International Journal of Mathematics and Computer Research

ISSN: 2320-7167

Volume 13 Issue 06 June 2025, Page no. – 5342-5345

Index Copernicus ICV: 57.55, Impact Factor: 8.615

DOI: 10.47191/ijmcr/v13i6.09

The Role of Multimedia for Solving Integer Linear Programming

Hardi Tambunan

Department of Mathematics Education, Faculty of Mathematics and Science, Universitas HKBP Nommensen, Medan

ARTICLE INFO

16 June 2025

Corresponding Author:
Hardi Tambunan

ABSTRACT

The purpose of this study is to discuss the solving of linear integer programming (ILP) with the simplex and Branch and Bound methods, and multimedia. So that solvers or students can get practical results with accurate results and shorter time, and as a means of checking the correctness of the answers manually to find solutions to ILP problems. Some multimedia software that can be used for solving ILP, including LINDO and LINGO. Judging from the accuracy of the results and the efficiency of ILP solving time, the two multimedia are better used compared to the simplex and Branch and Bound methods.

KEYWORDS: Multimedia, integer linear programming

I. INTRODUCTION

Linear programming (LP) is a science in the field of optimization that is used to solve real problems. LP is a mathematical model that has objective functions and linear constraints. Solving linear programming with the simplex method was developed by George Dantzig in 1947 [1]. The method is designed to solve LP problems through several iterations to obtain the optimum solution.

Integer linear programming (ILP) is an extension of LP with one additional limitation, namely the value of the variable must be integer. ILP is a mathematical model with integer variable values, and linearity in objective and constraint functions. The ILP model can be stated as follows.

$$\text{Maximum } f(x) = c^T x$$
$$\text{Constraint } Ax \le b$$
$$l \le x \le u$$
$$x_i \text{ integer, } j \in J' \subset J$$

where A is the matrix m × n, c is the vector n × 1, c^T is the transpose c, and J = {1,2, . . ., n}. where A is the matrix m × n, c is the vector n × 1, c^T is the transpose c, and J = {1,2, . . ., n}.

Some problems that might occur, such as the accuracy of the integer solutions obtained, and require a lot of time. As a result, students are less interested in solving ILP problems. The purpose of this study is to demonstrate the accuracy and time efficiency of solving ILP with multimedia software. So that solvers or students can use multimedia as a tool to obtain practical results, and as a means of checking the correctness of the answers manually to find solutions to ILP.

II. LITERATURE REVIEW

The Branch-and-Bound (B&B) method was developed based on the idea of the simplex method in a linear program by Land and Doing in 1960 [2], then modified by Dakin in 1965 [3]. This method is used for solving integer linear programs, specifically the search for integer variable values. The process of finding an integer solution from a continuous solution x_j, then for some $j \in J$ can be obtained $x_j = [x_j] + f_j$, $0 \le f_j \le 1$, where $[x_j]$ is the smallest integer that does not exceed x_j, and f_j is the smallest division component. The approach is carried out by forming two new subproblems by adding the proving limit $L_j \le x_j \le [x_j]$, and $[x_j] + 1 \le x_j \le u_j$, for certain variables $j \in J$. The branching process of a sequence of continuous problems is repeated for the integer variable $j \in J$, and for different integers $[x_j]$. The branch of the subproblem ends if the subproblem does not have a feasible solution, or the subproblem solution is no better than the best feasible integer solution before, or the solution is already integer feasible. Another approach to finding integer values can be done by integer processes [4-6]. In the process of solving ILP starting from the simplex method to the search for integers through B&B takes a lot of time. The more variables and constraints, the more iterations will be performed. As a result, solving ILP will be problematic for solvers or students. Therefore, in terms of results and process aspects, it is necessary to innovate in solving ILP by utilizing available facilities, such as multimedia.

Multimedia use is media related to the use of technology such as computers and software [7]. Multimedia can be interpreted as an interactive communication system [8]. The benefits of multimedia in data processing are time efficiency [7], improving student performance [9], and solving difficult problems [10].

III. METHODOLOGY

The ILP problem solving method used in this study is the use of multimedia, namely the application of simplex method algorithms into software, such as LINDO and LINGO packages. LINDO is a convenient, but powerful tool for solving integer linear programming problems [11]. LINGO is a comprehensive tool designed to make building and solving linear programming problems [12].

IV. RESULTS

One practical example to show ILP problem solving with the simplex method up to Branch and Bound, and multimedia with the LINDO and LINGO programs are discussed below. The ILP model can be stated as follows.

$$\text{Max } Z = 3X_1 + 6X_2 + 4X_3$$

$$\text{Constraints: } 3X_1 + 4X_2 + X_3 \leq 60$$

$$2X_1 + 3X_2 + X_3 \leq 50$$

$$X_1 + 2X_2 + 2X_3 \leq 44$$

$$X_1, X_2, X_3 \in \text{Integer}$$

$$X_1, X_2, X_3 \geq 0$$

Simplex Method

Solving ILP problems with the simplex method requires three iterations. Based on the results of the third iteration obtained the value of the objective function, max Z = 113.33 with X1 = 0, X2 = 12.67, and X3 = 9.33 [7].

Branch and Bound

Based on solving linear programs using the simplex method, the variable values X1 = 0, X2 = 12.67, X3 = 9.33 and Z=113.33 are obtained. So, to find a feasible solution for integers, it is solved by branching and bounding, as in the following steps.

Step 1: Create bounding for the Z value.

The initial linear programming (LP0) with optimal values from the simplex table has an upper bound (UB) = 113.33 (X1 = 0, X2 = 12.67, and X3 = 9.33), and a lower bound (LB) = 108 (X1 = 0, X2=12, and X3 = 9). So, the restrictions for Z become;

$$LP\,0$$

$$\boxed{108 \leq Z \leq 113.33}$$

Step 2: Create a branch.

Based on the variable values, X2 = 12.67 and X3 = 9.33 are not integers. To determine the branching, the largest comma number is selected, namely the variable value X2 = 12.67. So, a branch is made for X2 ≤ 12 and X2 ≥ 13.

Step 3: Create nodes.

The LP1 solution is obtained by limiting X2 ≤ 12 for the following constraint function.

Constraint 1: 3X1 + 4X2+ X3 ≤ 60.

If X1 = 0 and X2 = 12 for constraint 1 than X3 = 12, so the value of Z = 120. Because the variable value Z = 120 exceeds the upper bound of Z, the branch for the first sub-constraint is not feasible and the branch is stopped. If X2 = 12 and X3 = 9.33 for constraint 1 then X1 = 0.89, so Z = 111.99. Because the variable value Z = 111.99 does not exceed the upper bound of Z and is less than the bound of Z, the branch for the first sub-constraint is feasible, and can be branched to find the optimal result from the decision variable value in the form of an integer with new bound for Z

Constraint 2: 2X1+3X2+X3≤50.

If X1 = 0 and X2 = 12 for constraint 2 then X3 = 14, so Z = 128. Because the variable value Z = 128 exceeds the upper bound of Z, the branch for the second sub-constraint is not feasible and the branch is terminated. If X2 = 12 and X3 = 9.33 for constraint 2 then X1 = 2.335, so Z = 116.325. Because the variable value Z = 116.325 exceeds the upper limit of Z, the branch for the second sub-constraint is not feasible and the branch is stopped.

Constraint 3: X1 + 2X2+ 2X3 ≤ 44.

If X1 = 0 and X2 = 12 for constraint 3, then X3=10, so Z=112. Because the variable value Z = 112 does not exceed the upper bound of Z or less than the bound of Z and the value of the decision variable is an integer number, the branch for the third sub-constraint is feasible and can be stopped. If X2 = 12 and X3 = 9.33 for constraint 3, then X1 = 1.34, so Z = 113.34. Because the variable value Z = 113.34 exceeds the upper bound of Z, the branch for the third sub-constraint is not feasible and the branch is terminated.

The solution in LP2 is obtained by making the constraint X2 ≥ 13 from the constraint function as follows.

Constraint 1: 3X1 + 4X2+ X3 ≤ 60.

If X1 = 0 and X2 = 13 for constraint 1, then X3 = 8. So, Z= 110. Because the variable value Z = 110 does not exceed the upper bound of Z or less than the bound of Z and the value of the decision variable is an integer number, the branch for the first sub-constraint is feasible and can be stopped. If X2 = 13 and X3 = 9.33 for constraint 1, then X1 = -0.44, so the value of Z= 114. Because the variable value Z = 114 exceeds the upper bound of Z, the branch for the first sub-constraint is not feasible and the branch is terminated.

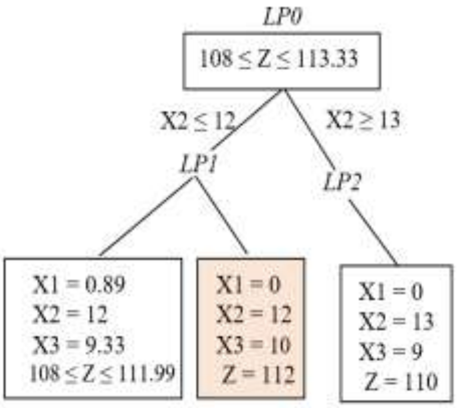
Constraint 2: 2X1 + 3X2+ X3 ≤ 50.

If X1 = 0 and X2 = 13 for constraint 2, then X3 = 11. So, the value of Z = 122. Because the variable value Z=122 exceeds the upper bound of Z, the branch for the second sub-constraint is not feasible and the branch is terminated. If X2 = 13 and X3 = 9.33 for constraint 2, then X1 = 0.835. So, the value of Z = 117.825. Because the variable value Z = 117.825 exceeds the upper bound of Z, the branch for the second sub-constraint is not feasible and the branch is terminated.

Constraint 3: X1 + 2X2+ 2X3 ≤ 44.

If X1 = 0 and X2 = 13 for constraint 3, then X3 = 9. So, the value of Z = 114. Because the variable value Z=114 exceeds the upper bound of Z, the branch for the third sub-constraint is not feasible and the branch is terminated. If X2 = 13 and X3 = 9.33 for constraint 3, then X1 = -0.66. So, the value of Z = 113.34. Because the variable value Z=113.34 exceeds the upper bound of Z, then the branch for the third sub-constraint is not feasible and the branch is terminated.

Based on these calculations, the LP1 and LP2 nodes in branching and bounding are as follows.



Carrying out the same iteration for 22 steps, and 7 times bounding and branching, a feasible integer solution was found, namely X1=10, X2=12, and X3=10 with a maximum value of Z=112.

LINDO

The ILP problem solving above is done in two stages, namely entering the ILP problem model in the Windows version of LINDO, and solving it. The results are shown in Figure 1 below.

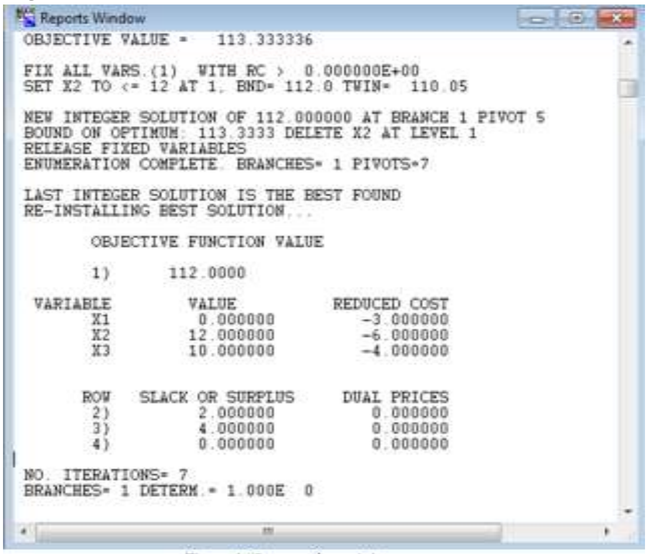


Figure 1: Output after solving

The results of solving with LINDO show that the maximum value for the objective function, max Z = 112, with X1 = 0, X2 = 12, and X3 = 10.

LINGO

Solving ILP models using LINGO software is done in two stages, namely entering the ILP problem model in the windows version of LINGO and solving. The results are shown are shown in Figure 2 below.

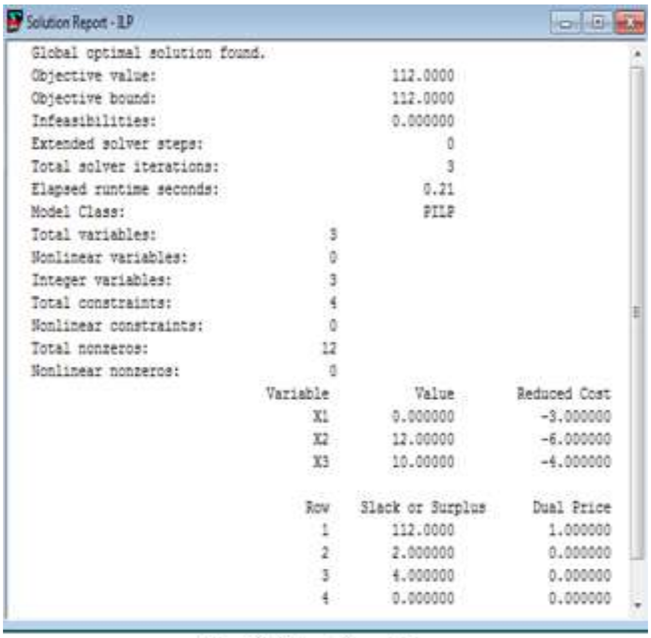


Figure 2: Output after solving

The results of solving ILP problems with LINGO show that the value of the objective function, max Z = 112, with X1 = 0, X2 = 12, and X3 = 10

V. CONCLUSIONS

Solving ILP requires accuracy and long iteration. Solving ILP starts with the simplex method until Branch and Bound requires a lot of time. So, it is less thorough and the time required will allow accuracy of the values obtained. Therefore, for accuracy of results and time efficiency, the use of multimedia is very necessary.

Some multimedia software that can be used for solving ILP, including LINDO and LINGO. Judging from the accuracy of the results and the efficiency of ILP solving time, the two multimedia are better used compared to the simplex and Branch and Bound methods.

REFERENCES

1. Danzing G.B and Thapa, M. N. Linear Programming. Spinger, 1997
2. Land, A. H and Doig, A. G. An Automatic Method of Solving Discrete Programming Problems. Econometrica, vol. 28, no. 3, pp. 497-520, 1960
3. Dakin, R.J. A Tree-Search Algorithm for Mixed Integer Programming Problems. Computer Journal, 8, pp. 250-255, 1965
4. Tambunan, H. Mathematical Model for Mapping Students Cognitive Capability. International Journal of Evaluation and Research in Education, vol.5, no.3, pp. 221–226, 2016

5. Tambunan, H and Mawengkang, H. Solving Mixed Integer Non-Linear Programming Using Active Constraint. Global Journal of Pure and Applied Mathematics, vol 13, no.7, pp. 2965-2973, 2016
6. Tambunan, H and Mawengkang, H. Integer Linear Programming Approach for Detection Learning Outcomes Achievement. Far East Journal of Mathematical and Sciences, vol.5, no.1, pp. 95-109, 2018
7. Tambunan, H. Designing Multimedia Learning for Solving Linear Programming. Global Journal of Pure and Applied Mathematics, vol. 12, no. 6, pp. 5265-5281, 2017
8. Reisman, S. Multimedia Computing: Preparing for the 21st Century. Harrisburg: IDEA Group Publishing, 1994
9. Ayub, A.F.M, Tarmizi, R.A, Bakar, K.A and Luan, W.S. Adoption of Wxmaxima Software in the Classroom: Effect on Students’ Motivation and Learning of Mathematics. Malaysia Journal of Mathematical Sciences, vol. 8, no.2, pp. 311-323, 2014
10. Panjaitan, B. Designing a Learning Media using Matlab for Matrix and Its Operations. Global Journal of Pure and Applied Mathematics, vol. 12, no. 5, pp. 4431-4441. 2016
11. LINDO. LINDO User’s Manual. Chicago: LINDO Systems, Inc, 2003
12. LINGO. The Modeling Language and Optimizer. Chicago, Illinois: LINDO Systems Inc, 2016