

An Experiment with Simplex Method for Solving Linear Programming Problems

by

Samiha Shameem
18101007

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2022

© 2022. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

A handwritten signature in black ink, appearing to read 'Shameem', written in a cursive style.

Samiha Shameem
18101007

Approval

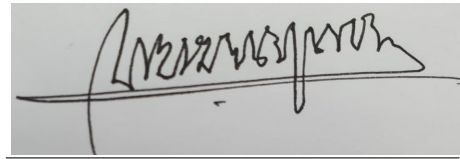
The thesis/project titled “An Experiment with Simplex Method for Solving Linear Programming Problems” submitted by

1. Samiha Shameem (18101007)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 20, 2022.

Examining Committee:

Supervisor:
(Member)



Dr. Mohammad Kaykobad
Distinguished Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Linear programming (LP) is the most popular among optimization techniques used in production, industry, planning and in other areas. Simplex method for solving LP problems has been in use for more than 70 years. Despite the fact that pathological examples can be created in which simplex algorithm requires exponential number of iterations, in practice it has been found very efficient. In this research, we aim to devise an improvised algorithm for the simplex method so as to reach optimality in fewer iterations. In a feasible region with differentiable surface, optimal solution will correspond to a point in which gradient of the objective function will coincide with the normal of the surface. LP theory asserts that if an LP has an optimal solution there is one at a vertex of the polyhedron. Unfortunately vertices of the polyhedron are in the intersection of n or more hyperplanes and therefore are not differentiable. Moreover, LP theory asserts that there is an optimal vertex at which gradient of the objective function will lie in the cone determined by normals of hyperplanes intersection of which is the vertex itself. Unfortunately finding the right combination of the hyperplanes is a combinatorial problem. Therefore, we may think of starting simplex iterations from a point where gradient of the objective function makes minimum angles with the normals of hyperplanes determining the point. It may be noted here that such a point may well be beyond feasible region, and we may need iterations to reach feasibility. We would like to carry out simulation of this algorithm and compare its performance with the existing simplex algorithm. Furthermore, we have noted two other issues of LP problem. Firstly, we show that revenue maximizing and profit maximizing LP problems are equivalent. Secondly, LP duality is robust in the sense that even if non-negativity constraints are included into the main constraints duality results hold.

Keywords: Linear Programming; Simplex Method; Basic Feasible Solution; Feasibility; Infeasibility; Duality

Acknowledgement

I would like to thank my supervisor, Dr. Mohammad Kaykobad, and my co-supervisor, Mr. M Tanvir Kaykobad, for their relentless support and guidance. I would not have been able to get as far as I have without their advice and dedication.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Acknowledgment	iv
Table of Contents	v
List of Figures	vii
List of Tables	viii
Nomenclature	ix
1 Introduction	1
1.1 Motivation for Linear Programming	1
1.2 Basics of Linear Programming	1
1.3 An Example of Linear Programming	3
1.4 Research Problem	4
1.5 Research Objectives	4
2 Literature Review	6
2.1 Introduction to Simplex Method	6
2.2 Steps of the Simplex Algorithm	6
2.3 A Problem Solved by Simplex Method	10
2.4 Related Works	16
3 Proposed Variation of Simplex Model	17
3.1 Steps to create the initial simplex tableau	17
3.2 Implementation	20
3.3 Testing	23
3.4 Results and Analysis	23
4 Some Issues on Linear Programming	26
4.1 Consistency of Primal and Dual Problems	27
4.2 Profit and Revenue Maximizing LP problems	28

5	Concluding Statements	30
5.1	Conclusion	30
5.2	Future Work	30
	Bibliography	32

List of Figures

1.1	Geometric representation of the linear inequalities	4
3.1	A general representation of the proposed simplex model	19
3.2	Flowchart of our implementation of the proposed simplex model . . .	22
3.3	Performance comparison (starting from infeasibility)	25

List of Tables

2.1	An example of initial simplex tableau	8
2.2	After first iteration	9
2.3	Production Requirements in one period (one month).	11
2.4	Production Availability in one period (one month).	11
2.5	Initial Simplex Tableau.	13
2.6	Simplex Tableau for Iteration 1.	14
2.7	Simplex Tableau for Iteration 4.	15
2.8	Simplex Tableau for Iteration 5.	15
3.1	Initial Simplex Tableau of the proposed Simplex Method.	20
3.2	Results of randomly generated test cases.	24

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

BV Basic Variable

LP Linear Programming

LPP Linear Programming Problem

LU Lower-Upper

NBV Non-basic Variable

NDA Non-degeneracy Assumption

RHS Right Hand Side

SLE System of Linear Equations

Chapter 1

Introduction

1.1 Motivation for Linear Programming

Human society has always been in short of resources to meet the growing demand. This is true not only for underdeveloped or developing countries even the most affluent societies like to satisfy demand using as less resources as possible. In the second world war USA had to support allied forces with resources that had to be transported across the Atlantic. They thought about ways and means to do that. In fact linear Programming techniques got developed around the time. In daily life we want to produce goods using minimum resources, in minimum time or given resources we would like to maximize profit by producing different goods. Finding an improved solution to a problem in a mathematical way is known as optimization. In practically every aspect of decision-making, optimization models are employed widely. From maximizing the profit in businesses, minimizing the labor costs in industries to optimal planning for radiation therapy in cancer treatments, mathematical optimization plays a significant role. In simple words, given a scenario, they are used to find the best solution from various alternatives. But of course, profit of a business cannot be maximized infinitely because various costs such as production costs, labor costs, cost of raw materials and others have to be considered. Some required materials may not be available all seasons, laborers have limited working hours per day, deadlines for certain products may need to be met, etc. Hence, in mathematical terms, we need to optimize our outcome within a given set of constraints. Mathematical optimization is widely needed in different industries. So, it is necessary to find the best method to solve such problems accurately and efficiently. This is why it is one of the most important research topics in operational research. There are many types of optimization problems. In this thesis, we will be focusing on solving problems with linear inequalities, which is known as linear programming (LP).

1.2 Basics of Linear Programming

One of the founders of LP problems is Russian mathematician Leonid Kantorovich who developed it in 1939, and it remained unknown to the western scientific community. In 1947, Dantzig was the first to publish the simplex algorithm, and in the same year, John von Neumann devised the duality theory. Dantzig was tasked with designing a mechanism for the Air Force to adopt in order to improve their planning

process [7]. This led to his initial example of using linear programming to determine the best way to assign 70 persons to 70 tasks, demonstrating its utility. The amount of processing power needed to evaluate all of the permutations in order to choose the best assignment is enormous; the number of conceivable configurations exceeds the number of particles in the cosmos. However, presenting the problem as a linear program and using the Simplex algorithm to discover the best solution takes only a few seconds. The linear programming theory minimizes the number of possible optimal solutions that must be checked dramatically. Dantzig's Simplex algorithm [2] is one of the most popular methods to solving LP problems. It has been utilized successfully in the optimization community for over 70 years. While the simplex algorithm has proven to be quite efficient in practice, Klee and Minty [3] and Zadeh [4] demonstrate that in the worst situation, it can take exponential time for the simplex algorithm to run. Russian mathematician Khachiyan [6] and later, Karmarkar [8], solved this problem by introducing polynomial time methods for tackling LP problems.

Linear programming problems (LPP) are optimization problems where *feasible regions* are defined by a set of linear inequalities usually including non-negativity constraints and one needs to find out the maximum or minimum value of a linear function known as *objective function*. Linear programming is the method of considering different inequalities relevant to a situation and calculating the best value in those conditions. The following are the basic components of a linear programming problem:

- **Decision Variables:** They are the variables in determining the outcome. The solution depends on them. To solve any problem, the decision variables must be identified first.
- **Objective Function:** This is the function to optimize. Usually objective is to maximize profit or revenue or minimize associated costs.
- **Constraints:** They are the limitations or restrictions placed on the decision variables. They typically set a limit on the value of the decision variables.
- **Feasible Region:** This is the set of all potential points that satisfy the constraints of an optimization problem. An optimal solution is a point in the feasible region that produces the optimal value of the objective function. An infeasible solution is any point outside the feasible region.

For a problem to be identified as a linear programming problem, it needs to fulfill certain criteria. The characteristics of a linear programming problem are:

1. **Constraint:** Constraints must be in the form of linear inequalities that decision variables must satisfy.
2. **Objective Function:** The objective function should be a linear function of the concerned variables.
3. **Linearity:** The relationship between the variables in the function must be linear. It means that the degree of the variables must be one.

4. Non-negativity: Usually the values of the variables should be non-negative. In most cases, the variables represent the amounts of some resource used or the levels of a set of activities, hence this non-negativity requirement is necessary. If at least one of the variables is negative or zero, then we can transform the problem to a problem with only non-negative variables. This helps bring the equations to a standard form.

1.3 An Example of Linear Programming

There are numerous approaches to solving linear programming problems, including the graphical method, the simplex method, and the use of software such as R, open solver, and others. One of the simplest ways is the graphical method. It is most used when the LP problem has two decision variables. The set of inequalities is subjected to constraints in this approach. The inequalities are then plotted on an XY plane. After plotting all the inequalities in the XY plane, the region enclosed by all the inequalities denote all the valid solutions to the problem. Thus this region is the feasible region of this LPP. So, our optimal solution lies in this region. To understand this method better, let us solve a simple linear programming problem.

Example: Suppose we have a furniture store and we only sell chairs and tables. It takes 8 hours to make a single chair and 18 hours to make a table. The cost of making a chair is \$50 and a table is \$80. Our workers work 60 hours per week. We can afford to spend \$300 for costs per week. We have decided to sell the chairs for \$80 each and tables for \$150 each. How many of each should we make so we can maximize our profit?

Solution: Let us first translate this problem into a linear programming problem. Let x be the number of chairs we will make and y be the number of tables. Creating the linear inequalities based on the statements we get,

$$\begin{array}{rcl}
 \text{Objective function:} & 80x + 150y & = z \\
 \text{Constraints:} & 8x + 18y & \leq 60 \\
 & 50x + 80y & \leq 300 \\
 & x, y & \geq 0
 \end{array}$$

So, here $80x + 150y$ is the objective function to be maximized and is denoted by Z . The coefficients 80 and 150 are known as the cost coefficients and x and y are the decision variables to be determined. The conditions $x \geq 0, y \geq 0$ are the non-negative restrictions. The solution to an optimization problem is always a corner point or vertex on the bounded region represented by some linear inequalities. Meaning, the solution will lie at the intersections (corner points) of the inequalities. Let us look at the graphical representation of the given inequalities in a 2D space. From the graph, we can see that the linear inequalities have formed a polygon defined by the vertices $(0,0)$, $(0,3.33)$, $(2.308,2.308)$ and $(6,0)$. Since we cannot produce fractional amount of furniture, we will be rounding down our solutions. This is known as integer linear programming. So, the resulting vertices are $(0,3)$, $(2,2)$ and $(6,0)$. Using our max objective function, we find the values of our objective function, denoted by Z . We find $Z = 0$ for solution $(x,y) = (0,0)$, $Z = 450$ for $(0,3)$, $Z = 460$ for $(2,2)$ and $Z = 480$ for $(6,0)$ respectively. As 480 is the maximum value,

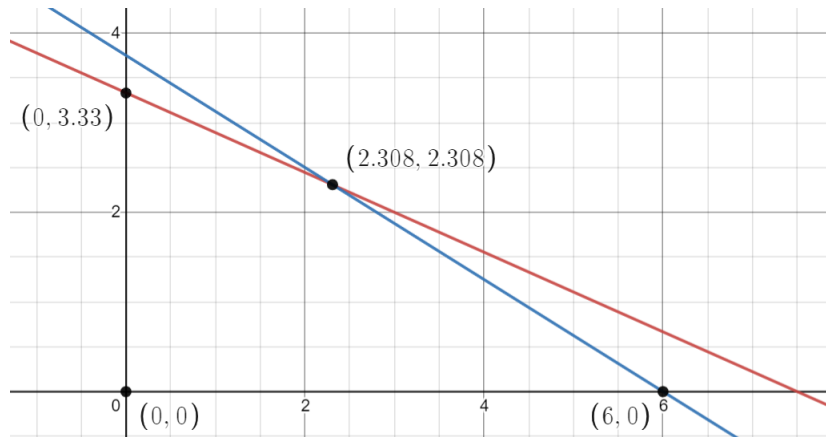


Figure 1.1: Geometric representation of the linear inequalities

we can say our optimized solution is for $(x,y) = (6,0)$. This means that we will be able to maximize our profit if we make 6 chairs and 0 tables per week.

The above problem can be solved in a 2D space, where the linear inequalities form a polygon. We can find the solution on the corner points of the polygon. But when there are many variables, we can imagine the linear inequalities enclosing a region in a multi-dimensional space. The bounded region can be thought of as a *polytope*. A polytope is a geometric object with "flat" sides in elementary geometry. It is a three-dimensional polyhedron generalization of a shape in any number of dimensions. In real life, linear programming problems include many variables and many constraints and as such, solving it geometrically becomes impossible. For large companies, the slightest mistake may cost them millions of dollars. Hence, it was of utmost need to come up with a more feasible method to solving such LP problems. Some popular algorithms that solve LP problems are the Simplex Method, Ellipsoid Method, Interior Point Method and the Cutting Plane Method, each of which have their own strengths and weaknesses. In this thesis we will be having a closer look at the Simplex Method.

1.4 Research Problem

The Simplex method is a simple iterative process. In real life circumstances, linear programming problems may end up in having large number of variables and constraints. Hence it takes a large number of iterations to solve those LP problems. This can be very costly. An improvised simplex where optimality is reached in fewer iterations can make the algorithm much more efficient.

1.5 Research Objectives

In a feasible region with differentiable surface, optimal solution corresponds to a point in which gradient of the objective function coincides with the normal of the surface. But in LP problems, the feasible region is a polytope, whose surface is not differentiable in the intersections of the *hyperplanes*. So, we intend on starting the simplex iterations from such a point where the gradient of the objective function makes minimum angles with the normals of the hyperplanes determining the point.

We would then like to carry out a simulation of this algorithm and compare its performance against the usual simplex algorithm.

We have also tried to show the robustness of the duality theory of LP through some examples. We have shown that even if we include the non-negativity constraints in the main constraints, the conversion to its dual problem does not affect the results.

We have further shown that profit and revenue maximizing LP problems are equivalent in the sense that profit maximizing solution simultaneously maximizes revenue and vice versa.

Chapter 2

Literature Review

The Simplex Method was introduced almost seven decades ago. It's been a highly efficient computational tool for tackling linear programming issues since then. For many years, the approach has been the focus of significant research. However, there are certain major aspects of its behavior that have yet to be clearly comprehended. Studies are still being conducted on solving its limitations and increasing its efficiency.

2.1 Introduction to Simplex Method

Leonid Kantorovich, a Russian mathematician who introduced linear programming problems in 1939, Dantzig, who published the simplex approach in 1947, and John von Neumann, who developed the theory of duality in the same year, are considered the founders of this topic.

Since then, many individuals have contributed to the field of linear programming in various ways, including theoretical developments, computational aspects, and exploration of new applications of the subject. According to Bazaara et al. (2009) [15], the simplex method of linear programming is universally acknowledged because of its capacity to simulate critical and complicated management decision problems and its ability to produce solutions in a fair period of time.

2.2 Steps of the Simplex Algorithm

The Simplex Method is one of the most powerful and widely used linear programming algorithms. This strategy employs an iterative approach to reach the best possible solution. In this procedure, we keep modifying the values of the basic variables until the objective function reaches its maximum value.

We start our study of the simplex algorithm by stating that if there is an optimal solution, there must also be an optimal extreme point. These extreme points are defined in terms of basic solutions that are feasible. We improve these basic solutions using the simplex method until they are optimal, or until we determine that the optimal value is unbounded. Assume R is a feasible LPP region. If the objective function has no maximum or minimum value, R is said to be unbounded. The feasible region here stretches indefinitely in any direction. If R is bounded, the objective function has a maximum and minimum value on R , with each occurring

at a vertex or corner point of R. Here the feasible region is enclosed or bounded by the system of linear inequalities.

Suppose, an SLE contains n variables and m equations where $n \geq m$. A *basic solution* for this system is obtained in the following way:

1. Set the $n - m$ variables to a value of zero. These are the non-basic variables (NBV).
2. Solve the system for the m variables that remain. These refer to the basic variables (BV).
3. The basic solution is the resultant vector of the variables. A basic feasible solution is a solution in which all variables have non-negative values.

The simplex method employs a very efficient technique. It does not compute the objective function's value at each location. It starts at the feasible region's corner point, where all of the basic variables are zero. Then it advances progressively from one corner point to the next, increasing the objective function's value at each stage. The process is repeated until the optimal solution is discovered. Let us look at the steps in details.

Step 1) Setting Up the Problem: The first step is to write the objective function, z and the inequality constraints from the given scenario. Because the simplex approach is used to solve problems with a large number of variables, variables such as x, y, z , etc. So we use symbols $x_1, x_2, x_3, \dots, x_n$ and so on.

Step 2) Converting to Standard Form: The problem must be expressed in standard form before the simplex algorithm may be used to solve it. All constraints must be equations in standard form, and all variables must be non-negative. To convert the linear inequalities of the constraints to equations, *slack variables* and *surplus variables* are introduced.

1. For each constraint i of type (\leq), a **slack** variable, s_i is added such that s_i is non-negative. For example: $3x_1 + 2x_2 \leq 5$ will translate to $3x_1 + 2x_2 + s_i = 5$ where $s_i \geq 0$.
2. For each constraint i of type (\geq), a **surplus** variable, s_i is subtracted such that s_i is non-negative. For example: $3x_1 + 2x_2 \geq 5$ will translate to $3x_1 + 2x_2 - s_i = 5$ where $s_i \geq 0$.

The objective function is then rewritten in the form,

$$-c_1x_1 - c_2x_2 - \dots - c_nx_n + 0s_1 + 0s_2 + \dots + 0s_m + z = 0$$

Step 3) Forming Initial Tableau: The simplex method is implemented by executing basic row operations on a matrix known as the simplex tableau. We may express the problem in an augmented matrix called the initial simplex tableau now that the inequalities have been transformed into equations. Each inequality constraint is represented by a separate row. In the simplex tableau, the non-negativity constraints do not appear in the rows. The objective function is written on the

bottom row.

Example: Suppose we are given the following maximizing LPP,

$$\begin{aligned} \text{Objective function, } \max, Z &= 10x_1 + 20x_2 \\ \text{Constraints, } x_1 + 2x_2 &\leq 6 \\ 3x_1 + 3x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

After following step 1 and 2, we get,

$$\begin{aligned} \text{Objective function, } -10x_1 - 20x_2 + 0s_1 + 0s_2 + z &= 0 \\ \text{Constraints, } x_1 + 2x_2 + s_1 &= 6 \\ 3x_1 + 3x_2 + s_2 &= 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Then the initial simplex tableau is shown in Table 2.1.

B	C_b	P	x_1	x_2	s_1	s_2
s_1	0	6	1	2	1	0
s_2	0	12	3	3	0	1
	1	0	-10	-20	0	0

Table 2.1: An example of initial simplex tableau

The B column indicates the basic variables. The C_b column contains the coefficients of the basic variables in the objective function, z . The P column contains the right-hand side of the constraint equations. We may also refer to this as the solution column. The remaining columns represent the left-hand side of the constraint equations. The bottom row shows the objective function.

The basic solution for the initial tableau will be $s_1 = 6, s_2 = 12, Z = 0$.

Step 4) Identifying Pivot: After setting up the initial simplex tableau for an LPP, the simplex method checks for optimality. It improves the current solution if it is not optimal. We introduce a new basic variable into the solution to improve the current solution. This new variable is known as the *entering variable*. This means that one of the current basic variables must be replaced; otherwise, the basic solution would contain too many variables. This leaving variable is known as the *departing variable* or *exiting variable*. The variables for entering and leaving are chosen as follows:

1. The most negative entry in the bottom row of the tableau corresponds to the entering variable.
2. The leaving variable is determined by the ratio of the smallest non-negative entry of the solution column with the column corresponding to the entering variable.
3. The pivot element is the entry in the entering variable's column or pivot column and the exiting variable's row.

So, from Table 2.1, we can identify the entering variable as x_2 , since it has the most negative value, -20. The column of x_2 will be the pivot column or key column. So then we find the ratios of the values of P column and x_2 column. For the first row, we get $6 \div 2 = 3$. For the second row, we get $12 \div 3 = 4$. So the minimum ratio, 3 is found in first row. Hence the first row is the departing variable's row, s_1 is the departing variable. The entry in the key column and departing variable's row is 2. Therefore 2 is our pivot or key element of the current simplex tableau.

Q. Why do we pick the bottom-most row's most negative entry?

Ans: The largest coefficient in the objective function is represented by the most negative entry in the bottom row. As a result, this coefficient will have the greatest impact on the objective function's value. So in the above objective function, $z = 10x_1 + 20x_2$, it makes more sense that we increase x_2 rather than x_1 .

Q. Why do we calculate ratios and why does the minimum ratio identify the leaving variable row?

Ans: The pivot column identifies which variable would enter the basis and increase z . In the example above, x_2 will enter. But what value of x_2 should we consider? If we take $x_2 = 50$, it violates the first condition, $x_1 + 2x_2 \leq 6$. If we take $x_2 = 6$, then it violates the second condition, $3x_1 + 3x_2 \leq 12$. Hence when we take the smallest ratio $6 \div 2 = 3$, which is the maximum value x_2 can be without breaking any of the given constraints.

Q. Why do we identify the pivot element?

Ans: The simplex approach improves the value of the objective function by starting with a corner point and then proceeding to the next corner point. Changing the number of units in the variables improves the objective function's value. We can increase the number of units in one variable while decreasing the number of units in another. This process is allowed through *pivoting* by the pivot element.

Step 5) Pivoting: We use the Gauss-Jordan Elimination method to find the improved solution in the column that contains the pivot. This is referred to as pivoting. We put 1 in the location of the pivot element by dividing the whole row with the pivot element. We make all other entries of the pivot column 0 using row operations. We see the transformation of Table 2.1 after pivoting in Table 2.2.

B	C_b	P	x_1	x_2	s_1	s_2
x_2	20	3	1/2	1	1/2	0
s_2	0	3	3/2	0	-3/2	1
	1	60	0	0	10	0

Table 2.2: After first iteration

After pivoting, step 3 is repeated until no negative entries are found at the bottom row.

Step 6) Determining Solution: When no negative entries are found in the bottom row, the iteration stops. It indicates that we have reached our optimal solution. In Table 2.2, there are no negative entries so this is our final table. From here we get the basis solution $x_2 = 3, s_2 = 3$ and

$$Z = (10 \times 0) + (20 \times 3) + (0 \times 0) + (0 \times 3) = 60.$$

So $Z=60$ is our maximized solution. The slack variable s_2 indicates that 3 units have remained unused.

Q. Why do we stop when the bottom-most row no longer contains negative entries?

Ans: The bottom row of Table 2.2 tells us the answer. The corresponding equation of the bottom row is,

$$0x_1 + 0x_2 + 10s_1 + 0s_2 + Z = 60$$

$$\text{or } Z = 60 - 10s_1$$

Since all the variables are non-negative, Z can achieve a maximum value of 60 while s_1 remains zero.

2.3 A Problem Solved by Simplex Method

Let us now solve a real life LP problem using the simplex method. The following problem has been collected from the paper by Anggoro et al. (2019) [16].

Problem

In Sukarame, Bandar Lampung, a study was conducted at the Star Bakery home industry. Here, three sorts of bread are made. Flavored bread, mattress bread, and fresh bread are the three types. Six different types of flavored bread are available: chocolate, chocolate cheese, cheese milk, blueberry, strawberry, and pineapple. The cost of this flavored bread is Rp.2,500 per package. The Mattress bread combines three distinct flavors inside a single loaf. Chocolate, chocolate cheese, cheese milk, strawberry, blueberry, and pineapple flavors are all present. The cost of each mattress bread box is Rp.6,000. Fresh Bread does not come in a variety of flavors. 12 pieces of fresh bread are contained in one package of fresh bread. Each package of freshly baked bread costs Rp.5,000.

Product Requirements:

An interview was conducted by the researchers [16] with the owners of the Bintang Bakery to know about the product requirements. The materials required to produce each of the types of bread in a one-month period is shown in Table 2.3.

Available Resources:

The availability of products for a one-month period is shown in Table 2.4. For one month, the capacity of the engine's operating hour is less than 475,200 seconds. For one month, labor hour capacity is less than 748,800 seconds.

If the current output of each bread is 3640 packs, 1300 packs, and 520 packs, how much more do we need to manufacture to optimize profit?

SL no.	Production Factor	Flavored Bread	Mattress Bread	Fresh Bread	Unit
1	Raw Materials				
	Flour	28	100	250	g
	Sugar	7	25	62	g
	Developer	1	9	4	g
	Softener	1	6	2	g
	Yellow Butter	5	20	50	g
	Salt	1	3		g
	Milk Powder	1	3	2	g
	Liquid Milk	5	20		g
	BOS Butter	5	20		g
	Egg	4	15	25	g
	Filling	14	20		g
	White butter			5	g
	Calcium			2	g
2	Production machine	32	132	336	seconds
3	Labour Hours	65	209	450	seconds

Table 2.3: Production Requirements in one period (one month).
Source: Bintang Bakery home industry, 2018

SL no.	Production Factor	Availability	Unit
1	Raw Materials		
	Flour	400	kg
	Sugar	250	kg
	Developer	90	kg
	Softener	40	kg
	Yellow Butter	90	kg
	Salt	10	kg
	Milk Powder	60	kg
	Liquid Milk	60	kg
	BOS Butter	90	kg
	Egg	70	kg
	Filling	200	kg
	White butter	90	kg
	Calcium	20	kg
2	Production machine		
	Mixer	20	hours
	Divider Machine	7	hours
	Oven	105	hours
3	Labour Hours	208	hours

Table 2.4: Production Availability in one period (one month).
Source: Bintang Bakery home industry, 2018

Solution using Simplex Method:

We first determine the decision variables and set up the objective function and the constraints. This is a maximizing LPP as we intend to increase the earnings.

Decision variables: x_1 = number of packs of flavored bread
 x_2 = number of packs of mattress bread
 x_3 = number of packs of fresh bread

Objective function: $Z = 2500x_1 + 6000x_2 + 5000x_3$

Constraints:

Flour,	$28x_1$	+	$100x_2$	+	$250x_3$	\leq	4400000
Sugar,	$7x_1$	+	$25x_2$	+	$62x_3$	\leq	250000
Developer,	$1x_1$	+	$9x_2$	+	$4x_3$	\leq	90000
Softener,	$1x_1$	+	$6x_2$	+	$2x_3$	\leq	40000
Yellow butter,	$5x_1$	+	$20x_2$	+	$50x_3$	\leq	90000
Salt,	$1x_1$	+	$3x_2$			\leq	10000
Milk powder,	$1x_1$	+	$3x_2$	+	$2x_3$	\leq	60000
Liquid milk,	$5x_1$	+	$20x_2$			\leq	60000
BOS butter,	$5x_1$	+	$20x_2$			\leq	90000
Egg,	$4x_1$	+	$15x_2$	+	$25x_3$	\leq	70000
Filling,	$14x_1$	+	$20x_2$			\leq	200000
White butter,					$5x_3$	\leq	90000
Calcium,					$2x_3$	\leq	20000
Machine,	$32x_1$	+	$132x_2$	+	$336x_3$	\leq	475200
Labor,	$65x_1$	+	$209x_2$	+	$450x_3$	\leq	748800
	x_1	,	x_2	,	x_3	\geq	0

Standard Form: Introducing slack variables on the left hand side to convert (\leq) equations to ($=$) equations,

$$\begin{aligned}
 28x_1 + 100x_2 + 250x_3 + s_1 &= 400000 \\
 7x_1 + 25x_2 + 62x_3 + s_2 &= 250000 \\
 x_1 + 9x_2 + 4x_3 + s_3 &= 90000 \\
 x_1 + 6x_2 + 2x_3 + s_4 &= 40000 \\
 5x_1 + 20x_2 + 50x_3 + s_5 &= 90000 \\
 x_1 + 3x_2 + s_6 &= 10000 \\
 x_1 + 3x_2 + 2x_3 + s_7 &= 60000 \\
 5x_1 + 20x_2 + s_8 &= 60000 \\
 5x_1 + 20x_2 + s_9 &= 90000 \\
 4x_1 + 15x_2 + 25x_3 + s_{10} &= 70000 \\
 14x_1 + 20x_2 + s_{11} &= 200000 \\
 &+ 5x_3 + s_{12} = 90000 \\
 &+ 2x_3 + s_{13} = 20000 \\
 32x_1 + 132x_2 + 336x_3 + s_{14} &= 475200 \\
 65x_1 + 209x_2 + 450x_3 + s_{15} &= 748800
 \end{aligned}$$

Therefore, objective function,

$$\begin{aligned}
 Z &= 2500x_1 + 6000x_2 + 5000x_3 + 0s_1 + 0s_2 + 0s_3 + 0s_4 + 0s_5 \\
 &+ 0s_6 + 0s_7 + 0s_8 + 0s_9 + 0s_{10} + 0s_{11} + 0s_{12} + 0s_{13} + 0s_{14} + 0s_{15}
 \end{aligned}$$

Iteration 0: Now, we create the initial tableau (Table 2.5). In Table 2.5, Z_j row is calculated as sum of the product of $C_{b_i} \times a_{ij}$ where $i = 1, 2, \dots, 15$ indicate the rows of basic variables and $j = 0, 1, 2, \dots, 15$ indicate the columns P to s_{15} . C_j indicate the coefficients of the variables in the objective function, Z which are shown in the

			2500	6000	5000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Min ratio	
B	C_b	P	x_1	x_2	x_3	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	P/key col.	
s_1	0	48666.67	0	0	0	1	0	0	0	0	16.33	0	-1.64	0	0	0	0	0	0	0	-0.56	-29594.59
s_2	0	162245.33	0	0	0	0	1	0	0	0	4.03	0	-0.41	0	0	0	0	0	0	0	-0.14	-391686.7
s_3	0	67370.67	0	0	0	0	0	1	0	0	5.45	0	-1.18	0	0	0	0	0	0	0	-0.01	-57331.32
s_4	0	23685.33	0	0	0	0	0	0	1	0	2.23	0	-0.59	0	0	0	0	0	0	0	0	-40311.65
s_5	0	22133.33	0	0	0	0	0	0	0	1	5.67	0	-0.69	0	0	0	0	0	0	0	-0.11	-32129.03
s_{14}	0	30336	0	0	0	0	0	0	0	0	42.08	0	-5.11	0	0	0	0	0	0	1	-0.75	-5937.37
s_7	0	49685.33	0	0	0	0	0	0	0	0	-0.77	1	0.01	0	0	0	0	0	0	0	0	3992571.43
x_2	6000	2000	0	1	0	0	0	0	0	0	-1	0	0.2	0	0	0	0	0	0	0	0	10000
s_9	0	30000	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	-30000
s_{10}	0	20066.67	0	0	0	0	0	0	0	0	1.83	0	-0.44	0	1	0	0	0	0	0	-0.06	-45150
s_{11}	0	104000	0	0	0	0	0	0	0	0	-36	0	4.4	0	0	1	0	0	0	0	0	23636.36
s_{12}	0	89213.33	0	0	0	0	0	0	0	0	0.57	0	0.03	0	0	0	1	0	0	0	-0.01	2867571.43
s_{13}	0	19685.33	0	0	0	0	0	0	0	0	0.23	0	0.01	0	0	0	0	1	0	0	0	1581857.14
x_3	5000	157.33	0	0	1	0	0	0	0	0	-0.11	0	-0.01	0	0	0	0	0	0	0	0	-25285.71
x_1	2500	4000	1	0	0	0	0	0	0	0	4	0	-0.6	0	0	0	0	0	0	0	0	-6666.67
Z_j		22786666.67	2500	6000	0	0	0	0	0	0	3433.33	0	-331.11	0	0	0	0	0	0	0	11.11	
$Z_j - C_j$			0	0	0	0	0	0	0	0	3433.33	0	-331.11	0	0	0	0	0	0	0	11.11	

Table 2.7: Simplex Tableau for Iteration 4.

			2500	6000	5000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	C_b	P	x_1	x_2	x_3	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	
s_1	0	65111.11	0	8.22	0	1	0	0	0	0	8.11	0	0	0	0	0	0	0	0	0	-0.56
s_2	0	166387.56	0	2.07	0	0	1	0	0	0	1.96	0	0	0	0	0	0	0	0	0	-0.14
s_3	0	79121.78	0	5.88	0	0	0	1	0	0	-0.42	0	0	0	0	0	0	0	0	0	-0.01
s_4	0	29560.89	0	2.94	0	0	0	0	1	0	-0.71	0	0	0	0	0	0	0	0	0	0
s_5	0	29022.22	0	3.44	0	0	0	0	0	1	2.22	0	0	0	0	0	0	0	0	0	-0.11
s_{14}	0	81429.33	0	25.55	0	0	0	0	0	0	16.53	0	0	0	0	0	0	0	0	1	-0.75
s_7	0	49560.89	0	-0.06	0	0	0	0	0	0	-0.71	1	0	0	0	0	0	0	0	0	0
s_8	0	10000	0	5	0	0	0	0	0	0	-5	0	1	0	0	0	0	0	0	0	0
s_9	0	40000	0	5	0	0	0	0	0	0	-5	0	0	1	0	0	0	0	0	0	0
s_{10}	0	24511.11	0	2.22	0	0	0	0	0	0	-0.39	0	0	0	1	0	0	0	0	0	-0.06
s_{11}	0	60000	0	-22	0	0	0	0	0	0	-14	0	0	0	0	1	0	0	0	0	0
s_{12}	0	88902.22	0	-0.16	0	0	0	0	0	0	0.72	0	0	0	0	0	1	0	0	0	-0.01
s_{13}	0	19560.89	0	-0.06	0	0	0	0	0	0	0.29	0	0	0	0	0	0	1	0	0	0
x_3	5000	219.56	0	0.03	1	0	0	0	0	0	-0.14	0	0	0	0	0	0	0	0	0	0
x_1	2500	10000	1	3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Z_j		26097777.78	0	7655.56	0	0	0	0	0	0	1777.78	0	0	0	0	0	0	0	0	0	11.11
$Z_j - C_j$			0	1655.56	0	0	0	0	0	0	1777.78	0	0	0	0	0	0	0	0	0	11.11

Table 2.8: Simplex Tableau for Iteration 5.

number of packs of flavored bread to be produced, $x_1 = 10000$
number of packs of mattress bread to be produced, $x_2 = 0$
number of packs of fresh bread to be produced, $x_3 = 219.56 \approx 219$

Therefore, in optimal production, earnings = Rp. $(2500 \times 10000) + (5000 \times 219) =$
Rp. 26,095,000

So, earnings will increase by = Rp. $(26,095,000 - 19,500,000) =$ Rp. 6,595,000.

2.4 Related Works

Dantzig's Simplex algorithm [2] has been used in the optimization community very successfully for about 70 years now. While this method has proved to be very efficient practically, some pathological examples created by Klee and Minty [3] and Zadeh [4] show that in the worst case, the simplex algorithm can take exponential time to execute. This problem was addressed by Russian mathematician Khachiyan [6] and later on, Karmarkar [8] who introduced polynomial time algorithms for solving LP problems. In addition, some more research works were done in [11] and [14] to improve the simplex algorithm. In a paper, Murshed et al [9] initiated some thoughts of starting simplex method from a point that is obtained by the intersection of the normals of the hyperplanes which make minimum angles with the gradient of the objective function. Though the paper had some flaws, the same idea was used by Junior and Lins [10], to accelerate the simplex method by selecting suitable initial solutions. Further work was done by Hu [13], who was able to improve the solution by applying LU decomposition. LU decomposition refers to the factorization of a square matrix into one upper triangular matrix and one lower triangular matrix, such that the original matrix can be obtained from the product of these two triangular matrices.

One of the most important discovery in the field is the duality theory of Linear Programming. According to George Dantzig, John von Neumann conjectured the duality theorem for linear optimization shortly after Dantzig presented him with the linear programming issue in 1947. Neumann also developed a link between the Linear Programming issue and game theory at the time. At the very first conference on Linear Programming in Chicago in 1949, Tucker and two of his graduate students, David Gale and Harold W. Kuhn, presented the first rigorous proof of the duality theorem in their first article "Linear Programming and the Theory of Games" [12]. The theorem states that on the same set of data, one can formulate another linear minimizing programming problem for a linear maximizing programming problem, or a linear maximizing programming problem for a linear minimizing programming problem. Hence the term dual problem. The existence of a finite optimal solution to one of them assures the existence of an optimal solution to the other, and the values of both objective functions are the same. Albert W Tucker and his group presented the rigorous evidence of duality theory as early as 1948, according to Dantzig's prologue to Nering and Tucker, 1993.

On pathological examples of [3] and [4], Dantzig's simplex algorithm takes exponential time. Through the development of polynomial time algorithms for handling LP issues, Khachiyan [6] and subsequently Karmarkar [8] properly addressed this issue. Furthermore, Bland [5] has eliminated the unfavorable cycling problem.

Chapter 3

Proposed Variation of Simplex Model

In a maximizing LPP whose objective function lies in a *convex feasible region* with a differentiable surface, the gradient of the surface would coincide with the gradient of the objective function at the optimal point. Here, convex feasible region indicates a region in multi-dimensional space, bounded by hyperplanes with vertices as its corners. However, in LP problems, we have a finite amount of hyperplanes and the surface is not differentiable at the basic feasible points (vertices of the polyhedron). According to LP theory, at optimality, gradient of the objective function lies in the cone determined by the normals of the hyperplanes the optimal solution lies on. But computing all the combinations of the normals of the hyperplanes is not easy neither efficient. So, intuitively we can look for the optimal solution on n hyperplanes, whose normals make minimum angles with the gradient of the objective function. Now the vertex at such a region may not be feasible, but it should possibly be near the optimal solution in a combinatorial sense. But first, it is necessary to remove redundant half spaces from the system of linear inequalities to ensure the existence of solution to the obtained system of linear equations. Assuming that vectors a_i 's are normalized by multiplying the corresponding rows by an appropriate constant, we compute $s^t \cdot a_i, \forall i = \{1, 2, \dots, m\}$. Arrange $s^t \cdot a_i$ and $s_j, \forall j = \{1, 2, \dots, n\}$ in descending order of their values. Choose the set S of variables corresponding to the last m half spaces as candidates for basic variables. Starting with the first simplex tableau, we forcibly select entering variables $x_k, k \in S$ from this list, and as exiting variable we select one that is not in S , ignoring the usual rules of selecting rows and columns. Note that it is possible for this solution to be infeasible. So, necessary steps need to be taken to bring the basis to feasibility. Finally, this basic solution as the initial tableau can be passed to an existing version of the simplex method to find the optimal value.

3.1 Steps to create the initial simplex tableau

The steps to create the initial simplex tableau are given below.

1. Initially, for n variables and m constraints, the constraints and objective function are converted into unit vectors for ease of calculation.

2. Then the angle made by the normal of the hyperplane determining each of the constraints with the objective function gradient is calculated by using the formula of dot product, $\vec{a} \cdot \vec{b} = |\vec{a}| \cdot |\vec{b}| \cdot \cos\theta$.
3. Then the angles are sorted in ascending order, or the cosine values are sorted in descending order.
4. The initial non-basic variables are taken from the top n hyperplanes of the sorted list.
5. The remaining m variables from the bottom will be the initial basic variables.
6. Now, the initial simplex tableau is to be created with this set of basic variables. The basic variables are entered one by one and row reduction operations are applied to ensure that each row contains only one basic variable.
7. If coefficient of an entering basic variable becomes zero in the row due to previous row reductions, the basic variable can be replaced with a non-basic variable, whose coefficient is not zero in that row. This will prevent zero-division error when pivoting.
8. The resulting basis may be infeasible. In that case, necessary measure need to be taken to bring the basis to feasibility.
9. After the initial tableau is created, it is passed on to the regular simplex algorithm to compute the optimal solution.

Step-by-step demonstration on an LP example Let us take a maximizing LPP to demonstrate how to create the initial simplex tableau.

$$\begin{array}{ll}
 \text{Objective function:} & 2x_1 + 3x_2 \\
 \text{Constraints:} & 4x_1 + 3x_2 \leq 10 \\
 & 2x_1 + 7x_2 \leq 20 \\
 & x_1 \geq 0 \\
 & x_2 \geq 0
 \end{array}$$

Converting the constraints to standard form, we get,

$$\begin{array}{l}
 4x_1 + 3x_2 + s_1 = 10 \\
 2x_1 + 7x_2 + s_2 = 20
 \end{array}$$

First, finding the normals and converting them to unit vectors, we get,

$$\begin{array}{ll}
 \text{Objective function:} & \frac{2}{\sqrt{13}}x_1 + \frac{3}{\sqrt{13}}x_2 \\
 \text{Subject to,} & \frac{4}{5}x_1 + \frac{3}{5}x_2 \quad (1) \\
 & \frac{2}{\sqrt{53}}x_1 + \frac{7}{\sqrt{53}}x_2 \quad (2) \\
 & x_1 \quad (3) \\
 & x_2 \quad (4)
 \end{array}$$

Now, we find the angle between the objective function and the normal of the hyperplanes.

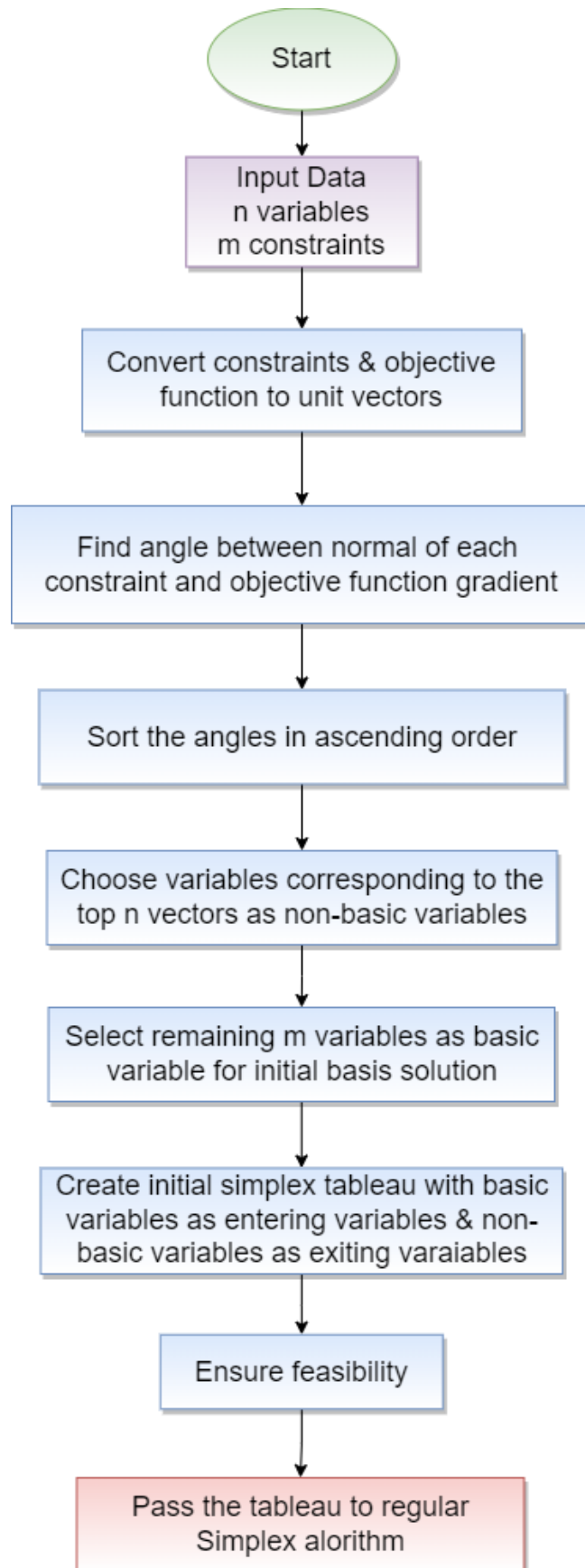


Figure 3.1: A general representation of the proposed simplex model

i	Dot product between gradient of objective function and normal i	$\cos\theta$	θ
1	$\frac{2}{\sqrt{13}} \cdot \frac{4}{5} + \frac{3}{\sqrt{13}} \cdot \frac{3}{5}$	0.94	19.44°
2	$\frac{2}{\sqrt{13}} \cdot \frac{2}{\sqrt{53}} + \frac{3}{\sqrt{13}} \cdot \frac{7}{\sqrt{53}}$	0.95	17.74°
3	$\frac{2}{\sqrt{13}} \cdot 1 + \frac{3}{\sqrt{13}} \cdot 0$	0.55	56.31°
4	$\frac{2}{\sqrt{13}} \cdot 0 + \frac{3}{\sqrt{13}} \cdot 1$	0.83	33.69°

From the above table, we see top 2 hyperplanes are eqn (i) and (ii). They are the closest to the objective function. So from there we get our first solution, $s_1 = 0$ and $s_2 = 0$. So, our linear system of equation becomes,

$$\begin{aligned} 4x_1 + 3x_2 &= 10 \\ 2x_1 + 7x_2 &= 20 \end{aligned}$$

So our initial non-basic variables are s_1 and s_2 , and our initial basic variables are x_1 and x_2 . We get our first basic feasible solution by solving the current system of linear equations. We get,

$$(x_1, x_2, s_1, s_2) = \left(\frac{10}{22}, \frac{30}{11}, 0, 0\right)$$

Now we need to create the initial simplex tableau. To do that, we need to eliminate the other basic variables in each equation as only one basic variable can exist in one row. We currently have,

$$\begin{aligned} 4x_1 + 2x_2 + s_1 &= 10 & \text{(i)} \\ 2x_1 + 7x_2 + s_2 &= 20 & \text{(ii)} \\ \text{Performing } (i) \times 7 - (ii) \times 3, & \quad x_1 + \frac{7}{22}s_1 - \frac{3}{22}s_2 = 10/22 & \text{(iii)} \\ \text{Performing } (i) - (ii) \times 2, & \quad x_2 - \frac{1}{11}s_1 + \frac{2}{11}s_2 = 30/11 & \text{(iv)} \end{aligned}$$

Now, we create the initial simplex tableau using (iii) and (iv) as shown in Table 3.1.

			2	3	0	0
B	C_b	P	x_1	x_2	s_1	s_2
x_1	2	10/22	1	0	7/22	-3/22
x_2	3	30/11	0	1	-1/11	2/11
Z_j		100/11	2	3	4/11	3/11
$Z_j - C_j$			0	0	4/11	3/11

Table 3.1: Initial Simplex Tableau of the proposed Simplex Method.

In Table 3.1, we see that bottom row has no negative values, that means we have reached optimality. From this LP problem, we can observe that we get the optimal solution directly from the initial simplex tableau using the improvised simplex algorithm, i.e 0 iterations were required. The regular simplex algorithm would take 3 iterations to reach optimality.

3.2 Implementation

The efficiency of this improvised model requires comprehensive experiments with random data, but intuitively this proposed algorithm is expected to produce good results.

Initially after taking input, a regular simplex table is created with the objective function in the first row, followed by the constraints. The last row is kept for $Z_j - C_j$. The first column is for C_b and the last is the solution column. Constraints having \leq inequalities are added as they are, \geq inequalities are multiplied by -1 to convert to \leq inequalities and equal constraints, ($=$), are broken down to \leq and \geq inequalities and handled accordingly. We know, for n variables and m constraints, there will be a total of $n + m$ constraints. We now convert the hyperplane vectors and objective function to unit vectors by dividing them by their lengths. The resulting vectors are the normals of the hyperplanes and objective function gradient respectively. We store them in a different table for ease of calculation. Here, the first m vectors indicate the normal of the first m hyperplanes of given constraints and the last n vectors indicate the same for the non-negativity constraints for all decision variables. We then use the dot product formula $\vec{a} \cdot \vec{b} = |\vec{a}| \cdot |\vec{b}| \cdot \cos\theta$ to find the values of $\cos\theta$. We have already precalculated $\frac{\vec{a}}{|\vec{a}|}$ and $\frac{\vec{b}}{|\vec{b}|}$ where \vec{a} are the normals of hyperplanes and \vec{b} is the objective function gradient. The cosine values indicate the angles between them. The larger the value of cosine, smaller the angle. So, we sort the cosine values in descending order. Then we take top n hyperplanes from the sorted list and from there, the corresponding numbered variables are stored as initial non-basic variables and the remaining m are stored as initial basic variables. Corresponding numbered variable of given constraint indicate its slack variable and that of non-negativity constraint indicate its decision variable. Now, we forcibly enter these basic variables as entering variables, if it isn't already there, by replacing the non-basic variables, taken as exiting variables. We apply row reduction operations to ensure that each row contains only one basic variable. While entering a basic variable, if we notice that its coefficient in that row has become zero due to previous row reductions, we can replace this basic variable with a non-basic variable to avoid zero-division error when pivoting. We must also ensure that the non-basic variable that we are entering into the basis should also have a non-zero coefficient in that row. In this way our initial simplex tableau is created. All values are stored as fractions to minimize data loss in calculations.

But this initial basis may be infeasible. If the solution column contains negative values, we know that it is infeasible because of non-negativity restriction of LP. To ensure feasibility, we slightly change the rules for selecting entering and exiting variables. Usually we consider only the positive entries when finding the minimum ratio for selecting exiting variables. But to eliminate the negative entries in solution column, we consider the ratios of each of the negative entry in solution column as numerator, with negative entries in its corresponding row as denominator. This maintains our positive minimum ratio rule in pivoting. We can deduce the entering and exiting variable from the minimum ratio as before and pivot. We continue this process until our basis becomes feasible, i.e all negative entries are eliminated from the solution column.

Figure 3.2 shows how we have implemented the proposed simplex model.

Theoretically, our proposed simplex method should take fewer iterations to reach the optimal solution than the regular simplex method.

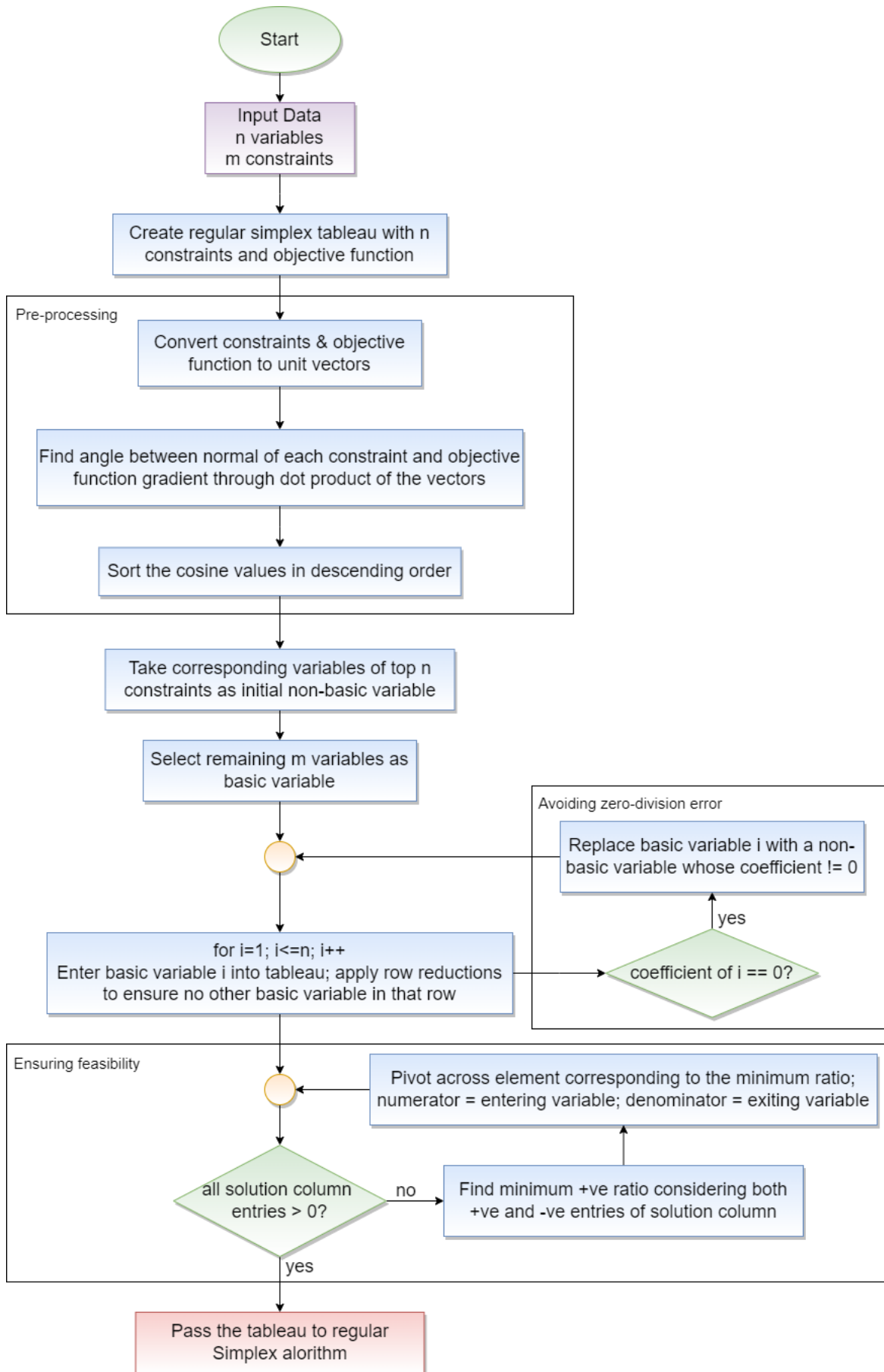


Figure 3.2: Flowchart of our implementation of the proposed simplex model

3.3 Testing

The proposed initial solution was put to the test by creating a series of test cases with n variables and m ' \leq ' inequality constraints. Each coefficient in the objective function and the restrictions was chosen at random from a uniform distribution within a range of $[-100, 100)$. We also randomly created a point \bar{x} and calculated the right hand side vector that meets all the conditions, including the non-negativity ones, to ensure that the system is feasible. That is, $b = A\bar{x}$ has been used to determine the right hand side of the restrictions. Random test cases were produced in this way for each combination of variables and constraints, which were then solved using the normal simplex approach and the proposed variant of the simplex method. The average number of iterations required to achieve optimality has then been recorded for each size of test case. The regular simplex technique performed worse than our proposed version for LP problems with negative right hand side values.

To discover the initial basic solution, repeat the steps below until we have these m basic variables in our tableau, as shown at the end of the previous subsection.

1. We select basic variables as our entering variables and non-basic variables as our outgoing variables to replace undesired non-basic variables.
2. We substitute an entering basic variable with any non-basic variable to prevent the pivot element from becoming zero.

We reach feasibility by iterating our tableau using the following principles once we've reached our goal tableau:

1. As with the simplex approach, select the column of the entering variable.
2. If the current solution is infeasible, instead of considering solely ratios of positive entries on the right hand side and in the entering column, choose the smallest of the positive ratios to shift from infeasibility to feasibility.

We repeat this process until our basis is feasible, i.e. all negative items in the solution column are removed. We then apply the normal simplex approach to attain optimality once feasibility has been established.

3.4 Results and Analysis

We summarized the simulation findings in the following table by presenting key parameters, and then showed the results in graphs.

We conducted various experiments using our implementation¹ of the proposed simplex algorithm, the results of which are displayed in Table 3.2. The number of test cases we conducted for a combination of m , n values, as well as the average number of iterations it took to reach optimality, are listed in each row. Our findings were then divided into two groups: those in which the first tableau related to a fundamental feasible solution and those in which it was in the infeasible region. Because the

¹<https://github.com/shameme97/A-Variation-of-Simplex-Algorithm>

Table 3.2: Results of randomly generated test cases.

Cases	n	m	Average number of iterations					
			General Simplex	Proposed Simplex	With Feasibility		With Infeasibility	
					General	Proposed	General	Proposed
100	10	10	12.27	15.05	9.27	16.25	15.39	13.8
100	10	20	19.59	18.72	16.04	20.71	22.49	17.09
100	20	10	18.70	21.23	14.11	22.26	24.51	19.62
100	20	20	33.58	43.91	24.38	44.85	43.96	42.85
100	20	40	53.68	48.77	42.06	48.33	56.23	48.87
100	30	20	51.88	59.72	32.81	59.83	65.69	59.64
100	30	30	84.22	82.31	43.90	91.30	88.99	84.67
100	30	60	106.51	92.87	73.67	92.07	112.31	93.01
100	40	30	95.04	112.28	55.62	117.00	118.19	109.51
100	40	40	144.53	137.78	77.18	147.12	158.33	135.87
100	40	60	163.24	133.24	98.09	150.64	171.29	131.09
100	50	40	153.71	170.58	87.83	175.70	181.94	168.39
100	50	50	203.33	206.98	118.71	220.43	225.39	198.13
100	50	70	252.10	189.98	157.00	179.50	254.04	190.19
100	60	40	153.20	200.92	96.26	202.12	210.14	199.72
100	60	60	300.25	277.06	160.83	305.17	319.26	273.23
50	60	80	335.84	262.34	188.50	287.50	341.98	261.29
50	70	50	245.60	281.00	134.69	292.25	297.79	275.71
50	70	70	392.82	370.67	183.33	423.00	425.05	362.62
20	70	90	437.30	320.80	209.0	252.00	462.67	328.44
20	80	50	236.23	314.95	143.09	307.45	329.36	322.45
20	80	60	296.45	349.40	160.00	357.00	354.93	374.71
20	80	80	531.65	460.00	319.25	442.50	584.75	464.38
50	90	70	433.56	464.30	246.33	474.44	474.66	462.07
50	90	90	679.98	583.54	332.00	625.60	720.44	578.65
20	100	50	286.05	318.30	157.22	314.44	391.45	321.45
20	100	60	484.55	471.25	320.00	464.50	525.69	472.94

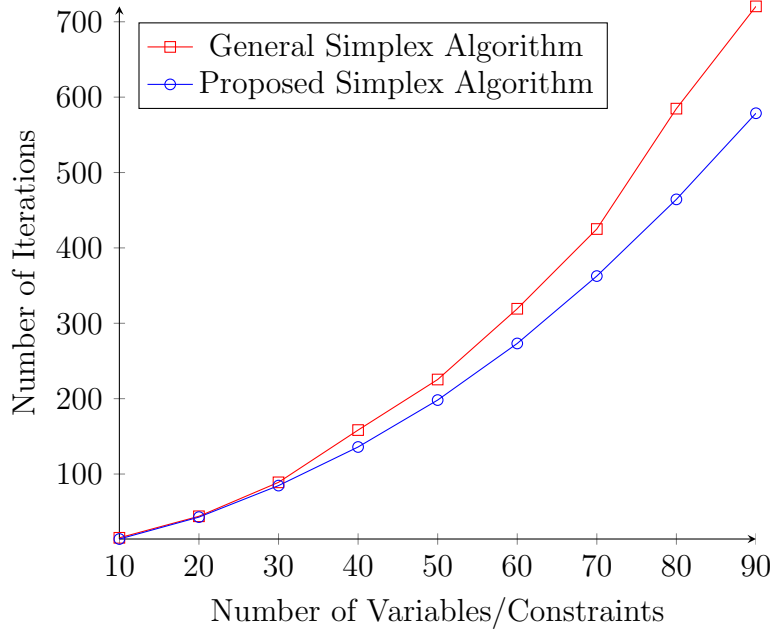


Figure 3.3: Performance comparison (starting from infeasibility)

right hand side was calculated using matrix A and the specific point \bar{x} we wished to make feasible, it could contain some negative values, in which case the first simplex tableau represents an infeasible solution. The general simplex, on the other hand, iterates towards optimality if the very initial tableau corresponds to a basic feasible solution. However, in this case, the proposed variant takes a detour through the point of intersection of hyperplanes that make the smallest angle with the objective function gradient, despite the fact that this detour may result in iterating from a basic feasible solution to a basic infeasible solution, then iterating for feasibility, and finally for optimality. This is why, in the scenario where the first tableau corresponds to a basic feasible solution, the proposed variant requires a higher number of iterations. It is worth noting that only a small percentage of the cases, the tests we did resulted in right-hand side vectors with all-positive components. Needless to say, in most cases, the right hand side vector will have non-positive components, therefore the conventional simplex algorithm will begin with an infeasible solution and perform poorly in comparison to the new technique.

Figure 3.3 has been prepared from Table 3.2. It displays the amount of iterations required by the general simplex and suggested variation for LP problems, with each point (x, y) indicating that on average, for LP problems with x variables and x constraints, y iterations were required to obtain optimality. The proposed simplex method reaches optimality in fewer iterations when the LP problems start from an infeasible region, as shown in the figure. The difference between the average number of iterations taken by each approach grows as the number of variables/constraints grows. When dealing with a high number of variables and constraints, the proposed simplex appears to be more efficient. On the other hand, for LP problems starting from a feasible region, the general simplex method seems to outperform our proposed simplex algorithm. While LP problems typically include many more variables than inequality, we couldn't incorporate the results of such tests because $m \ll n$.

Chapter 4

Some Issues on Linear Programming

The result of duality theory in LP theory is noteworthy. We give two results in this note. To begin, this report shows that even when non-negativity constraints are included in the collection of main constraints, duality conclusions remain valid. Second, it illustrates how revenue maximization and profit maximization are logically similar problems.

Let us consider the following LP problem for maximizing sales of an industry:

$$\begin{aligned} s^t x &\rightarrow \text{maximize} & (4.1) \\ Ax &\leq b \\ x &\geq 0 \end{aligned}$$

where A is an $m \times n$ matrix, x and b are vectors having respectively n and m components. Each entry i of the (unit) vector s represents unit sale's price of the produce i .

Its dual problem is

$$\begin{aligned} b^t y &\rightarrow \text{minimize} & (4.2) \\ A^t y &\geq c \\ y &\geq 0 \end{aligned}$$

There is a dual variable in (4.2) for each constraint in (4.1). The equivalent dual variable is unlimited if the constraint is equality, and vice versa. Similarly, there is a dual constraint in (4.2) for every variable in (4.1). The related dual constraint is equality if the variable is unconstrained, and vice versa.

LP theory asserts that if the inequalities in (4.1) and (4.2) are consistent, that is if there exist x_1 and y_1 such that

$$Ax_1 \leq b, \quad x_1 \geq 0$$

and

$$A^t y_1 \geq c, \quad y_1 \geq 0$$

are both satisfied then there are x^* and y^* that are optimal to corresponding systems, and $s^t x^* = b^t y^*$.

Furthermore, the solutions satisfy linear complimentary slackness characteristics. That is

$$y^{*t}(b - Ax^*) = 0, \quad x^{*t}(c - A^t y^*) = 0$$

The equivalent dual variables in (4.2) are unconstrained if some of the restrictions in (4.1) are equality, and vice versa. The following small examples demonstrate how the unboundedness of one problem's solution leads to the infeasibility of the other.

$$\begin{array}{ll}
 x_1 + x_2 \rightarrow \text{maximize} & y_1 \rightarrow \text{minimize} \\
 x_1 - x_2 = 1 & y_1 = 1 \\
 x_1, x_2 \text{ unrestricted} & -y_1 = -1 \\
 & y_1 \text{ unrestricted}
 \end{array} \tag{4.3}$$

It's also possible that both LP issues are impossible to solve, as in Gale, Kuhn, and Tucker's example [1].

$$\begin{array}{ll}
 2x_1 + 5x_2 \rightarrow \text{maximize} & 2y_1 + 5y_2 \rightarrow \text{minimize} \\
 x_1 - x_2 = 2 & y_1 + 2y_2 = 2 \\
 2x_1 + 2x_2 = 5 & y_1 + 2y_2 = 5 \\
 x_1, x_2 \text{ unrestricted} & y_1, y_2 \text{ unrestricted}
 \end{array} \tag{4.4}$$

In this chapter we discuss some special properties of linear programming in LPP. In Section 4.1 we show that if even nonnegativity constraints are inserted in main inequalities LP theory connecting primal and dual problems remain consistent. In Section 4.2, we establish equivalence of profit and revenue maximizing LP problems.

4.1 Consistency of Primal and Dual Problems

In this section, we formulate the primal problem by merging the nonnegativity constraints into the main inequalities as follows:

$$\begin{array}{lll}
 s^t x & \rightarrow & \text{maximize} \\
 Ax & \leq & b \\
 -Ix & \leq & 0 \\
 x & \text{unrestricted} &
 \end{array} \tag{4.5}$$

Its dual problem is

$$\begin{array}{ll}
 b^t y + 0^t z & \rightarrow \text{minimize} \\
 A^t y - Iz & = c \\
 y, z & \geq 0
 \end{array} \tag{4.6}$$

where y represents constraints $Ax \leq b$ and z represents constraints corresponding to $-Ix \leq 0$. The system (4.6) can also be written as

$$\begin{aligned} b^t y &\rightarrow \text{minimize} \\ A^t y &\geq c \\ y &\geq 0 \end{aligned} \tag{4.7}$$

which is identical to (4.2) implying that inequality constraints can be considered part of main constraints without causing any anomalies.

4.2 Profit and Revenue Maximizing LP problems

Let's look at the second issue, which is how revenue and profit maximization difficulties are linked. The e_j in (4.8) is a unit vector with j th component being 1 and all α 's and β 's are non-negative values. The α_i 's are known as shadow prices related to the resource i , and they imply that if the resource i is increased by a marginal amount, the objective function value will increase by α_i per unit increase of the resource i .

$$\begin{aligned} a_i^t x &= b_i, \quad \forall i \in I \\ x_j &= 0, \quad \forall j \in J \\ |I| + |J| &\geq n \text{ and } s = \sum_{i \in I} \alpha_i a_i + \sum_{j \in J} \beta_j e_j, \end{aligned} \tag{4.8}$$

Naturally, shadow pricing have nothing to do with the actual costs of these commodities. However, because most goods are sold for a profit,

$$\sum_{i=1}^m a_{ij} p_i \leq s_j, \quad \forall j \in \{1, \dots, n\} \tag{4.9}$$

where the unit price of resource i is p_i . Dual of this LP is

$$\begin{aligned} y^t b &\rightarrow \text{minimize} \\ A^t y &\geq s \\ y &\geq 0 \end{aligned} \tag{4.10}$$

If the primal problem in (4.1) has an optimal solution, the dual problem must as well, and the values of the primal and dual problems' optimal solutions must be the same. Let the optimal solutions to the primal and dual problems, respectively, be x^* and y^* . As a result, $s^t x^* = y^{*t} b$. On the other hand, the entire value of the sold and leftover resources is not always equal to the value of the objective function of the dual of the LP, or the primal for that matter. Rather, it is bigger, equaling $s^t x^* + (b - Ax^*)^t p$, where p is the resource price vector. Now rewrite the preceding expression as $(s - Ap)^t x^* + b^t p$, where $b^t p$ is constant and the objective function becomes $(s - Ap)^t x$ without the constant, where $s - Ap = r$ is the profit vector. If

x^{**} is the best solution to the objective function $r^t x$ with the same restrictions as below, then

$$\begin{aligned} r^t x &\rightarrow \text{maximize} && (4.11) \\ Ax &\leq b \\ x &\geq 0 \end{aligned}$$

then $r^t x^{**} \geq r^t x^*$ is true. In reality, in the LP above, the objective function is now profit maximization. It signifies that profit-maximizing production adds the highest value to input resources. As a result, maximizing revenues and profits are the same thing.

Chapter 5

Concluding Statements

5.1 Conclusion

In our research we have tried to come up with a more efficient approach to solving LP problems with the help of the Simplex algorithm. We have demonstrated that we can begin simplex iterations from a vertex that must be close to the optimal in some way, albeit this is not always possible. This is the basic solution for hyperplanes that make minimal angles with the objective function's gradient. The proposed initial basic solution provides an advantage over the traditional initial solution, according to our simulation results. On a side note, we demonstrated that profit and revenue maximizing objective functions can be inferred logically from one another. We have also demonstrated the robustness of LP theory in terms of duality results, even when nonnegativity constraints can be regarded of as part of primary inequalities, and the results are unaffected.

5.2 Future Work

So far, we have been able to prove that our proposed simplex algorithm can produce optimized results in fewer iterations than the general simplex method in cases of infeasibility. But our simulation has only been tested on a relatively smaller scale, with randomly generated test cases. We would like to test our algorithm further on benchmark test cases to assess its efficiency. We would also be working on making the algorithm more efficient in feasible LP problems as well. Moreover, the algorithm we devised has been tested with maximizing problems. But using the theory of duality, we can convert the minimizing problems to maximizing problems by transposing the matrix of the minimizing problem and solve the same way. For simplicity, we have tested the algorithm with LP problems having less than (\leq) inequalities only. We intend on handling greater than (\geq) inequalities by multiplying the equations with -1 , thus converting them to \leq inequalities. Furthermore, we can handle equal constraints ($=$) by splitting them into two constraints, one \leq constraint and one \geq constraint. Linear programming enthusiasts are also welcome to work on reducing the time complexity of our proposed algorithm.

Bibliography

- [1] D. Gale, H. W. Kuhn, and A. W. Tucker, “Linear programming and the theory of games,” *Activity analysis of production and allocation*, vol. 13, pp. 317–335, 1951.
- [2] L. MACCOLL, *DANTZIG, GB-LINEAR PROGRAMMING AND EXTENSIONS*, 1967.
- [3] V. Klee and G. J. Minty, “How good is the simplex algorithm,” *Inequalities*, vol. 3, no. 3, pp. 159–175, 1972.
- [4] N. Zadeh, “A bad network problem for the simplex method and other minimum cost flow algorithms,” *Mathematical Programming*, vol. 5, no. 1, pp. 255–266, 1973.
- [5] R. G. Bland, “New finite pivoting rules for the simplex method,” *Mathematics of operations Research*, vol. 2, no. 2, pp. 103–107, 1977.
- [6] L. G. Khachiyan, “A polynomial algorithm in linear programming,” in *Doklady Akademii Nauk*, Russian Academy of Sciences, vol. 244, 1979, pp. 1093–1096.
- [7] G. B. Dantzig, “Reminiscences about the origins of linear programming,” in *Mathematical Programming The State of the Art*, Springer, 1983, pp. 78–86.
- [8] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” in *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, 1984, pp. 302–311.
- [9] M. Murshed, B. Sarwar, M. Sattar, and M. Kaykobad, “A new polynomial algorithm for linear programming problem, www.gscit.monash.edu.au,” 1993.
- [10] H. V. Junior and M. P. E. Lins, “An improved initial basis for the simplex algorithm,” *Computers & Operations Research*, vol. 32, no. 8, pp. 1983–1993, 2005.
- [11] H. Arsham, “A big-m free solution algorithm for general linear programs,” *International Journal of Pure and Applied Mathematics*, vol. 32, no. 4, p. 549, 2006.
- [12] R. Cottle, E. Johnson, and R. Wets, “George b. dantzig (1914–2005),” *Notices of the AMS*, vol. 54, no. 3, pp. 344–362, 2007.
- [13] J.-F. Hu, “A note on “an improved initial basis for the simplex algorithm”,” *Computers & operations research*, vol. 34, no. 11, pp. 3397–3401, 2007.
- [14] S. Ru, M. Shen, and X. Xue, “A concise way of determination for uppercase LP initial feasible basis of simplex method,” *Scientia Magna*, vol. 4, no. 1, p. 142, 2008.

- [15] H. D. S. Mokhtar S. Bazaraa John J. Jarvis, *Linear Programming and Network Flows*. Wiley, 2009, ISBN: 9780470462720.
- [16] B. S. Anggoro, R. M. Rosida, A. M. Mentari, C. D. Novitasari, and I. Yulista, "Profit optimization using simplex methods on home industry bintang bakery in sukarama bandar lampung," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 1155, 2019, p. 012010.