Jurnal Matematika Vol. 14, No.2, Desember 2024, pp. 117-124 ISSN: 1693-1394
Article DOI: 10.24843/JMAT.2024.v14.i02.p176

A Depth-First Branch and Bound Algorithm for Integer Knapsack Problem in Freight Transportation

Sasha Fahrunisha
Jl. Majapahit No.62. Gomong. Mataram, NTB
e-mail: sashafhrn@gmail.com

Lailia Awalushaumi
Jl. Majapahit No.62. Gomong. Mataram, NTB
e-mail: *awalushaumi@unram.ac.id*

Zata Yumni Awanis
Jl. Majapahit No.62. Gomong. Mataram, NTB
e-mail: *zata.yumni@unram.ac.id*

Muhammad Rijal Alfian
Jl. Majapahit No.62. Gomong. Mataram, NTB
e-mail: *rijal_alfian@unram.ac.id*

Abstract: *E-commerce growth so rapidly in Indonesia and giving great opportunity for logistics service in order to give easiness for customer to make deliveries. Pos Indonesia (Persero) is an Indonesian state-owned companies engaged in delivery of goods and letter. In sending a goods, its important to choose the right goods that will be lifted into the freight transportation. The more optimal load of goods, the more income will be gotten. To find an optimal selection of solution, it can be modeled as Integer Knapsack problem by optimizing the maximum capacity of knapsack with constraint the weight of goods. The purpose of the research is to get an optimal solution by selecting a right goods based from weight using Depth-First Branch and Bound in freight transportation of Pos Office Mataram. The result provided quite significant because it use varied data within one day of delivery. Total weight from 23-31 august 2022 is 11,563.3 kg and get Rp. 16,259,430.00 income can be delivered in 11 days using Depth-First Branch and Bound better than an actual shipment that need 12 days to distributed all of goods. So we can see that the Depth-First Branch an Bound can be an alternative solution to avoid overload and minimize an operating cost. Therefore, the accumulation of weight using Depth-First Branch and Bound more effective to get an optimal selection of goods in freight transportation.*

Keywords: *Depth-First Branch and Bound, Freight Transportation, Integer Knapsack Problem.*

1. Introduction

We all know that mathematics is certainly a science for modeling many phenomena which then allows prediction to solve a problem such as optimizing load of goods in freight transportation. Should be noted before that the growth of logistic business has increased every year especially during the pandemic covid-19. Therefore, a digital activity also increased in community that's the reason emergence of competition between logistic businesses. One of strategy to face a logistic business competition by optimizing load of goods in the distribution, so that the more optimal load of goods the more income will be gotten.

The logistic company that use as a sample for optimizing load of goods in this research is freight transportation by Pos Mataram that the office located in Jl. Sriwijaya, Punia, Kec. Mataram, Mataram City. Previously, there are several complaints by customers in the company cause they have'nt received their shipment. It happened because the distribution spend a long time of delivery. Thus, we hope for this research can provide an information and could be an alternative solution for optimize the load of freight transportation.

In this research, the volume of goods is ignored because the selection of freight transportation only contain a small dimension but varied weight of goods. This case only focused on the weight of each item that will be loaded into the freight transportation, so it can be modeled as Knapsack problem which is an optimization problem where someone should be choose some item or object to be placed in knapsack container, provided in the right way the item can be load optimally and then the company will get an income as much as possible (Irmeilyana dkk, 2017). If the goods loading not optimal, it can spend more cost and time certainly. And also if the goods over the knapsack (overload), the risk of unwanted things such as damage or even accidents will only give a big loss for the company.

Model for representing an Integer Knapsack problem is Integer Programming problem which is another form of linear programming problem where the asumption of divisibility is weakned or completely lost [3]. There are many ways to solve an Integer Knapsack problem, such as Genetic algorithm, Dynamic programming, etc. But in this research the one way choosen to solve the Integer Knapsack problem is using a Depth-First Branch and Bound (DFBnB). A Depth-First Branch and Bound algorithm was choosen on this research because it very well to solve some problem with the decision variable are integers, where Integer Knapsack problem certainly must give an integers result also. Branch and Bound is general algorithm for finding an optimal solution to various optimization problems, especially for discrete and combinatorial optimization [7]. A Depth First Search is a searching tree structure of Branch and Bound that used in this research, the ilustrated shown on Figure 1 below.

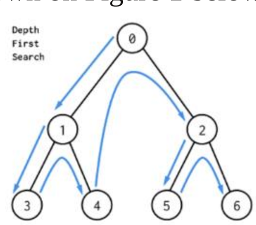

Figure 1. Depth First Search Structure

Jurnal Matematika Vol. 14, No.2, Desember 2024, pp. 117-124 ISSN: 1693-1394
Article DOI: 10.24843/JMAT.2024.v14.i02.p176

Branch and Bound algorithm has three basic steps for each iteration like branching, bounding and pruning. In branching, we deviding a set of feasible solution into some of subproblem. From the some of problem that still exist, select the last generate subproblem. Then stop the iteration until a greater limit has found. After that, branch the node to get two subproblem by setting the branch variable to 0 or 1. A Depth-First Branch and Bound works to explore the state space in depth first and then finds a lot of suboptimal solution with slowly improving quality where the solution will be an approximation to the optimal solution if DFBnB stop prematurely.

2. Method

The iteration steps of Depth-First Bracnh and Bound algorithm are:

i. Branching (Dividing the problem into some subproblem)
ii. Bounding (Find the estimated optimization value with the best solution in the subproblem)
   - Maximization (Upper Bound)
   - Minimization (Lower Bound)

   In obtaining optimization estimates we must using a Linear Relaxation to initialization problem.

1. The first step is modeling Integer Knapsack problem as an Integer Programming problem.

$$\max \sum_{j=1}^{n} b_j X_j = b_1 X_1 + b_2 X_2 + w_3 X_3 + \ \dots + b_{n-1} X_{n-1} + b_n X_n$$

$$\text{s.t.} \quad \sum_{j=1}^{n} w_j X_j \leq V = w_1 X_1 + w_2 X_2 + w_3 X_3 + \ \dots + w_{n-1} X_{n-1} + w_n X_n$$

$$X_j \in \{0,1\}, \qquad j = 1, 2, \dots, n. \tag{1}$$

   where b_j is shipping cost, w_j weight of item, X_j decision value (binary), and V is maximum capacity of knapsack.

2. Sort all items then reorder of items according to their weight to get near the maximum as soon as possible.
3. Select items one by one. Note that the last item may be partially selected. To determined the approximate value of X_j, we can use the formula below.

$$s = min\left\{k: \sum_{j=1}^{k} w_j > V\right\}. \tag{2}$$

$$\bar{X}_j = 1, where\ j = 1, \dots s-1$$

$$\bar{X}_j = 0, where\ j = s+1, \dots n. \tag{3}$$

$$\bar{X}_s = \frac{\bar{V}}{w_s}. \tag{4}$$

$$\bar{V} = V - \sum_{1}^{s-1} w_j\,. \tag{5}$$

Then the approximate optimal solution :

$$\bar{z} = \sum_{j=1}^{s-1} b_j + \left\lfloor \bar{V} \frac{b_s}{w_s} \right\rfloor \quad (6)$$

4. If we get a fractional value of X_j, we able be approached it into 0 or 1 and then branch with two constraint $X_j \leq 0$ or $X_j \geq 1$.
5. Repeat step 4 if the decision variable still have a fractional value.
6. Do this until all of the decision variables are binary numbers. Therefore, we get an optimal solution of Knapsack problem.
7. A subproblem removed from further consideration if it fulfills the following requirements below:
   - Bound $\leq z^*$
   - LP relaxation doesn't have a feasible solution
   - The optimal solution of LP relaxation is an integer

The following flowchart below gives us steps the how research method process.

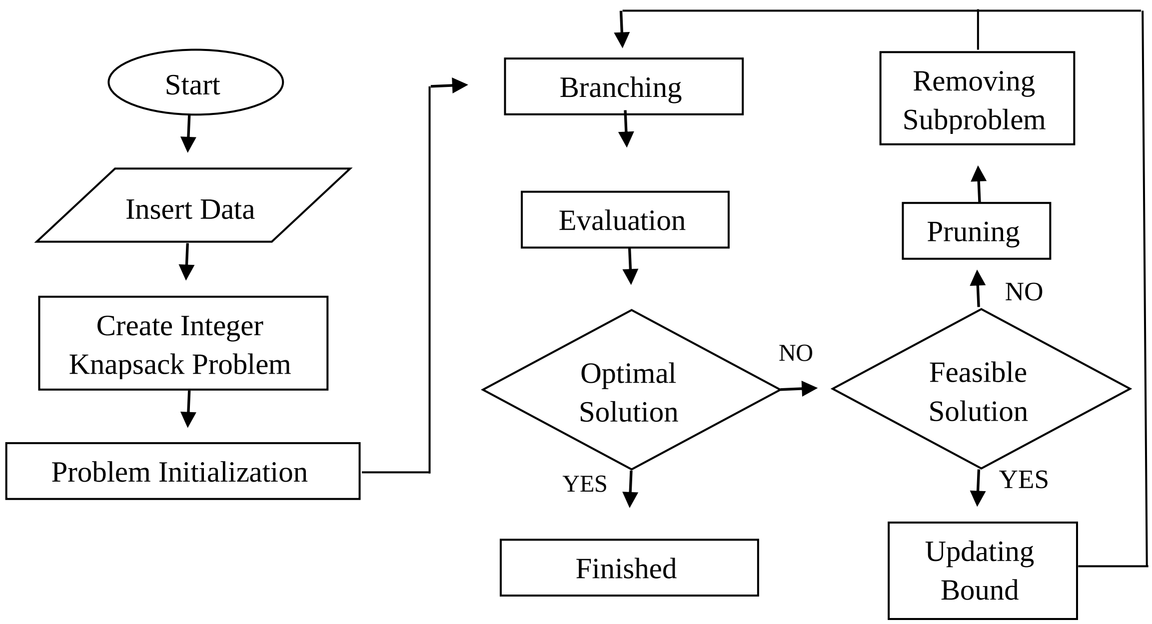

Figure 2. Flowchart Depth-First Branch and Bound

3. Result and Discussion

Example 3.1 The following example below show how the algorithm works. Assume X_n is an item that will load to the knapsack and we have the maximum capacity of knapsack is 10 kg. So which item will be load to the knapsack? We just have two option, if an item give a value of 1 it mean that the item will be load, but if an item give a value of 0 that was not selected to be load.

$$\begin{aligned} &\max \quad 18X_1 + 9X_2 + 9X_3 + 3X_4 \\ &\text{s.t.} \quad 5X_1 + 3X_2 + 3X_3 + X_4 \leq 10\ (kg) \\ &X_j \in \{0,1\}, \qquad j = 1,2,3,4. \end{aligned} \tag{7}$$

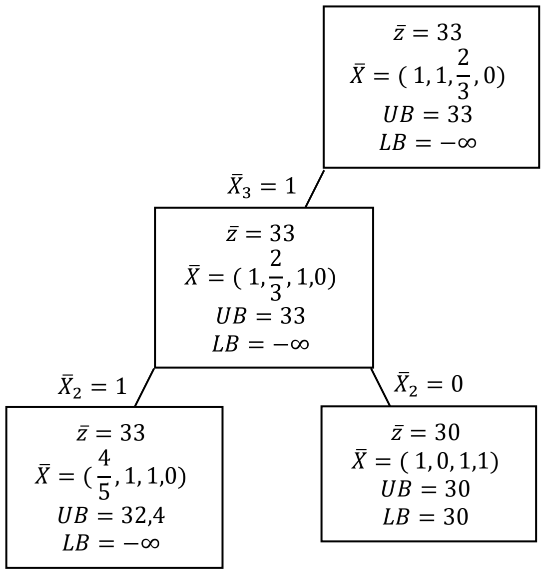

Figure 3. An Optimal Solution

The answer is $\bar{X} = (1, 0, 1, 1)$ means that the first, third, and fourth item will be load.

Table 1. The calculation Result Using Depth-First Branch and Bound Algorithm

Date	Total Weigth	Total Shipping cost	Amount of Data
23/08/22	1114.21 kg	Rp 1,552,595	15
24/08/22	1199.16 kg	Rp 1,684,760	10
25/08/22	1199.97 kg	Rp 1,701,535	7
26/08/22	1185.62 kg	Rp 1,835,460	9
27/08/22	1199 kg	Rp 1,697,080	9
28/08/22	1199.93 kg	Rp 1,665,840	6
29/08/22	81.59 kg	Rp 110,275	3
30/08/22	1199.95 kg	Rp 1,677,390	4
31/09/22	1137.09 kg	Rp 1,591,975	13
01/09/22	970.99 kg	Rp 1,347,940	12
02/09/22	1075.79 kg	Rp 1,540,000	1

Here we have the result of data calculation using the program that has been compiled according to the Depth-First Branch and Bound algorithms shown in Table 1. The structured of algorithm based on steps in Figure 2.

In this research, we use a report of goods shipment from Mataram to Denpasar, Banyuwangi, Jember, and Surabaya. The freight transportation have maximum capacity 1200 kg. Data obtained from Pos Office Mataram in a week of August 2022. Table 2. shown a data collecting in a week of August.

From Table 2. above, there are several shipment that have overload. From this case, we can apply a Knapsack problem to find a best solution how to optimizing the load of goods on the limited knapsack using Depth-First Branch and Bound. In this research, we are using data record from Pos Office Mataram through 9 days.

Table 2 Data record of goods in 24 – 31 of August 2022

Date	Total Weigth	Total Shipping cost	Amount of Data
23/08/22	1114.21 kg	Rp 1,552,595.00	15
24/08/22	1504.26 kg	Rp 2,111,340.00	12
25/08/22	1013.96 kg	Rp 1,435,555.00	12
26/08/22	1220.01 kg	Rp 1,724,690.00	10
27/08/22	1524.66 kg	Rp 2,119,040.00	11
28/08/22	792.43 kg	Rp 1,128,710.00	11
29/08/22	9.95 kg	Rp 15,015.00	3
30/08/22	2799.34 kg	Rp 3,945,205.00	18
31/08/22	1584.48 kg	Rp 2,227,280.00	12

After we apply a Knapsack problem to the case, the comparison result between actual shipment with Depth-First Branch and Bound, can be seen at Table 3. below.

Table 3. Different Result of Shipment

Date	Shipment	Total Weight	Total Shipping Cost
23/08/22	Actual	1114.21 kg	Rp 1,552,595
	Depth-First Branch and Bound	1114.21 kg	Rp 1,552,595
24/08/22	Actual	1151.67 kg	Rp 1,250,755
	Depth-First Branch and Bound	1199.16 kg	Rp 1,684,760
25/08/22	Actual	1198.47 kg	Rp 1,674,805
	Depth-First Branch and Bound	1199.97 kg	Rp 1,701,535

26/08/22	Actual	1192.95 kg	Rp 1,699,335
	Depth-First Branch and Bound	1185.62 kg	Rp 1,835,460
27/08/22	Actual	1191.26 kg	Rp 1,680,800
	Depth-First Branch and Bound	1199 kg	Rp 1,697,080
28/08/22	Actual	749.81 kg	Rp 1,010,460
	Depth-First Branch and Bound	1199.93 kg	Rp 1,665,840
29/08/22	Actual	581.11 kg	Rp 831,545
	Depth-First Branch and Bound	81.59 kg	Rp 110,275
30/08/22	Actual	1100.47 kg	Rp 1,555,345
	Depth-First Branch and Bound	1199.95 kg	Rp 1,677,390
31/08/22	Actual	789.73 kg	Rp 1,089,990
	Depth-First Branch and Bound	1137.09 kg	Rp 1591975
01/09/22	Actual	1199.34 kg	Rp 1,685,805
	Depth-First Branch and Bound	970.99 kg	Rp 1,347,940
02/09/22	Actual	1192.13 kg	Rp 1,704,120
	Depth-First Branch and Bound	1075.79 kg	Rp 1,540,000
03/09/22	Actual	101.89	Rp 137,225
	Depth-First Branch and Bound	-	-

From Table 3. above, the result using Depth-First Branch and Bound better than the actual shipment. All of goods during 9 days can be send only 11 days. As we know that the actual shipment need 12 days to distributed all of goods on a freight transportation. Therefore, we can minimize the operating cost and get more income. This calculation can be an information for company and alternative solution to optimizing load of goods in a freight transportation.

4. Conclusion

The solution using Depth-First Branch and Bound give an optimal load of goods. Total shipment on 9 days can be sending only in 11 days, better than the actual that need 12 days to distributed all of goods. From this, we can take a coclusion that the result of Depth-First Branch an Bound to solve a Knapsack problem can be an alternative solution to load goods when the all of goods have overload. Therefore, a company can get more income because the calculation give more effective solution to minimize an operating cost.

References

Coniglio, Stefano, Fabio Furini dan Pablo San Segundo. (2021). A New Combinatorial Branch and Bound Algorithm for Knapsack Problem with Conflict. European Journal of Operational Research. Amsterdam: Elsevier.

Husein, Abdullah, Dewi Murni, dan Meira Parma Dewi. (2021). Algoritma Genetika Pada Optimasi Persoalan Knapsack 0/1. UNPjoMath Vol. 4 No. 2. Padang: Universitas Negeri Padang.

Rois, Muhammad Abdurrahman, Siti Maslihah, dan Budi Cahyono. (2019). Penyelesaian Integer Knapsack Problem Menggunakan Algoritma Greedy, Dynamic Programming, Brute Force dan Genetic. Telematika Vol. 12 No. 2. Semarang: Universitas Brawijaya.

Pratiwi, Mulyono and Rochmad. (2014). Implementasi Algoritma Branch and Bound pada 0-1 Knapsack Problem untuk Mengoptimalkan Muatan Barang. *Journal of Mathematics*. UNNES: Semarang.

Malone and C. Yuan. (1992) A Depth-First Branch and Bound for Learning Optimal Bayesian Network. Springer International Publising: Switzerland.

Dimyati dan Tjutju Tarliah. (1992). Operations research : Model-model Pengambilan Keputusan. Sinar Baru Algensindo.

Sampurno, E. Sugiharti and Alamsyah. (2018). Comparison of Dynamic Programming Algorithm and Greedy Algorithm on Integer Knapsack Problem in Freight Transportation. *Scientifict Journal of Informatics*: Semarang.

H. Kellerer, U. Pferschy and D. Pisinger. (2004). Knapsack Prooblem. Springer: New York.

Irmeilyana, P. B. J. Bangun and H. Izzah. (2017). Solution of Multiple Constraints KnapsackProblem (MCKP) by Using Branch and Bound Method and Greedy Algorithm. *Journal of Modeling and Optimization*: Indralaya.

Suyanto. (2010). Algoritma Optimasi (Deterministik atau Probabilistik). Graha Ilmu: Yogyakarta.