JOURNAL OF DATA SCIENCE
eISSN:2805-5160
Vol.2022:21

Load Optimization with Shortest Distance Approach

Yusria Lenitasari[1], Tri Basuki Kurniawan[1*], Edi Surya Negara[1], Tata Sutrabri[2], Misinem[3]

[1]Magister of Information Technology, University of Bina Darma, Palembang, Indonesia
[2]Faculty of Computer Science, University of Bina Darma, Palembang, Indonesia
[3]Faculty of Vocational, University of Bina Darma, Palembang, Indonesia

[*]**Email**: tribasukikurniawan@binadarma.ac.id

Abstract

The most beneficial results or values are produced through an optimization technique. Load optimization is a problem that the logistics sector faces, despite the fact that there are many other optimization-related concerns. This problem has a connection to the knapsack problem, which is the combination of the number of items that can fit into a container with a capacity when one set of items has both weight and volume. The following problem is referred to as "Bin packing," which is an optimization problem in which objects of various sizes must be packed into a finite number of bins or containers, each of which has a specific capacity, while utilizing the fewest number of bins. By merging these two issues, the best payload value will be produced. In order to optimize the volume and weight of product preparation and arrangement based on delivery destinations (in terms of distance) on previously operational vehicles, a program will simulate the combination of these concerns. The initial load item carried by the original driver was compared in this study using experimental data to the outcomes of the load optimization approach. Results after the shortest distance approach's improvement were compared to the results obtained before. According to the comparison, the shortest-distance approach led to better outcomes.

Keywords

Load Optimization, Shortest Distance, Bin-Packing, Knapsack Problem, Artificial Intelligence

Introduction

The Covid 19 epidemic that swept the globe has sparked widespread participation in the use of information technology. Companies in the logistics industry that offer shipping and delivery services for packages have also been impacted by the explosion of online commerce. Logistics businesses may use this momentum to increase sales by maximizing the number of shipments or parcels delivered in a single truck.

Technology may be used to boost the effectiveness and efficiency of satisfying demands in the case of logistics business players when goods transport vehicles have restricted numbers and

Submission: 15 December 2022
Acceptance: 24 December 2022

http://ipublishing.intimal.edu.my/jods.html

types of vehicles to meet needs. Similar adjustments must be made for vehicles transported by vehicles based on their capacity, shape, and size. One of the efficiency indications that businesses may use is the arrangement of products, even down to the order in which they are grouped by address, to maximize the volume and capacity of items in truck containers.

Considerable attention must go into optimizing the volume, capacity, and layout of the contents within the vehicle container to reduce the number of operating vehicles, the number of staff, the delivery time, and other expenditures. The current issue is a result of the conventional arrangement of item items, where officers must estimate without taking into account or use trial and error in terms of placement and arrangement of item items, provided that the total amount of each item is put into the whole container weighs less than or equal to the container's capacity limit and without taking into account the address of the item to be delivered. The knapsack problem is the name given to this issue.

The "knapsack problem," or the difficulty of filling truck containers, was explored by several prior researchers, most notably (Gazali, Ngarap, & Manik, 2010), who discovered that the Greedy Algorithm might be used to find the best solutions to the three-dimensional container loading problem. The arrangement of commodities in accessible containers or containers can be optimized using a genetic algorithm to solve the knapsack problem, according to Supriana (2016) research. Sampurno, Sugiharti, & Alamsyah (2018) compared the Dynamic Programming Algorithm with the Greedy Algorithm in their study on algorithm comparison. Dynamic Programming Algorithm outperformed Greedy Algorithm in his research's outcomes.

By contrasting four algorithms; greedy, dynamic programming, brute force, and genetics; Abdurrahman Rois, Maslihah, & Cahyono (2019) concluded that dynamic programming is the most effective and efficient approach for application on both small and big-size data. Then, in their research on knapsack issue optimization strategies, Devita & Wibawa (2020) compared 5 (five) knapsack problem algorithms, including the greedy algorithm, dynamic programming, branch and bound brute force, and genetics. According to his research, the dynamic programming method offers the best solution and typical processing time.

Wungguli, Ibrahim, & Yahya (2021) also compared the greedy and branch and bound algorithms for the knapsack problem and generated research. Moesya, Cahyaningrum, & Khairunnisa (2019), who reached the dynamic programming and branch and bound algorithms, found that the branch and the bound algorithm was superior to the dynamic programming algorithm in terms of strategy complexity but inferior to the branch and bound algorithm in terms of execution time.

Methodology

In this paper, we conducted several steps before the experiment, such as collecting and pre-processing data. The methodology in this research is as follows in Figure 1.

Submission: 15 December 2022
Acceptance: 24 December 2022

JOURNAL OF DATA SCIENCE
eISSN:2805-5160
Vol.2022:21

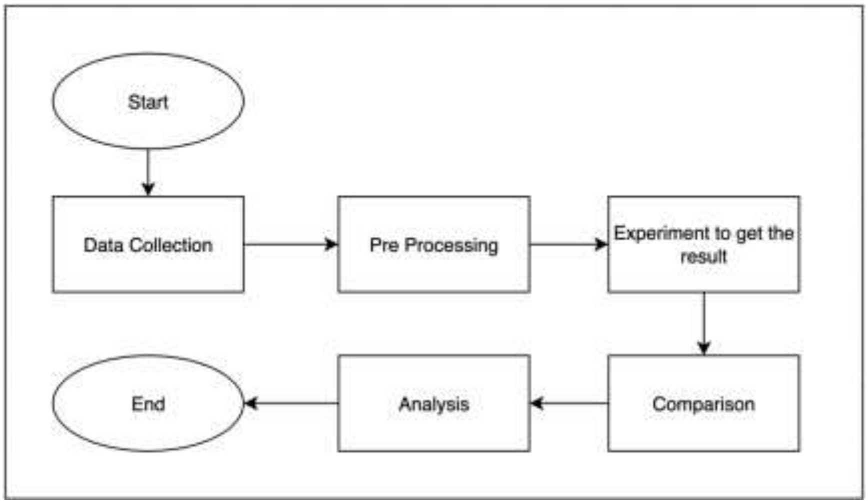

Figure 1. The Methodology

Figure 1 shows the processes that start with data collection. In this process, real data from the transportation company was collected. The process continues with pre-processing the dataset to clean and remove not completed data or change the wrong data type during process data collecting. The following procedure is done for the experiment, shown in the sub-discussion. After getting the result, the comparison process and the analysis were conducted.

Collecting Data

The actual data from the industry has already been gathered and prepared, which includes a week's worth of transactional data for the customer's item that should be delivered, as shown in Table 1.

Table 1. Experiment data

Day #	Date	No of drivers	Total of Items	Total Weight (kg)	Total Volume (cm^3)
1	2022-01-24	7	154	9,193.83	70,251,481.25
2	2022-01-25	5	159	11,221.90	85,927,503.16
3	2022-01-26	5	143	7,368.01	61,719,722.28
4	2022-01-27	9	221	12,798.80	85,948,824.65
5	2022-01-28	8	192	9,166.57	65,154,072.69
6	2022-01-29	5	111	3,287.27	15,931,953.08
7	2022-01-30	6	93	2,560.73	16,924,824.39
	Total	45	1073	55,597.10	401,858,386.49

Table 1 shows each day consists of several numbers of drivers carrying an item with different numbers of parcels, weights, and volumes. Each driver will bring that item in their lorry, which different type of vehicle depends on their capacity to carry all items belonging to that driver.

As a reference, Table 2 shows the vehicle type used by that transportation company.

Submission: 15 December 2022
Acceptance: 24 December 2022

Table 2. Vehicle type

No	Code	Description	Length (inches)	Height (inches)	Width (inches)	Max Weight (kg)	Max Volume (cm^3)
1	4x4	4x4 Pickup	4	3	3.5	500	1,189,307.56
2	VAN	Van	8	3	3.5	500	2,378,615.11
3	LORRY-S	1-tonne lorry	10	5	5	1,000	7,079,211.65
4	LORRY-M	3-tonne lorry	14	7.2	7	3,000	19,980,366.96
5	LORRY-L	5-tonne lorry	17	7.2	7	5,000	24,261,874.16

Table 2 shows the vehicle type used by the company. The length, height, and width of the lorry are in inches. First, we convert it into cm, and then we calculate the volume of the lorry—the weight of the lorry in kg.

Experiment Setup

The experiment began by deciding the type of vehicle for each driver based on the weight and volume of items belonging to the driver. Based on that, we can try to decrease the number of the lorry and their capacity to reduce the cost of items delivered for the company for each day dataset.

The first comparison result is the number of the lorry needed by an original driver with the result proposed by load optimization only based on their weights dan volumes. In this experiment, we want to know how much the number of lorries will be reduced.

The second comparison was made with the result from load optimization with the shortest distance constraint applied. In this experiment, we considered not only the number of items with proper weight and volume that should be put in a specific container. We also consider the distance between one parcel to another in one container / the lorry, as shown in Figure 2.

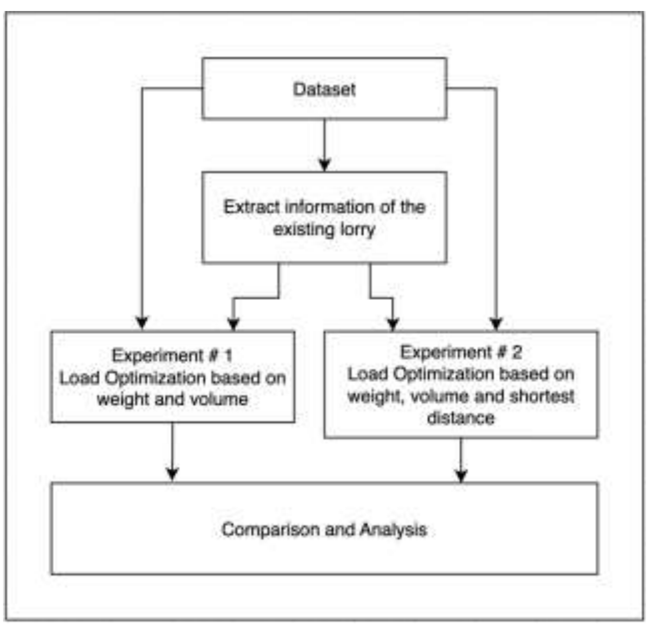

Figure 2. Experiment setup

Figure 2 shows the experiment was conducted, and the comparison result was reported. The conclusion will be based on two (2) experiments and comparison results. The investigation was conducted using Python at The Jupiter notebook on workstations with Intel Xeon Processor 3.3 GHz and 8 GB RAM.

Google OR Tools

OR-Tools is an open-source optimization software package designed to tackle the world's most challenging issues in vehicle routing, flows, integer and linear programming, and constraint programming (Google, 2022).

After modeling the issue in your preferred programming language, you may solve it with one of a half-dozen solvers: commercial solvers like Gurobi or CPLEX or open-source solvers like SCIP, GLPK, or Google's GLOP and award-winning CP-SAT.

The initial version was uploaded on GitHub in September 2015, and version 7.2 was released in July 2019. While the Google AI-based solution was developed in C++, the suite may also be utilized with Python, Java, and C#. Linux, Mac, and Windows versions are all accessible for download and installation.

K-means Clustering

The clustering algorithm in this project is proposed to keep the cluster member items close to each other before the optimization route is proposed. One of most algorithms is K-means (Du, Huang, & Qiu, 2014). The goal of K-means clustering, a vector quantization technique that originated in signal processing is to divide n observations into *k* clusters, where each observation belongs to the cluster that has the closest mean (also known as the cluster centroid or cluster center), which serves as a prototype for the cluster. As a result, the data space is divided into Voronoi cells. The geometric median is the only one that minimizes Euclidean distances; K-means clustering reduces within-cluster variances (squared Euclidean distances) but not regular Euclidean distances, which would be the more challenging Weber issue. For instance, K-medians and K-medoids can be used to find better Euclidean solutions.

Although the issue is computationally challenging (NP-hard) (Li et al., 2022), effective heuristic methods quickly reach a local optimum. These often follow an iterative refining strategy used by both K-means and Gaussian mixture modeling, comparable to the expectation-maximization procedure for mixtures of Gaussian distributions. They both employ cluster centers to represent the data, but the Gaussian mixture model allows for different-shaped clusters. In contrast, K-means clustering tries to discover clusters of equivalent spatial dimensions.

Submission: 15 December 2022
Acceptance: 24 December 2022

Jupyter Notebook

A server-client program called the Jupyter Notebook App enables editing and executing notebook papers from a web browser. The Jupyter Notebook App may be used locally on a computer without an internet connection or installed on a remote server and viewed online (Notebook, 2017).

The Jupyter Notebook App contains a "Dashboard" (Notebook Dashboard), a "control panel" exposing local files and letting to open notebook papers or shutting down their kernels, in addition to displaying, editing, and executing notebook documents.

The "computational engine" that runs the code in a Notebook document is called a notebook kernel. Python code is executed by the ipython kernel, which is mentioned in this manual. There are kernels for several more languages.

A Notebook document's related kernel is immediately launched when you open it. The kernel does the computation and generates the results when the notebook is run (either cell-by-cell or through the menu Cell -> Run All). Depending on the sort of computations, the kernel may use a lot of CPU and RAM. Keep in mind that RAM is not freed until the kernel has terminated.

Results and Discussion

The first step is to extract information about the possible vehicle used by each driver in one day. The total weight and volume for all items belonging to that driver are calculated, and then we compare and decide which lorry can carry all items by refers the vehicle type from Table 2. The result of the used vehicles every day is shown in Table 3.

Table 3. The list of used vehicles

Day #	Date	No of drivers	Total of Items	Total Weight (kg)	Total Volume (cm^3)		
1	2022-01-24	7	154	9,193.83	70,251,481.25		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	13	107.18	622,619.93	4x4	500 - 1,189,307.56
		2	18	916.60	7,099,072.00	LORRY-M	3,000 - 19,980,366.96
		3	12	221.58	5,004,940.00	LORRY-S	1,000 – 7,079,211.65
		4	53	6,930.00	46,052,640.00	LORRY-L	5,000 - 24,261,874.16
						LORRY-L	5,000 - 24,261,874.16
		5	19	343.74	2,758,888.00	LORRY-S	1,000 - 7,079,211.65
		6	25	424.00	2,454,084.33	LORRY-S	1,000 - 7,079,211.65
		7	14	251.73	6,259,237.00	LORRY-S	1,000 - 7,079,211.65
		Total	154	9,193.83	70,251,481.25	4x4 (1); LORRY-S (4); LORRY-M (1); LORRY-L (2) = 8 vehicles	
2	2022-01-25	5	159	11,221.90	85,927,503.16		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	33	698.13	15,057,891.27	LORRY-M	3,000 - 19,980,366.96
		2	62	8,720.00	57,088,000.00	LORRY-L	5,000 - 24,261,874.16
						LORRY-L	5,000 - 24,261,874.16

Submission: 15 December 2022
Acceptance: 24 December 2022

						LORRY-M	3,000 - 19,980,366.96
		3	37	1,469.08	7,860,608.90	LORRY-M	3,000 - 19,980,366.96
		4	20	278.50	5,465,163.99	LORRY-S	1,000 - 7,079,211.65
		5	7	56.19	455,839.00	4x4	500 - 1,189,307.56
		Total	159	11,221.90	85,927,503.16		4x4 (1); LORRY-S (1); LORRY-M (3); LORRY-L (2) = 7 vehicles
3	2022-01-26	5	143	7,368.01	61,719,722.28		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	45	1,034.71	16,888,194.57	LORRY-M	3,000 - 19,980,366.96
		2	8	67.80	883,620.80	4x4	500 - 1,189,307.56
		3	18	2,650.00	17,527,480.00	LORRY-M	3,000 - 19,980,366.96
		4	29	1,611.90	11,962,734.00	LORRY-M	3,000 - 19,980,366.96
		5	43	2,003.60	14,457,692.00	LORRY-M	3,000 - 19,980,366.96
		Total	143	7,368.01	61,719,722.28		4x4 (1); LORRY-M (4) = 5 vehicles
4	2022-01-27	9	221	12,798.80	85,948,824.65		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	24	1,255.60	8,620,152.00	LORRY-M	3,000 - 19,980,366.96
		2	18	216.85	6,344,213.00	LORRY-S	1,000 - 7,079,211.65
		3	22	299..31	1,892,248.59	VAN	500 - 2,378,615.11
		4	24	387.82	2,741,124.00	LORRY-S	1,000 - 7,079,211.65
		5	63	8,690.00	57,514,600.00	LORRY-L	5,000 - 24,261,874.16
						LORRY-L	5,000 - 24,261,874.16
						LORRY-M	3,000 - 19,980,366.96
		6	18	222.14	790,663.59	4x4	500 - 1,189,307.56
		7	19	333,08	3,158.891.30	LORRY-S	1,000 - 7,079,211.65
		8	14	113,20	747,050.24	4x4	500 - 1,189,307.56
		9	19	1,280.00	4,219,882.11	LORRY-M	3,000 - 19,980,366.96
		Total	221	12,798.80	85,948,824.65		4x4 (2); VAN (1); LORRY-S (3); LORRY-M (3); LORRY-L (2) = 11 vehicles
5	2022-01-28	8	192	9,166.57	65,154,072.69		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	28	358.97	3,456,100.50	LORRY-S	1,000 - 7,079,211.65
		2	41	872.19	7,972,876.86	LORRY-M	3,000 - 19,980,366.96
		3	12	595.55	6,366,174.00	LORRY-S	1,000 - 7,079,211.65
		4	34	5,060.00	34,248,000.00	LORRY-L	5,000 - 24,261,874.16
						LORRY-M	3,000 - 19,980,366.96
		5	12	556.39	2,069,162.22	LORRY-S	1,000 - 7,079,211.65
		6	12	144.80	1,249,316.00	VAN	500 - 2,378,615.11
		7	29	1,274.10	8,539,830.00	LORRY-M	3,000 - 19,980,366.96
		8	27	304.56	1,252,613.00	VAN	500 - 2,378,615.11
		Total	192	9,166.57	65,154,072.69		VAN (2); LORRY-S (3); LORRY-M (3); LORRY-L (1) = 9 vehicles
6	2022-01-29	5	111	3,287.27	15,931,953.08		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	38	426.28	2,528,633.00	LORRY-S	1,000 - 7,079,211.65
		2	12	663.80	2,417,306.20	LORRY-S	1,000 - 7,079,211.65

Submission: 15 December 2022
Acceptance: 24 December 2022

JOURNAL OF DATA SCIENCE
eISSN:2805-5160
Vol.2022:21

		3	13	217.44	1,091,542.20	4x4	500 - 1,189,307.56
		4	20	831.90	5,966,432.00	LORRY-S	1,000 - 7,079,211.65
		5	28	1,147.85	3,928,039.68	LORRY-M	3,000 - 19,980,366.96
		Total	111	3,287.27	15,931,953.08		4x4 (1); LORRY-S (3); LORRY-M (1) = 5 vehicles
7	2022-01-30	6	93	2,560.73	16,924,824.39		
		Driver #	Item #	Max Weight (kg)	Max Volume (cm^3)	Code	Max Weight (kg) - Volume (cm^3)
		1	13	142.80	697,402.50	4x4	500 - 1,189,307.56
		2	20	269.03	2,185,392.00	VAN	500 - 2,378,615.11
		3	13	379.16	5,017,758.75	LORRY-S	1,000 - 7,079,211.65
		4	18	1,064.11	3,388,140.15	LORRY-M	3,000 - 19,980,366.96
		5	13	200.45	789,662.00	4x4	500 - 1,189,307.56
		6	16	505.18	4,846,474.00	LORRY-S	1,000 - 7,079,211.65
		Total	93	2,560.73	16,924,824.39		4x4 (2); VAN (1); LORRY-S (2); LORRY-M (1) = 6 vehicles
					Total		51 vehicles

From Table 3, we can see what vehicle type and their number for each day. It calculates by comparing each driver's total weight and volume with what vehicle type can adequately carry out all items. Based on this data, we try to determine the reduced number of vehicles that should be used when we use load optimization algorithms.

The first experiment used load optimization based on weight and volume. This algorithm is based on the OR Tools library, provided by Google, using solving constraint integer programming (SCIP), one of the linear solvers approaches available in that library. The results are shown in Table 4.

Table 4. The list of used vehicles based on weight and volume constraint

Day #	Date	Total Vehicles		Total Weight (kg)	Total Volume (cm^3)	Total Items	Weight (kg)	Volume (cm^3)
1	2022-01-24	8		17,500	98,010,269.44	154	9,193.83	70,251,481.25
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	54	3, 687.34	24,243,955.50
		2	LORRY-L	5,000.00	24,261,874.16	57	3,161.16	24,249,172.83
		3	LORRY-M	3,000.00	19,980,366.96	41	2, 045.33	19,124,352.93
		4	LORRY-S	1,000.00	7,079,211.65	1	120.00	1,482,000.00
		5	LORRY-S	1,000.00	7,079,211.65	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	4x4	500.00	1,189,307.56	1	180.00	1,152,000.00
			Total	17,500	98,010,269.44	154	9,193.83	70,251,481.25
2	2022-01-25	7		20,500	116,733,368.41	159	11,221.90	85,927,503.16
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	69	3,257.30	24,257.537.94
		2	LORRY-L	5,000.00	24,261,874.16	45	3,894.78	24,246,241.95
		3	LORRY-M	3,000.00	19,980,366.96	35	1,836.52	19,966,723.27

Submission: 15 December 2022
Acceptance: 24 December 2022

		4	LORRY-M	3,000.00	19,980,366.96	9	2,053.00	16,305,000.00
		5	LORRY-M	3,000.00	19,980,366.96	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	4x4	500.00	1,189,307.56	1	180.00	1,152,000.00
			Total	20,500	116,733,368.41	159	11,221.90	85,927,503.16
3	2022-01-26	5		12,500.00	81,110,775.40	143	7,368.01	61,719,722.28
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	69	2,797.55	19.970,517.05
		2	LORRY-M	3,000.00	19,980,366.96	35	2,608.05	19,975,164.65
		3	LORRY-M	3,000.00	19,980,366.96	37	1,683.11	19,019,040.57
		4	LORRY-M	3,000.00	19,980,366.96	1	250.00	1,600,000.00
		5	4x4	500.00	1,189,307.56	1	29.30	1,155,000.00
			Total	12,500.00	81,110,775.40	143	7,368.01	61,719,722.28
4	2022-01-27	11				221	12,798.80	85,948,824.65
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	66	4,095.15	24,260,326.29
		2	LORRY-L	5,000.00	24,261,874.16	83	3,999.23	24,243,499.13
		3	LORRY-M	3,000.00	19,980,366.96	52	2,425.31	19,965297.50
		4	LORRY-M	3,000.00	19,980,366.96	18	1,919.00	15,175,701.72
		5	LORRY-M	3,000.00	19,980,366.96	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	LORRY-S	1,000.00	7,079,211.65	0	0	0
		9	VAN	500.00	2,378,615.11	0	0	0
		10	4x4	500.00	1,189,307.56	1	180.00	1,152,000.00
		11	4x4	500.00	1,189,307.56	1	180.00	1,152,000.00
			Total	23,500.00	134,459,714.38	221	12,798.80	85,948,824.65
5	2022-01-28	9		18,000.00	110,197,840.21	192	9,166.57	65,154,072.69
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	90	2,956.95	24,260,351.36
		2	LORRY-M	3,000.00	19,980,366.96	25	2,999.65	19,899,014.22
		3	LORRY-M	3,000.00	19,980,366.96	76	2,959.97	18,594,707.10
		4	LORRY-M	3,000.00	19,980,366.96	1	250.00	2,400,000.00
		5	LORRY-S	1,000.00	7,079,211.65	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	VAN	500.00	2,378,615.11	0	0	0
		9	VAN	500.00	2,378,615.11	0	0	0
			Total	18,000.00	110,197,840.21	192	9,166.57	65,154,072.69
6	2022-01-29	5		6,500.00	42,407,309.47	111	3,287.27	15,931,953.08
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	106	2,997.87	15,030,754,08
		2	LORRY-S	1,000.00	7,079,211.65	5	289.40	901,199.00
		3	LORRY-S	1,000.00	7,079,211.65	0	0	0
		4	LORRY-S	1,000.00	7,079,211.65	0	0	0
		5	4x4	500.00	1,189,307.56	0	0	0
			Total	6,500.00	42,407,309.47	111	3,287.27	15,931,953.08

Submission: 15 December 2022
Acceptance: 24 December 2022

7	2022-01-30	6		6,500.00	38,896,020.49	93	2,560.73	16,924,829.39
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	93	2,560.73	16,924,829.39
		2	LORRY-S	1,000.00	7,079,211.65	0	0	0
		3	LORRY-S	1,000.00	7,079,211.65	0	0	0
		4	VAN	500.00	2,378,615.11	0	0	0
		5	4x4	500.00	1,189,307.56	0	0	0
		6	4x4	500.00	1,189,307.56	0	0	0
			Total	6,500.00	38,896,020.49	93	2,560.73	16,924,829.39
Total Lorry		51	lorries		Total Empty Lorry	23	lorries	28 used lorries

The precise number of items, their weight, and their volume that can fit inside each vehicle are listed in Table 4. It demonstrates how the optimization outcome based on their truck capacity can eliminate 23 trucks. Only 28 out of the 51 available lorries are actually utilized.

Based on the items' individual weights, volumes, and shortest lengths inside the same container, the final experiment was conducted. This approach makes use of new weight and volume restricted K-Means algorithms. The result is shown in Table 5.

Table 5. The list of used vehicles based on weight, volume, and shortest distance approach

Day #	Date	Total Vehicles		Total Weight (kg)	Total Volume (cm^3)	Total Items	Weight (kg)	Volume (cm^3)
1	2022-01-24	8		17,500	98,010,269.44	154	9,193.83	70,251,481.25
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	43	2,494.00	20,618,469.00
		2	LORRY-L	5,000.00	24,261,874.16	37	3,055.00	24,123,667.00
		3	LORRY-M	3,000.00	19,980,366.96	54	2,744.00	19,614,128.93
		4	LORRY-S	1,000.00	7,079,211.65	20	899.70	7,079,211.65
		5	LORRY-S	1,000.00	7,079,211.65	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	4x4	500.00	1,189,307.56	0	0	0
			Total	17,500	98,010,269.44	154	9,193.83	70,251,481.25
2	2022-01-25	7		20,500	116,733,368.41	159	11,221.90	85,927,503.16
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	60	3,017.22	21,994,269.90
		2	LORRY-L	5,000.00	24,261,874.16	34	3,278.70	24,254,004.00
		3	LORRY-M	3,000.00	19,980,366.96	30	2,998.70	19,728,374.00
		4	LORRY-M	3,000.00	19,980,366.96	35	1,927.28	19,950,854.27
		5	LORRY-M	3,000.00	19,980,366.96	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	4x4	500.00	1,189,307.56	0	0	0
			Total	20,500	116,733,368.41	159	11,221.90	85,927,503.16
3	2022-01-26	5		12,500.00	81,110,775.40	143	7,368.01	61,719,722.28

Submission: 15 December 2022
Acceptance: 24 December 2022

JOURNAL OF DATA SCIENCE
eISSN:2805-5160
Vol.2022:21

		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	53	2,743.85	19,288,694.25
		2	LORRY-M	3,000.00	19,980,366.96	39	2,043.46	19,695,094.57
		3	LORRY-M	3,000.00	19,980,366.96	36	1,978.75	17,408,102.00
		4	LORRY-M	3,000.00	19,980,366.96	15	601,95	5,327,831.45
		5	4x4	500.00	1,189,307.56	0	0	0
			Total	12,500.00	81,110,775.40	143	7,368.01	61,719,722.28
4	2022-01-27	11				221	12,798.80	85,948,824.65
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	83	3,826.03	22,701,390.00
		2	LORRY-L	5,000.00	24,261,874.16	27	3,750.00	24,032,000.00
		3	LORRY-M	3,000.00	19,980,366.96	58	2,947.63	19,504,284.46
		4	LORRY-M	3,000.00	19,980,366.96	53	2,275.00	19,675,150.00
		5	LORRY-M	3,000.00	19,980,366.96	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	LORRY-S	1,000.00	7,079,211.65	0	0	0
		9	VAN	500.00	2,378,615.11	0	0	0
		10	4x4	500.00	1,189,307.56	0	0	0
		11	4x4	500.00	1,189,307.56	0	0	0
			Total	23,500.00	134,459,714.38	221	12,798.80	85,948,824.65
5	2022-01-28	9		18,000.00	110,197,840.21	192	9,166.57	65,154,072.69
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-L	5,000.00	24,261,874.16	76	2,850.72	24,016,398.00
		2	LORRY-M	3,000.00	19,980,366.96	36	2,995.99	19,813,677.23
		3	LORRY-M	3,000.00	19,980,366.96	28	321.56	1,380,213.00
		4	LORRY-M	3,000.00	19,980,366.96	52	2998.29	19,943,784.31
		5	LORRY-S	1,000.00	7,079,211.65	0	0	0
		6	LORRY-S	1,000.00	7,079,211.65	0	0	0
		7	LORRY-S	1,000.00	7,079,211.65	0	0	0
		8	VAN	500.00	2,378,615.11	0	0	0
		9	VAN	500.00	2,378,615.11	0	0	0
			Total	18,000.00	110,197,840.21	192	9,166.57	65,154,072.69
6	2022-01-29	5		6,500.00	42,407,309.47	111	3,287.27	15,931,953.08
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	72	2,287	11,360,316.88
		2	LORRY-S	1,000.00	7,079,211.65	39	999.81	4,571,636.20
		3	LORRY-S	1,000.00	7,079,211.65	0	0	0
		4	LORRY-S	1,000.00	7,079,211.65	0	0	0
		5	4x4	500.00	1,189,307.56	0	0	0
			Total	6,500.00	42,407,309.47	111	3,287.27	15,931,953.08
7	2022-01-30	6		6,500.00	38,896,020.49	93	2,560.73	16,924,829.39
		Vehicle #	Code	Max Weight (kg)	Max Volume (cm^3)	No of Items	Weight (kg)	Volume (cm^3)
		1	LORRY-M	3,000.00	19,980,366.96	66	1,596.22	13,636,046.88
		2	LORRY-S	1,000.00	7,079,211.65	27	964.51	3,288,782.52
		3	LORRY-S	1,000.00	7,079,211.65	0	0	0

Submission: 15 December 2022
Acceptance: 24 December 2022

	4	VAN	500.00	2,378,615.11	0	0	0
	5	4x4	500.00	1,189,307.56	0	0	0
	6	4x4	500.00	1,189,307.56	0	0	0
		Total	6,500.00	38,896,020.49	93	2,560.73	16,924,829.39
Total Lorry	51	lorries		Total Empty Lorry	27	lorries	24 used lorries

Table 5 details the precise quantity of items, together with their weight and volume, that can fit into each lorry. It demonstrates that the optimization outcome based on their capacity for lorries can eliminate 27 lorries. Only 24 out of the 51 available trucks are used. It demonstrates that the load optimization based on weight and volume was not as effective as the shortest technique, which produced better results.

Conclusion

This study focuses on load optimization using a program that simulates the interaction of load optimization issues in order to optimize the volume and weight of product preparation and arrangement based on delivery destinations (in terms of distance) on previously operating vehicles. In this study, the initial load item carried by the original driver was compared using experimental data and the outcomes of the load optimization approach. Results obtained before the shortest distance approach was improved were compared to the results currently. The comparison showed that the fastest route delivered better results.

References

Abdurrahman Rois, M., Maslihah, S., & Cahyono, B. (2019). Penyelesaian Integer Knapsack Problem Menggunakan Algoritma Greedy, Dynamic Programming, Brute Force dan GeneticDY, DYNAMIC PROGRAMMING, BRUTE FORCE DAN GENETIC. *Telematika*, *12*(2), 87–97. https://doi.org/10.35671/telematika.v12i2.841

Devita, R. N., & Wibawa, A. P. (2020). Teknik-teknik Optimasi Knapsack Problem. *Sains, Aplikasi, Komputasi Dan Teknologi Informasi*, *2*(1), 35. https://doi.org/10.30872/jsakti.v2i1.3299

Du, M., Huang, X., & Qiu, B. (2014). K-means-based processing of GPS data. *Sensors and Transducers*, *163*, 148–154.

Gazali, W., Ngarap, & Manik, I. (2010). Perancangan Program Simulasi Optimasi Penyusunan Barang Dalam Kontainer Menggunakan Algoritma Greedy. *Jurnal Mat Stat*, *10*(2), 100–113.

Google. (2022). Google OR Tools. Retrieved October 23, 2022, from google.com website: https://developers.google.com/optimization

Li, Y., Zhou, X., Gu, J., Guo, K., & Deng, W. (2022). A Novel K-Means Clustering Method for Locating Urban Hotspots Based on Hybrid Heuristic Initialization. *Applied Sciences (Switzerland)*, *12*(16). https://doi.org/10.3390/app12168047

Submission: 15 December 2022
Acceptance: 24 December 2022

Moesya, A. R., Cahyaningrum, K., & Khairunnisa, S. (2019). *Komparasi Strategi Dynamic Programming dan Branch and Bound untuk Permasalahan 0 / 1 Knapsack Program Studi Sarjana Informatika Fakultas Informatika Universitas Telkom Bandung*. 0–6.

Notebook, J. (2017). Jupyter. Retrieved October 10, 2022, from jupyter website: https://jupyter.org/

Sampurno, G. I., Sugiharti, E., & Alamsyah, A. (2018). Comparison of Dynamic Programming Algorithm and Greedy Algorithm on Integer Knapsack Problem in Freight Transportation. *Scientific Journal of Informatics*, *5*(1), 49. https://doi.org/10.15294/sji.v5i1.13360

Supriana, I. W. (2016). Optimalisasi Penyelesaian Knapsack Problem Dengan Algoritma Genetika. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *7*(3), 182. https://doi.org/10.24843/lkjiti.2016.v07.i03.p06

Wungguli, D., Ibrahim, S., & Yahya, L. (2021). Perbandingan Algoritma Greedy Dan Metode Branch And Bound Pada Penyelesaian Knapsack 0-1 Untuk Mengoptimalkan Muatan Barang. *JURNAL ILMIAH MATEMATIKA DAN TERAPAN*, *18*, 188–198. https://doi.org/10.22487/2540766X.2021.v18.i2.15605

Submission: 15 December 2022
Acceptance: 24 December 2022