

# ALGORITMA SWEEP – LOCAL SEARCH PADA CVRP DENGAN PEMROGRAMAN PYTHON (STUDY KASUS : DISTRIBUSI SURAT KABAR)

**Riska Indrayana**

Program Studi Matematika, FMIPA, Universitas Negeri Malang,  
\*email : [riska.indrayana.2003126@students.um.ac.id](mailto:riska.indrayana.2003126@students.um.ac.id)

**Asmianto**

Program Studi Matematika, FMIPA, Universitas Negeri Malang,  
email : [asmianto.fmipa@um.ac.id](mailto:asmianto.fmipa@um.ac.id)

## Abstrak

Surat kabar adalah salah satu sumber informasi yang masih sangat diminati oleh masyarakat meskipun perkembangan teknologi digital telah merambah berbagai aspek kehidupan termasuk surat kabar elektronik. Tantangan utama dalam menjalankan proses distribusi surat kabar adalah menentukan rute perjalanan terpendek untuk mengoptimalkan waktu dan jarak tempuh dengan mempertimbangkan kapasitas setiap kendaraan yang biasa disebut *Capacitated Vehicle Routing Problem* (CVRP). Solusi pada permasalahan tersebut yaitu menggunakan algoritma *Sweep – Local Search* untuk menentukan rute terpendek yang optimal. Tujuan dari dilakukannya penelitian ini adalah untuk menerapkan algoritma *Sweep – Local search* dalam penentuan rute terpendek yang dilakukan di Jawa Pos Radar Malang untuk distribusi surat kabar harian. Metode penelitian yang digunakan yaitu menggunakan pendekatan kuantitatif deskriptif dengan pendekatan lapangan. Penelitian dilakukan menggunakan perhitungan dengan alat bantu *Python*. Tahapan algoritma *Sweep – Local Search* yaitu mengubah koordinat kartesius menjadi koordinat polar, melakukan *clustering*, pengoptimalan menggunakan algoritma *Local Search* dengan pertukaran dua agen dalam satu rute yang sama hingga mendapatkan hasil yang optimal. Hasil penentuan rute terpendek diperoleh dua *clustering* sesuai dengan kapasitas kendaraan yang menempuh total jarak 142 Km dengan waktu tempuh 117 menit dan 19,5 Km dengan waktu tempuh 21,4 menit. Dari proses perhitungan didapatkan hasil rute yang lebih optimal dibandingkan rute yang diterapkan perusahaan.

**Kata Kunci:** Surat Kabar, CVRP, Rute Terpendek, Algoritma *Sweep – Local Search*

## Abstract

Newspapers are one source of information that is still very much in demand by the public even though the development of digital technology has penetrated various aspects of life including electronic newspapers. The main challenge in carrying out the newspaper distribution process is determining the shortest travel route to optimize travel time and distance by considering the capacity of each vehicle commonly called the *Capacitated Vehicle Routing Problem* (CVRP). The solution to the problem is to use the *Sweep - Local Search* algorithm to determine the optimal shortest route. The purpose of this research is to apply the *Sweep - Local search* algorithm in determining the shortest route conducted at Jawa Pos Radar Malang for daily newspaper distribution. The research method used is using a descriptive quantitative approach with a field approach. The research was conducted using calculations with Python tools. The stages of the *Sweep - Local Search* algorithm are converting Cartesian coordinates into polar coordinates, clustering, and optimization using the *Local Search* algorithm by exchanging two agents in the same route until getting optimal results. The results of determining the shortest route obtained two clustering according to the capacity of the vehicle which covered a total distance of 142 Km with a travel time of 117 minutes and 19.5 Km with a travel time of 21.4 minutes. From the calculation process, the results of the route are more optimal than the route applied by the company.

**Keywords:** Newspaper, CVRP, Shortest Route, *Sweep - Local Search* Algorithm

Figure 1.

## PENDAHULUAN

Surat kabar adalah salah satu sumber informasi yang masih sangat diminati oleh masyarakat. Meskipun perkembangan teknologi digital telah merambah berbagai aspek kehidupan, distribusi surat kabar masih menjadi bagian penting dari rutinitas harian masyarakat. Untuk menjalankan proses distribusi surat kabar, transportasi sangat

penting bagi perusahaan. Salah satu tantangan utama yang sering dihadapi adalah menentukan rute perjalanan terpendek untuk distribusi surat kabar. Hal itu dilakukan untuk mengurangi waktu tempuh dalam pengantaran surat kabar ke sejumlah agen agar bisa minimum.

Permasalahan tentang pengoptimalan rute distribusi barang dengan mempertimbangkan kapasitas kendaraan, biasa disebut dengan

*Capacitated Vehicle Routing Problem (CVRP)*. CVRP termasuk dalam kategori NP-Hard problem yang berarti secara teori maupun praktik pada dunia nyata memiliki permasalahan yang sangat banyak dan kompleks sehingga sulit untuk dipecahkan secara teoritis dan praktis. Permasalahan ini bisa diselesaikan dengan dua metode, yaitu metode metaheuristik dan metode konvensional (Toth & Vigo, 2002).

Perkembangan CVRP pada kehidupan sehari-hari yang bermanfaat yaitu pada persoalan distribusi barang. Salah satu perusahaan yang selalu menyalurkan barang setiap harinya dengan jumlah permintaan barang pada setiap agennya berbeda yaitu perusahaan surat kabar Radar Malang.

Radar Malang merupakan bagian dari Jawa Pos Group. Radar Malang berdiri pada tanggal 15 Desember 1999 dan telah berkembang pesat selama delapan belas tahun. Perusahaan ini merupakan media terbesar di Kota Malang dari 15 Radar di Jawa Timur. Saat ini, Radar Malang menyampaikan berita dari seluruh wilayah Malang Raya, yaitu Kota Malang, Kabupaten Malang, dan Kota Batu. Radar Malang juga saat ini merupakan koran lokal terbesar di daerah Malang Raya, dengan penjualan harian rata-rata sebesar 15.000 eksemplar.

Dengan begitu, Pasti dalam proses distribusi surat kabar Radar Malang ada banyak rute yang dilalui perusahaan, sehingga perusahaan harus tepat dalam menentukan rute yang optimal agar proses distribusi memiliki waktu yang lebih baik dan efisien.

Permasalahan *Capacitated Vehicle Routing Problem* sudah banyak dilakukan penelitian oleh beberapa peneliti seperti yang dilakukan oleh Ega Arifta, Fibri Rakhmawati (2023). Penulis menyelesaikan permasalahan CVRP untuk penentuan rute kendaraan distribusi buku menggunakan algoritma *Sweep* didapatkan hasil jarak yang lebih optimal dari rute sebelumnya yaitu sebesar 35,35%. Selanjutnya, penelitian yang dilakukan oleh Arunya Boonkleaw, Nanthi Suthikarnnarunai, PhD., Rawinkhan Srinon, PhD. (2017). Peneliti menyelesaikan permasalahan VRP dengan menggunakan algoritma *Sweep* untuk menentukan rute distribusi koran di mana menghasilkan waktu tempuh lebih cepat sebesar 41,5% dibandingkan rute yang diterapkan oleh Perusahaan *Morning Newspaper*, Bangkok, Thailand. Dan yang terakhir, penelitian yang dilakukan oleh Muhammad Ruben, Arif Imran (2020). Penelitian

yang mereka lakukan yaitu usulan rute distribusi menggunakan algoritma *Sweep* dan *Local Search* pada Industri Makanan Sizi dengan mendapatkan solusi penyelesaian permasalahan yaitu mengurangi total jarak sebesar 27,923 km atau sebesar 15%.

Penyelesaian masalah CVRP dapat menggunakan berbagai metode yaitu Algoritma Penyebaran (*Sweep Algorithm*), Algoritma Koloni Semut, *Local Search*, *Tabu Search Algorithm*, *African Buffalo Optimization (ABO)* dan sebagainya. Untuk menyelesaikan permasalahan CVRP pada distribusi surat kabar Radar Malang, dilakukan analisis rute dengan menggunakan metode Algoritma Penyebaran (*Sweep Algorithm*) - *Local Search* untuk mencari rute terpendek yang optimal dalam pendistribusian surat kabar karena Algoritma Penyebaran (*Sweep Algorithm*) memiliki kelebihan yaitu dapat menghasilkan solusi yang tepat dengan keadaan yang sebenarnya di lapangan untuk permasalahan rute optimasi, dan *Local Search* dapat memperbaiki dari hasil solusi sebelumnya dengan melakukan pengubahan urutan rute atau modifikasi lainnya.

Metode Algoritma Penyebaran (*Sweep Algorithm*) memiliki tahapan yaitu dengan mengelompokkan agen, membuat koordinat kartesius, mengubah koordinat kartesius menjadi koordinat polar, membuat kluster, dan penerapan *Local Search* untuk memperbaiki solusi awal dengan eksplorasi perubahan dalam rute seperti *1- insertion intra route* yang akan digunakan dalam penelitian ini dalam mencari rute distribusi surat kabar Radar Malang.

## KAJIAN TEORI

### A. Distribusi

Distribusi merupakan suatu bentuk aktivitas pemasaran dalam proses pengiriman produk dari produsen kepada pelanggan atau agen dengan waktu yang tepat dan cepat (Fatnita & Lukmandono, 2020). Kegiatan distribusi sangat penting dalam pemasaran di suatu perusahaan. Hal ini disebabkan oleh fakta bahwa tanpa distribusi, produk tidak akan bisa dengan mudah sampai ke pelanggan baik dari wilayah yang mudah terjangkau hingga wilayah - wilayah yang kurang terjangkau, karena jika tidak ada proses distribusi maka barang tidak akan sampai dengan mudah ke para agen atau konsumen (Fauziah dkk., 2022).

Semua aspek pengiriman produk kepada agen termasuk dalam distribusi. Distribusi juga dapat dianggap sebagai bagian dari material *handling* karena material *handling* berpindah setiap saat. Berikut, merupakan beberapa masalah yang dihadapi selama proses distribusi yang harus diperbaiki untuk meningkatkan jaringan distribusi (Nono dkk., 2020).

1. Titik depot memastikan proses pendistribusian produk berjalan dengan lancar, sehingga produk dapat sampai ke agen tepat pada waktunya.
2. Keputusan yang paling penting dalam manajemen distribusi adalah penentuan jadwal pengiriman dan rute pengiriman produk dari satu lokasi ke lokasi tujuan yang lain.

Perusahaan yang melakukan pengiriman produk harus membuat keputusan seperti ini. Biaya pengiriman akan sangat dipengaruhi oleh keputusan tentang jadwal pengiriman dan rute yang akan ditempuh untuk setiap jenis kendaraan. Namun, banyak faktor yang harus dipertimbangkan juga selama proses pengiriman. Rute dan jadwal seringkali harus mempertimbangkan beberapa masalah tambahan, seperti armada transportasi dan kapasitas kendaraan masing - masing. Permasalahan penjadwalan atau penentuan rute pengiriman harus mencapai beberapa tujuan, seperti meminimalkan waktu atau jarak tempuh dan juga biaya pengiriman.

#### B. *Capacitated Vehicle Routing Problem (CVRP)*

*Capacitated Vehicle Routing Problem (CVRP)* adalah salah satu bentuk yang sangat umum dari permasalahan VRP mengenai penentuan rute yang optimal dengan keterbatasan kapasitas kendaraan tertentu. Setiap kendaraan melakukan distribusi dari depot ke setiap wilayah agen sesuai dengan permintaan barang dari masing - masing agen tersebut (Desaulniers dkk., 2014). Dalam CVRP, Transportasi terdiri dari distribusi barang yang berasal dari depot dilambangkan sebagai titik 0, untuk dikirim ke sekumpulan  $n$  titik lainnya, biasanya disebut sebagai agen  $N = \{1, 2, \dots, n\}$ . Jumlah yang harus dikirimkan ke agen  $i \in n$  atau sebagai permintaan agen, untuk berat barang yang

harus dikirim  $q_i \geq 0$  sesuai kapasitas setiap armadanya. Sehingga  $K = \{1, 2, \dots, |K|\}$  dimana  $K$  merupakan kendaraan yang diasumsikan homogen, berarti  $|K|$  kendaraan tersedia di depot. (Desaulniers dkk., 2014). Tujuan dari permasalahan CVRP yaitu untuk mengurangi total jarak tempuh rute distribusi barang dari depot ke setiap agennya. Dari tujuan CVRP adapun cara perhitungan jarak dibagi menjadi dua :

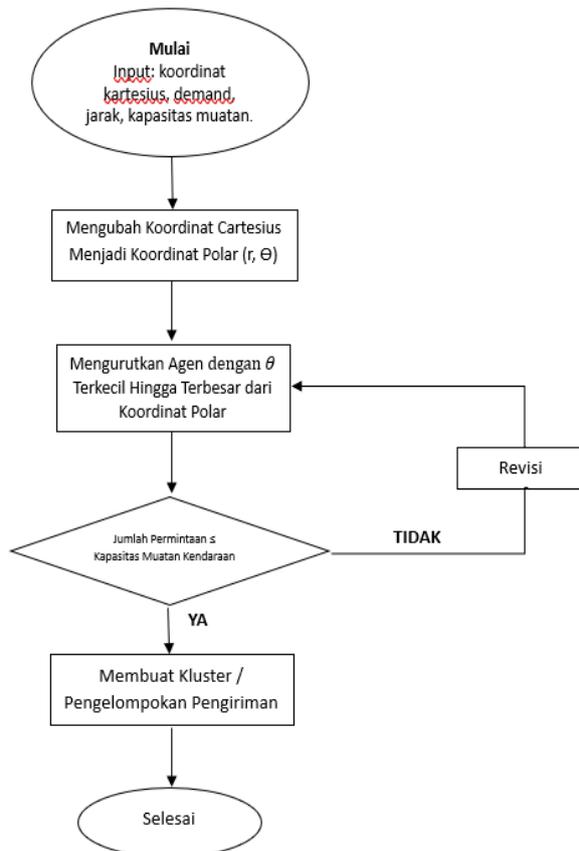
1. Jarak simetris (*Symmetric distance*)  
Yaitu konsep perhitungan jarak dengan menganggap besarnya jarak dari kota M ke kota N sama besarnya dengan jarak dari kota N ke kota M.
2. Jarak tidak simetris (*Asyymetric distance*)  
Merupakan kebalikan dari jarak simetris, perhitungan jarak dengan mengasumsikan besarnya jarak dari kota M ke kota N tidak sama dengan jarak kota N ke kota M.

#### C. Algoritma Penyebaran (*Sweep Algorithm*)

Algoritma Penyebaran (*Sweep Algorithm*) adalah Algoritma yang dirancang dengan khusus untuk pengelompokan dengan beberapa implementasi dimana simpul dipertukarkan antara *cluster* yang berdekatan, dan rute dioptimalkan kembali (Desaulniers dkk., 2014). Algoritma penyebaran melakukan pengelompokan dengan tahapan pengelompokan agen yang disebut *nodes*, dengan menggabungkan agen - agen dalam satu kelompok atau *cluster* dengan batas jumlah maksimal pada setiap kendaraan yang digunakan (Boonkleaw dkk., 2009).

Manfaat dari algoritma Sweep dalam penelitian ini yaitu sebagai tahapan pertama dalam pengelompokan agen yang relatif sederhana dan mudah diimplementasikan karena dalam pengurutannya berdasarkan sudut atau koordinat sehingga dapat menciptakan rute yang saling berdekatan agar efisien dalam pengiriman barangnya.

Tahapan - tahapan untuk menyelesaikan permasalahan dengan algoritma penyebaran (*Sweep Algorithm*) ditunjukkan dengan diagram alir dibawah ini :

Gambar 1. Diagram Alir *Sweep Algorithm*

Dalam mengubah koordinat kartesius  $(x,y)$  menjadi koordinat polar dilakukan dengan rumus sebagai berikut :

$$r = \sqrt{x^2 + y^2} \quad (1)$$

$$\theta = \arctan \frac{y}{x} \quad (2)$$

Keterangan :

$x$  = Nilai *Latitude*

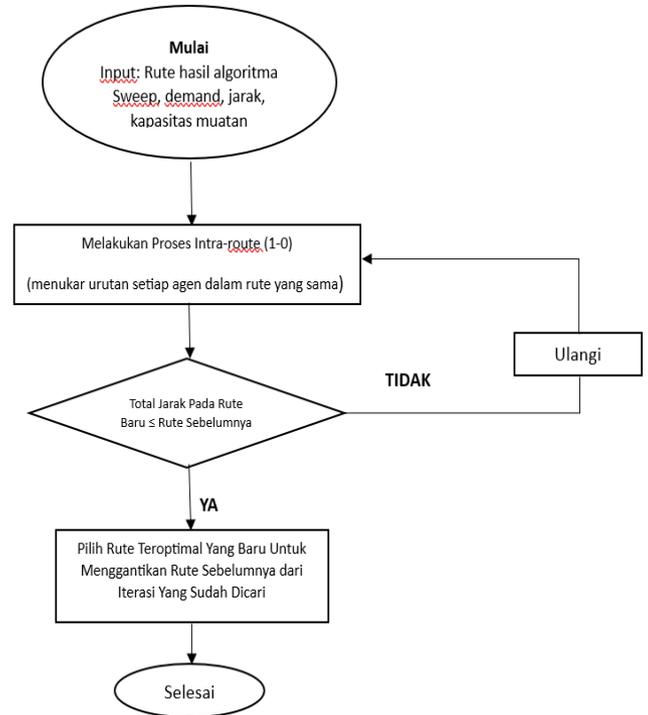
$y$  = Nilai *Longitude*

#### D. *Local Search*

*Local Search* adalah metode heuristik yang penyelesaiannya menggunakan beberapa cara kombinasi dari teknik optimasi (Toth & Vigo, 2002). Metode *Local Search* mencari di dalam ruang yang sama dengan solusi kandidat (yang mungkin dibuat secara acak) bergerak ke "tetangga" yang "lebih baik" dan berhenti ketika semua tetangga lebih rendah dari solusi sebelumnya (Gendreau & Potvin, 2010). *Local search* Dalam penelitian ini menggunakan salah satu operator dari *local search* yaitu *Insertion Intra-Route (1-0)* yang merupakan proses perbaikan rute dilakukan dengan cara

memindahkan satu titik agen ke titik agen lainnya dalam satu rute yang sama.

Tahap – tahap untuk memperbaiki rute yang sudah ada dari penyelesaian sebelumnya agar lebih optimal dengan algoritma *local search* menggunakan operator *Insertion Intra-Route (1-0)* ditunjukkan dengan diagram alir di bawah ini :

Gambar 2. Diagram Alir *Local Search*

## METODE

Penelitian ini menggunakan pendekatan kuantitatif deskriptif dengan pendekatan lapangan. Pendekatan ini dilakukan untuk menggambarkan karakteristik rute distribusi dalam pencarian rute terpendek pada situasi yang nyata di perusahaan surat kabar Radar Malang. Jenis penelitian ini difokuskan dengan mengumpulkan data secara langsung yaitu dengan cara melakukan observasi langsung ke perusahaan, wawancara dengan manajer logistik, serta pengambilan data terkait sistem distribusi surat kabar secara langsung, dan melakukan pengamatan jalur dengan menggunakan *Google maps*.

Populasi dalam penelitian ini terdapat 16 Agen untuk distribusi surat kabar yang lokasinya berada di Malang Raya. Pada penelitian ini terdapat beberapa tahapan dalam proses penyelesaiannya yaitu dengan

melakukan pengumpulan data yang didapat dari Perusahaan Radar Malang diikuti identifikasi permasalahan distribusi logistik serta membuat rumusan masalah dengan metode penyelesaiannya, kemudian dilakukan penerapan metode Algoritma *Sweep - Local Search* untuk menyelesaikan permasalahan agar mendapat rute teroptimal. Setelah itu dilakukan analisis hasil, jika hasil menunjukkan rute distribusi yang lebih optimal maka dilanjut untuk membuat kesimpulan, namun jika sebaliknya, dilakukan evaluasi ulang pada tahapan penerapan algoritma untuk mencapai solusi yang lebih optimal.

Dalam proses analisis data, penulis menggunakan alat bantu *python* dengan *Sweep Algorithm – Local Search* untuk menentukan rute teroptimal termasuk total jarak dan waktu tempuh. Dalam tahapan ini dilakukan tahapan awal dengan membentuk model matematis yang digunakan dalam penelitian ini yaitu sebagai berikut :

**Variabel Keputusan**

$$X_{ijk} = \begin{cases} 1, & \text{Jika terjadi perjalanan dari node } i \text{ ke node } j \\ & \text{oleh armada } k \\ 0, & \text{Jika tidak atau lainnya} \end{cases}$$

**Fungsi Tujuan**

$$\text{Min } Z = \sum_{i \in N} \sum_{j \in N, j \neq i} \sum_{k \in K} X_{ijk} D_{ij} \quad (1)$$

**Fungsi Kendala**

$$\sum_{i \in N} \sum_{k \in K} X_{ijk} = 1 \quad \forall j \in N, j \neq i, j \neq 1 \quad (2)$$

$$\sum_{j \in N} \sum_{k \in K} X_{ijk} = 1 \quad \forall i \in N, j \neq i, j \neq 1 \quad (3)$$

$$\sum_{i \in N} X_{ijk} - \sum_{j \in N} X_{ijk} = 0 \quad \forall k \in K \quad (4)$$

$$\sum_{i \in N} p_i \sum_{j \in N} X_{ijk} \leq w_k \quad \forall k \in K, i \neq j \quad (5)$$

$$X_{ijk} \in \{0,1\} \quad \forall i, j \in N, \forall k \in K \quad (6)$$

Keterangan :

**Himpunan**

N : Himpunan keseluruhan *node* termasuk depot, *drop out*, dan pelanggan

K : Himpunan dari keseluruhan armada

**Indeks**

- i : Indeks pelanggan i
- j : Indeks pelanggan j
- k : Indeks armada

**Parameter**

- $p_i$  : Banyaknya *demand* dari pelanggan i
- $D_{ij}$  : Jarak dari pelanggan i ke pelanggan j
- $w_k$  : Kapasitas maksimum dari armada k
- R : Bilangan Riil

Fungsi tujuan yang ingin dicapai terdapat pada persamaan (1) yaitu untuk meminimalkan jarak pengantaran. Batasan (2) untuk memastikan bahwa setiap pelanggan j hanya didatangi oleh 1 armada yang berasal dari salah satu agen i dan (3) untuk memastikan bahwa armada dari pelanggan i hanya menuju ke satu pelanggan j. Batasan (4) memastikan bahwa setiap kendaraan yang datang ke pelanggan i akan menuju ke pelanggan yang lain. Batasan (5) memastikan bahwa total permintaan pelanggan dalam satu rute tidak melebihi kapasitas dari armada. Dan Batasan (6) menunjukkan bahwa keputusan bernilai integer. N merupakan jumlah keseluruhan agen yaitu 17 agen dan K merupakan jumlah keseluruhan yaitu 2 armada

**HASIL DAN PEMBAHASAN**

Algoritma *Sweep-Local Search* pada CVRP dapat digunakan dalam pengelompokan atau *clustering* rute awal yang dilakukan dengan menggunakan algoritma *Sweep* setelah itu dilanjut proses pengoptimalan dan perbaikan rute menggunakan algoritma *Local Search*. Proses dari implementasi algoritma *Sweep-Local Search* pada CVRP dapat dijabarkan sebagai berikut :

**Input** : Depot (dilambangkan dengan kode A0), Agen (dilambangkan dengan kode A1,A2,...,A14), *Dropout* (dilambangkan dengan kode A\*\*), Jarak dari depot (Km), Jarak antar agen ke agen (Km), Permintaan setiap agen (eksemplar), Koordinat Polar ( $r, \theta$ ), Kecepatan setiap kendaraan (Km/jam), Kapasitas setiap kendaraan (eksemplar), Rute awal algoritma *Sweep*.

**Proses :**

1. Menentukan koordinat polar dengan mengubah koordinat kartesius.
2. Memasukkan kode agen, *demand*, jarak dari depot, serta koordinat polar untuk membuat *clustering*.
3. Membuat *cluster* awal menggunakan algoritma *Sweep* sesuai kapasitas kendaraan.
4. Memperbaiki rute yang telah diperoleh menggunakan algoritma *Local Search*.
5. Memasukkan data tambahan yaitu rute awal algoritma *Sweep* untuk dioptimalkan.
6. Membandingkan hasil optimasi algoritma *Local Search* dengan algoritma sebelumnya, hasil pengoptimalan diterima berdasarkan kriteria pengoptimalan rute yang lebih baik.

**Output :** Program yang telah dibuat akan menampilkan hasil dari rute yang terbentuk, total jarak tempuh, waktu tempuh, visualisasi *graph* dari rute yang diperoleh.

Untuk langkah – langkah yang lebih lengkap dapat dilihat pada penerapan masing – masing algoritma selanjutnya.

Desain *system* pada penelitian ini dengan menggunakan *Google Colab* dalam bahasa pemrograman *Python*. Dari penelitian yang telah dilakukan terdapat beberapa *coding* yang berbeda dalam pemrosesannya sesuai ketentuan setiap algoritmanya.

```
[ ] !import numpy as np

# Data
agen_code = ['A0', 'A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7', 'A8', 'A9', 'A10', 'A11', 'A12', 'A13', 'A14', 'A**']
demand = [0, 552, 308, 48, 192, 2400, 1020, 576, 1272, 1200, 264, 240, 480, 48, 360, 1200]
distance_from_depot = [0, 83.8, 97.8, 98.9, 99.2, 101, 102, 109, 105, 105, 107, 111, 114, 118, 126, 105]
angles = [0, 93.97434467704669, 94.02792898581497, 94.02810671447651, 94.03370202057215, 94.03858871754335, 94.04348144704669, 94.048144704669, 94.05280144704669, 94.057458144704669, 94.062115144704669, 94.0667728144704669, 94.0714305144704669, 94.0760882144704669, 94.0807459144704669, 94.0854036144704669, 94.0900613144704669, 94.0947190144704669, 94.0993767144704669, 94.1040344144704669, 94.1086921144704669, 94.1133498144704669, 94.1180075144704669, 94.1226652144704669, 94.1273229144704669, 94.1319806144704669, 94.1366383144704669, 94.1412960144704669, 94.1459537144704669, 94.1506114144704669, 94.1552691144704669, 94.1599268144704669, 94.1645845144704669, 94.1692422144704669, 94.1738999144704669, 94.1785576144704669, 94.1832153144704669, 94.1878730144704669, 94.1925307144704669, 94.1971884144704669, 94.2018461144704669, 94.2065038144704669, 94.2111615144704669, 94.2158192144704669, 94.2204769144704669, 94.2251346144704669, 94.2297923144704669, 94.2344500144704669, 94.2391077144704669, 94.2437654144704669, 94.2484231144704669, 94.2530808144704669, 94.2577385144704669, 94.2623962144704669, 94.2670539144704669, 94.2717116144704669, 94.2763693144704669, 94.2810270144704669, 94.2856847144704669, 94.2903424144704669, 94.2950001144704669, 94.2996578144704669, 94.3043155144704669, 94.3089732144704669, 94.3136309144704669, 94.3182886144704669, 94.3229463144704669, 94.3276040144704669, 94.3322617144704669, 94.3369194144704669, 94.3415771144704669, 94.3462348144704669, 94.3508925144704669, 94.3555502144704669, 94.3602079144704669, 94.3648656144704669, 94.3695233144704669, 94.3741810144704669, 94.3788387144704669, 94.3834964144704669, 94.3881541144704669, 94.3928118144704669, 94.3974695144704669, 94.4021272144704669, 94.4067849144704669, 94.4114423144704669, 94.4160999144704669, 94.4207576144704669, 94.4254153144704669, 94.4300730144704669, 94.4347307144704669, 94.4393884144704669, 94.4440461144704669, 94.4487038144704669, 94.4533615144704669, 94.4580192144704669, 94.4626769144704669, 94.4673346144704669, 94.4719923144704669, 94.4766500144704669, 94.4813077144704669, 94.4859654144704669, 94.4906231144704669, 94.4952808144704669, 94.5000000144704669]

# Constants
speed = 90 # Dengan asumsi kecepatan konstan dalam km/jam
depot_time = 0 # Waktu yang dihabiskan di depot untuk setiap rute (dalam jam)

# Hitung koordinat polar (jarak, sudut) untuk setiap pelanggan
polar_coordinates = list(zip(distance_from_depot, angles))

#Urutkan pelanggan berdasarkan sudut dalam urutan menaik
sorted_customers = [agen_code for _, agen_code in sorted(zip(angles, agen_code))]
```

Gambar 3. Tampilan Coding pada Google Colab Algoritma Sweep

```
# Membuat matriks jarak
distances = np.array([
    [0, 83.8, 97.8, 98.9, 99.2, 101, 102, 109, 105, 105, 107, 111, 114, 118, 126, 105],
    [83.8, 0, 13.6, 14.6, 14.9, 17.3, 17.8, 20.9, 21.3, 23.2, 22.9, 31.5, 35.3, 38.6, 47.2, 21],
    [97.8, 14.6, 0, 2.9, 1.6, 3.7, 4.3, 7.4, 7.5, 8.5, 9.4, 10.8, 16.5, 17.9, 26.4, 7.5],
    [98.9, 14.7, 2.5, 0, 3.5, 4.2, 6.7, 9.8, 10, 11, 11.9, 12.8, 18.5, 19.9, 28.4, 9.9],
    [99.2, 14.4, 2.1, 2.8, 0, 3.7, 3.8, 6.9, 6.4, 6.5, 7.4, 9.2, 14.9, 16.3, 24.8, 6.4],
    [101, 15.8, 2.9, 4.7, 2.4, 0, 2.1, 7.2, 7.2, 6.3, 7.2, 9.2, 14.9, 16.3, 24.8, 6.8],
    [102, 17.1, 4.2, 6.4, 3.6, 2, 0, 3.8, 4, 3.4, 4.4, 7.2, 12.9, 14.3, 22.8, 3.9],
    [109, 19.7, 6.8, 8.7, 5.9, 4.6, 3.2, 0, 0.4, 1, 2, 5.5, 11.2, 11.5, 19, 0.4],
    [105, 19.7, 6.8, 8.7, 5.9, 4.6, 3.2, 0.4, 0, 1, 2, 5.9, 11.6, 11.8, 19.3, 0.4],
    [105, 20.3, 7.3, 9.2, 6.7, 5.3, 3.8, 1.3, 1.3, 0, 1, 6.8, 13.5, 12.7, 22.4, 0.9],
    [107, 20.8, 7.9, 9.7, 7.3, 5.7, 4.4, 1.8, 1.8, 1.2, 0, 7.3, 13, 12, 19.4, 1.4],
    [111, 30, 11.4, 12, 9.2, 8.6, 7.2, 5.4, 5.8, 6.4, 6.7, 0, 5.7, 7.1, 15.7, 5.8],
    [114, 33.7, 17.1, 17.7, 14.9, 14.3, 14.2, 11.1, 11.5, 12.3, 12.4, 5.7, 0, 7.7, 14.8, 11.5],
    [118, 37.1, 18.5, 19.1, 16.3, 15.4, 14.1, 10.8, 11.2, 11.8, 11.8, 7.1, 7.7, 0, 7.5, 12.9],
    [126, 46, 27, 27.8, 24.8, 22.9, 21.6, 18.3, 18.7, 19.3, 19.3, 14.6, 14.8, 7.5, 0, 18.7],
    [103, 19.5, 6.5, 8.7, 5.9, 4.3, 3, 0.4, 0.4, 0.95, 1.6, 5.9, 11.6, 11.9, 19.7, 0]
])

# Membuat kamus untuk memetakan nama ke ke indeks
node_indices = {agen_code[i]: i for i in range(len(agen_code))}

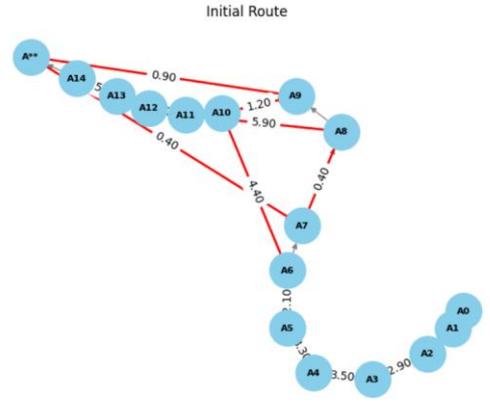
# Kecepatan armada dalam km/jam
fleet_speed = 90

# Rute awal dari algoritma sweep
sweep_result = ['A0', 'A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A10', 'A8', 'A**', 'A7', 'A9', 'A11', 'A12', 'A13', 'A14']
```

Gambar 4. Tampilan Coding pada Google Colab Algoritma Local Search

Seperti pada gambar 3 dan 4 dimana merupakan salah satu contoh yang menunjukkan bahwa terdapat perbedaan input data pada masing – masing algoritmanya yaitu pada tampilan *coding* untuk algoritma *Sweep* input yang di masukkan Agen kode, *Demand*, Jarak dari depot, koordinat polar. Dan untuk input data pada tampilan *coding* untuk algoritma *Local Search* yaitu input *Demand*, Agen kode, Matriks jarak, Rute awal dari algoritma sebelumnya.

Selanjutnya, terdapat salah satu contoh gambar hasil visualisasi *Graph* dengan menggunakan pemrograman *Python*.



Gambar 5. Visualisasi Graph Pemrograman Python

**Studi Kasus Penerapan Algoritma**

Studi kasus yang diangkat dalam penerapan algoritma *Sweep-Local Search* pada CVRP yaitu dalam kasus penentuan rute teroptimal untuk pendistribusian surat kabar.

Data yang digunakan dalam penelitian ini yaitu berupa data distribusi perusahaan surat kabar Jawa Pos Radar Malang tahun 2023 untuk wilayah Malang Raya yang ditampilkan pada tabel 1. Diketahui untuk proses pendistribusian harian Radar Malang menggunakan 2 buah armada yaitu truk box besar dengan kapasitas sekitar 11.000 eksemplar dan mobil Gran Max dengan kapasitas sekitar 1500 eksemplar. Armada ini digunakan sebagai sarana transportasi distribusi harian Radar Malang ke 16 agen yang dimiliki. Banyaknya permintaan atau *demand* harian surat kabar Radar Malang tiap agen berbeda - beda berkisar antara 48 hingga 2.400 eksemplar. Distribusi harian Radar Malang setiap harinya dimulai dari pukul 01.30 WIB dan berakhir pada pukul 04.30 WIB.

Tabel 1. Data Kode Agen, Agen, dan Demand

Kode Agen	Nama Agen	Demand (eksemplar)
A0	Percetakan Sumengko	0
A1	Agen Gianto Lawang	552
A2	Agen Owi	360
A3	Agen Terminal	48
A4	Agen Mickey Mouse 2	192
A5	Agen Erwan	2.400
A6	Agen Rubiani	1.020
A10	Agen Classa	264
A9	Agen Hercules	1.200
A**	Radar Malang	1.200
A7	Agen Mickey Mouse 1	576
A8	Agen Kuswadi	1.272
A11	Agen Putri	240
A12	Agen Duaja Bululawang	480
A13	Agen Classa Pakis	48
A14	Agen Elite Kepanjen	360
A15	Agen Suharto	360
A16	Agen Batu Alun-alun	840

Pada kode A\*\* yaitu Radar Malang untuk *demand* nya menyesuaikan permintaan dari Agen Suharto dan Agen Batu Alun - Alun, karena Radar Malang merupakan tempat drop out surat kabar untuk pengiriman yang menggunakan mobil Gran Max.

Dalam proses pengolahan data dilakukan dengan menggunakan program *python* mulai dari penerapan algoritma *sweep*, dilakukan optimalisasi dengan algoritma *Local Search*, dan yang terakhir membuat gambar rute.

a. Penerapan Algoritma *Sweep*

Tahap - tahap pengelompokan (*Clustering*) menggunakan algoritma *sweep* :

1. Menentukan masing - masing titik (agen) dalam koordinat kartesius dan mengubah koordinat kartesius menjadi koordinat polar  $(r, \theta)$  dengan menetapkan titik depot sebagai pusat koordinat.

Pada koordinat kartesius untuk nilai  $x$  = nilai *Latitude* dan nilai  $y$  = nilai *Longitude* yang diperoleh dengan alat bantu *Google maps*. Untuk mencari  $\theta$  pada koordinat polar  $(r, \theta)$  dengan mengubah koordinat kartesius  $(x, y)$  dalam perhitungan normal di kasus ini menggunakan rumus:

$$\theta = \arctan \frac{y}{x} \quad (1)$$

$$\theta = \theta + 180^\circ \quad (2)$$

Rumus diatas didapatkan dari formulasi algoritma *Sweep* sebagai berikut :

$$\theta_i = \begin{cases} \tan^{-1} \frac{y_i}{x_i}, & x_i > 0, y_i > 0 \\ \pi + \tan^{-1} \frac{y_i}{x_i}, & x_i < 0 \\ 2\pi + \tan^{-1} \frac{y_i}{x_i}, & x_i > 0, y_i < 0 \end{cases}$$

Dalam penghitungan koordinat polar penulis menggunakan program *Python* dengan Hasil yang ditampilkan pada tabel 2.

Tabel 2. Data Koordinat kartesius dan Koordinat Polar

Kode Agen	Titik drop	X (Latitude)	Y (Longitude)	$\theta(^{\circ})$
A0	Percetakan Sumengko	-7.38891	112.56468	0
A1	Agen Gianto Lawang	-7,82984	112,69708	93.97434468
A2	Agen Owi	-7,93225	112,64734	94.02792899
A3	Agen Terminal	-7,93343	112,65911	94.02810671
A4	Agen Mickey Mouse 2	-7,94372	112,64824	94.03370202
A5	Agen Erwan	-7,9525	112,63584	94.03858872
A6	Agen Rubiani	-7,96316	112,63251	94.0441035
A7	Agen Mickey Mouse 1	-7,98091	112,62755	94.05326575
A8	Agen Kuswadi	-7,98076	112,62504	94.05327985
A9	Agen Hercules	-7,9758	112,62049	94.05093226
A10	Agen Classa	-7,97299	112,61348	94.04976104
A11	Agen Putri	-8,02265	112,62749	94.0743952
A12	Agen Duaja Bululawang	-8,06862	112,63993	94.09721129
A13	Agen Classa Pakis	-8,07013	112,59673	94.09954236
A14	Agen Elite Kepanjen	-8,1316	112,56924	94.13166604
A15	Agen Suharto	-7,91283	112,58456	94.02033326
A16	Agen Batu Alun-alun	-7,87116	112,52694	94.00127172
A**	radar malang	-7,9789	112,6246	94.05235412

- Membentuk pengelompokan (*clustering*) sesuai dengan kedekatan lokasi agen yaitu dimulai dari urutan agen yang mempunyai  $\theta$  dari koordinat polar yang terkecil hingga terbesar dengan mempertimbangkan kapasitas setiap kendaraan.
- Membentuk iterasi dengan pemisalan iterasi 0 yaitu diambil  $A_0$  yang

dimasukkan kedalam *cluster* C menjadi  $C = \{A_0\}$ . Dan  $i = 1$  sampai seterusnya.

Lalu pada  $i = 1$  sampai seterusnya, diambil  $A_i$ . Dilihat apakah total permintaan  $A_i + C >$  kapasitas kendaraan, jika tidak maka  $A_i$  di masukkan ke *cluster* C sehingga  $C = C \cup A_i$ . Dan dilakukan iterasi selanjutnya sampai kapasitas kendaraan terpenuhi.

- Clustering* dihentikan jika terdapat *cluster* yang melebihi kapasitas maksimal kendaraan.
- Lalu, membuat *cluster* baru dengan tahap yang sama seperti tahap ke 2.
- Mengulangi tahap 2 sampai 4, hingga semua agen (titik) telah dimasukkan ke dalam sebuah *cluster*.
- Membuat tabel untuk setiap *cluster* yang sudah dilakukan dari tahap 2 sampai 4, ditunjukkan pada tabel 3 dan tabel 4.

- CLUSTER 1 dari depot (Truk Besar Kapasitas 11.000 eksemplar)

Tabel 3. Cluster 1 Algoritma Sweep

Kode Agen	Nama Agen	$\theta (^{\circ})$	Demand (eksemplar)
A0	Percetakan Sumengko	0	0
A1	Agen Gianto Lawang	93.97434468	552
A2	Agen Owi	94.02792899	360
A3	Agen Terminal	94.02810671	48
A4	Agen Mickey Mouse 2	94.03370202	192
A5	Agen Erwan	94.03858872	2.400
A6	Agen Rubiani	94.0441035	1.020
A10	Agen Classa	94.04976104	264
A9	Agen Hercules	94.05093226	1.200
A**	Radar Malang	94.05235412	1.200
A7	Agen Mickey Mouse 1	94.05326575	576
A8	Agen Kuswadi	94.05327985	1.272
A11	Agen Putri	94.0743952	240
A12	Agen Duaja Bululawang	94.09721129	480
A13	Agen Classa Pakis	94.09954236	48
A14	Agen Elite Kepanjen	94.13166604	360

Cluster 1 = {A0, A1, A2, A3, A4, A5, A6, A10, A9, A\*\*, A7, A8, A11, A12, A13, A14}

Dengan total kapasitas angkut sebesar 10.212 eksemplar.

- CLUSTER 2 dari Radar Malang (Gran Max Kapasitas 1.500 eksemplar)

Tabel 4. Cluster 2 Algoritma Sweep

Kode Agen	Nama Agen	$\theta$ (°)	Demand (eksemplar)
A**	Radar Malang	0	0
A16	Agen Batu Alun-alun	94.00127172	840
A15	Agen Suharto	94.02033326	360

Cluster 2 = {A\*\*, A16, A15}.

Dengan total kapasitas angkut sebesar 1.200 eksemplar.

Didapatkan *output* dari tahap – tahap algoritma *sweep* untuk penentuan rute distribusi sesuai kapasitas angkutnya dengan perhitungan menggunakan *Python* ditunjukkan pada tabel 5 :

Tabel 5. Hasil Algoritma Sweep

Cluster	Rute	Jarak Tempuh (Km)	Waktu Tempuh (Menit)	Demand
1	A0 - A1 - A2, A3 - A4 - A5, A6 - A10 - A9, A** - A7 - A8, A11 - A12 - A13 - A14	143.3	119	10.212 eksemplar
2	A** - A16 - A15	28	21.6	1.200 eksemplar

Tabel diatas merupakan hasil akhir yang didapatkan dari proses pendistribusian dari depot untuk *cluster* 1 dan dari *dropout* untuk *cluster* 2

sesuai kapasitasnya dan juga sesuai dari data perusahaan di mana untuk armada 2 yaitu mobil Gran Max berada di kantor Radar Malang sehingga untuk cluster2 di buat titik utama dari Radar Malang. Lalu untuk waktu tempuh didapatkan berdasarkan kecepatan (Km/h) setiap kendaraan yang sudah di tentukan oleh perusahaan.

b. Penerapan Algoritma *Local Search*

Tahap – tahap perbaikan rute sebelumnya dengan algoritma *Local Search (Insertion Intra-Route (1-0))*:

1. Masukkan tur (iterasi) dan rute hasil algoritma *Sweep*, matriks jarak, permintaan *demand*, kapasitas setiap kendaraan.
2. Dimulai dari tur ke 1,  $i = 1$ .
3. Melakukan penukaran dua urutan tujuan agen atau titik dalam rute yang sama untuk setiap *cluster* rute kendaraan, dilanjut ke tahap 4.

Contoh dalam penukaran :

A	B	C	D	E
---	---	---	---	---

Dengan menukar agen B dengan C dalam *cluster* yang sama didapatkan perubahan rute menjadi :

A	C	B	D	E
---	---	---	---	---

4. Jika hasil total jarak pada *cluster* rute baru lebih kecil dari sebelumnya maka dilanjutkan ke tahap 5, jika belum maka kembali ke tahap 3.
5. Memilih *cluster* rute baru untuk menggantikan *cluster* rute sebelumnya, kembali ke tahap 3.
6. Jika semua tur (iterasi) sudah di cari maka lanjut ke tahap 8, jika belum maka lanjut ke tahap 7.
7. Hitung  $i, i = i+1$ , lalu kembali ke tahap 3.
8. Membuat tabel Hasil perbaikan rute setelah semua tahap dikerjakan.

- CLUSTER 1 dari depot (Truk Besar Kapasitas 11.000 eksemplar)

Tabel 6. Perbaikan Rute Local Search Untuk Cluster 1

Algoritma	Rute	Jarak Tempuh (Km)	Waktu tempuh (menit)
Algoritma <i>Sweep</i>	A0 - A1 - A2 - A3 - A4 - A5 -A6- A10 - A9 - A** - A7 - A8 - A11 - A12 - A13 - A14	143.3	119
Algoritma <i>Local Search</i>	A0 - A1 - A3 - A2 - A4 - A5 - A6 - A10 - A9 -A** - A7 - A8 - A11 - A12 - A13 - A14	142	117

- CLUSTER 2 dari Radar Malang (Gran Max Kapasitas 1.500 eksemplar)

Tabel 7. Perbaikan Rute Local Search Untuk Cluster 2

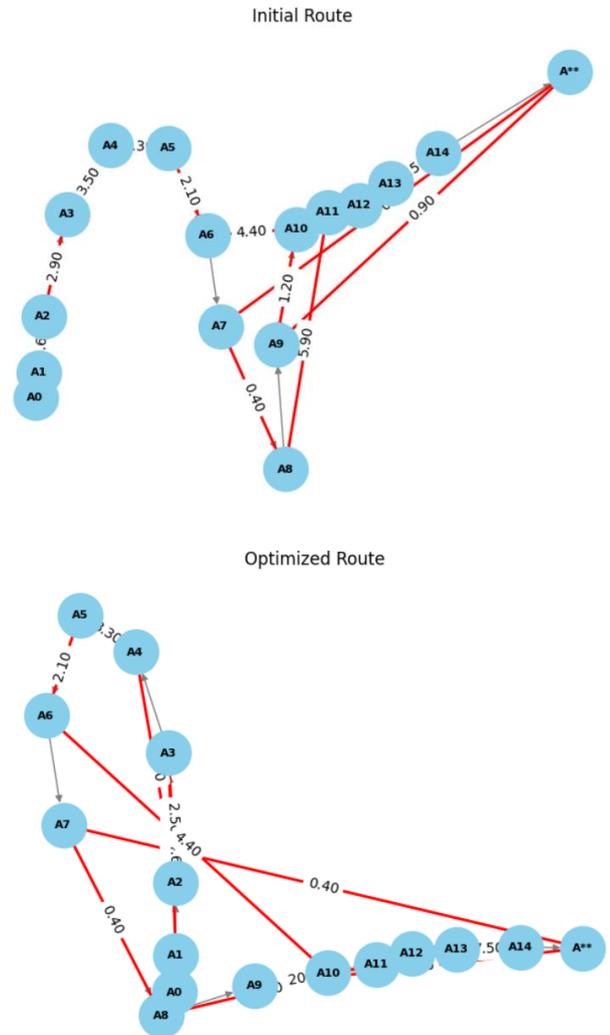
Algoritma	Rute	Jarak Tempuh (Km)	Waktu tempuh (menit)
Algoritma <i>Sweep</i>	A**- A16 - A15	28	21.6
Algoritma <i>Local Search</i>	A** - A15 - A16	19.5	21.4

Dari tabel – tabel diatas, didapatkan optimalisasi rute menggunakan algoritma *Local Search* dengan alat bantu *python* dalam menghitung jarak tempuh serta waktu tempuh dan dalam menentukan jarak yang paling optimal dari rute sebelumnya.

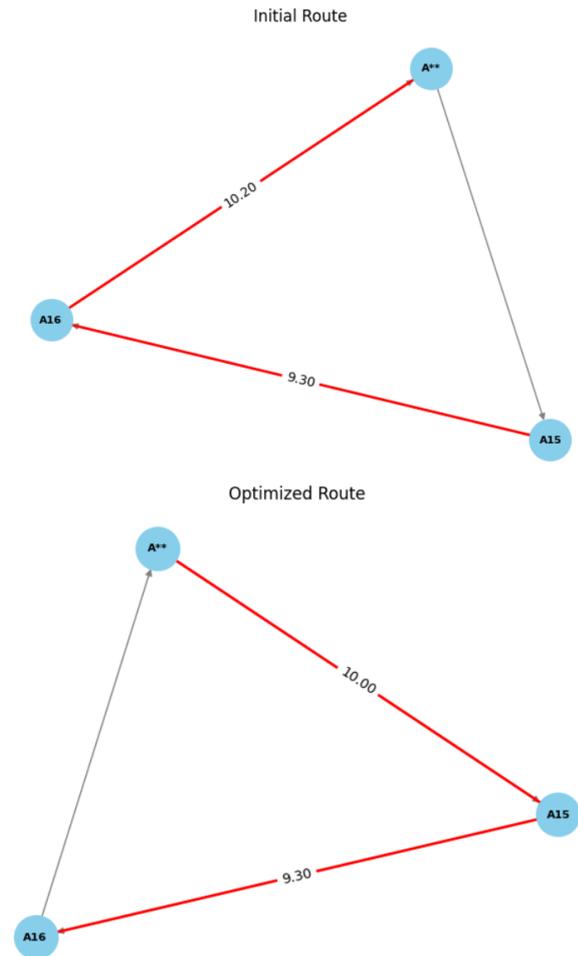
Sehingga didapatkan tabel hasil akhir dari kedua penyelesaian menggunakan algoritma *Sweep* dan di optimalisasi dengan algoritma *Local Search* ditunjukkan pada tabel 8 serta terdapat rute dari perusahaan ditunjukkan pada tabel 9 untuk perbandingan hasil penelitian.

Tabel 8. Hasil Akhir Penyelesaian Rute Distribusi

cluster	Nodes / cluster	Jarak Tempuh (Km)	Waktu Tempuh (menit)	Kapasitas Angkut Vehicle (demand)
1	A0 - A1- A3- A2 - A4 -A5- A6-A10- A9 - A**- A7- A8- A11-A12 -A13 - A14	142	117	10.212 eksemplar
2	A**- A15 -A16	19.5	21.4	1.200 eksemplar



Gambar 6. Rute Cluster 1 Dengan Python



Gambar 7. Rute Cluster 2 Dengan Python

Tabel 9. Rute Perusahaan

cluster	Nodes / cluster	Jarak Tempuh (Km)	Waktu Tempuh (menit)	Kapasitas Angkut Vehicle (demand)
1	A0 -A1- A3-A2- A4-A5 - A6-A7- A8-A**- A9-A10- A11-A12- A13-A14	143.5	140	10.212 eksemplar
2	A**-A15- A16	19.5	21.4	1.200 eksemplar

Setelah melakukan penelitian penerapan algoritma *Sweep - Local Search*, study kasus mengenai distribusi surat kabar di Jawa Pos Radar Malang dengan alat bantu program

*python* peneliti mendapatkan hasil penelitian dengan pencarian titik lokasi setiap agen dari data perusahaan dengan bantuan *Google maps* untuk memperoleh koordinat kartesius serta jarak antar setiap agennya. Alat bantu *Google maps* sangat membantu dalam pencarian jarak dan juga koordinat kartesius karena sangat menghemat waktu, biaya serta cukup akurat. Hasil perhitungan solusi yang optimal dengan menggunakan metode algoritma *Sweep - algoritma Local Search* dilakukan perhitungan dengan menggunakan alat bantu *python*. Dalam perhitungan dilakukan dengan mencari rute awal menggunakan algoritma *sweep* lalu setelah mendapatkan rute sesuai kapasitas kendaraan dilakukan pengoptimalan rute menggunakan algoritma *Local Search* untuk mendapatkan hasil akhir rute yang lebih optimal.

Dalam pencarian solusi optimal pertama dengan menggunakan algoritma *Sweep* dilakukan pencarian koordinat polar terlebih dahulu untuk menentukan agen yang memiliki koordinat terkecil hingga terbesar dari titik awal depot lalu dilanjut ke rute agen yang memiliki koordinat polar terkecil hingga memenuhi kapasitas kendaraan, jika dalam *cluster* rute pertama belum mencakup semua agen maka dilakukan *clustering* selanjutnya sesuai kapasitas kendaraannya masing - masing hingga semua agen mendapatkan *cluster* rute distribusinya.

Setelah mendapatkan 2 *cluster* rute dari hasil perhitungan dengan menggunakan *python* untuk algoritma *Sweep*, selanjutnya dilakukan ke pencarian solusi optimal kedua untuk mengoptimisasi rute awal dengan menggunakan algoritma *Local Search (Insertion Intra-Route (1-0))* yaitu dengan menukar dua agen dalam rute yang sama untuk setiap iterasinya hingga mendapatkan hasil rute yang paling optimal. Dalam perhitungan ini juga menggunakan alat bantu *python* dengan hasil rute akhir yang paling optimal yaitu untuk *cluster* pertama menggunakan kendaraan truk Box besar untuk kapasitas 11.000 eksemplar didapatkan rute dari Depot - Agen Gianto Lawang - Agen Terminal - Agen Owi - Agen Micky Mouse 2 - Agen Erwan - Agen Rubiani - Agen Classa - Agen Hercules - Radar Malang - Agen Micky Mouse 1 - Agen Kuswadi - Agen

Putri – Agen Duaja Bululawang – Agen Classa Pakis – Agen Elite Kepanjen. Total jarak tempuh sekitar 142 Km dan waktu tempuh 1 jam 57 menit dengan kecepatan rata – rata 90 Km/h. Dan *cluster* kedua menggunakan kendaraan mobil Gran Max untuk kapasitas 1.500 eksemplar didapatkan rute dari Radar Malang – Agen Suharto – Agen Batu Alun – Alun. Total jarak tempuh 19,5 Km dan waktu tempuh 21 menit 24 detik dengan kecepatan rata – rata 90 Km/h.

Berdasarkan hasil perhitungan solusi rute terpendek dengan menggunakan 2 metode algoritma mendapatkan hasil lebih optimal dibandingkan dengan rute asli milik perusahaan. Ada pula kelebihan dan kekurangan dalam menggunakan perhitungan algoritma *Sweep* dan algoritma *Local Search*.

Kelebihan dari algoritma *Sweep* dalam perhitungan ini yaitu dalam perhitungan masalah berurutan dan efisien sehingga pemrosesan masalah bisa selesai dilakukan dengan cara yang teratur dan dapat dipahami. Sedangkan kelemahannya yaitu tidak selalu optimal hasilnya sehingga harus dioptimalkan kembali dengan algoritma yang lain.

Kelebihan dari algoritma *Local Search* yaitu mampu memberikan pembaharuan rute yang lebih efektif dalam waktu yang lebih singkat di mana dengan perubahan kecil yang menghasilkan solusi yang lebih baik. Sedangkan kelemahannya yaitu memiliki keterbatasan dalam permasalahan solusi optimal yang lebih kompleks sehingga tidak mampu mengeksplorasi seluruh solusinya dengan baik.

## PENUTUP

### SIMPULAN

Berdasarkan hasil dan pembahasan dalam penelitian ini dapat disimpulkan bahwa :

1. Hasil penerapan algoritma *Sweep* – *Local Search* pada rute distribusi surat kabar harian Radar Malang didapatkan 2 *cluster* rute teroptimal dengan kapasitas angkut yang maksimal pada setiap kendaraan.
2. Hasil dari penelitian mendapatkan rute yang lebih efisien dan lebih optimal dibandingkan rute yang diterapkan oleh

perusahaan sebesar 1,05% untuk jarak dan 16,43% untuk waktu.

3. Algoritma *Sweep* dan algoritma *Local Search* merupakan algoritma metaheuristik yang dapat menjadi salah satu alternatif untuk menyelesaikan permasalahan CVRP, salah satunya permasalahan yang dapat diselesaikan yaitu dalam penelitian ini mengenai rute distribusi surat kabar.

### SARAN

Berdasarkan hasil dan pembahasan penelitian ini, ada beberapa saran yang kedepannya diharapkan dapat dikembangkan dan menjadi bahan pertimbangan guna penelitian selanjutnya dan juga bagi perusahaan :

1. Bagi perusahaan  
Diharapkan penelitian ini dapat dijadikan pertimbangan oleh departemen pendistribusian atau kurir yang bertugas dalam melakukan proses pendistribusian surat kabar melalui rute yang optimal.
2. Bagi penelitian selanjutnya  
Diharapkan dapat mengembangkan penelitian CVRP mengenai rute distribusi surat kabar ini dengan menggunakan metode yang berbeda seperti algoritma *nearest neighbour-reversal heuristic* serta menggunakan alat bantu selain *python* agar lebih optimal hasilnya.  
Bisa juga melakukan pengembangan algoritma *Local search* dalam penelitian ini dengan menggunakan salah satu jenis algoritma *Local Search* yang lain seperti *swap intra route* agar mendapatkan hasil yang lebih optimal.

### DAFTAR PUSTAKA

- Akhand, M. A. H., Peysa, Z. J., Sultana, T., & Al-Mahmud. (2016). Solving Capacitated Vehicle Routing Problem with route optimization using Swarm Intelligence. *2nd International Conference on Electrical Information and Communication Technologies, EICT 2015, Eict*, 112–117.  
<https://doi.org/10.1109/EICT.2015.7391932>.
- Arifita, E., & Rakhmawati, F. (2023). Analysis of Book Distribution Routes Using the Capacity Vehicle Routing Problem (CVRP) Method Using the

- Sweep Algorithm. *Sinkron*, 8(1), 360–367. <https://doi.org/10.33395/sinkron.v8i1.12013>.
- Aurachman, R., Baskara, D. B., & Habibie, J. (2021). Vehicle routing problem with simulated annealing using python programming. *IOP Conference Series: Materials Science and Engineering*, 1010(1). <https://doi.org/10.1088/1757-899X/1010/1/012010>
- Basriati, S., & Sunarya, R. (2015). Optimasi distribusi Koran menggunakan metode saving matriks (studi kasus: PT. Riau Pos Intermedia). *Seminar Nasional Teknologi ...*, November, 448–453. <http://ejournal.uin-suska.ac.id/index.php/SNTIKI/article/view/3027>.
- Bernardino, R., & Paia, A. (2023). The family capacitated vehicle routing problem. *European Journal of Operational Research*, October. <https://doi.org/10.1016/j.ejor.2023.10.042>
- Boonkleaw, A., Suthikarnnarunai, N., & Srinon, R. (2009). Strategic Planning and Vehicle Routing Algorithm for Newspaper Delivery Problem: Case study of Morning Newspaper, Bangkok, Thailand. *Lecture Notes in Engineering and Computer Science*, 2179(1), 1067–1071.
- Dereci, U., & Karabekmez, M. E. (2022). The applications of multiple route optimization heuristics and meta-heuristic algorithms to solid waste transportation: A case study in Turkey. *Decision Analytics Journal*, 4(June), 100113. <https://doi.org/10.1016/j.dajour.2022.100113>
- Desaulniers, G., Madsen, O. B. G., & Ropke, S. (2014). Chapter 5: The Vehicle Routing Problem with Time Windows. In *Vehicle Routing*. <https://doi.org/10.1137/1.9781611973594.ch5>
- Fatnita, A. V., & Lukmandono. (2020). Optimasi Rute Distribusi Tabung LPG 3 Kg Dengan Menggunakan Alogaritma Genetika Pada Penyelesaian Capacitated Vehicle Routing Problem (CVRP) (Studi kasus pada PT. Jana Pusaka Migas). *Prosiding Seminar Nasional Sains Dan Teknologi Terapan*, 1(1), 39–46. <https://ejournal.itats.ac.id/sntekpan/article/view/1213>.
- Fauziah, A. S., Cholissodin, I., & Rahayudi, B. (2022). Optimasi Pendistribusian Air Mineral menggunakan Algoritma Genetika. 6(2), 966–972.
- Gendreau, M., & Potvin, J.-Y. (2010). *International Series in Operations Research and Management Science: Handbook of Metaheuristics*.
- Hanafi, R., Rusman, M., Mardin, F., Parenreng, S. M., & Azzazli, A. (2020). Distribution Route Optimization of a Capacitated Vehicle Routing Problem by Sweep Algorithm. *IOP Conference Series: Materials Science and Engineering*, 875(1). <https://doi.org/10.1088/1757-899X/875/1/012066>
- Kumari, M., De, P. K., Chaudhuri, K., & Narang, P. (2023). Utilizing a hybrid metaheuristic algorithm to solve capacitated vehicle routing problem. *Results in Control and Optimization*, 13(September), 100292. <https://doi.org/10.1016/j.rico.2023.100292>
- Kurnia, G., Kurniawan, A. C., Nawadir, M., Yasmin, M. S., & Hibatullah, M. (2020). Route Optimization of Oil Country Tubular Goods Distribution Using Sweep and Savings Algorithm. *IPTEK Journal of Proceedings Series*, 0(5), 28. <https://doi.org/10.12962/j23546026.y2020i5.7927>.
- Nono, V., Sofitra, M., & Wijayanto, D. (2020). Penyelesaian Capacitated Vehicle Routing Problem Dengan Menggunakan Algoritma Sweep Untuk Penentuan Rute Distribusi Untuk Depo Pt. Abc Kubu Raya. *Jurnal TIN Universitas Tanjungpura*, 4(2), 232–238.
- Nusantara, K. G., Alfi, N., Farikha, N., Nagari, T. B., & Oktyajati, N. (2023). Determining Newspaper Distribution Routes using Cheapest Insertion Heuristic Algorithm with Sweep Clustering: A Case Study. 2918–2929. <https://doi.org/10.46254/an12.20220533>.
- Pangestu, A. D., Munib, A. A., Fitri, T. N., & Oktyajati, N. (2023). Determining the Optimal Route for Newspaper Distribution by Using the Sweep Algorithm Method (Case Study: PT Aksara Solopos). 2909–2917. <https://doi.org/10.46254/an12.20220532>
- Ruben, M., & Imran, A. (2020). Usulan Rute Distribusi Menggunakan Algoritma Sweep Dan Local Search (Studi Kasus Di Perusahaan X). *Jurnal Rekayasa Sistem Industri*, 6(1), 40–44. <https://doi.org/10.33884/jrsi.v6i1.2491>.
- Sirait, Z. A., & Lubis, R. S. (2023). Numerical: Jurnal Matematika dan Pendidikan Matematika Completion of CVRP with a Combination of Sweep and Local Search Algorithms (Case Study: PT. AJW). 7(1), 59–70. <http://journal.iaimnumetrolampung.ac.id/index.php/numericalDOI:https://doi.org/10.25217/numerical.v7i1>.
- Thammano, A., & Rungwachira, P. (2021). Hybrid modified ant system with sweep algorithm and path relinking for the capacitated vehicle routing problem. *Heliyon*, 7(9), e08029. <https://doi.org/10.1016/j.heliyon.2021.e08029>
- Tohari, A., & Astuti, Y. P. (2023). Penerapan Algoritma Genetika Dalam Menentukan Rute

Terpendek PT. Pos Cabang Lamongan. *Jurnal Ilmiah Matematika*, 11(3), 458–467.

Toth, P., & Vigo, D. (2002). Discrete Mathematics and Applications The Vehicle Routing Problem. *The Vehicle Routing Problem*, 1–26.

Wibisono, I. B., & Hafidza, L. A. (2023). *Determining Newspaper Distribution Routes Using Sweep Algorithm and Local Search to Solve the Capacitated Vehicle Routing Problem and Minimizing Cost*. 2860–2872.

<https://doi.org/10.46254/an12.20220528>.