

OPTIMASI PENGIRIMAN Q9 SAMEDAY SERVICE DI POSTAL PROCESSING CENTRE BANDUNG 40400

Saepudin Nirwan¹⁾, Marwanto Rahmatuloh²⁾

Program Studi D III Teknik Informatika, Politeknik Pos Indonesia

Jl. Sariasih no. 54, Bandung 40151, Indonesia Tlp. +6222 200 9570, Fax. +6222 200 9568

Email: ¹nirwansaepudin@poltekpos.ac.id, ²mrahmatuloh@poltekpos.ac.id

ABSTRAK

Pos Indonesia adalah perusahaan terbesar di Indonesia yang melayani pengiriman barang maupun jasa. Dalam meningkatkan layanan pengiriman, Pos Indonesia melakukan berbagai inovasi salah satunya adalah layanan Q9 Sameday Service yang merupakan layanan dalam kota dimana barang yang dikirim harus sampai kepada tujuan dalam waktu maksimum 9 jam. Dalam proses distribusi Pos Indonesia telah memiliki model pengantaran dengan penjadwalan kendaraan dan rute yang sudah berjalan. Namun dalam menangani Q9 Sameday Service yang memiliki karakteristik tertentu diperlukan inovasi untuk menyesuaikan kemampuan mengantar kepada pelanggan dengan tepat waktu. Implementasi Traveling Salesman Problem menjadi salah satu model dengan pertimbangan ketepatan waktu, jarak, dan jumlah kendaraan yang diperlukan dalam melakukan distribusi kiriman. Dalam studi kasus ini, agar fleksibilitas penyelesaian TSP lebih mudah maka digunakan varian selanjutnya yaitu dengan metoda penyelesaian Vehicle Routing Problem (VRP). Masalah VRP ini diselesaikan dengan linear programming menggunakan paket PuLP Python untuk menghasilkan solusi optimal. Hasil optimum pengiriman dilakukan dengan 4 kendaraan dengan waktu perjalanan terlama 1,646 jam dan total jarak yang ditempuh 136,11 km.

Kata kunci : Vehicle Routing Problem, Pulp, Python, Pos Indonesia

I. Pendahuluan

1.1 Latar Belakang

Dalam beberapa tahun terakhir pertumbuhan jasa pengiriman di Indonesia semakin meningkat, pada tahun 2020 Asosiasi Logistisk dan Forwarder Indonesia (ALFI) memprediksi tumbuh lebih dari 30% [1] terutama pengiriman barang e-commerce. Pos Indonesia adalah perusahaan terbesar di Indonesia yang melayani pengiriman barang maupun jasa. Pos Indonesia sebagai perusahaan terbesar dibidang jasa logistik juga memanfaatkan peluang ini dengan menawarkan produk-produk inovasi baru sehingga mampu bersaing dan memenuhi kebutuhan pelanggannya. Dalam meningkatkan layanan pengiriman, Pos Indonesia melakukan berbagai inovasi salah satunya adalah layanan Q9 Sameday Service yang merupakan layanan dalam kota dimana

barang yang dikirim harus sampai kepada tujuan dalam waktu maksimum 9 jam. Dalam proses distribusi Pos Indonesia telah memiliki model pengantaran dengan penjadwalan kendaraan dan rute yang sudah berjalan. Namun dalam menangani Q9 Sameday Service yang memiliki karakteristik tertentu diperlukan inovasi untuk menyesuaikan kemampuan mengantar kepada pelanggan dengan tepat waktu [2].

1.2 Identifikasi Masalah

Q9 Sameday Service adalah produk yang memiliki karakteristik sendiri, merupakan layanan dalam kota yang harus sampai kepada pelanggan dalam waktu 9 jam dengan berat kiriman maksimum 5 kg. Pada saat ini pengelolaan kiriman Q9 Sameday Service diluar jadwal pengiriman reguler yang dilakukan oleh PPC, hal ini disebabkan fokus pada waktu pengiriman yang

merupakan standar layanan yang harus dipenuhi.

1.3 Tujuan

Untuk melakukan optimasi kiriman digunakan model Vehicle Routing Problem sederhana yang dikembangkan dari model Traveling Salesman Problem dengan Linier Programming.

1.4 Ruang Lingkup

Pengiriman Q9 Service Deliver mencakup distribusi kiriman dari Postal Processing Center Bandung 40400 ke 13 Delivery Centre di wilayah Bandung Raya.

II. Tinjauan Pustaka

2.1 Traveling Salesman Problem

Salah satu bentuk konsep optimisasi yang terus berkembang adalah optimisasi kombinatorial. Optimisasi kombinatorial adalah salah satu bentuk konsep optimisasi yang terus berkembang, optimisasi ini yang mengatur seluruh solusi yang feasible dan layak dalam bentuk diskrit. Salah satu jenis optimisasi kombinatorial adalah Traveling Salesman Problem, yaitu model untuk menemukan rute perjalanan seorang salesman yang dimulai dari satu titik awal ke beberapa titik yang telah ditentukan, setelah itu akan kembali ke titik awal sehingga jarak yang ditempuh minimum dengan setiap titik dikunjungi tepat satu kali [4]. Tujuan akhirnya adalah untuk menentukan panjang lintasan terbaik dengan jarak total atau biaya yang paling minimum.

2.2 Vehicle Routing Problem

Vehicle Routing Problem adalah sebuah varian dari Traveling Salesman Problem yang diinvestigasi lebih luas. Model dasar dari Vehicle Routing Problem adalah Traveling Salesman Problem, dimana sebuah kendaraan bergerak dari titik awal dan akan kembali ke titik awal tersebut [3]. Diketahui sebuah jaringan $G = (N, L)$ dengan N menunjukkan sekumpulan $N = \{0, 1, \dots, n\}$ dan

$L = \{(i, j); i, j \in N, i \neq j\}$ adalah himpunan busur. Titik O mengindikasikan titik awal dengan sejumlah kendaraan NV . Jarak matrik $D =$ yang ditentukan dalam busur L . Jika $d_{ij} = d_{ji}$ untuk seluruh (i, j) , maka masalah bisa tidak simetris, dimana busur tidak berarah. Permintaan pelanggan i dimana q_i dan jumlah permintaan pelanggan dalam satu rute tidak melebihi kapasitas Q_k kendaraan. Tujuan dasar dari Vehicle Routing Problem adalah untuk menentukan NV kendaraan dengan total jarak minimal [4].

2.3 Pemrograman Linier

Pemrograman linier [1] dikembangkan selama perang dunia kedua, ketika sistem memaksimalkan efisiensi pemanfaatan sumberdaya menjadi sangat penting untuk menghasilkan capaian terbaik dalam menghadapi berbagai permasalahan yang sangat kompleks. Saat ini pemrograman linier sangat membantu banyak organisasi bisnis dan perusahaan dengan memaksimalkan keuntungan melalui pemodelan matematika. Secara matematis pemrograman linier adalah teknik yang digunakan untuk fungsi memaksimalkan atau meminimalkan tujuan. Dengan kata lain melakukan optimisasi linier dengan berbagai batasan yang ditentukan, yaitu dalam rangka mengoptimalkan fungsi obyektif. Fungsi obyektif dan kendala harus deterministik dan dapat diekspresikan dalam bentuk linier.

2.4 PuLP Python

Pemrograman linier banyak digunakan untuk mengalokasikan sumber daya dalam mencapai sebuah objektif. Dalam kasus VRP ini sebagai sebuah permasalahan routing akan diselesaikan dengan menggunakan python, bahasa pemrograman python adalah interpreter, interaktif, berorientasi objek dan merupakan bahasa tingkat tinggi yang memudahkan dalam penggunaannya. Penggunaan paket PuLP juga bertujuan untuk memudahkan dalam memodelkan pertidaksamaan dalam pemrograman linear,

PuLP sendiri berasal dari Python Programming Linear. PuLP sendiri adalah pemodelan untuk Linear Programming dan merupakan perangkat lunak open source yang ditulis dalam bahasa python, biasanya digunakan untuk menggambarkan masalah optimasi sebagai masalah matematika. PuLP juga dapat memanggil berbagai pemodelan solver external LP lainnya misalnya GPLK, CPLEX, Gurobi serta untuk menampilkan solusi yang dihasilkan biasanya menggunakan library Numpy dan Matplotlib untuk menampilkan area yang fisibel dalam bentuk grafik [2].

2.5 Q9 Sameday Delivery Pos Indonesia

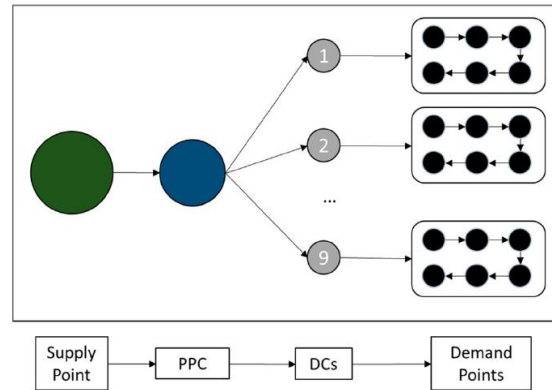
Q9 Sameday Service adalah layanan yang diberikan oleh Pos Indonesia dalam mendukung pelaku bisnis online dan usaha kecil dan menengah berupa pengiriman dagangan online, paket, dokumen, dan surat dengan fitur kecepatan pengiriman. Layanan tersedia khusus di Pulau Jawa dan ibukota propinsi diluar Pulau Jawa dengan batas waktu pengiriman maksimal 9 jam dan batas akhir pengiirman pukul 16.00 waktu setempat. Q9 memiliki beberapa keunggulan diantaranya memiliki tarif premium dihitung per-kg dengan berat maksimal 5 kg [2].

III. Studi Kasus

3.1 Postal Processing Centre (PPC)

Postal Processing Center (PPC) Bandung 40400 adalah bagian dari Pos Indonesia yang diantaranya memiliki fungsi pemrosesan, pendistribusian, dan pengantaran kiriman pos secara efektif dan efisien di wilayah kerjanya. Jaringan tersier adalah kepanjangan tangan dari PPC, dimana PPC melakukan pengiriman pendistribusian kiriman dari PPC ke delivery centre-nya (DC) di wilayah Bandung Raya, DC yang menjadi tujuan pengiriman pada jaringan tersier, yang secara terjadwal memiliki armada dan waktu pengiriman yang ditetapkan. Struktur jaringan yang ada di Bandung 40400 mengelola jaringan

distribusi yang terdiri dari 1 titik suplai biasanya berasal dari suplai poin dari kota lainnya baik melalui udara, laut, maupun darat, 1 Pusat Distribusi utama yang disebut Postal Processing Centre (PPC) dan 13 pusat distribusi perantara atau disebut Delivery Centre (DC), dan selanjutnya kantor pos atau agen pos, seperti terlihat pada gambar di bawah ini.



Gambar 1 Supply network structure, sumber [5]

Permasalahan yang muncul adalah pengiriman Q9 biasanya dilakukan diluar armada dan jadwal yang tersedia, dengan spesifikasi produk yang sudah dijelaskan di atas, fokus utama untuk pengiriman Q9 adalah waktu pengiriman.

3.2 Dataset

Delivery Centre adalah tujuan dan asal pengantaran produk Q9 Sameday Service, DC terdiri dari 13 lokasi.

DC_center	latitude	longitude
DC_SEKEJATI	-6.942732	107.650631
DC_SITUSAEUR	-6.936028	107.595377
DC_ASIAAFRIKA	-6.920710	107.606176
DC_KATAPANG	-7.028577	107.523677
DC_DAYEUEHKOLOT	-6.989556	107.626977
DC_CIPEDES	-6.894072	107.590046
DC_CIKUTRA	-6.899788	107.641506
DC_CIKERUH	-6.874496	107.600817
DC_LEMBANG	-6.811667	107.616643
DC_MAJALAYA	-7.048117	107.759724
DC_PADALARANG	-6.882824	107.503859
DC_CIMAH	-6.885874	107.538212
DC_UJUNGBERUNG	-6.913661	107.694689

Tabel 1 Lokasi Delivery Centre

Dalam hal ini DC_Sekejati sebagai PPC, sehingga dalam proses pengantaran sebagai titik awal.

Setiap DC akan dipetakan dalam jarak antar PPC dengan DC maupun antar DC sesuai dengan jarak yang dihitung berdasarkan jarak berdasarkan google map.

DC_SEKEJATI	DC_SITUSAEUR	DC_ASIAAFRIKA	DC_KATAPANG	DC_DAYEUKHOLOT	DC_CIPEDES	DC_CIKUTRA	DC_CIKERUH	DC_LEMBANG	DC_MAJALAYA
0.00	6.90	5.90	11.20	5.60	12.20	5.26	12.60	18.30	7.25
6.90	0.00	4.82	6.20	9.24	6.12	5.22	19.00	10.20	11.70
5.90	4.82	0.00	10.80	8.98	7.46	2.50	9.40	13.70	11.48
11.20	6.20	10.80	0.00	8.90	12.12	11.22	23.10	16.20	2.10
5.60	9.24	8.98	8.90	0.00	15.36	8.46	17.70	18.46	2.10
12.20	6.12	7.46	12.12	15.36	0.00	8.54	22.20	10.00	17.46
5.26	5.22	2.50	11.22	8.46	8.54	0.00	12.56	13.44	10.94
12.60	19.00	9.40	23.10	17.70	22.20	12.56	0.00	26.46	14.40
18.30	10.20	13.70	16.20	18.46	10.00	13.44	26.46	0.00	21.20
8.80	11.70	11.48	12.40	2.10	17.46	10.94	14.40	21.20	0.00
13.70	7.70	10.44	8.40	15.90	13.10	13.66	25.70	16.20	17.80
14.40	9.00	8.34	11.50	16.44	8.80	11.12	20.34	15.00	16.40
8.32	11.48	6.66	19.32	13.92	13.72	4.16	8.50	19.50	13.62

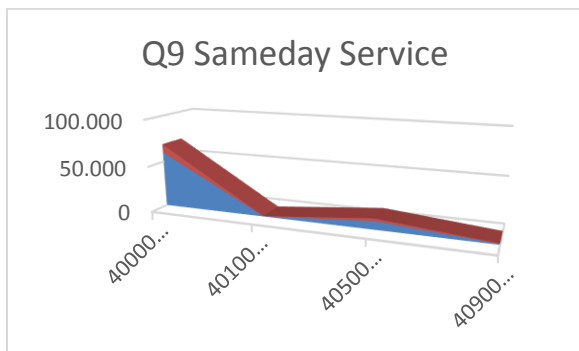
Tabel 2 Jarak PPC – DC dan antar DC

Sesuai dengan jarak antar DC juga dihitung berapa waktu tempuh yang diperlukan baik dari PPC ke DC maupun antara DC.

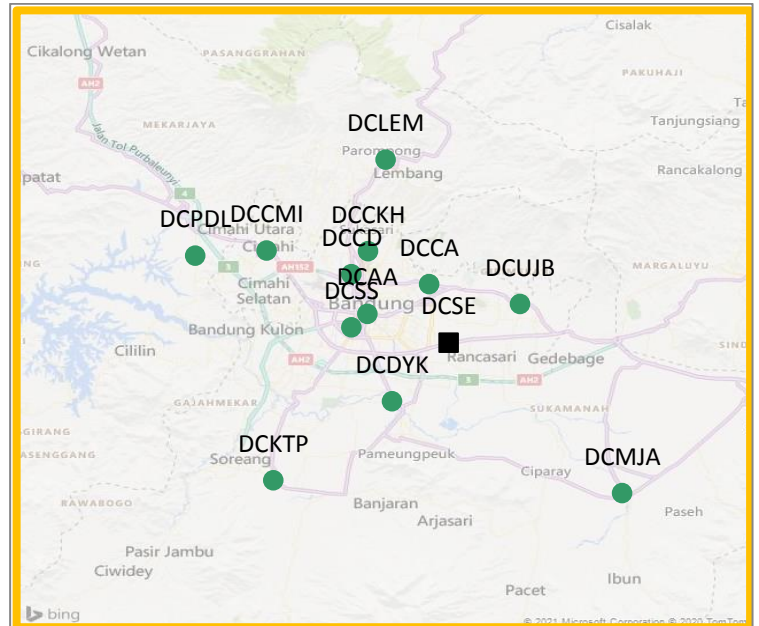
DC_SEKEJATI	DC_SITUSAEUR	DC_ASIAAFRIKA	DC_KATAPANG	DC_DAYEUKHOLOT	DC_CIPEDES	DC_CIKUTRA	DC_CIKERUH	DC_LEMBANG	DC_MAJALAYA	DC_PADALARANG	DC_CIMAH	DC_UJUNGBERUNG
0.000	0.314	0.281	0.457	0.271	0.491	0.259	0.504	0.694	0.314	0.281	0.457	0.271
0.314	0.000	0.245	0.291	0.392	0.288	0.258	0.717	0.424	0.281	0.245	0.444	0.383
0.281	0.245	0.000	0.444	0.383	0.333	0.167	0.397	0.541	0.457	0.291	0.444	0.000
0.457	0.291	0.444	0.000	0.381	0.488	0.458	0.854	0.624	0.271	0.392	0.383	0.381
0.271	0.392	0.383	0.381	0.000	0.596	0.366	0.674	0.699	0.491	0.288	0.333	0.488
0.491	0.288	0.333	0.488	0.596	0.000	0.369	0.824	0.417	0.259	0.258	0.167	0.458
0.259	0.258	0.167	0.458	0.366	0.369	0.000	0.503	0.532	0.504	0.717	0.397	0.541
0.504	0.717	0.397	0.541	0.674	0.824	0.503	0.000	0.966	0.694	0.424	0.541	0.624
0.694	0.424	0.541	0.624	0.824	0.503	0.000	0.966	0.000	0.377	0.474	0.467	0.497
0.377	0.474	0.467	0.497	0.154	0.666	0.449	0.564	0.791	0.541	0.341	0.432	0.364
0.541	0.341	0.432	0.364	0.614	0.521	0.539	0.941	0.691	0.564	0.384	0.362	0.467
0.564	0.384	0.362	0.467	0.832	0.377	0.455	0.762	0.584	0.361	0.467	0.306	0.728
0.361	0.467	0.306	0.728	0.548	0.541	0.223	0.367	0.734				

Tabel 1 Waktu Tempuh PPC-DC dan antar DC

Selanjutnya dipetakan ke dalam plot sesuai dengan tabel lokasi.



Gambar 3 Produksi Q9 Sameday Service untuk bulan Februari 2020, sumber P3M PPC.



Gambar 2 Peta Lokasi Delivery Centre

IV. Pemodelan

4.1 Model Matematika

Untuk memodelkan pendistribusian Q9 Sameday Service menggunakan TSP yang dimodifikasi yang mendukung perjalanan beberapa salesman dan dikembangkan menjadi VRP, dalam studi kasus ini adalah kendaraan kurir, dimana setiap Delivery Centre akan dikunjungi sekali kecuali untuk PPC yang mungkin dapat dikunjungi beberapa kali.

Dimana:

K adalah jumlah kendaraan

Dengan beberapa variabel yang ditetapkan:

$$x_{ij} = \begin{cases} 1 & \text{rute yang dilewati dari DC } i \text{ to } j \\ 0 & \text{jika tidak} \end{cases}$$

Variabel tiruan untuk perintah

$$u_i - u_j + C * x_{ij} \leq C - d_j, C$$

$$= \frac{N}{K} \begin{cases} u_i & \text{perintah bahwa DC } i \text{ dikunjungi} \\ d_j & \text{cost untuk mengunjungi DC } j, 0 \leq u_i \leq C - d_j, \forall i \in V \setminus \{0\} \end{cases}$$

Tujuan

$$\text{Min} \sum_{i=0}^n \sum_{j \neq i, j=0}^n c_{ij} x_{ij} \begin{cases} c_{ij}: \text{jarak dari DC } i \text{ ke DC } j \\ x_{ij}: \text{rute antara } i \text{ dan } j \end{cases}$$

Batasan

$$\begin{aligned} \sum_{i \in V} x_{ij} &= 1, \forall j \in V \setminus \{0\} \\ \sum_{j \in V} x_{ij} &= 1, \forall i \in V \setminus \{0\} \\ \sum_{i \in V} x_{i0} &= K \\ \sum_{j \in V} x_{0j} &= K \end{aligned}$$

4.2 Implementasi

Implementasi menggunakan platform google collab, dengan python dan paket pulp, matplotlib, seaborn, numpy, dan pandas. Program sumber diadaptasi dari [6].

Memetakan plot untuk DC:

Menentukan masalah

```
prob=LpProblem("vehicle", LpMinimize)
```

Menentukan objektif

```
cost = lpSum([x[(i,j)]*distances[(i,j)] for (i,j) in distances])
prob+=cost
```

Menetapkan batasan

```
for k in sites:
    cap = 1 if k != 'DC_SEKEJATI'
else K
    #inbound connection
    prob+= lpSum([ x[(i,k)] for i
in sites if (i,k) in x]) ==cap
    #outbound connection
    prob+=lpSum([ x[(k,i)] for i i
n sites if (k,i) in x]) ==cap
```

Eliminasi Subtour

```
N=len(sites)/K
```

```
for i in sites:
    for j in sites:
        if i != j and (i != 'DC_SE
KEJATI' and j!= 'DC_SEKEJATI') and
(i,j) in x:
            prob += u[i] - u[j] <=
(N)*(1-x[(i,j)]) - 1

%time prob.solve()

totalTime = 0;
for t in tours:
    times = 0
    for i in range(0, len(t)):
        times += time.loc[t[i][0],
t[i][1]]
        if times > totalTime:
            totalTime = times
print(totalTime)
```

4.3 Hasil dan Pembahasan

Dari hasil menjalankan program dengan dataset yang tersedia didapatkan hasil sebagai berikut:

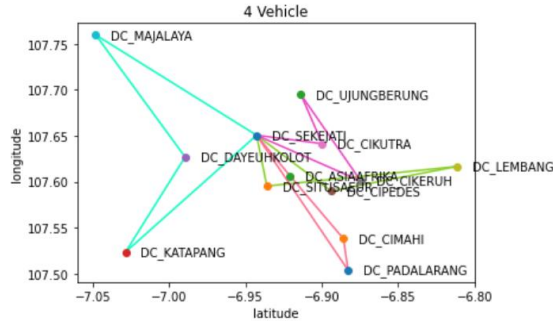
Jumlah kendaraan yang digunakan = 4

Longest time spent: 1.646 (hrs)

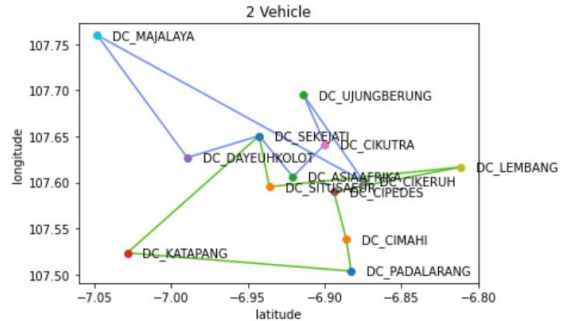
Total distance: 136.11 (km)

Dengan rute yang terbentuk:

```
DC_SEKEJATI -> DC_CIPEDES ->
DC_LEMBANG -> DC_SITUSAEUR ->
DC_SEKEJATI
DC_SEKEJATI -> DC_CIKERUH ->
DC_UJUNGBERUNG -> DC_CIKUTRA ->
DC_SEKEJATI
DC_SEKEJATI -> DC_MAJALAYA ->
DC_DAYEUHKOLOT -> DC_KATAPANG ->
DC_SEKEJATI
DC_SEKEJATI -> DC_PADALARANG ->
DC_CIMAHI -> DC_ASIAAFRIKA ->
DC_SEKEJATI
```



Gambar 3 Rute dengan 4 kendaraan



Gambar 3 Rute dengan 2 kendaraan

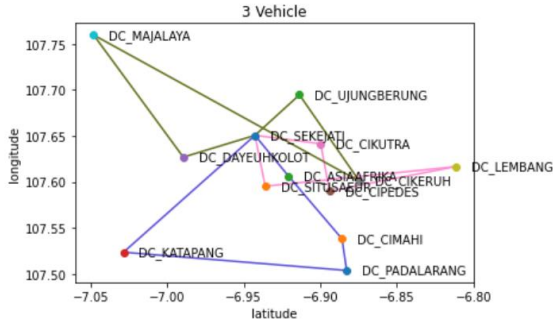
Jumlah kendaraan yang digunakan = 3

Longest time spent: 1.845 (hrs)

Total distance: 122.56 (km)

Dengan rute yang terbentuk:

- DC_SEKEJATI -> DC_SITUSAEUR ->
- DC_LEMBANG -> DC_CIPEDES ->
- DC_CIKUTRA -> DC_SEKEJATI
- DC_SEKEJATI -> DC_KATAPANG ->
- DC_PADALARANG -> DC_CIMPAHI ->
- DC_ASIAAFRIKA -> DC_SEKEJATI
- DC_SEKEJATI -> DC_UJUNGBERUNG ->
- DC_CIKERUH -> DC_MAJALAYA ->
- DC_DAYEUKOLOT -> DC_SEKEJATI



Gambar 3 Rute dengan 3 kendaraan

Jumlah kendaraan yang digunakan = 2

Longest time spent: 2.734 (hrs)

Total distance: 107.56 (km)

Dengan rute yang terbentuk:

- DC_SEKEJATI -> DC_KATAPANG ->
- DC_PADALARANG -> DC_CIMPAHI ->
- DC_CIPEDES -> DC_LEMBANG ->
- DC_SITUSAEUR -> DC_SEKEJATI
- DC_SEKEJATI -> DC_DAYEUKOLOT ->
- DC_MAJALAYA -> DC_CIKERUH ->
- DC_UJUNGBERUNG -> DC_CIKUTRA ->
- DC_ASIAAFRIKA -> DC_SEKEJATI

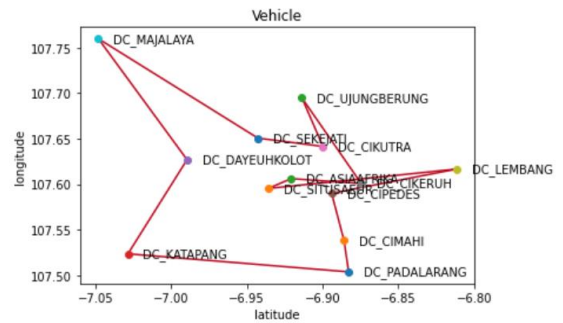
Jumlah kendaraan yang digunakan = 1

Longest time spent: 4.315 (hrs)

Total distance: 96.69 (km)

Dengan rute yang terbentuk:

- DC_SEKEJATI -> DC_MAJALAYA ->
- DC_DAYEUKOLOT -> DC_KATAPANG ->
- DC_PADALARANG -> DC_CIMPAHI ->
- DC_CIPEDES -> DC_LEMBANG ->
- DC_SITUSAEUR -> DC_ASIAAFRIKA ->
- DC_CIKERUH -> DC_UJUNGBERUNG ->
- DC_CIKUTRA -> DC_SEKEJATI



Gambar 3 Rute dengan 1 kendaraan

Secara umum pola distribusi terbaik yang dapat dilakukan dengan memanfaatkan 4 kendaraan dengan waktu tempuh terpanjang 1,646 jam dengan jarak 136,11 km.

V. Kesimpulan dan Saran

5.1 Kesimpulan

Pemanfaatan VRP yang dikembangkan dari TSP dilakukan untuk melakukan optimasi kiriman Q9 Sameday Service yang fokus pada pengiriman tepat waktu dalam 9 jam. Melihat dari jumlah produksi yang masih kecil, tidak memperhitungkan

kapasitas kendaraan, secara umum PPC mengijinkan Less Than Truckload (LTL) dalam melakukan distribusi.

5.2 Saran

Jika pada waktunya dengan jumlah produksi meningkat diperlukan model optimasi yang memperhitungkan kapasitas kendaraan dan permintaan dari DC, dengan memanfaatkan model optimasi yang sesuai.

Daftar Pustaka

- [1] C. M. Annur, "Prospeknya Cerah, Bisnis Logistik Diprediksi Tumbuh Lebih 30% di 2020," *Katadata*, 2019. <https://katadata.co.id/sortatobing/berita/5e9a50d967cca/prospeknya-cerah-bisnis-logistik-diprediksi-tumbuh-lebih-30-di-2020>.
- [2] P. Indonesia, "Q9 Sameday Service," *Pos Indonesia*, 2020. <https://www.posindonesia.co.id/id/content/q9-sameday-service>.
- [3] M. M. Bidgoli and A. S. Kheirkhah, "An arc interdiction vehicle routing problem with information asymmetry," *Comput. Ind. Eng.*, vol. 115, pp. 520–531, 2018, doi: 10.1016/j.cie.2017.11.019.
- [4] K. Belachgar, "Vehicle Routing Problem with Dinstance Constraints and Clustering," Al Akhwayn University, 2017.
- [5] G. Timperio, S. Tiwari, J. M. Gaspar Sánchez, R. A. García Martín, and R. de Souza, "Integrated decision support framework for distribution network design," *Int. J. Prod. Res.*, vol. 58, no. 8, pp. 2490–2509, Apr. 2020, doi: 10.1080/00207543.2019.1680894.
- [6] JWang, "Vehicle-routing-problem," *github*, 2019. <https://github.com/jwang0306/vehicle-routing-problem/blob/master/vrp.ipynb>.