. After updating the tableau, we get the following tableau.

Iteration 2:

	z	x_1	x_2	x_3	x_4	x_5	RHS
	1	0	0	2.67	4	1.67	-13
x_2	0	0	1	-0.33	-1	-0.33	3
x_1	0	1	0	0.33	-1	-0.67	4

From the above tableau, we found that the dual and primal problem are feasible. Therefore, the optimal solution is found which $(x_1^*, x_2^*, x_3^*) = (4, 3, 0)$.

$\blacksquare$

In conclusion, the simplex method starts when the primal problem has a basic feasible solution while the dual simplex method starts when the dual problem has a basic feasible solution. However, if both solutions of the primal problem and the dual problem are infeasible, then we can not start the simplex method and the dual simplex method. So in this thesis, we would like to construct the algorithm for solving the linear programming problem when this case occurs.

CHAPTER 3

THE PROPOSED METHOD

In this chapter, we will present the main idea of our method. Then, the algorithm for solving linear programming problems is shown. In addition, some examples that show the efficiency of our algorithm are proposed.

3.1 Negative relaxation of dual problem

We would like to construct the relaxed problem without using artificial variables which are introduced when the primal solution can not be obtained easily. On the other hand, if the dual problem is feasible while the primal is not, then we can solve the problem by the dual simplex without using artificial variables. Therefore, if we have the dual feasible point, then the simplex method can start without using artificial variables. Consider Figure 3.1 which presents the feasible region of the dual problem:

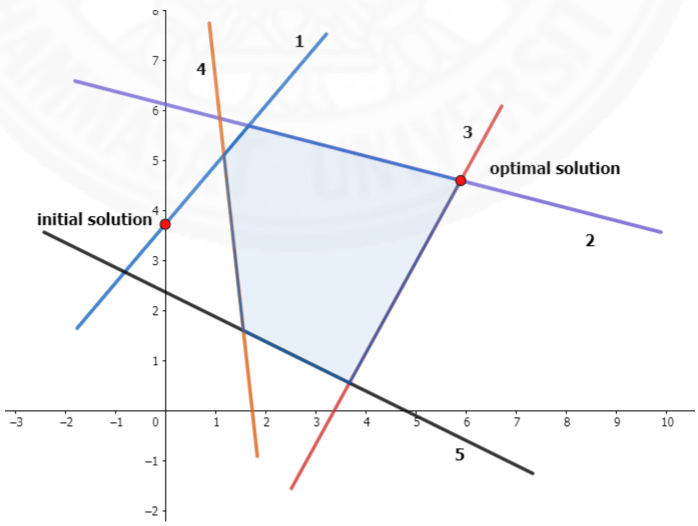


Figure 3.1: Example of feasible region of the dual problem

From Figure 3.1, if we choose the initial solution as the marked point, we can see that the constraint 4 is not satisfied, then the simplex method can not start. If we would like to start from this point, then an artificial variable will be added to constraint 4. Therefore, if we wolud like to obviate the use of an artificial variable, the constraint 4 would be relaxed as Figure 3.2. So, this solution is a feasible point of the dual problem.

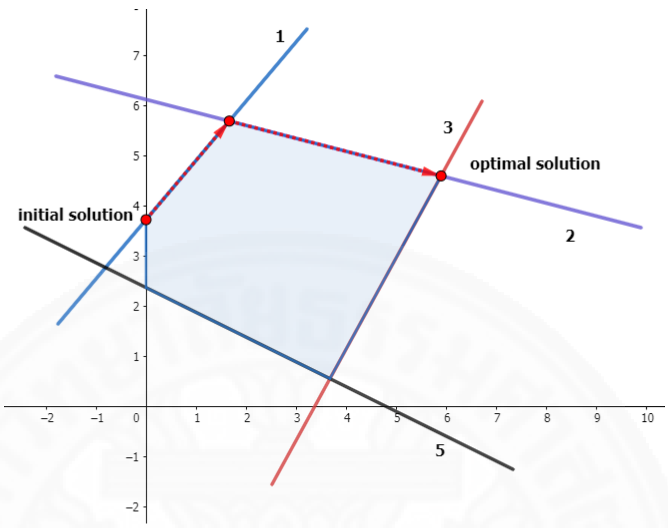


Figure 3.2: The relaxed problem

Since the constraint 4 is associated with the variable in primal problem, if we will relax this variable in primal then the dual simplex method can start without using artificial variables. In general, the question arises that how can we identify the unsatisfied constraints in the dual problem?

Consider a linear programming problem in the standard form:

$$\text{maximize} \quad z = \mathbf{c}^{\mathbf{T}}\mathbf{x} \tag{3.1}$$

$$\text{subject to } \mathbf{Ax} = \mathbf{b},$$

$$\mathbf{x} \geq \mathbf{0},$$

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

In this thesis, we will use the primal-dual relationships to construct the algorithm. From (3.1), we can write the dual problem as follows:

$$\text{maximize} \quad z = \mathbf{b}^{\mathbf{T}}\mathbf{w} \tag{3.2}$$

$$\text{subject to } \mathbf{A}^{\mathbf{T}}\mathbf{w} \geq \mathbf{c},$$

where $\mathbf{w} \in \mathbb{R}^m$.

Let $\mathbf{A} = [\mathbf{A}_{:1}, \mathbf{A}_{:2}, ..., \mathbf{A}_{:n}]$ where $\mathbf{A}_{:j}$ is the j^{th} column of matrix $\mathbf{A}$ and $\mathbf{A} = [\mathbf{B}, \mathbf{N}]$ where $\mathbf{B} \in \mathbb{R}^{m \times m}$, $\mathbf{N} \in \mathbb{R}^{m \times (n-m)}$ and $\mathbf{B}$ is a nonsingular matrix. Let $\mathbf{I_B}$ be an index set of the basic variables and $\mathbf{I_N}$ be an index set of the nonbasic variables.

For any basis $\mathbf{B}$, the problem (3.1) can be written as follows:

$$\text{maximize} \quad z + (\mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_{\mathbf{N}}^{\mathbf{T}})\mathbf{x_N} = \mathbf{c}_{\mathbf{b}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{b} \tag{3.3}$$

$$\text{subject to} \quad \mathbf{x_B} + \mathbf{B}^{-1}\mathbf{N}\mathbf{x_N} = \mathbf{B}^{-1}\mathbf{b},$$

$$\mathbf{x_B}, \mathbf{x_N} \geq 0.$$

Therefore, we can write the initial tableau as follows:

	z	$\mathbf{x_B}$	$\mathbf{x_N}$	RHS
	1	0	$\mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_{\mathbf{N}}^{\mathbf{T}}$	$z_0 = \mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{b}$
$\mathbf{x_B}$	0	$\mathbf{I}$	$\mathbf{B}^{-1}\mathbf{N}$	$\mathbf{B}^{-1}\mathbf{b}$

From the initial tableau, we can consider the solution of primal and dual problems by considering the value of $\mathbf{B}^{-1}\mathbf{b}$ for the primal problem, and $\mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_{\mathbf{N}}^{\mathbf{T}}$ for the dual problem .

- For the primal problem, if $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$, then $\mathbf{x} = \begin{bmatrix} \mathbf{B}^{-1}\mathbf{b} \\ \mathbf{0} \end{bmatrix}$ is the basic feasible solution.

- For the dual problem, if $\mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c}_{\mathbf{N}}^{\mathbf{T}} \geq \mathbf{0}$, then $\mathbf{w}^{\mathbf{T}}$ is the feasible solution for the dual problem where $\mathbf{w}^{\mathbf{T}} = \mathbf{c}_{\mathbf{B}}^{\mathbf{T}}\mathbf{B}^{-1}$.

Therefore, for any basis $\mathbf{B}$, there are four cases for finding the optimal solution as follows:

Case 1: If $\mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c_N^T} \geq \mathbf{0}$ and $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$, then the primal and dual solutions are feasible. Therefore, we get the optimal solution by KKT conditions.

Case 2: If $\mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c_N^T} \geq 0$ and $\mathbf{B}^{-1}\mathbf{b} \ngeq \mathbf{0}$. The dual solution is feasible while the primal solution is infeasible. Then, the dual simplex method can be performed to find the optimal solution.

Case 3: If $\mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c_N^T} \ngeq 0$ and $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$, then the dual solution is an infeasible solution while the primal solution is feasible. Therefore, the simplex method can be used to solve it.

Case 4: If $\mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c_N^T} \ngeq 0$ and $\mathbf{B}^{-1}\mathbf{b} \ngeq \mathbf{0}$, then both solutions of the primal and dual problems are infeasible. In this case, we can not start the simplex method and the dual simplex method.

For Case 4, the simplex method can start when artificial variables are added. So, the problem is bigger, and it may waste some computational time. Therefore, in this thesis, we propose the improvement of the simplex method without using artificial varibles by constructing the relaxed problem.

Form the problem (3.3), if $\mathbf{B}^{-1}\mathbf{b} \ngeq \mathbf{0}$, then the primal solution is not satisfied some constraints in the primal problem while if $\mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{N} - \mathbf{c_N^T} \ngeq 0$, then the dual solution is not satisfied the constraints in the dual problem. If we wolud like to start the dual simplex method, then we will relax the variables associated with the unsatisfied dual constraints.

Let $z_j - c_j = \mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{A}_{:j} - c_j$ for all $j \in \mathbf{I_N}$. For any basis $\mathbf{B}$, we can rewrite the initial tableau as follows:

	z	$\mathbf{x_B}$	$\mathbf{x_N}$	RHS
	1	0	$z_j - c_j = \mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{A}_{:j} - c_j, \forall j \in \mathbf{I_N}$	$z_0 = \mathbf{c_B^T}\mathbf{B}^{-1}\mathbf{b}$
$\mathbf{x_B}$	0	$\mathbf{I}$	$\mathbf{B}^{-1}\mathbf{N}$	$\mathbf{B}^{-1}\mathbf{b}$

Consider the initial tableau, if $z_j - c_j < 0$ for some $j \in \mathbf{I_N}$, then the dual solution is infeasible. To make the dual solution feasible, the variables which correspond to $z_j - c_j < 0,$ for all $j \in \mathrm{I_N}$ will be relaxed. So the dual simplex method can start. By relaxing the negative values which corespond to the dual problem, this relaxeation is called **Negative Relaxation of Dual Problem**.

After the relaxed problem is solved, the relaxed variables will be restored. According, we can conclude the steps of algorithm below.

3.2 Negative relaxation of dual problem algorithm

The steps of the algorithm can be summarized as follows:

Initial step:

Choose the initial basis $\mathbf{B}$ and compute the initial simplex tableau

(see section 2.3.4).

Let $G = \{j \in \mathbf{I_N} | z_j - c_j \geq 0\}$,

$L = \{j \in \mathbf{I_N} | z_j - c_j < 0\}$,

and $b_{min} = \underset{1 \leq i \leq m}{\text{minimum}}\{\overline{b_i}\}$ where $\overline{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b}$.

Step 1:

If $G \neq \varnothing$, then

if $L \neq \varnothing$, then

if $b_{min} \geq 0$, then the simplex method is performed.

else relax variables in L and perform

the dual simplex method.

if the optimal solution is found, then restore variables

in L and perform the simplex method, then **stop**.

else restore variables in L and go to **Step 2**.

else if $b_{min} \geq 0$, then the optimal solution is found and **stop**.

else the dual simplex method is performed, then **stop**.

else if $b_{min} \geq 0$, then the simplex method is performed and then **stop**.

else go to **Step 2**.

Step 2: The perturbation simplex method [6] is performed.

Next, the **Negative relaxation of dual problem** algorithm can be written the flowchart as Figure 3.3.

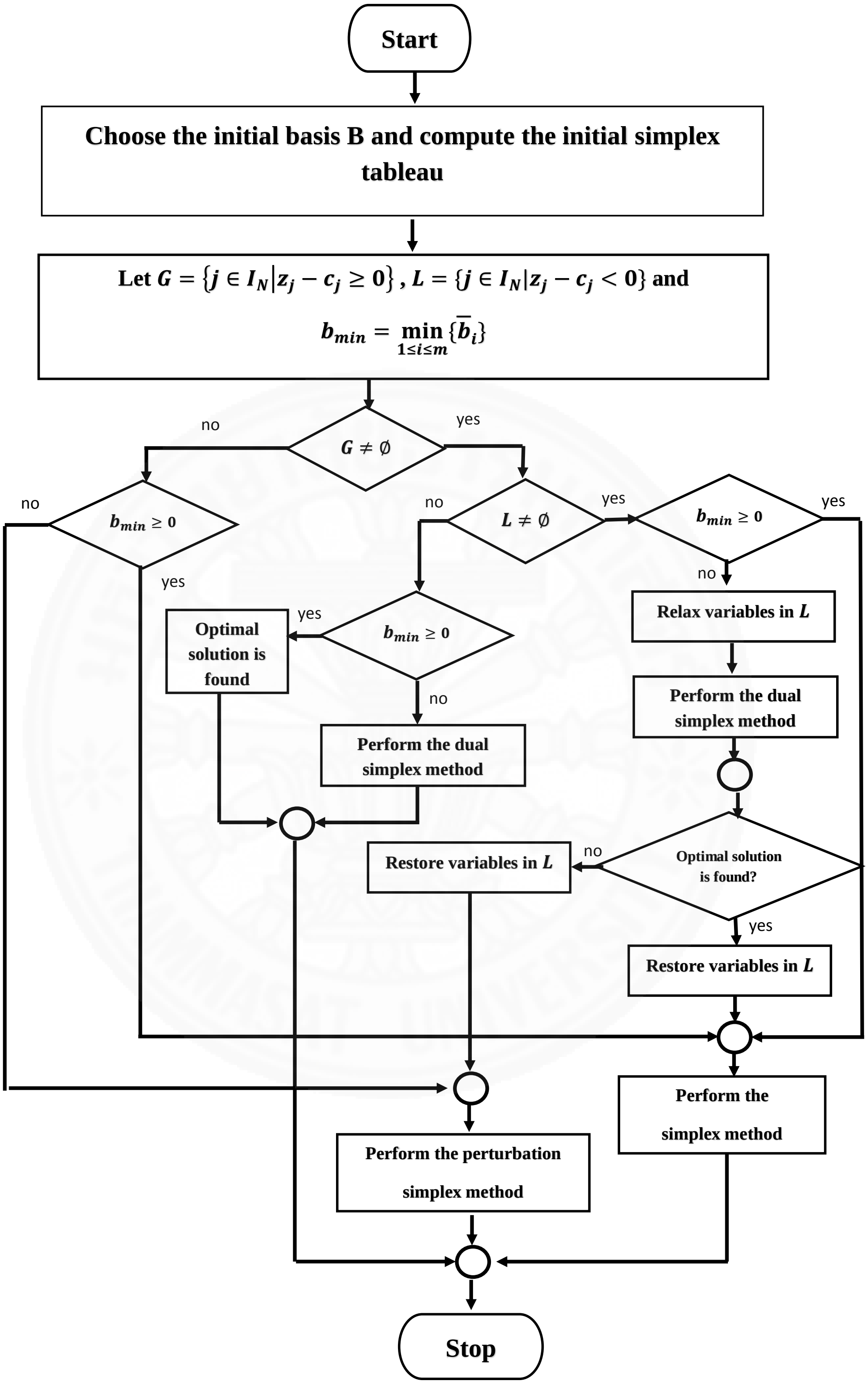


Figure 3.3: The flowchart of **Negative relaxation of dual problem** algorithm.

Example 3.2.1. Consider the following linear programming problem:

$$\begin{array}{llllll} \text{maximize} & z = 2x_1 & & +2x_3 & -5x_4 & \\ \text{subject to} & x_1 & +x_2 & +x_3 & & \leq 8 \\ & -2x_1 & +x_2 & -3x_3 & +5x_4 & \leq -5 \\ & -x_1 & +2x_2 & +x_3 & & \leq -6 \\ & 3x_1 & +x_2 & -2x_3 & +5x_4 & \leq -4 \\ & x_1, & x_2, & x_3, & x_4 & \geq 0. \end{array}$$

Before the algorithm starts, we transform this problem to the standard form as follows:

$$\begin{array}{llllllllll} \text{maximize} & z = 2x_1 & & +2x_3 & -5x_4 & & & & & \\ \text{subject to} & x_1 & +x_2 & +x_3 & & +x_5 & & & & = 8 \\ & -2x_1 & +x_2 & -3x_3 & +5x_4 & & +x_6 & & & = -5 \\ & -x_1 & +2x_2 & +x_3 & & & & +x_7 & & = -6 \\ & 3x_1 & +x_2 & -2x_3 & +5x_4 & & & & +x_8 & = -4 \\ & x_1, & x_2, & x_3, & x_4, & x_5, & x_6, & x_7, & x_8 & \geq 0. \end{array}$$

First, we will choose the initial basis **B** by using the cosine criterion. Then, the indices correspondings to variables are $\alpha_1 = 9.295$, $\alpha_2 = -4.913$, $\alpha_3 = 6.454$, $\alpha_4 = -6.363$, $\alpha_5 = 8$, $\alpha_6 = -5$, $\alpha_7 = -6$ and $\alpha_8 = -4$ respectively. So we choose the index of a basic feasible solution as $\mathbf{I_B} = \{1,5,3,8\}$. Then, the initial tableau can be written below:

	z	x_1	x_5	x_3	x_8	x_2	x_6	x_7	x_4	RHS
	1	0	0	0	0	-1.6	-0.8	-0.4	1	6.4
x_1	0	1	0	0	0	-1.4	-0.2	-0.6	-1	4.6
x_5	0	0	1	0	0	1.8	0.4	0.2	2	4.8
x_3	0	0	0	1	0	0.6	-0.2	0.4	-1	-1.4
x_8	0	0	0	0	1	-2	-1	-1	0	7

From the initial tableau, we get $G = \{4\}, L = \{2,6,7\}$ and $b_{min} = -1.4$. So variables x_2, x_6, and x_7 are relaxed. Hence, we can write the initial tableau for the relaxation problem as follows:

	z	x_4	x_1	x_5	x_3	x_8	RHS
	1	1	0	0	0	0	6.4
x_1	0	-1	1	0	0	0	4.6
x_5	0	2	0	1	0	0	4.8
x_3	0	-1	0	0	1	0	-1.4
x_8	0	0	0	0	0	1	7

This relaxed problem can solve by the dual simplex method. After pivoting, we get the following tableau.

	z	x_4	x_1	x_5	x_3	x_8	RHS
	1	0	0	0	1	0	5
x_1	0	0	1	0	-1	0	6
x_5	0	0	0	1	2	0	2
x_4	0	1	0	0	-1	0	1.4
x_8	0	0	0	0	0	1	7

We see that the optimal solution for the relaxed problem is found. Next, we will restore variables in L and the tableau can be updated as follows:

	z	x_4	x_1	x_5	x_3	x_8	x_2	x_6	x_7	RHS
	1	0	0	0	1	0	1	-1	0	5
x_1	0	0	1	0	-1	0	-1	0	-1	6
x_5	0	0	0	1	2	0	2	0	1	2
x_4	0	1	0	0	-1	0	-1	0.2	-0.4	1.4
x_8	0	0	0	0	0	1	0	-1	-1	7

For the above tableau, the simplex method can start. After pivoting 2 iterations to slove it, we get the following tableau.

	z	x_4	x_1	x_5	x_3	x_8	x_2	x_6	x_7	RHS
	1	5	0	2	0	0	6	0	0	16
x_1	0	0	1	0.5	0	0	0.5	0	−0.5	7
x_3	0	0	0	0.5	1	0	1.5	0	0.5	1
x_6	0	5	0	2.5	0	0	8.5	1	0.5	12
x_8	0	5	0	2.5	0	1	6.5	0	−0.5	19

Therefore, the optimal solution is found that is $(x_1^*, x_2^*, x_3^*, x_4^*) = (7,0,1,0)$ with $z^* = 16$. Moreover, the number of iterations and size of matrix are compared with the two-phase simplex method are shown as below.

	Our method		Two-phase method	
	Dual simplex	Primal simplex	Phase I	Phase II
The number of iterations	1	2	3	3
Size of matrix	4×5	4×8	4×11	4×8

From Example 3.2.1, we found that our method can reduce the number of iterations. Additionally, the matrix size solved by our method is smaller than the matrix size solved by two-phase simplex method.

■

The proposed algorithm can start when G is not empty, so the special case of linear programming problems which can guarantee that the algorithm will be used is the unrestricted variable problem. Since the unrestricted variables will be replaced by the difference of two nonnegative variables, the negation of the cost vector $\mathbf{c}$ causing G is not empty.

Example 3.2.2. Consider the following linear programming problem:

$$\begin{array}{llrcrcl} \text{maximize} & z = & -7x_1 & + & 9x_2 & & \\ \text{subject to} & & 6x_1 & - & 8x_2 & \leq & 9 \\ & & 8x_1 & - & 4x_2 & \leq & 7 \\ & & -7x_1 & + & x_2 & \leq & 6 \\ & & 8x_1 & + & 9x_2 & \leq & -7 \\ & & 3x_1 & + & 9x_2 & \leq & -1 \\ & & x_1, & & x_2 & \text{are unrestricted variables.} & \end{array}$$

First, we transform variables in this problem by letting $x_j = x_j^+ - x_j^-$ for all $j = 1,2$ where $x_j^+, x_j^- \geq 0$, then we get the following problem.

$$\begin{array}{llrrrrl} \text{maximize} & z = & -7x_1^+ & +9x_2^+ & +7x_1^- & -9x_2^- & \\ \text{subject to} & & 6x_1^+ & -8x_2^+ & -6x_1^- & +8x_2^- & \leq 9 \\ & & 8x_1^+ & -4x_2^+ & -8x_1^- & +4x_2^- & \leq 7 \\ & & -7x_1^+ & +x_2^+ & +7x_1^- & -x_2^- & \leq 6 \\ & & 8x_1^+ & +9x_2^+ & -8x_1^- & -9x_2^- & \leq -7 \\ & & 3x_1^+ & +9x_2^+ & -3x_1^- & -9x_2^- & \leq -1 \\ & & x_1^+, & x_2^+, & x_1^-, & x_2^- & \geq 0 \end{array}$$

Then, the problem is converted to the standard form as below.

$$\begin{array}{llrrrrrrrrrl} \text{maximize} & z = & -7x_1^+ & +9x_2^+ & +7x_1^- & -9x_2^- & & & & & & \\ \text{subject to} & & 6x_1^+ & -8x_2^+ & -6x_1^- & +8x_2^- & +x_3 & & & & & = 9 \\ & & 8x_1^+ & -4x_2^+ & -8x_1^- & +4x_2^- & & +x_4 & & & & = 7 \\ & & -7x_1^+ & +x_2^+ & +7x_1^- & -x_2^- & & & x_5 & & & = 6 \\ & & 8x_1^+ & +9x_2^+ & -8x_1^- & -9x_2^- & & & & x_6 & & = -7 \\ & & 3x_1^+ & +9x_2^+ & -3x_1^- & -9x_2^- & & & & & x_7 & = -1 \\ & & x_1^+, & x_2^+, & x_1^-, & x_2^-, & x_3 & x_4, & x_5, & x_6, & x_7 & \geq 0. \end{array}$$

Then, we will choose the initial basis $\mathbf{B}$ as $\mathbf{I_B} = \{3,4,5,6,7\}$ that is identity matrix. Then, the initial tableau can be written below:

	z	x_1^+	x_2^+	x_1^-	x_2^-	x_3	x_4	x_5	x_6	x_7	RHS
	1	7	−9	−7	9	0	0	0	0	0	0
x_3	0	6	−8	−6	8	1	0	0	0	0	9
x_4	0	8	−4	−8	4	0	1	0	0	0	7
x_5	0	−7	1	7	−1	0	0	1	0	0	6
x_6	0	8	9	−8	−9	0	0	0	1	0	−7
x_7	0	3	9	−3	−9	0	0	0	0	1	−1

From the initial tableau, we found that G and L are not empty and $b_{min} = -7$. So variables in L are relaxed. Then, the initial tableau for the relaxed problem can be written as follows:

	z	x_1^+	x_2^-	x_3	x_4	x_5	x_6	x_7	RHS
	1	7	9	0	0	0	0	0	0
x_3	0	6	8	1	0	0	0	0	9
x_4	0	8	4	0	1	0	0	0	7
x_5	0	−7	−1	0	0	1	0	0	6
x_6	0	8	−9	0	0	0	1	0	−7
x_7	0	3	−9	0	0	0	0	1	−1

After the dual simplex method is performed that x_2^- is the entering variable and x_6 is the leaving variable, the optimal tableau for the relaxed problem is found as below.

	z	x_1^+	x_2^-	x_3	x_4	x_5	x_6	x_7	RHS
	1	15	0	0	0	0	1	0	−7
x_3	0	13.11	0	1	0	0	0.89	0	2.78
x_4	0	11.56	0	0	1	0	0.44	0	3.89
x_5	0	−7.89	0	0	0	1	−0.11	0	6.78
x_2^-	0	−0.89	1	0	0	0	−0.11	0	0.78
x_7	0	−5	0	0	0	0	−1	1	6

Next, variables x_2^+ and x_1^- will be restored which the negation of coefficients of x_2^- and x_1^+. Then, we get

	z	x_1^+	x_2^-	x_3	x_4	x_5	x_6	x_7	x_2^+	x_1^-	RHS
	1	15	0	0	0	0	1	0	0	−15	−7
x_3	0	13.11	0	1	0	0	0.89	0	0	−13.11	2.78
x_4	0	11.56	0	0	1	0	0.44	0	0	−11.56	3.89
x_5	0	−7.89	0	0	0	1	−0.11	0	0	7.89	6.78
x_2^-	0	−0.89	1	0	0	0	−0.11	0	−1	0.89	0.78
x_7	0	−5	0	0	0	0	−1	1	0	5	6

From the above tableau, x_1^- is the entering variable and x_5 is the leaving variable, and the primal simplex is performed. Then, we get the optimal tableau as below.

	z	x_1^+	x_2^-	x_3	x_4	x_5	x_6	x_7	x_2^+	x_1^-	RHS
	1	0	0	0	0	1.90	0.78	0	0	0	5.89
x_3	0	0	0	1	0	1.66	0.70	0	0	0	14.04
x_4	0	0	0	0	1	1.46	0.28	0	0	0	13.82
x_1^-	0	−1	0	0	0	0.13	−0.01	0	0	1	0.86
x_2^-	0	0	1	0	0	0.11	−0.09	0	−1	0	0.01
x_7	0	0	0	0	0	−0.63	−0.93	1	0	0	1.70

Therefore, the optimal solution is found as $(x_1^*, x_2^*) = (-0.86, -0.01)$ with $z^* = 5.89$. Furthermore, the number of iterations and size of matrix which are compared to the two-phase simplex method are shown below.

	Our method		Two-phase method	
	Dual simplex	Primal simplex	Phase I	Phase II
The number of iterations	1	1	5	1
Size of matrix	5×7	5×9	5×11	5×9

From Example 3.2.2, we found that our method can reduce the number of iterations. In addition, the matrix size solved by our method is smaller than the matrix size solved by the two-phase simplex method.

■

CHAPTER 4

EXPERIMENTAL RESULTS

In this chapter, the efficiency of our algorithm is presented by comparing the average number of iterations and the average CPU time solving generated tested problems by the propsed method with respect to the two-phase simplex method. The linear programming problem which is tested is in the following form:

$$\begin{aligned} \text{maximize} \quad & \mathbf{c^T x} \\ \text{subject to} \quad & \mathbf{Ax} \leq \mathbf{b}, \end{aligned}$$

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

All parameters are randomly generated according to these rules:

- vector $\mathbf{c}$ with range of values between $c_i \in [-9, 9]$ for all $i = 1, 2, ..., n$,
- range of values of a_{ij} between $a_{ij} \in [-9, 9]$ for all $i = 1, 2, ..., m$ and for all $j = 1, 2, ..., n$,
- vector $\mathbf{b}$ with range of values between $b_i \in [-9, 9]$ for all $i = 1, 2, ..., m$,
- 50 problems for each size of matrix,
- having around 50 percent of artificial variables in each problem,
- matrix $\mathbf{A}$ having size 10×10, 10×30, 20×20, 20×60, 40×40, 40×60, and 60×60.

For generated problems, we implemented our algorithm and two-phase method by MATLAB R 2014 programming, and these tests were run in an Intel(R) Core(TM) i5-460M 2.53 GHz and 2 GB of RAM. The algorithm started by adding slack variables which were chosen to be basic variables.

The average number of iterations for solving each problem and the average CPU time for solving each problem by our algorithm and the two-phase simplex method are shown as TABLE 4.1 and TABLE 4.2, respectively.

TABLE 4.1: The average number of iterations for solving each problem

Problem type	Size of matrix	Our algorithm	Two-phase algorithm	$\frac{\text{Our algorithm}}{\text{Two-phase algorithm}}$
1	10×10	**11.12**	13.02	0.85
2	10×30	**12.76**	14.52	0.88
3	20×20	**24.04**	26.00	0.92
4	20×60	**28.76**	30.76	0.93
5	40×40	**57.84**	70.92	0.82
6	40×60	**77.76**	80.70	0.96
7	60×60	**112.20**	125.86	0.89
Average total		46.35	51.68	0.89

TABLE 4.2: The average of CPU time for solving each problem

Problem type	Size of matrix	Our algorithm	Two-phase algorithm	$\frac{\text{Our algorithm}}{\text{Two-phase algorithm}}$
1	10×10	**0.03173** s	0.04868 s	0.65
2	10×30	**0.03306** s	0.04948 s	0.67
3	20×20	**0.04010** s	0.05178 s	0.77
4	20×60	**0.04902** s	0.07136 s	0.69
5	40×40	**0.06124** s	0.18282s	0.33
6	40×60	**0.06646** s	0.20468 s	0.32
7	60×60	**0.10144** s	0.38336 s	0.26
Average total		0.055	0.141	0.39

From TABLE 4.1 and TABLE 4.2, they can be plotted with their standard deviations as Figure 4.1 and 4.2, respectively.

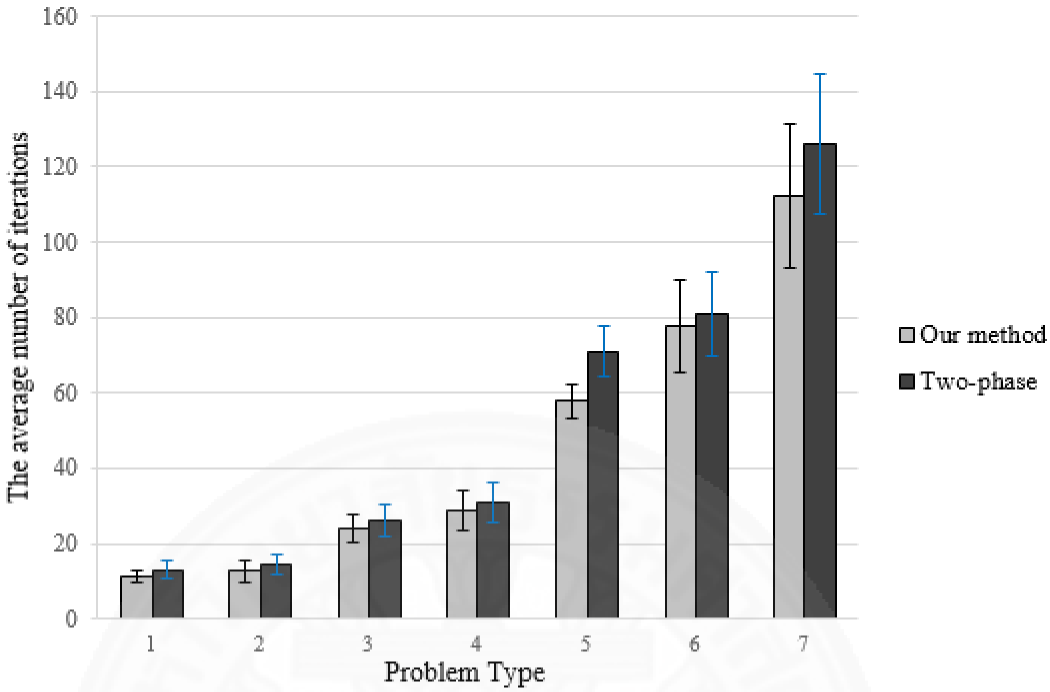


Figure 4.1: The average number of iterations for our method and the two-phase simplex method

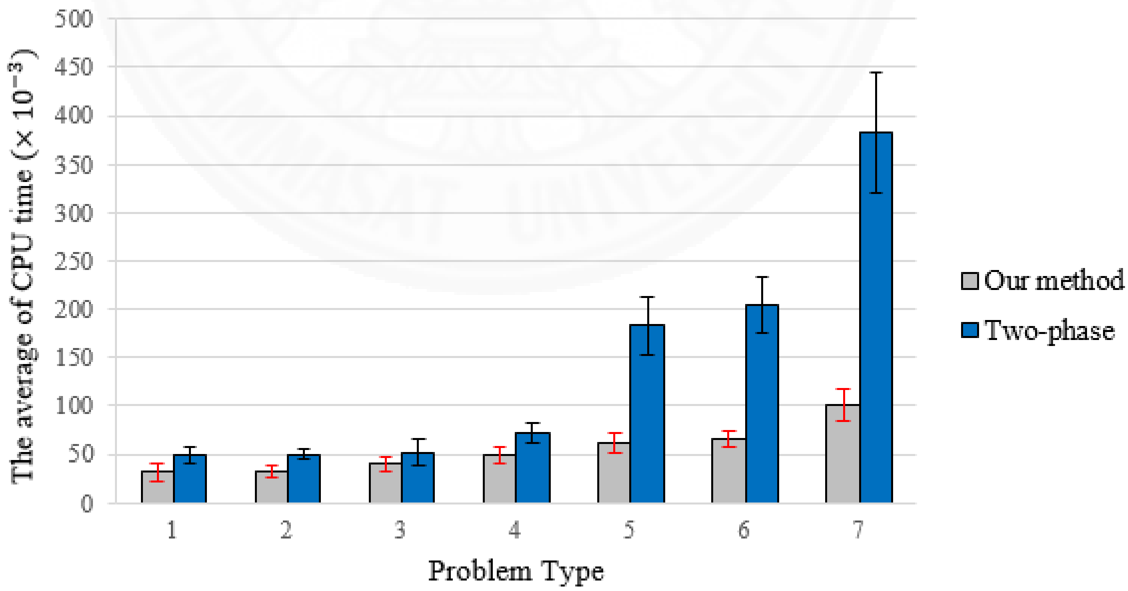


Figure 4.2: The average of CPU time for our method and the two-phase simplex method

From TABLE 4.1, we found that the average number of iterations solving by our method is less than the average number of iterations solving by the two-phase method for all problem sizes. Moreover, the maximum improvement is 18% for 40×40 matrix $\mathbf{A}$ while the minimum improvement is 4% for 40×60 matrix $\mathbf{A}$. However, the average improvement is 11% which it means that we can reduce the number of iterations in about 11%.

Additionally, we found that the average of CPU time solving by our method is less than the average of CPU time solving by the two-phase method for all problem sizes as TABLE 4.2. Furthermore, the maximum improvement is 74% for 60×60 matrix $\mathbf{A}$ while the minimum improvement is 23% for 20×20 matrix $\mathbf{A}$. However, the average improvement is 61% which it means that we can reduce the CPU time in about 61%.

CHAPTER 5

CONCLUSIONS

In this thesis, we present the improvement of the simplex method for solving a linear programming problem without using artificial varibles. Our algorithm starts by choosing an initial basis. If it gives the primal and dual infeasible solutions, then variables that cause its dual infeasible are relaxed, and the dual simplex method can be performed.

From the computational results, it indicates that our algorithm is efficient than the two-phase method. Both the average number of iterations and the average of CPU time solved by our method are less than the two-phase method for every generated problems. Since we solve the smaller relaxed problem for finding the primal feasible solution while solving by the two-phase method maintains with full matrix for finding the primal feasible solution, the computational time can be reduced. However, our algorithm could not start when the reduced costs are negative which we will use the perturbation simplex method to solved it. Nevertheless, unrestricted variable problems could always be solved by our algorithm since the negation of $\mathbf{c}$ will be added for converting it to the standard problem. So, the relaxed problem can be constructed certainly.

Since this algorithm starts by constructing the relaxed problem by choosing variables which associated the unsatisfied constraints in the dual problem. For the future work, we will construct the relaxed problem by ignoring some constraints that are satisfied the primal solution by starting at any basis. By this relaxation, the algorithm can start without using artificial variables.

REFERENCES

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

[2] G. Dantzig, Linear Programming and Extensions, Princeton University Press, New Jersey, 6(1963).

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

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

[5] A. Enge, P. Huhn, A counterexample to H. Arsham Initialization of the simplex algorithm: an artificial-free approach, SIAM Review, 40(1998), 1-6, (online).

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

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

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

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

[10] A. Boonperm and K. Sinapiromsaran, The artificial-free technique along the objective direction for the simplex algorithm, Journal of Physics: Conference Series, 490(2014) 1-4.

[11] M.S. Bazaraa, J.J. Javis, H.D. Sherail, Linear Programming ans Network Flows, John Wiley and Sons, New York, 1990.

[12] W. Li and H. Li , On simplex method with most-obtuse-angle rule and cosine rule, Applied Mathematics and Computation, 217(2011) 7867-7873.

[13] Inayatullah, Touheed and Imtiaz, A Streamlined Artificial Variable Free Version of Simplex Method, PLOS ONE Journal, 10(2015)1-28.

[14] H.V. Junior, M. Lins, An improved initial basis for the Simplex algorithm, Computers & Operations Research, 32(2005) 1983 -1993.

[15] R.G. Bland, A Combinatorial Abstraction of Linear Programming, Journal of Combinatorial Theory (Ser.B), 23(1977) 33-57.

BIOGRAPHY

Name	Miss Chanisara Prayonghom
Date of Birth	January 6, 1994
Educational Attainment	Academic Year 2015: Bachelor of Science (Mathematics), Thammasat University, Thailand
	Academic Year 2017: Master of Science (Mathematics), Thammasat University, Thailand
Scholarships	2016: Graduate Scholarship from the Faculty of Science and Technology (Thammasat University)

Publications

1. C. Prayonghom, A. Boonperm, Artificial-free Simplex Method by Dual Infeasible Variable Relaxation for LP Problems, Proceedings of The Operations Research Network of Thailand (OR-NET 2018), (2018), 296-302.

2. C. Prayonghom, A. Boonperm, X- and Y-intercepts Consideration Algorithm for solving Linear Programming Problems, Proceedings of 10th International Conference on Advences in Science, Engineering and Technology (ICASET-18), (2018), 39-44.

Oral Presentation

1. C. Prayonghom, A. Boonperm, Artificial-free Simplex Method by Dual Infeasible Variable Relaxation for LP Problems, in Operations Research Network of Thailand (OR-NET 2018), 23-24 April, 2018, Chonburi, Thailand.

Poster Presentation

1. C. Prayonghom, A. Boonperm, X- and Y-intercepts Consideration Algorithm for solving Linear Programming Problems, in 10th International Conference on Advences in Science, Engineering and Technology (ICASET-18), 20-21 June 2018, Paris, France.