

IMPLEMENTASI LINEAR PROGRAMMING PADA MODEL CVRPP UNTUK PENGELOLAAN OPERASIONAL LOGISTIK

Nur Tri Ramadhanti Adiningrum¹⁾, Cahyo Prianto²⁾, Muhammad Yusril Helmi Setyawan³⁾

^{1,2,3} D4 Teknik Informatika, Universitas Logistik dan Bisnis Internasional, Jl. Sarimanah No.14

Co Responden Email: nurtrira06@gmail.com

Abstract

A logistics company is a company that specializes in providing logistics services, which helps in managing supply chain functions including warehousing, distribution, and transportation. One of the logistics companies in the city of Bandung, which is engaged in logistics services and has a pickup service that is tasked with marketing products and providing goods pick-up services. In this service, the activity planning stage such as routes and vehicle capacity is an important stage. However, in its implementation, this company has not implemented travel activities with the best routes or only based on driver experience, and has not maximized the vehicle's carrying capacity. Capacitated Vehicle Routing Problem with Pickup (CVRPP) is the method used in handling this problem. This study aims to create a distance search model and maximize vehicle capacity that can improve operational efficiency in managing logistics capacity and routes. To achieve the research objectives, Linear Programming with the Python programming language is used as the calculation process used and produces the best solution. The results of the study show that the route formed using Linear Programming produces the shortest distance among other routes with a distance savings of 19.99% in the analysis and 31.92% in the application. This is also supported by an evaluation with an Optimality Gap value of 0% or the solution found is optimal or very good.

Abstrak

Perusahaan logistik merupakan perusahaan yang memiliki kekhususan dalam penyediaan layanan logistik, yang membantu dalam mengelola fungsi rantai pasokan termasuk pergudangan, distribusi, dan transportasi. Salah satu perusahaan logistik di kota Bandung, yang bergerak di bidang layanan jasa logistik dan memiliki layanan *pickup* yang bertugas untuk memasarkan produk serta melakukan layanan penjemputan barang. Pada layanan ini, tahap perencanaan aktivitas seperti rute dan kapasitas kendaraan merupakan tahapan yang penting. Namun, pada penerapannya perusahaan ini belum menerapkan aktifitas perjalanan dengan rute terbaik atau hanya berdasar pengalaman *driver*, serta kurang memaksimalkan kapasitas angkut kendaraan. *Capacitated Vehicle Routing Problem with Pickup (CVRPP)* adalah metode yang digunakan dalam penanganan masalah ini. Penelitian ini bertujuan pada pembuatan model pencarian jarak dan pemaksimalan kapasitas kendaraan yang dapat meningkatkan efisiensi operasional dalam pengelolaan kapasitas dan rute logistik. Untuk mencapai tujuan penelitian, *Linear Programming* dengan bahasa pemrograman Python digunakan sebagai proses perhitungan yang digunakan dan menghasilkan solusi terbaik. Hasil penelitian menunjukkan bahwa rute yang terbentuk menggunakan *Linear Programming* menghasilkan jarak paling pendek diantara rute lainnya dengan penghematan jarak sebesar 19.99% pada analisis dan 31.92% pada aplikasi. Hal itu juga didukung dengan evaluasi dengan *Optimality Gap* yang bernilai 0% atau solusi yang ditemukan adalah optimal atau sangat baik.

Article history

Received 04 Jun 2024

Revised 19 Sep 2024

Accepted 11 Oct 2024

Available online 31 Oct 2024

Keywords

Logistic,

Best Route,

Vehicle Capacity,

Linear Programming,

Pickup

Riwayat

Diterima 04 Jun 2024.

Revisi 19 Sep 2024

Disetujui 11 Okt 2024

Terbit online 31 Okt 2024

Kata Kunci

Logistik,

Rute terbaik,

Kapasitas kendaraan,

Pemrograman Linier,

Penjemputan

PENDAHULUAN

Logistik merupakan aliran barang atau jasa dari sumber ke tujuan untuk memenuhi kebutuhan pelanggan. Fokus logistik adalah pergerakan barang atau jasa (Ridwan dkk., 2020). Tujuan logistik adalah menyediakan ketepatan pada jumlah barang, waktu, tempat dan biaya. Perusahaan logistik memiliki kekhususan dalam penyediaan layanan logistik, yang membantu dalam mengelola fungsi rantai pasokan termasuk pergudangan, distribusi, dan transportasi (Ngo, 2023). Pendekatan terbaik untuk logistik adalah memprioritaskan efektivitas untuk mengelola ketepatan ketika memenuhi permintaan pelanggan (Bantacut & Fadhil, 2018). Adapun salah satu perusahaan logistik di kota Bandung sebagai objek penelitian ini, yang bergerak dibidang layanan jasa logistik. Perusahaan ini memiliki layanan *pickup* yang bertugas untuk memasarkan produk serta melakukan *pickup service*.

Beberapa dekade terakhir terjadi peningkatan pemanfaatan optimasi, berdasarkan riset operasi dan pemrograman matematika, untuk manajemen penyediaan barang dan jasa yang efektif dalam sistem distribusi dan transportasi (Moghdani dkk., 2021). Bidang tersebut dapat dioptimalkan dengan menerapkan *Vehicle Routing Problem* (VRP), yang digambarkan sebagai masalah perancangan rute pengiriman atau pengumpulan yang optimal dari satu depot ke sejumlah kota atau pelanggan yang tersebar secara geografis, dengan adanya kendala samping (Sitompul & Manasye Horas, 2021). Masalah perutean kendaraan dapat didefinisikan sebagai masalah pencarian rute optimal dengan tujuan meminimalkan jarak tempuh, waktu, dan biaya yang digunakan dalam suatu proses pada bidang distribusi dan transportasi (Liu dkk., 2023). Banyak algoritma telah diusulkan untuk mengatasi VRP seperti *Heuristic Algorithm* (Claudiu Pop dkk., 2011) sesuai dengan kebutuhan proyek bisnis seperti optimasi rute dan kapasitas untuk menemukan jarak optimal berdasarkan permintaan pelanggan dan kapasitas kendaraan (Moudya dkk., 2023) (Chi & He, 2023). Adapun contoh lain penerapannya adalah masalah perutean kendaraan berkapasitas *pickup* (Koc dkk., 2020).

Pada layanan *pickup*, tahap perencanaan aktivitas dan manajemen merupakan tahapan yang penting. Perencanaan rute yang efektif dapat secara signifikan mengurangi biaya transportasi keseluruhan (Chi & He, 2023). Selain dari rute, tentunya perusahaan logistik memiliki moda transportasi yang tentunya isi barang harus sesuai dengan kapasitasnya (Limei dkk., 2018). Atribut berat barang digunakan untuk mencerminkan pembatasan kapasitas kendaraan pada barang yang dimuat (Chi & He, 2023). Meskipun terdapat berbagai algoritma pencarian rute terbaik dan pemaksimalan kapasitas, namun penerapannya dalam konteks layanan *pickup* pada perusahaan logistik yang menjadi objek penelitian ini masih belum menerapkan perjalanan rute terbaik. Dimana perusahaan ini belum menerapkan aktifitas perjalanan dengan rute terpendek atau hanya berdasar pengalaman *driver* dan belum memaksimalkan barang pada kapasitas kendaraan. Baik rute yang ditempuh tersebut jauh atau tidak, *pickup* tetap dilakukan untuk perjalanan tersebut.

Dalam konteks tersebut, penggunaan rute logistik terbaik dan pemaksimalan kapasitas kendaraan menjadi krusial pada pengaturan *pickup*. Dengan mempertimbangkan variabel-variabel seperti jarak dan kapasitas kendaraan, hasil penelitian ini ditujukan untuk memberikan solusi yang terbaik dalam pelayanan *pickup*. Sehingga, variabel yang dipertimbangkan dalam penelitian ini adalah hanya jarak dan kapasitas kendaraan dengan faktor-faktor lain seperti waktu, biaya, atau kondisi lalu lintas tidak dipertimbangkan, dengan asumsi bahwa faktor-faktor lain tidak signifikan atau dapat diabaikan. Maka, penelitian ini bertujuan pada pembuatan model pencarian jarak terbaik dan pemaksimalan kapasitas kendaraan yang dapat meningkatkan efisiensi operasional dalam pengelolaan rute logistik. Karena itu, penelitian ini direkomendasikan untuk mengisi gap pengetahuan ini dengan fokus pada pencarian jarak dan kapasitas kendaraan untuk layanan *pickup* dalam konteks spesifik di perusahaan logistik yang diteliti.

Pada penelitian ini pencarian yang dilakukan adalah meminimalkan jumlah kendaraan yang digunakan dan meminimalkan total jarak tempuh rute perjalanan, maka jenis VRP yang dapat digunakan adalah

Capacitated VRP (CVRP) (Jiang dkk., 2022) dan *Vehicle Routing Problem Pickup Delivery* (VRPPD) (Mazzuco D. dkk., 2017). Namun, dalam kasus ini, penelitian dispesifikan menjadi *pickup*. Sedangkan *delivery* diabaikan. Maka dalam penelitian ini model yang digunakan adalah *Capacitated Vehicle Routing Problem with Pickup* (CVRPP).

Untuk mencapai tujuan penelitian, penggunaan algoritma *Linear Programming* (LP) yang menggunakan paket PuLP dengan bahasa pemrograman Python yang dilakukan sebagai proses perhitungan yang digunakan dan menghasilkan solusi terbaik. Untuk memaksimalkan hasil jarak pada rute yang dilalui, digunakan media Google API Key agar jarak dihitung sesuai dengan jalan sebenarnya atau bukan dalam bentuk garis lurus antar titik lokasi. Pembuatan antar muka sederhana juga dapat digunakan sebagai visualisasi dari model yang telah dibuat dalam pengelolaan rute logistik dengan menggunakan *framework* Flask.

Kondisi perusahaan logistik sebagai objek yang diteliti saat ini adalah, pada layanan *pickup*, proses yang dilakukan adalah kurir yang berada di depot (perusahaan logistik) menerima permintaan *pickup* dari konsumen. Kemudian kurir akan *mengambil* barang ke

lokasi konsumen tersebut. Setelah kurir menerima barang dari konsumen, maka kurir akan kembali ke depot untuk meletakkan barang yang diterima oleh kurir.

Penelitian ini menerapkan proses perhitungan dengan memanfaatkan variabel yang didapat dari perusahaan logistik, yaitu data lokasi kantor depot. Lalu untuk variabel lainnya adalah lokasi konsumen dan ketentuan kapasitas maksimal transportasi. Dengan demikian, penelitian ini diharapkan dapat memberikan manfaat untuk perusahaan bidang logistik, khususnya pada jasa layanan *pickup* serta diharapkan dapat membantu dalam peningkatan kinerja layanan *pickup* pada perusahaan logistik.

METODE PENELITIAN

Metodologi penelitian adalah suatu cara untuk memecahkan masalah penelitian secara sistematis. Peneliti perlu mengetahui tidak hanya metode/teknik penelitian tetapi juga metodologinya (Patel & Patel, 2019). Adapun dalam penelitian ini terdiri dari metode pengumpulan data dan diagram alur metodologi yang digunakan untuk memecahkan masalah. Adapun langkah-langkah dalam pengerjaan penelitian dipaparkan dalam bentuk diagram alur seperti pada Gambar 1.



Gambar 1. Diagram alur penelitian

Adapun penjelasan dari metodologi penelitian adalah sebagai berikut.

1. Tahap Perencanaan

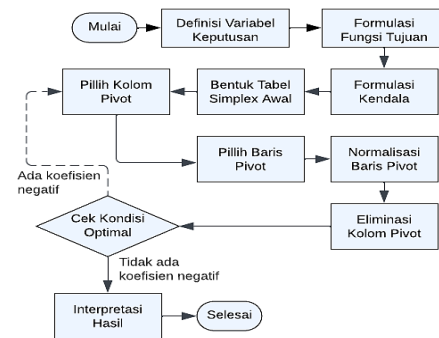
Penelitian ini menggunakan metode observasi dan wawancara untuk

mendeskripsikan proses bisnis dalam menentukan rute terbaik dan memaksimalkan kapasitas kendaraan *pickup*. Penelitian ini menggunakan algoritma *Linear Programming* (LP) untuk

meminimalkan jarak tempuh dan penggunaan kendaraan, dengan mempertimbangkan kapasitas kendaraan agar jumlah kendaraan yang digunakan seminimal mungkin tanpa melebihi kapasitas yang tersedia.

2. Tahap Pengumpulan Data
Pengumpulan data mencakup informasi titik koordinat lokasi kantor cabang utama dan alamat konsumen yang akan digunakan dalam penelitian ini.
3. Tahap Analisa
Tahap ini menggunakan dua pendekatan formula yaitu Haversine untuk mengukur jarak terpendek dan Euclidean untuk mengukur tingkat kemiripan jarak. Analisis mencakup perhitungan jarak antar titik.
4. Tahap Implementasi
Implementasi algoritma *Linear Programming* (LP) menggunakan library PuLP Python. Algoritma LP dipilih karena kemampuannya dalam menangani masalah optimasi dengan banyak variabel dan kendala, serta menemukan solusi optimal dengan efisiensi tinggi. Batasan kendala juga diterapkan dalam proses ini.
5. Evaluasi
Evaluasi dilakukan dengan *Optimality Gap* dan analisis komparatif jarak untuk mengetahui apakah hasil model lebih baik daripada kemungkinan hasil lainnya.
6. Visualisasi Antar Muka
Antarmuka diimplementasikan dengan *framework* Flask untuk membuat aplikasi web interaktif. Flask merupakan *micro web framework* yang menggunakan bahasa Python (Wijayanto & Susetyo, 2022). Flask dipilih karena kemampuannya dalam membangun aplikasi web yang ringan.

Linear Programming (LP) ialah teknik matematis yang digunakan untuk menemukan solusi optimal dari masalah yang melibatkan sejumlah batasan linier. *Linear programming* merupakan metode yang sangat efektif dalam bidang optimasi dan banyak digunakan dalam berbagai aplikasi seperti manajemen operasi, logistik, ekonomi, dan teknik (Nurmayanti & Sudrajat, 2021). Gambar 2 menunjukkan cara kerja dari *Linear Programming*.



Gambar 2. Alur kerja *Linear Programming*

Berikut adalah penjelasan alur kerja *Linear Programming*.

1. Definisi Variabel Keputusan
Tentukan variabel-variabel yang akan digunakan dalam *Linear Programming* yang mewakili keputusan yang harus diambil. Contoh: x_{ij} adalah jumlah unit yang diangkut oleh kendaraan i ke lokasi j .
2. Formulasi Fungsi Tujuan
Rumuskan fungsi tujuan yang ingin dioptimalkan, baik untuk memaksimalkan maupun meminimalkan suatu nilai.
Minimalkan atau Maksimalkan Z
$$= c_1x_1 + c_2x_2 + \dots + c_ix_{ij} \quad (1)$$

Dimana:
 Z : Fungsi tujuan
 c_i : Koefisien dari variabel keputusan x_i
 x_{ij} : Variabel keputusan
3. Formulasi Kendala
Tentukan batasan-batasan atau kendala yang harus dipatuhi oleh variabel keputusan.
$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1 \quad (2)$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2 \quad (3)$$

$$\vdots$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ij}x_j \leq b_i \quad (4)$$

$$x_i \leq 0 \text{ untuk semua } i$$

Dimana:
 a_{ij} : Koefisien dari variabel keputusan x_i dalam kendala ke- j
 b_j : Batasan kendala ke- j
4. Bentuk Tabel Simplex Awal
Buat tabel Simplex awal berdasarkan fungsi tujuan dan kendala yang telah diformulasikan.
5. Pilih Kolom Pivot
Identifikasi kolom pivot, yaitu kolom yang memiliki koefisien negatif terbesar dalam baris fungsi tujuan (untuk masalah minimisasi). Pilih kolom j dengan $\min(c_j)$.

6. Pilih Baris Pivot

Tentukan baris pivot dengan membagi nilai konstanta dalam kendala dengan koefisien positif dari kolom pivot dan memilih nilai terkecil yang positif.

Pilih baris i dengan untuk $\left(\frac{b_i}{a_{ij}}\right)$ untuk $a_{ij} > 0$

Atau

$$\text{Hitung rasio} = \frac{RHS}{\text{Koefisien di kolom pivot}} \quad (5)$$

7. Normalisasi Baris Pivot

Lakukan normalisasi baris pivot dengan membagi semua elemen baris pivot dengan elemen pivot itu sendiri.

$$\text{Baris Pivot Baru} = \frac{\text{Baris Pivot Lama}}{\text{Elemen Pivot}} \quad (6)$$

8. Eliminasi Kolom Pivot

Lakukan operasi baris untuk membuat semua elemen di kolom pivot, selain elemen pivot, menjadi nol.

$$\begin{aligned} \text{Baris Baru} = & \\ & \text{Baris Lama} \\ & - (\text{Koefisien Kolom Pivot} \\ & \times \text{Baris Pivot yang Dinormalisasi}) \end{aligned} \quad (7)$$

9. Cek Kondisi Optimal

Periksa apakah semua koefisien di baris fungsi tujuan tidak ada yang negatif. Jika ada koefisien negatif, kembali ke langkah "Pilih Kolom Pivot". Jika tidak ada koefisien negatif, lanjut ke langkah berikutnya.

10. Kondisi Optimal Tercapai dan Interpretasi Hasil

Setelah kondisi optimal tercapai, interpretasikan hasil yang diperoleh dari tabel simplex terakhir. Hasil akhir yang didapat adalah nilai optimal fungsi tujuan (Z) dan nilai variabel keputusan (x_1, x_2, \dots, x_m).

Pada setiap kendaraan, proses *pickup* dilakukan sebanyak satu kali. Yaitu dari depot ke setiap konsumen, kemudian kembali lagi ke depot (Kristina dkk., 2020). Maka dalam hal ini, rute yang digunakan untuk pemodelan adalah depot → konsumen → depot. Kemudian untuk kapasitas digunakan total beban maksimal kapasitas kendaraan dalam satuan kilogram. Maka variabel yang digunakan untuk penelitian ini adalah titik lokasi untuk menentukan jarak, dan jumlah beban total yang dapat diangkut kendaraan untuk memaksimalkan kapasitas kendaraan dalam membawa barang.

Dari variabel yang digunakan, jarak tempuh yang minimum dipengaruhi oleh bagaimana rute perjalanan yang dilalui (Basriati & Aziza, 2017). Selain itu, penggunaan kapasitas kendaraan distribusi yang tidak maksimal mengakibatkan kebutuhan kendaraan menjadi tinggi. Hal ini tentunya berpengaruh terhadap jauhnya jarak tempuh kendaraan yang dilalui dan biaya yang dikeluarkan membesar (Efendi, 2022).

HASIL DAN PEMBAHASAN

Data lokasi depot yang digunakan adalah terletak pada koordinat *latitude* -6.919792 dan koordinat *latitude* 107.606014. Untuk lokasi konsumen dapat menggunakan lokasi koordinat konsumen. Total maksimal beban kapasitas yang dapat diangkut oleh kendaraan adalah 60 kilogram, dengan total kendaraan yang tersedia diasumsikan terdapat empat kendaraan yang tersedia pada depot.

Adapun lokasi dan *demand* permintaan yang dibuat secara sintesis dengan menggunakan bahasa pemrograman Python dengan data *dummy*. Untuk lokasi depot berada ditengah – tengah dan dikelilingi oleh lokasi konsumen sehingga membentuk seperti *hexagon*. *Demand* pada depot adalah 0 karena tidak terjadi kegiatan *pickup*. Adapun data yang dibuat adalah sebagai berikut.

Tabel 1. Data lokasi dan *demand*

Lokasi	<i>Latitude</i>	<i>Longitude</i>	<i>Demand</i>
Depot (0)	-6.919792	107.606014	0
1	-6.918875	107.606014	28
2	-6.918600	107.608079	24
3	-6.921718	107.609349	16
4	-6.925110	107.606014	10
5	-6.923185	107.600138	10
6	-6.915666	107.598867	12

Analisis Bentuk Grafik

Setelah dibuat dengan titik yang dihubungkan terbentuk hasil rute seperti pada Gambar 3(a).

Dalam menentukan jarak, tentunya diperlukan lokasi satu titik dengan titik yang lain. Adapun cara untuk menghitung jarak dapat dilakukan dengan menggunakan formula Haversine dan Euclidean. Haversine akan menghasilkan jarak terpendek antara dua titik, misalnya pada bola yang diambil dari garis bujur (*longitude*) dan garis lintang (*latitude*) (Al Amin & Wahyudiyono, 2021). Sedangkan nilai Euclidean diterapkan sebagai fungsi

heuristik yang diperoleh dari jarak langsung yang ada pada jarak koordinat(Eviana dkk., 2022).

Adapun formula Haversine adalah sebagai berikut.

$$\Delta lat = y = lat_2 - lat_1 \quad (8)$$

$$\Delta long = long_2 - long_1 \quad (9)$$

$$x = \Delta long \cdot \cos\left(\frac{lat_1 + lat_2}{2}\right) \quad (10)$$

$$d = \sqrt{(x.x + y.y)} \cdot R \quad (11)$$

Dimana :

R = jari-jari bumi sebesar 6371(km)

x = Longitude (Bujur)

Δlat = besaran perubahan latitude

Δlong = besaran perubahan longitude

y = besaran perubahan latitude

d = jarak (meter)

1 derajat = 0.0174532925 radian

Adapun formula Euclidean sebagai berikut.

$$h(n) = \sqrt{(x_{i1} - x_{i2})^2 + (y_{i1} - y_{i2})^2} \quad (12)$$

Dimana:

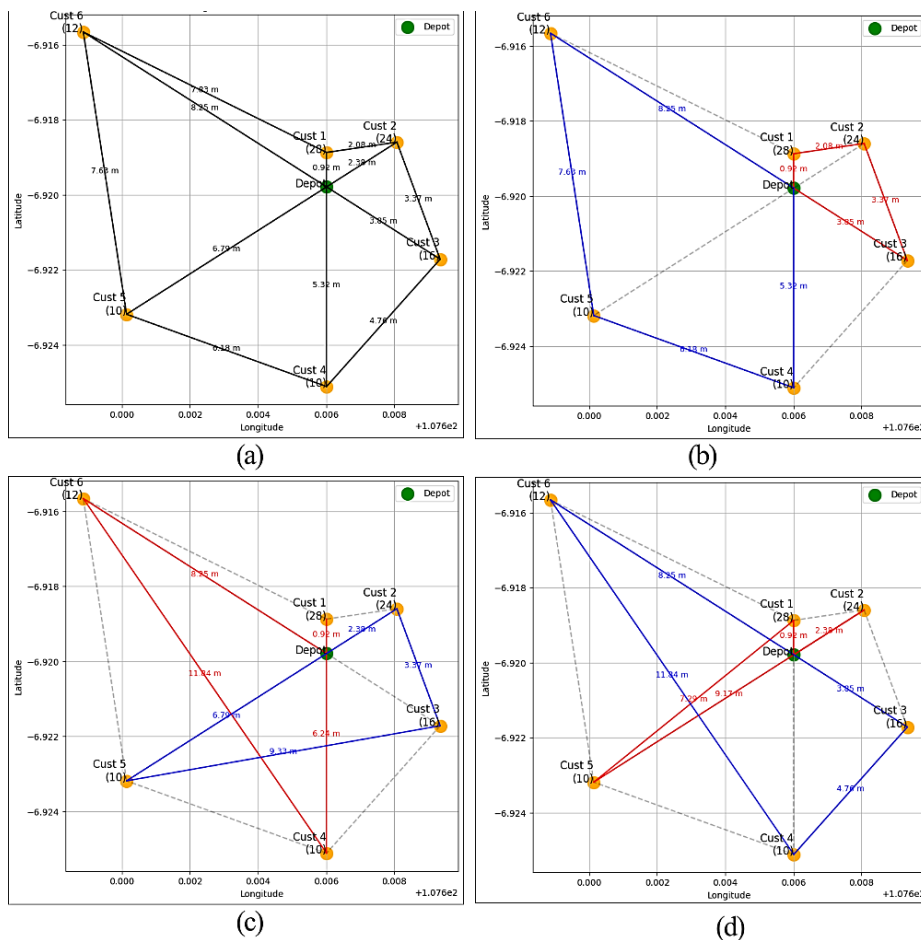
x_i (latitude) = garis lintang mengarah dari khatulistiwa (0) ke kutub selatan, atau khatulistiwa ke kutub utara (sudut 0-90 dan 0 -90).

y_i (longitude) = garis bujur adalah garis horizontal seperti dari khatulistiwa. Sudut 0 (Greenwich) ke arah Hawaii adalah 0-180, sedangkan kebalikannya dari 0 ke-180.

i = index dari atribut

n = jumlah data

Seluruh perhitungan yang digunakan menggunakan bahasa pemrograman Python. Dari hasil perhitungan Euclidean dan haversine hasil gambar yang terbentuk adalah pada Gambar 3.



Gambar 3. Bentuk Rute Yang Dihasilkan

Berdasarkan data lokasi dan demand, dibuatkan rute lain (rute random) yang

kemungkinan berpeluang untuk melakukan pickup ada pada Gambar 3(b), 3(c), dan 3(d).

Hasil rute-rute tersebut adalah rute yang dibuat tanpa menggunakan *Linear Programming*. Adapun hasil jarak tempuh total yang dihasilkan rute-rute tersebut ada pada Tabel 2 berikut.

Rute Random	Jarak Total (m)
1 (3(b))	37.60
2 (3(c))	49.11
3 (3(d))	48.46

Hal selanjutnya adalah pembentukan rute dengan *Linear Programming*. Pada perhitungan LP, dalam penelitian ini perlu ditentukan beberapa aspek, yaitu:

1. Inisialisasi variabel keputusan
 Dimana: x_{ij} adalah jumlah unit yang diangkut oleh kendaraan i ke lokasi j .
2. Fungsi tujuan
 Fungsi tujuan yang digunakan adalah fungsi *Minimize*. Fungsi ini digunakan untuk meminimalkan total jarak yang ditempuh oleh semua kendaraan.
3. Fungsi kendala (*constraint*)
 - a. Setiap node dikunjungi tepat sekali oleh satu kendaraan
 - b. Kendaraan harus berangkat dari depot dan kembali ke depot
 - c. Total permintaan *demand* yang diambil sama dengan yang diantar
 - d. Kendaraan tidak melebihi kapasitasnya
 - e. Mencegah terbentuknya rute parsial (*subtour*)

Pada kasus ini, perhitungan *Linear Programming* menggunakan metode simpleks karena berfungsi untuk mencari nilai optimum(Fikri dkk., 2021). Untuk menggunakan metode simpleks sebagai penentu nilai variabel keputusan optimum yang didapat dengan melakukan iterasi secara berulang terhadap tabel simpleks sampai ditemukan nilai yang optimum(Nurmayanti & Sudrajat, 2021). Pada tujuan minimisasi, tabel sudah optimal jika semua nilai pada baris Z sudah negatif atau 0. Jika belum, kembali ke langkah kedua; jika sudah optimal, baca solusi optimal(Suroso & Nugroho, 2023).

Proses perhitungan dilakukan dengan cara mengambil data dari koordinat lokasi dan data permintaan *demand*. Data lokasi dan *demand* tertera pada Tabel 1. Untuk mencapai tujuan

penelitian, penggunaan algoritma *Linear Programming* yang menggunakan paket PuLP Python dilakukan sebagai proses perhitungan untuk menghasilkan solusi terbaik.

Hasil perhitungan yang didapatkan, didapatkan nilai jarak yang ada pada Tabel 3 dengan hasil minimasi rute yang terbentuk. Variabel x_{ij} mengartikan jarak antara titik i dan titik j . Dimana $x_{ij} = 1$ menunjukkan bahwa rute dari i ke j diambil.

Variabel	Jarak(m)	Variabel	Jarak(m)
x_{03}	3.85	x_{60}	14.20
x_{34}	0.18	x_{01}	0.92
x_{45}	4.76	x_{12}	2.08
x_{56}	7.68	x_{20}	2.38

Dengan menggabungkan jarak-jarak rute tersebut, persamaan Z minimasi akhirnya adalah:

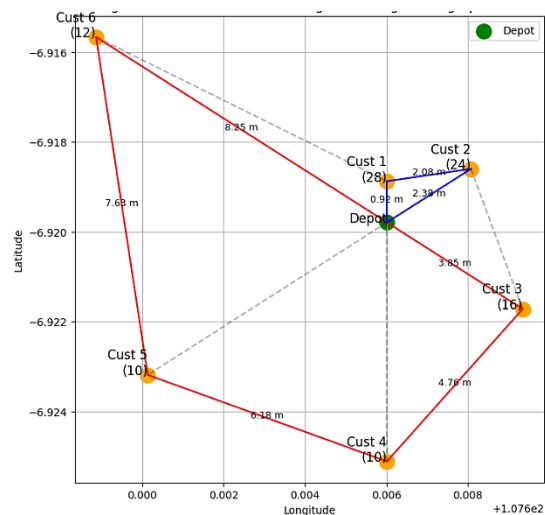
$$Z = (3.85x_{03} + 0.18x_{34} + 4.76x_{45} + 7.68x_{56} + 14.20x_{60} + 0.92x_{01} + 2.08x_{12} + 2.38x_{20}) \quad (13)$$

Jika di substitusi nilai jaraknya maka perhitungannya adalah:

$$Z = (3.85*1) + (0.18*1) + (4.76*1) + (7.68*1) + (14.20*1) + (0.92*1) + (2.08*1) + (2.38*1) \quad (14)$$

$$Z = 36.05 \text{ meter} \quad (15)$$

Hasil Z minimasi yang dipatkan didukung dengan hasil visualisasi pada Gambar 4 dan penjelasan pada Tabel 4



Gambar 4. Hasil rute yang didapatkan

Berdasarkan Gambar 4, terdapat hasil pencarian rute yang didapatkan. Dimana total kendaraan yang digunakan sebanyak dua kendaraan dari empat kendaraan yang tersedia.

Hal itu juga dibuktikan dengan rute yang terbentuk terbagi menjadi dua rute yang tertera pada warna merah dan warna biru. Hasil rute dipaparkan pada Tabel 3.

Kendaraan	Warna	Hasil Rute	Jarak (m)
1	Merah	0→3→4→5 →6→0	30.67
2	Hijau	0→1→2→0	5.38
Total jarak			36.05

Berdasarkan hasil jarak total pada Tabel 2 dan Tabel 4, dapat diketahui bahwa rute yang terbentuk menggunakan LP menghasilkan jarak yang paling pendek yaitu 36.05 meter atau hemat jarak sekitar 19.99% dari rata-rata jarak rute random. Hal ini menunjukkan bahwa rute yang dihasilkan menggunakan LP menghasilkan jarak paling pendek dibandingkan dengan jarak di rute-rute yang lainnya.

Untuk mengukur seberapa baik atau seberapa dekat sebuah solusi yang ada dengan solusi terbaik, digunakan evaluasi dengan *Optimality Gap*. Adapun formula *Optimality Gap* adalah sebagai berikut (Bernardino & Paias, 2024).

$$\text{Optimality Gap} = \left(\frac{\text{Jarak rute yang dihitung} - \text{Jarak Optimal}}{\text{Jarak Optimal}} \right) \times 100\% \quad (13)$$

Interpretasi persentase *Optimality Gap*:

- 0%: *Optimality gap* 0% berarti solusi yang ditemukan adalah optimal atau sangat baik mendekati nilai optimal yang sebenarnya.
- <1%: Sering dianggap sangat baik dalam banyak konteks optimasi yang membutuhkan presisi tinggi.

- 1-5%: Umumnya dianggap baik dan dapat diterima terutama untuk masalah yang tidak memerlukan solusi yang presisi.
- 5-10%: Masih dapat diterima dalam beberapa kasus tergantung pada kompleksitas masalah dan kebutuhan praktisnya.
- >10%: Menunjukkan bahwa solusi yang ditemukan jauh dari nilai optimal dan memerlukan perbaikan atau evaluasi lebih lanjut.

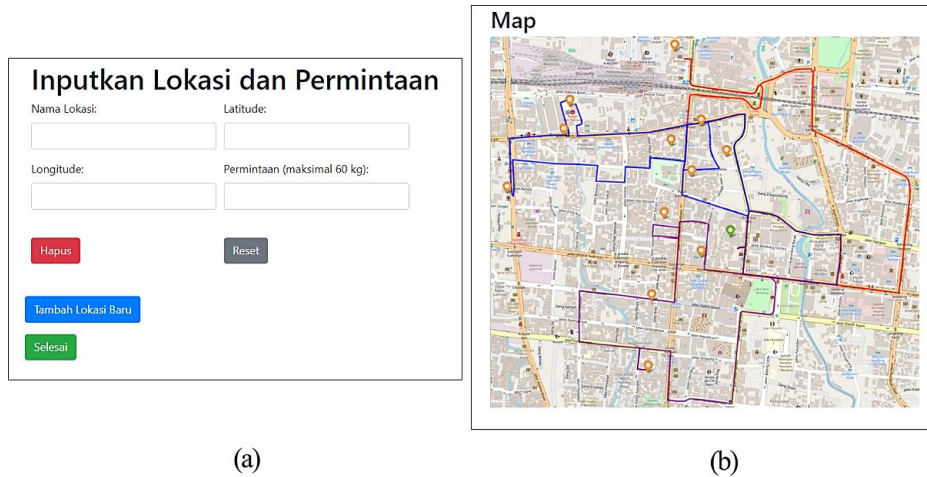
Setelah ditemukan jarak total rute random dan jarak total yang dibentuk oleh LP, *Optimality Gap* dapat dihitung dan dihasilkan yang tertera pada Tabel 4.

Rute	Jarak (m)	<i>Optimality Gap</i>
LP	36.05	0%
Random 1	37.60	4.28%
Random 2	49.11	36.22%
Random 3	48.46	34.42%

Berdasarkan Tabel 5, rute yang dihasilkan menggunakan LP memiliki *Optimality Gap* sama dengan 0%, atau hasil optimal yang dihasilkan sangat baik. Sedangkan rute lain memiliki *Optimality Gap* yang cukup jauh dari 0%. Maka hasil yang telah didapatkan menggunakan *Linear Programming* mampu menghasilkan solusi optimal yang baik.

Implementasi Aplikasi

Untuk memberikan hasil visualisasi yang lebih baik, perancangan sistem untuk pencarian rute terbaik dan memaksimalkan kapasitas kendaraan dibuat dengan menggunakan *framework* Flask. Adapun hasil implementasi aplikasi yang dibuat adalah tertera pada Gambar 5(a) dan Gambar 5(b).



Gambar 5. Implementasi Aplikasi

Flask adalah kerangka kerja web yang ditulis dalam bahasa Python. Flask juga menyediakan pustaka dan koleksi kode program yang dapat digunakan untuk membangun situs web (Novindri & Saian, 2022).

Pada Gambar 5(a), Halaman index digunakan untuk menginputkan nama lokasi, koordinat lokasi, dan berat barang permintaan konsumen. Pada Gambar 5(b) Halaman hasil menunjukkan hasil rute yang terbentuk dari inputan pada halaman index. Hasil yang ditampilkan adalah *report* rute masing-masing kendaraan beserta total permintaan dan jaraknya, dan juga map yang dibentuk menggunakan OpenStreetMap.

Inputan untuk aplikasi dapat menggunakan data lokasi secara bebas dengan Batasan lokasi sejauh 1 km dari depot sebagai area *pickup*. Adapun data terbaru yang digunakan sebanyak 12 lokasi pada Tabel 6.

Tabel 6. Data Lokasi dan Demand Untuk Aplikasi

Lokasi	Latitude	Longitude	Demand
Depot (0)	-6.919792	107.606014	0
1	-6.915593	107.601029	15
2	-6.927422	107.610256	15
⋮	⋮	⋮	⋮
12	-6.924538	107.603077	18

Pada 5(b) hasil menunjukkan terdapat tiga kendaraan yang berjalan dengan warna merah, biru dan ungu seperti yang tertera pada Tabel 7. Tentunya evaluasi tetap dilakukan pada aplikasi. Evaluasi yang digunakan sama seperti pada evaluasi grafik yaitu perbandingan jarak rute dan *Optimality Gap*. Adapun hasilnya adalah pada Tabel 7.

Tabel 7. Evaluasi rute pada aplikasi

Rute	Hasil Rute	Jarak (m)
Random	0→1→4→7→11→12 →0	9348.0
	0→3→6→8→0	5038.0
	0→2→5→9 →10→0	6641.0
	Total Jarak	21027.0
	<i>Optimality Gap</i>	46.91%
LP	0→5→8→5→ 9→7→0	4137.0
	0→10→1→6→ 12→0	5168.0
	0→11→3→2→0	3054.0
	Total Jarak	14312.9
	<i>Optimality Gap</i>	0%

Tabel 6 menunjukkan hasil dari jarak total rute random dan rute *Linear Programming* serta *Optimality Gap* setiap rute. Berdasarkan evaluasi, total jarak yang didapat menggunakan LP adalah jarak yang paling pendek yaitu 14312.9 meter atau menghemat jarak sebesar 31.92% dari jarak random. Kemudian untuk *Optimality Gap* yang memiliki solusi terbaik adalah rute yang dibentuk oleh *Linear Programming* yaitu 0%.

Maka hasil pencarian rute dan pemaksimalan kapasitas kendaraan yang dilakukan menggunakan *Linear Programming* yang telah diimplementasikan pada aplikasi mampu menghasilkan solusi optimal yang baik.

KESIMPULAN

Dapat disimpulkan penelitian ini bertujuan untuk menemukan rute dengan jarak tempuh terbaik pada layanan *pickup* dan memaksimalkan kapasitas kendaraan menggunakan *Linear Programming* yang

dievaluasi menggunakan perbandingan jarak dan *Optimality Gap* dengan bahasa pemrograman Python yang dapat diimplementasikan dalam bentuk aplikasi.

Berdasarkan hasil penelitian, didapatkan dua kendaraan yang digunakan dari empat kendaraan yang tersedia. Hal itu berarti dengan melakukan penerapan LP pada kapasitas kendaraan, dapat memberikan hasil berupa penggunaan kendaraan dapat diefektifkan atau diminimalisir. Selain itu, dari perbandingan jarak rute, rute yang dibentuk menggunakan LP memiliki rute terpendek diantara rute-rute lainnya. Hal itu juga didukung dengan hasil *Optimality Gap* untuk rute LP adalah 0% atau solusi yang ditemukan adalah optimal atau sangat baik. Penelitian ini menunjukkan bahwa model CVRPP menggunakan LP mampu mendapatkan hasil rute terbaik dan dapat memaksimalkan kapasitas kendaraan.

Implementasi dalam bentuk aplikasi untuk model ini menggunakan *framework* Flask. Hasil uji menggunakan aplikasi juga didapatkan bahwa rute yang terbentuk oleh *Linear Programming* adalah rute terpendek dan memiliki nilai *Optimality Gap* 0%.

REFERENSI

- Al Amin, I. H., & Wahyudiyono, W. (2021). Implementasi Metode Haversine Untuk Pencarian Optical Distribution Point. *Jurnal Ilmiah Teknologi Informasi*, 13(1).
- Bantacut, T., & Fadhil, R. (2018). Penerapan LOGISTIK 4.0 dalam Manajemen Rantai Pasok Beras Perum BULOG: Sebuah Gagasan Awal. *Jurnal Pangan*, 27(2). <https://doi.org/https://doi.org/10.33964/jp.v27i2.371>
- Basriati, S., & Aziza, D. (2017). Penentuan Rute Distribusi pada Multiple Depot Vehicle Routing Problem (MDVRP) Menggunakan Metode Insertion Heuristic (Studi Kasus: Orange Laundry di Kota Pekanbaru). *Jurnal Sains Matematika dan Statistika*, 3. <https://doi.org/10.24014/jsms.v3i1.4465>
- Bernardino, R., & Paiais, A. (2024). The family capacitated vehicle routing problem. *European Journal of Operational Research*, 314. <https://doi.org/10.1016/j.ejor.2023.10.042>
- Chi, J., & He, S. (2023). Pickup capacitated vehicle routing problem with three-dimensional loading constraints: Model and algorithms. *Transportation Research Part E: Logistics and Transportation Review*, 176. <https://doi.org/10.1016/j.tre.2023.103208>
- Claudiu Pop, P., Zelina, I., LupĂŃe, V., Pop Sitar, C., & Chira, C. (2011). Heuristic Algorithms for Solving the Generalized Vehicle Routing Problem. *International Journal of Computers Communications & Control*, 6(1). <https://univagora.ro/jour/index.php/ijccc/article/view/2210>
- Efendi, Y. (2022). *Perancangan Vehicle Routing Problem Menggunakan Algoritma Nearest Neighbor Dan Local Search Guna Optimasi Biaya Distribusi Pada Pt Madubaru*. dspace.uui.ac.id/handle/123456789/40110
- Eviana, A., Fauzan, Abd. C., Harliana, & Putra, F. N. (2022). Komparasi Jarak Euclidean dan Jarak Manhattan untuk Deteksi Covid-19 Melalui Citra CT-Scan Paru-Paru. *Jurnal Sistem Komputer*, 11(2). <https://doi.org/https://doi.org/10.34010/komputika.v11i2.5380>
- Fikri, A. J., Aini, S., Sukandar, R. S., Sfiyanah, I., & Listiasari, D. (2021). Optimalisasi Keuntungan Produksi Makanan Menggunakan Pemrograman Linear Melalui Metode Simpleks. *Jurnal Bayesian*, 1.
- Jiang, H., Lu, M., Tian, Y., Qiu, J., & Zhang, X. (2022). An evolutionary algorithm for solving Capacitated Vehicle Routing Problems by using local information. *Applied Soft Computing*, 17. <https://doi.org/https://doi.org/10.1016/j.asoc.2022.108431>
- Koc, C., Laporte, G., & Tukenmez, I. (2020). A review of vehicle routing with simultaneous pickup and delivery. *Computers & Operations Research*, 122. <https://doi.org/https://doi.org/10.1016/j.cor.2020.104987>
- Kristina, S., Sianturi, R. D., & Husnadi, R. (2020). Penerapan Model CVRP Menggunakan Google Or-Tools Untuk

- Penentuan Rute Pengantaran Obat Pada Perusahaan Pedagang Besar Farmasi. *Jurnal Telematika*, 15.
- Limei, H., Tannady, H., & Nurprihatin, F. (2018). Meminimumkan Biaya Transportasi pada Capacitated Vehicle Routing Problem dengan Metode Heuristik. *Seminar Rekayasa Teknologi Semrestek* 2018. <https://teknik.univpancasila.ac.id/semrestek/prosiding/index.php/12345/article/view/277>
- Liu, X., Chen, Y.-L., Por, Y. L., & Ku, C. S. (2023). A Systematic Literature Review of Vehicle Routing Problems with Time Windows. *Sustainability*, 15(15).
- Mazucco D., Oliveira D., & Frazzon E.M. (2017). State of the art in simulation-based optimization approaches for vehicle routing problems along manufacturing supply chains. *24th International Conference on Production Research*. [https://doi.org/State of the art in simulation-based optimization approaches for vehicle routing problems along manufacturing supply chains](https://doi.org/State%20of%20the%20art%20in%20simulation-based%20optimization%20approaches%20for%20vehicle%20routing%20problems%20along%20manufacturing%20supply%20chains)
- Moghdani, R., Salimifard, K., Demir, E., & Benyettou, A. B. (2021). The green vehicle routing problem: A systematic literature review. *Journal of Cleaner Production*, 279. <https://doi.org/https://doi.org/10.1016/j.jclepro.2020.123691>
- Moudya, F., Rarasati, N., & Syafmen, W. (2023). Optimisasi Rute Pada Cvrp Dalam Pendistribusian Gas Oksigen Menggunakan Algoritma Clarke And Wright Savings. *Jurnal Pendidikan Matematika dan Matematika*, 9. <https://doi.org/10.24853/fbc.9.1.105-118>
- Ngo, Q.-H. (2023). The effectiveness of market orientation in the logistic industry: A focus on SMEs in an emerging country. *Heliyon*, 9(7). <https://doi.org/https://doi.org/10.1016/j.heliyon.2023.e17666>
- Novindri, G. F., & Saian, P. O. N. (2022). Implementasi Flask Pada Sistem Penentuan Minimal Order Untuk Tiap Item Barang Di Distribution Center Pada PT XYZ Berbasis Website. *Jurnal MNEMONIC*, 5. <https://doi.org/10.36040/mnemonic.v5i2.4670>
- Nurmayanti, L., & Sudrajat, A. (2021). Implementasi linear programming metode simpleks pada home industry khasanah sari karawang. *Jurnal Manajemen*, 13(3). <https://doi.org/https://doi.org/10.30872/jmnn.v13i3.10085>
- Patel, M., & Patel, N. (2019). Exploring Research Methodology: Review Article. *International Journal of Research and Review*, 6(3). https://www.ijrrjournal.com/IJRR_Vol.6_Issue.3_March2019/IJRR0011.pdf
- Ridwan, Saputra, M. A., & Indriyanti, R. (2020). Penerapan Logistik 4.0 Dalam Pendistribusian Barang Produksi Pt. Solusi Bangun Indonesia Tbk. Cilacap. *Prosiding Seminar Sosial*, 2(1). <https://ejournal.akpelni.ac.id/index.php/prosiding-nsmis/article/view/154>
- Sitompul, C., & Manasye Horas, O. (2021). A Vehicle Routing Problem with Time Windows Subject to the Constraint of Vehicles and Good's Dimensions. *International Journal of Technology (IJTech)*, 12(4).
- Suroso, J. S. D., & Nugroho, P. (2023). Analisis Optimalisasi Produksi dengan Linear Programming Melalui Metode Simpleks (Studi Kasus UMKM Aqisa Rumah Rosella Surabaya). *Jurnal Kajian Ilmu Manajemen*, 3. <https://doi.org/10.21107/jkim.v3i2.18918>
- Wijayanto, C., & Susetyo, Y. A. (2022). Implementasi Flask Framework Pada Pembangunan Aplikasi Sistem Informasi Helpdesk (SIH). *Jurnal Ilmiah Penelitian dan Pembelajaran Informatika*, 7.