

## Algoritme *Sweep* dan *Particle Swarm Optimization* dalam Optimisasi Rute Kendaraan dengan Kapasitas

BIB PARUHUM SILALAH, KHOERUL FATIHIN,  
PRAPTO TRI SUPRIYO, SUGI GURITMAN

Departemen Matematika  
Fakultas MIPA Institut Pertanian Bogor  
University Kampus IPB Darmaga, Jl. Meranti Bogor 16680  
bibparuhum@gmail.com

### Abstrak

Masalah rute kendaraan dengan kapasitas (*capacitated vehicle routing problem*) adalah variasi dari masalah rute kendaraan (*vehicle routing problem*). Pada masalah rute kendaraan dengan kapasitas, kendaraan yang digunakan untuk distribusi produk memiliki batas daya angkut. Menentukan solusi optimal dari masalah rute kendaraan dan perluasannya adalah *NP-Hard*. Oleh karena itu untuk menyelesaikan masalah rute kendaraan dengan kapasitas ini banyak dikembangkan algoritme heuristik. Dalam paper ini, untuk mencari solusi masalah rute kendaraan dengan kapasitas, digunakan gabungan dua algoritme heuristik. Penyelesaian masalah dimulai dengan pembentukan kelompok (*clustering*) menggunakan algoritme *sweep*, kemudian setiap kelompok hasil algoritme *sweep* dioptimalkan menggunakan algoritme *particle swarm optimization*.

*Kata kunci:* algoritme *sweep*, algoritme heuristik, *particle swarm optimization*, masalah rute kendaraan dengan kapasitas.

### Abstract

*Capacitated vehicle routing problem is a variation of the vehicle routing problem. In the capacitated vehicle routing problem, the vehicle used for product distribution has a capacity limit. Determining the optimal solution of the vehicle routing problem and its variation, included the capacitated vehicle routing problem, is NP-Hard. Therefore, heuristic algorithms are developed. In this paper, for solving the capacitated vehicle routing problem, a combination of two heuristic algorithms are used. First, the sweep algorithm performs clustering, then the clustering results from the sweep algorithm are optimized using the particle swarm optimization algorithm.*

*Keywords:* sweep algorithm, heuristic algorithm, particle swarm optimization, capacitated vehicle routing problem.

## 1. PENDAHULUAN

Optimisasi adalah cabang ilmu matematika yang mempelajari teknik-teknik pencarian nilai maksimum atau minimum dari suatu problem. Teknik-teknik dan teori optimisasi dielaborasi, antara lain: penentuan *step size* metode *steepest descent* [4, 14], penggunaan kombinasi metode [8, 15], mengelaborasi metode Newton [16], menentukan batas atas iterasi algoritme primal-dual langkah penuh Newton [10]. Optimisasi dapat diterapkan hampir pada semua aspek kehidupan, tetapi banyak penerapan optimisasi dilakukan pada sistem distribusi dan transportasi [1, 7, 21].

*Traveling salesman problem* (TSP) merupakan masalah optimisasi yang berhubungan dengan distribusi dan transportasi. Pada dasarnya, TSP adalah masalah untuk menemukan jarak minimum dari simpul awal ke simpul berikutnya dengan aturan bahwa setiap simpul harus dikunjungi tepat sekali. Selanjutnya, setelah semua simpul dikunjungi tepat satu kali, harus kembali ke simpul awal. Masalah rute kendaraan (*Vehicle routing problem* (VRP)) adalah generalisasi dari TSP. Masalah rute kendaraan adalah suatu problem optimisasi yang mencari himpunan optimal rute kendaraan dalam mendistribusikan suatu produk/barang kepada konsumen. Pada masalah rute kendaraan, produsen ataupun distributor berada pada suatu lokasi yang disebut depot dan mendistribusikan produk ke konsumen-konsumen mereka dengan tujuan biaya perjalanan yang minimum [17].

Terdapat beberapa variasi dari *vehicle routing problem* seperti *vehicle routing problem* dengan *time windows* [5], *multi depot vehicle routing problem* [6], serta masalah rute kendaraan dengan kapasitas (*Capacitated vehicle routing problem* (CVRP)) [19, 20]. Pada masalah rute kendaraan dengan kapasitas, kendaraan yang digunakan untuk distribusi produk memiliki batas daya angkut.

Menentukan solusi optimal dari masalah rute kendaraan dan perluasannya adalah *NP-Hard*. Oleh karena itu untuk menyelesaikan masalah rute kendaraan ini banyak dikembangkan metode penyelesaian pendekatan yaitu menggunakan algoritme-algoritme heuristik. Algoritme heuristik untuk optimisasi yang dikembangkan diantaranya adalah metode titik interior [11, 12], algoritme *particle swarm optimization* (PSO) [2, 18, 19], algoritme firefly [9], algoritme koloni semut (ant colony algorithm) [13], algoritme genetika [21].

Dalam paper ini, metode heuristik digunakan untuk mencari solusi masalah rute kendaraan dengan kapasitas. Penyelesaian masalah dimulai dengan pembentukan kelompok (clustering) menggunakan algoritme *sweep* [3], kemudian setiap kelompok hasil algoritme *sweep* dioptimalkan menggunakan *particle swarm optimization* (PSO).

**1.1. Masalah Rute Kendaraan dengan Kapasitas.** Masalah rute kendaraan dengan kapasitas adalah suatu masalah optimisasi dengan tujuan meminimumkan biaya pendistribusian produk sesuai jumlah permintaan konsumen dengan menggunakan kendaraan yang memiliki batas daya angkut/kapasitas. Hal yang harus dipenuhi dalam masalah ini adalah:

- (1) Setiap konsumen memiliki permintaan jumlah produk dan hanya boleh dikunjungi tepat satu kali oleh satu kendaraan untuk mengantarkan permintaan konsumen.
- (2) Kendaraan berangkat dari depot awal mendistribusikan produk kepada konsumen dan harus kembali ke depot awal.
- (3) Kendaraan berangkat dari depot awal dengan membawa sejumlah produk yang tidak dapat melebihi jumlah daya angkut (kapasitas) kendaraan.
- (4) Jumlah total permintaan produk pada suatu rute yang dipilih tidak dapat melebihi kapasitas kendaraan.

**1.2. Algoritme Sweep.** Algoritme *sweep* pertama kali diperkenalkan oleh Gillet dan Miller pada tahun 1974 [3]. Algoritme ini termasuk dalam algoritme heuristik. Langkah pertama untuk menggunakan algoritme ini yaitu dengan menentukan koordinat polar lokasi setiap konsumen. Pada algoritme ini depot dianggap sebagai pusat koordinat. Oleh karena itu, perhitungan  $i$  setiap konsumen menggunakan persamaan:

$$\theta_i = \tan^{-1} \frac{y_1 - y_0}{x_1 - x_0}$$

dengan  $(x_i, y_i)$  sebagai titik lokasi konsumen sedangkan  $(x_0, y_0)$  sebagai titik lokasi depot. Langkah selanjutnya yaitu tahap *sweeping* konsumen. Tahap ini dilakukan dengan membuat *radial line* yang membentuk suatu sudut dengan depot sebagai pusat. Sudut ini diperbesar hingga mencakup konsumen pada titik  $i$ , sudut selanjutnya diperbesar hingga mencakup konsumen ke  $i + 1$ . Jika permintaan konsumen pada titik ke  $i + 1$  ditambahkan dengan permintaan konsumen pada titik  $i$  tidak melebihi kapasitas maksimal yang ditentukan, maka konsumen pada titik  $i$  dan  $i + 1$  termasuk dalam satu kelompok. Namun jika melebihi kapasitas maksimal yang telah ditentukan, maka *sweeping* dihentikan dan konsumen pada titik  $i + 1$  akan dimasukkan dalam kelompok berikutnya. Proses ini berlaku untuk  $i + 2, i + 3$ , dan seterusnya serta diulang sampai semua titik memiliki kelompok.

**1.3. Algoritme *Particle Swarm Optimization*.** *Particle swarm optimization* (PSO) merupakan algoritme berbasis populasi yang mengeksplorasi individu dalam pencarian solusi. *Particle* merujuk kepada individu (partikel) dan *swarm* merujuk kepada populasi. Individu-individu dibangkitkan secara acak dengan batasan nilai terkecil dan terbesar. Individu-individu yang merupakan kumpulan suatu populasi melakukan pencarian solusi. Individu bergerak mencari solusi menurut catatan posisi terbaik yang pernah dicapai. Individu bergerak berdasarkan penyesuaian terhadap posisi terbaik dari individu (*local best*) dan penyesuaian terhadap posisi individu terbaik dari seluruh kawanan (*global best*) selama melintasi ruang pencarian [2].

Prosedur standar algoritme PSO adalah sebagai berikut:

- (1) Misalkan ukuran populasi (jumlah individu/partikel) adalah  $N$ . Secara acak tentukan posisi awal dan kecepatan awal setiap individu.
- (2) Iterasi  $i = 1$ .
- (3) Nilai setiap individu diukur menurut fungsi sasaran. Jika nilai individu pada posisi saat ini lebih baik dari *local best*, maka *local best* diatur untuk posisi saat ini. Lebih baik disini berarti nilai individu saat ini pada fungsi tujuan lebih kecil/lebih besar dari nilai *local best* ketika fungsi tujuan adalah mencari nilai minimum/maksimum.
- (4) Nilai individu dibandingkan dengan *global best*. Jika nilai individu lebih baik, *global best* diperbaharui nilainya.
- (5) Kecepatan dan posisi setiap individu diperbaharui dengan menggunakan persamaan berikut:

$$\begin{aligned} V_k^{i+1} &= w \times V_k^i + c_2 \times rand_1 \times (P_k - X_k^i) + c_2 \times rand_2 \times (G_k - X_k^i) \\ X_k^{i+1} &= X_k^i + V_k^{i+1}, \end{aligned}$$

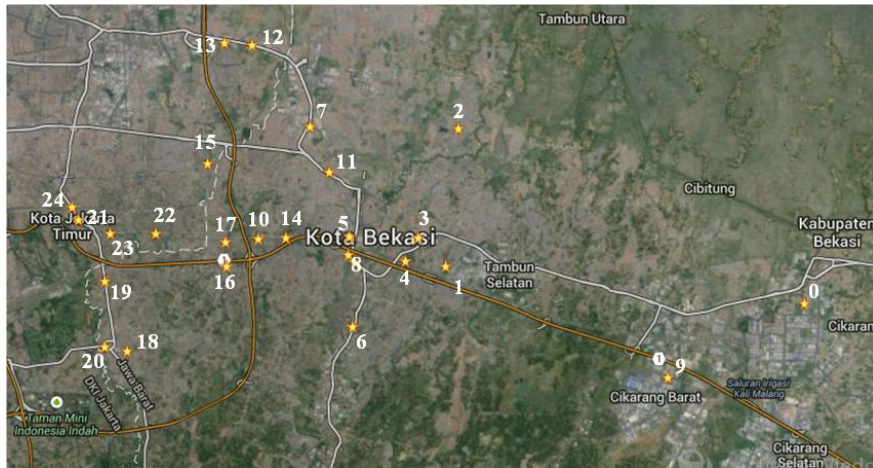
dimana:

- $V_k^{i+1}$  = kecepatan individu ke  $k$  pada iterasi  $i + 1$
- $X_k^{i+1}$  = posisi individu  $k$  pada iterasi  $i + 1$
- $w$  = parameter *inertia weight*, yaitu suatu parameter yang mengontrol seberapa pengaruh kecepatan
- $c_1, c_2$  = konstanta akselerasi
- $rand_1, rand_2$  = parameter acak antara 0 sampai 1
- $P_k$  = *local best* individu  $k$
- $G_k$  = *global best* dari semua individu

- (6) Periksa kriteria pemberhentian. Kriteria pemberhentian dapat berupa kekonvergenan solusi ataupun jumlah iterasi. Jika kriteria penghentian belum terpenuhi pergi ke langkah 2 untuk iterasi ke  $i = i + 1$ .

## 2. METODE PENELITIAN

Penelitian ini menggunakan data Wahyuni [20]. Data terdiri atas jarak distribusi dari depot ke setiap konsumen dan juga data jumlah permintaan untuk setiap konsumennya. Terdapat 24 konsumen yang akan dilayani dengan kendaraan yang memiliki batas kapasitas 135 boks. Lokasi depot dan semua konsumen dapat dilihat pada Gambar 1. Masing-masing konsumen memiliki jumlah permintaan masing masing, seperti ditampilkan pada pada Tabel 1. Nantinya akan dicari jalur terpendek untuk menentukan rute pendistribusian tersebut.



GAMBAR 1. Lokasi depot dan semua konsumen

Penyelesaian masalah dimulai dengan menerapkan algoritme *sweep* untuk pengelompokan konsumen dengan menggunakan bantuan *software Geogebra*, kemudian dilanjutkan dengan menggunakan algoritme PSO untuk mencari rute terbaik disetiap kelompok hasil algoritme *sweep*. Rute yang diperoleh kemudian dibandingkan dengan hasil pada [20] yang menggunakan *greedy randomized adaptive search procedure*.

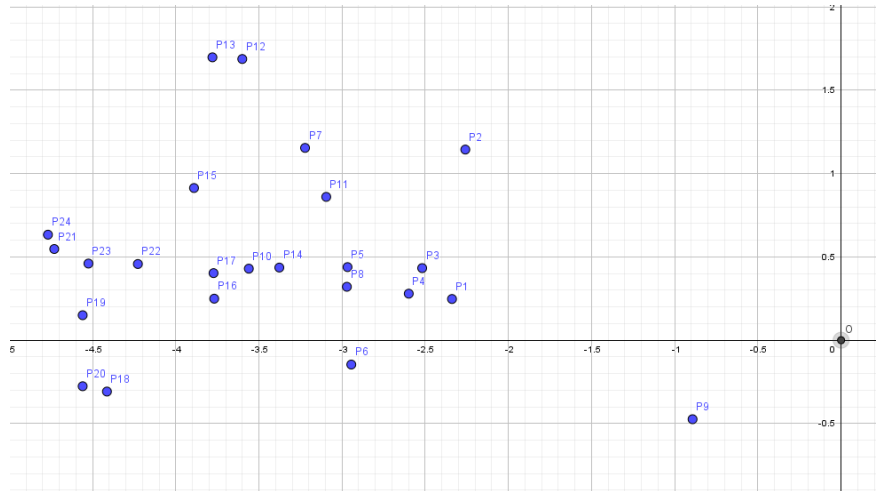
TABEL 1. Data permintaan konsumen

Konsumen ( $i$ )	Permintaan (boks)	Konsumen ( $i$ )	Permintaan (boks)
1	17	13	12
2	3	14	22
3	7	15	15
4	17	16	9
5	8	17	4
6	12	18	11
7	19	19	18
8	17	20	18
9	2	21	37
10	19	22	18
11	37	23	25
12	16	24	18

## 3. HASIL DAN PEMBAHASAN

**3.1. Clustering Menggunakan Algoritme Sweep.** Penggunaan algoritme *sweep* dimaksudkan untuk melakukan pengelompokan konsumen menjadi suatu kelompok konsumen yang memiliki kedekatan wilayah geografis. Selanjutnya dalam satu kelompok ini permintaan akan

didistribusikan menggunakan sebuah kendaraan. Dengan menggunakan bantuan *software Geogebra input* pada Gambar 1, menghasilkan *output* posisi semua konsumen dan depot (Gambar 2).



GAMBAR 2. *Output Geogebra* semua konsumen

Setelah semua titik sudah digambarkan, kemudian dengan bantuan *software Geogebra* setiap titik diubah menjadi sudut polar sehingga dihasilkan data sudut polar untuk setiap konsumennya seperti pada Tabel 2.

Tahap selanjutnya dilakukan *sweeping* dengan menggunakan algoritme *sweep* yang menghasilkan 3 kelompok yaitu:

- (1) Kelompok 1 terdiri atas konsumen 2, 12, 13, 7, 11, 15, 3, 5 dan 24.
- (2) Kelompok 2 terdiri atas konsumen 14, 10, 21, 8, 4, 17 dan 1.
- (3) Kelompok 3 terdiri atas konsumen 23, 22, 16, 19, 6, 20, 18 dan 9.

TABEL 2. Data sudut polar semua konsumen

Konsumen ( <i>i</i> )	Sudut Polar (°)	Konsumen ( <i>i</i> )	Sudut Polar (°)
1	173.96	13	155.81
2	153.15	14	172.65
3	170.25	15	166.79
4	173.87	16	176.22
5	171.59	17	173.91
6	182.84	18	183.99
7	160.3	19	178.12
8	173.84	20	183.47
9	207.95	21	173.4
10	173.12	22	174.83
11	164.47	23	174.19
12	154.89	24	172.44

**3.2. Penyelesaian Menggunakan Algoritme *Particle Swarm Optimization*.** Pada tahap ini setiap kelompok dicari rute terpendek menggunakan algoritme PSO. Pencarian rute optimal setiap kelompok menggunakan nilai parameter *N* (jumlah swarm) dan maksimum iterasi yang berbeda-beda. Selain itu ada parameter lain yang ditentukan dari awal untuk pencarian rute dengan menggunakan algoritme PSO yaitu  $c_1 = 2, c_2 = 2$ , dapat juga dipilih nilai  $c_1$  dan  $c_2$

yang lain. Kapasitas maksimum untuk kendaraan adalah 135 boks.

### Kelompok I

Kelompok 1 terdiri atas 9 konsumen dan 1 depot. Depot diberi indeks 1 sedangkan masing-masing konsumen diberi indeks sebagai berikut:

Konsumen	2	12	13	7	11	15	3	5	24
Indeks	2	3	4	5	6	7	8	9	10

Pada pencarian rute di kelompok 1 dilakukan percobaan menggunakan parameter sebagai berikut:

TABEL 3. Hasil percobaan dengan 30 *swarm* dan 500 iterasi pada data kelompok 1

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	9-1-8-7-10-3-4-6-5-2-9	95	135	0.1406
2	3-4-6-5-9-2-8-1-10-7-3	91	135	0.2031
3	4-3-6-7-10-9-2-1-8-5-4	91	135	0.1563
4	7-10-8-1-2-9-5-6-4-3-7	90	135	0.1875
5	7-9-2-1-8-5-6-4-3-10-7	93	135	0.1563
<b>6</b>	<b>3-10-7-5-9-8-1-2-6-4-3</b>	<b>87</b>	<b>135</b>	<b>0.1563</b>
7	3-4-6-2-1-5-9-8-7-10-3	96	135	0.1719
8	7-10-9-5-6-8-1-2-4-3-7	94	135	0.1563
9	3-6-7-10-9-5-8-1-2-4-3	91	135	0.1719
10	10-7-3-4-5-9-8-1-2-6-10	91	135	0.125

TABEL 4. Hasil percobaan dengan 50 *swarm* dan 500 iterasi pada data kelompok 1

Running ke-	Rute Optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	10-1-2-3-4-6-8-9-5-7-10	95	135	0.1719
2	2-1-8-10-7-3-4-6-5-9-2	90	135	0.1719
3	7-10-3-4-6-2-1-8-9-5-7	87	135	0.2188
4	1-10-7-6-4-3-2-9-5-8-1	93	135	0.2344
5	6-4-3-10-7-5-8-1-2-9-6	89	135	0.1875
6	6-2-9-8-1-10-7-5-3-4-6	93	135	0.2344
7	4-5-9-2-1-8-6-10-7-3-4	92	135	0.2344
8	9-10-7-5-6-4-3-2-1-8-9	91	135	0.25
9	10-7-5-8-1-9-2-6-4-3-10	90	135	0.2344
10	5-3-4-6-2-1-8-9-10-7-5	91	135	0.2344

Setelah melakukan beberapa kali percobaan dengan jumlah *swarm* dan maksimum iterasi yang berbeda-beda, diperoleh total jarak pendekatan terkecil distribusi pada kelompok 1 sebesar **87 km** dengan waktu eksekusi **0.1563 s** yakni pada percobaan dengan 30 *swarm* dan 500 iterasi. Rute kendaraan yang diperoleh adalah **0-2-11-13-12-24-15-7-5-3-0** dengan membawa muatan 135 boks.

### Kelompok II

Kelompok 2 terdiri atas 7 konsumen dan 1 depot. Depot diberi indeks 1 sedangkan masing-masing konsumen diberi indeks sebagai berikut:

TABEL 5. Hasil percobaan dengan 30 *swarm* dan 1000 iterasi pada data kelompok 1

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	7-10-6-4-3-2-9-1-8-5-7	92	135	0.25
2	2-9-5-7-10-1-8-3-4-6-2	96	135	0.3125
3	9-3-4-6-10-7-5-2-1-8-9	94	135	0.2656
4	6-10-7-3-4-2-1-8-9-5-6	91	135	0.3125
5	9-2-4-3-6-10-7-5-8-1-9	92	135	0.2656
6	10-7-3-4-6-5-8-1-2-9-10	89	135	0.25
7	2-5-6-4-3-7-10-9-8-1-2	91	135	0.25
8	10-3-4-2-1-8-9-5-6-7-10	91	135	0.2656
9	2-1-8-5-7-10-6-4-3-9-2	92	135	0.2656
10	6-4-3-5-7-10-9-2-1-8-6	92	135	0.2656

TABEL 6. Hasil percobaan dengan 50 *swarm* dan 1000 iterasi pada data kelompok 1

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (km)	Running time (detik)
1	6-3-4-2-1-8-7-10-9-5-6	94	135	0.4063
2	2-1-8-7-10-3-4-5-6-9-2	96	135	0.3281
3	8-1-9-10-7-5-6-4-3-2-8	94	135	0.3594
4	7-10-9-5-8-1-2-6-4-3-7	89	135	0.3438
5	9-8-1-2-5-3-4-6-7-10-9	93	135	0.5469
6	8-1-2-9-5-7-10-6-4-3-8	92	135	0.3281
7	9-2-1-8-5-7-10-3-4-6-9	89	135	0.375
8	10-3-4-6-5-9-2-1-8-7-10	90	135	0.4375
9	3-4-6-2-1-8-9-5-7-10-3	87	135	0.3438
10	8-1-2-4-3-6-7-10-9-5-8	91	135	0.4688

Konsumen	14	10	21	8	4	17	1
Indeks	2	3	4	5	6	7	8

Pada pencarian rute di kelompok 2 dilakukan percobaan menggunakan parameter sebagai berikut:

TABEL 7. Hasil percobaan dengan 30 *swarm* dan 500 iterasi pada data kelompok 2

Running ke-	Rute optimum	Total Jarak (km)	Kapasitas (boks)	Running time (detik)
1	2-6-1-8-5-7-4-3-2	71	133	0.1563
<b>2</b>	<b>7-4-3-2-5-8-1-6-7</b>	<b>71</b>	<b>133</b>	<b>0.1406</b>
3	4-3-2-8-1-6-5-7-4	73	133	0.1563
4	6-5-2-3-7-4-1-8-6	72	133	0.1875
5	4-7-3-2-5-6-8-1-4	72	133	0.2188
6	2-5-6-8-1-4-7-3-2	72	133	0.1406
7	5-7-4-3-2-6-1-8-5	71	133	0.1406
8	2-3-4-7-6-1-8-5-2	71	133	0.1719
9	8-1-6-2-3-4-7-5-8	71	133	0.1406
10	2-3-4-7-6-1-8-5-2	71	133	0.1094

TABEL 8. Hasil percobaan dengan 50 *swarm* dan 500 iterasi pada data kelompok 2

Running ke-	Rute Optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	6-5-2-3-7-4-1-8-6	72	133	0.25
2	1-4-7-3-2-5-6-8-1	72	133	0.2031
3	7-4-3-2-5-8-1-6-7	71	133	0.2188
4	2-3-4-7-6-1-8-5-2	71	133	0.1719
5	4-7-6-1-8-5-2-3-4	71	133	0.2344
6	4-3-2-6-1-8-5-7-4	71	133	0.2656
7	6-1-8-2-3-7-4-5-6	74	133	0.25
8	8-1-6-7-4-3-2-5-8	71	133	0.2656
9	2-3-4-7-5-8-1-6-2	71	133	0.1875
10	5-8-1-6-2-3-7-4-5	72	133	0.25

TABEL 9. Hasil percobaan dengan 30 *swarm* dan 1000 iterasi pada data kelompok 2

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	3-2-6-1-8-5-7-4-3	71	133	0.2813
2	2-3-4-7-5-8-1-6-2	71	133	0.3125
3	2-3-4-7-6-1-8-5-2	71	133	0.2969
4	1-8-5-7-4-3-2-6-1	71	133	0.375
5	7-5-8-1-6-2-3-4-7	71	133	0.2344
6	4-7-5-8-1-6-2-3-4	71	133	0.2344
7	7-6-1-8-5-2-3-4-7	71	133	0.2813
8	7-4-3-2-6-1-8-5-7	71	133	0.2656
9	6-1-8-5-2-3-4-7-6	71	133	0.2188
10	6-1-8-5-7-4-3-2-6	71	133	0.375

TABEL 10. Hasil percobaan dengan 50 *swarm* dan 1000 iterasi pada data kelompok 2

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	4-5-8-1-6-2-3-7-4	72	133	0.3594
2	8-5-7-4-3-2-6-1-8	71	133	0.3438
3	6-8-1-4-7-3-2-5-6	72	133	0.3594
4	3-4-7-6-1-8-5-2-3	71	133	0.5313
5	5-6-1-8-2-3-4-7-5	73	133	0.4688
6	6-7-4-3-2-5-8-1-6	71	133	0.4531
7	2-3-4-7-6-1-8-5-2	71	133	0.4844
8	1-6-7-4-3-2-5-8-1	71	133	0.4219
9	8-5-7-4-3-2-6-1-8	71	133	0.4375
10	5-2-3-4-7-6-1-8-5	71	133	0.3906

Setelah melakukan beberapa kali percobaan dengan jumlah *swarm* dan maksimum iterasi yang berbeda-beda, diperoleh total jarak pendekatan terkecil distribusi pada kelompok 2 sebesar **71 km** dengan waktu eksekusi **0.1406 s** yakni pada percobaan dengan 30 *swarm* dan 500 iterasi. Rute kendaraan yang diperoleh adalah **0-4-17-21-10-14-8-1-0** dengan dengan



membawa muatan 133 boks.

**Kelompok III**

Kelompok 3 terdiri atas 8 konsumen dan 1 depot. Depot diberi indeks 1 sedangkan masing-masing konsumen diberi indeks sebagai berikut:

Konsumen	23	22	16	19	6	20	18	9
Indeks	2	3	4	5	6	7	8	9

Pada pencarian rute di kelompok 3 dilakukan percobaan menggunakan parameter sebagai berikut:

TABEL 11. Hasil percobaan dengan 30 *swarm* dan 500 iterasi pada data kelompok 3

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	8-7-5-2-3-4-6-1-9-8	79	113	0.1563
2	4-6-1-9-3-2-5-7-8-4	80	113	0.0938
3	9-1-6-7-8-5-2-3-4-9	83	113	0.1719
4	<b>5-7-8-2-3-4-6-1-9-5</b>	<b>79</b>	<b>113</b>	<b>0.1406</b>
5	6-1-9-8-7-5-2-3-4-6	79	113	0.1719
6	4-6-1-9-3-2-7-8-5-4	81	113	0.1563
7	8-4-6-1-9-3-2-5-7-8	80	113	0.1719
8	6-4-3-2-5-7-8-9-1-6	79	113	0.1719
9	5-1-9-6-4-3-2-7-8-5	84	113	0.2969
10	5-7-8-2-3-4-9-1-6-5	82	113	0.1563

TABEL 12. Hasil percobaan dengan 50 *swarm* dan 500 iterasi pada data kelompok 3

Running ke-	Rute Optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	8-5-2-3-9-1-6-4-7-8	81	113	0.1719
2	7-5-2-3-4-6-9-1-8-7	82	113	0.2188
3	5-3-2-4-6-1-9-8-7-5	83	113	0.1875
4	4-3-2-5-8-7-9-1-6-4	80	113	0.2656
5	3-4-6-1-9-8-7-5-2-3	79	113	0.25
6	1-9-3-2-8-7-5-4-6-1	80	113	0.2813
7	7-5-2-3-4-6-9-1-8-7	82	113	0.3125
8	3-2-5-7-8-9-1-6-4-3	79	113	0.2301
9	2-3-9-1-6-4-7-8-5-2	81	113	0.1563
10	8-9-1-6-4-5-3-2-7-8	84	113	0.2344

Setelah melakukan beberapa kali percobaan dengan jumlah *swarm* dan maksimum iterasi yang berbeda-beda, diperoleh total jarak pendekatan terkecil pada kelompok 3 sebesar **79 km** dengan waktu eksekusi **0.1406 s** yakni pada percobaan dengan 30 *swarm* dan 500 iterasi. Rute kendaraan yang diperoleh adalah **0-9-19-20-18-23-22-16-6-0** dengan membawa muatan 113 boks.

Dalam studi kasus pendistribusian ini, algoritme *sweep* membagi konsumen menjadi tiga yaitu kelompok 1, 2 dan 3 yang masing-masing terdiri dari 9 konsumen, 7 konsumen dan 8 konsumen. Kemudian pada masing-masing diterapkan algoritme PSO, sehingga untuk kelompok 1, 2 dan 3 diperoleh total jarak tempuh optimal masing-masing 87 km, 71 km dan 79 km. Total

TABEL 13. Hasil percobaan dengan 30 *swarm* dan 1000 iterasi pada data kelompok 3

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	5-7-8-9-1-6-4-3-2-5	79	113	0.2969
2	5-8-7-4-6-1-9-2-3-5	84	113	0.4219
3	7-8-5-2-3-4-9-1-6-7	83	113	0.2301
4	3-4-6-1-9-8-7-5-2-3	79	113	0.2969
5	9-3-5-7-8-2-4-6-1-9	84	113	0.375
6	9-1-8-7-5-2-3-4-6-9	82	113	0.2969
7	7-8-2-4-6-1-9-3-5-7	84	113	0.4063
8	9-1-6-4-3-2-8-7-5-9	79	113	0.2656
9	7-8-9-1-6-4-3-2-5-7	79	113	0.3281
10	2-8-7-5-6-1-9-4-3-2	82	113	0.3125

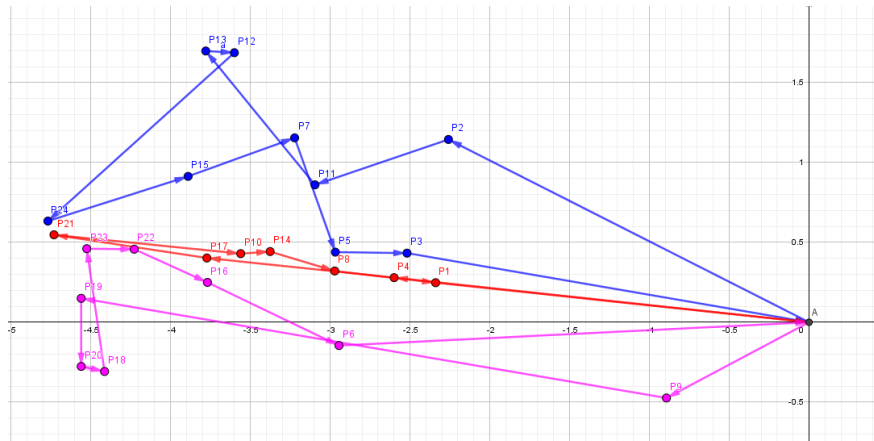
TABEL 14. Hasil percobaan dengan 50 *swarm* dan 1000 iterasi pada data kelompok 3

Running ke-	Rute optimum	Total jarak (km)	Kapasitas (boks)	Running time (detik)
1	8-7-9-1-6-4-3-2-5-8	80	113	0.5313
2	9-1-6-4-3-2-5-8-7-9	80	113	0.3594
3	2-8-7-5-9-1-6-4-3-2	79	113	0.5156
4	8-7-5-2-3-9-1-6-4-8	80	113	0.4531
5	8-3-4-6-1-9-2-5-7-8	83	113	0.4375
6	9-1-6-4-3-2-5-8-7-9	80	113	0.4375
7	1-9-2-8-7-5-3-4-6-1	83	113	0.375
8	9-1-6-3-2-8-7-5-4-9	81	113	0.2188
9	6-1-9-5-7-8-2-3-4-6	79	113	0.4844
10	5-7-8-1-9-6-4-3-2-5	82	113	0.3125

jarak keseluruhan adalah 237 km. Tampilan rute yang diperoleh diberikan pada Gambar 3, dengan jalur berwarna biru merupakan kelompok 1, merah merupakan kelompok 2, dan ungu untuk kelompok 3. Hasil yang diperoleh lebih baik dari pada rute optimal dengan menggunakan algoritme *greedy randomized adaptive search procedure* oleh Wahyuni dengan total jarak 267 km [20].

#### 4. SIMPULAN

Capacitated *vehicle routing problem* adalah variasi dari *vehicle routing problem*. Pada capacitated *vehicle routing problem*, kendaraan yang digunakan untuk distribusi produk memiliki batas daya angkut. Pada paper ini, metode heuristik digunakan untuk mencari solusi capacitated *vehicle routing problem*. Penyelesaian masalah dimulai dengan pembentukan kelompok menggunakan algoritme *sweep*, kemudian setiap kelompok hasil algoritme *sweep* dioptimalkan menggunakan *particle swarm optimization*. Dalam kasus yang dipilih, algoritme *sweep* membagi konsumen menjadi tiga. Kelompok 1 terdiri atas 9 konsumen, kelompok 2 terdiri atas 7 konsumen dan kelompok 3 terdiri atas 8 konsumen. Dengan menggunakan algoritme PSO dihasilkan total jarak tempuh optimal untuk kelompok 1 sampai 3 berturut-turut yaitu 87 km, 71 km dan 79 km, dengan total jarak adalah 237 km. Hasil yang diperoleh lebih baik dari pada rute optimal pada penelitian sebelumnya dengan menggunakan algoritme *greedy randomized adaptive search procedure*.



GAMBAR 3. Rute optimal yang diperoleh dengan algoritme *sweep* dan PSO

DAFTAR PUSTAKA

- [1] Muhammad Muhlis Al Kautsar, Bib Paruhum Silalahi, and Sugi Guritman. Paradox in a d-dimensional transportation problem. *International Journal of Pure and Applied Mathematics*, 118(3):547–557, 2018.
- [2] Franki Yusuf Bisilisin, Yeni Herdiyeni, and Bib Paruhum Silalahi. Optimasi k-means clustering menggunakan particle swarm optimization pada sistem identifikasi tumbuhan obat berbasis citra. 2014.
- [3] Billy E Gillett and Leland R Miller. A heuristic algorithm for the vehicle-dispatch problem. *Operations research*, 22(2):340–349, 1974.
- [4] S Idaman, BP Silalahi, and S Guritman. Penyelesaian arah vektor gradien untuk menentukan step size metode steepest descent pada fungsi nonlinear kuadratik banyak variabel. *Journal of Mathematics and Its Applications*, 17(1):47–60, 2018.
- [5] D Lalang, BP Silalahi, and F Bukhari. Vehicle routing problem time windows dengan pengemudi sesekali. *Journal of Mathematics and Its Applications*, 17(2):87–99, 2018.
- [6] SRM Making, BP Silalahi, and F Bukhari. Multi depot vehicle routing problem dengan pengemudi sesekali. *Journal of Mathematics and Its Applications*, 17(1):75–86, 2018.
- [7] H Mayyani, BP Silalahi, and A Aman. Frequency determination of bus rapid transit (brt) applied on service system of trans mataram metro bus to minimize the operational cost. *International Journal of Engineering and Management Research (IJEMR)*, 7(6):134–140, 2017.
- [8] DA Pratamasyari, BP Silalahi, and S Guritman. Kombinasi varian metode newton dan metode halley untuk menyelesaikan persamaan tak linier. *Journal of Mathematics and Its Applications*, 16(2):1–12, 2017.
- [9] D Rahmalia and AM Rohmah. Optimisasi perencanaan produksi pupuk menggunakan firefly algorithm. *Jurnal Matematika MANTIK*, 4(1), 2018.
- [10] Bib Paruhum Silalahi. Sharper analysis of upper bound for the iteration complexity of an interior-point method using primal-dual full-newton step algorithm. *Far East Journal of Mathematical Sciences*, 95(1):69–80, 2014.
- [11] Bib Paruhum Silalahi. Evaluation of interior-point method in scilab. In *IOP Conference Series: Earth and Environmental Science*, volume 299, page 012040. IOP Publishing, 2019.
- [12] Bib Paruhum Silalahi, Fahren Bukhari, Amril Aman, Elis Khatizah, and Nadhif Azmi Fahlevi. Comparison of interior point method execution time in solving linear optimization problems using mathematica and scilab. *International Journal of Statistics Economics*, 20(4):74–81, 2019.
- [13] Bib Paruhum Silalahi, Nurul Fathiah, and Prapto Tri Supriyo. Use of ant colony optimization algorithm for determining traveling salesman problem routes.
- [14] Bib Paruhum Silalahi, Djihad Wungguli, and Sugi Guritman. Steepest descent method with new step sizes. *World Academy of Science, Engineering and Technology, International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, 9(7):378–384, 2015.
- [15] BP Silalahi, R Laila, and IS Sitanggang. A combination method for solving nonlinear equations. In *IOP Conference Series: Materials Science and Engineering*, volume 166, page 012011. IOP Publishing, 2017.
- [16] BP Silalahi, S SISWANDI, and A AMAN. Tinjauan terhadap metode pengoptimasian pendekatan newton. *Journal of Mathematics and Its Applications*, 17(2):141–155, 2018.
- [17] Paolo Toth and Daniele Vigo. *The vehicle routing problem*. SIAM, 2002.
- [18] Frans Van Den Bergh and Andries Petrus Engelbrecht. A study of particle swarm optimization particle trajectories. *Information sciences*, 176(8):937–971, 2006.
- [19] SR Venkatesan, D Logendran, and D Chandramohan. Optimization of capacitated vehicle routing problem using pso. *International Journal of Engineering Science and Technology (IJEST)*, 3(10):7469–7477, 2011.

- [20] V Wahyuni. Penyelesaian capacited vehicle routing problem menggunakan greedy randomized adaptive search procedure [skripsi]. Bogor (ID): Institut Pertanian Bogor, 2015.
- [21] FD Wihartiko, A Buono, and BP Silalahi. Integer programming model for optimizing bus timetable using genetic algorithm. In *IOP Conference Series: Materials Science and Engineering*, volume 166, page 012016. IOP Publishing, 2017.