

# Penentuan Rute Optimal Penyiraman Tanaman Kota Yogyakarta Menggunakan *Genetic Algorithm*

Wuri Isdianto<sup>1</sup>, Utaminingsih Linarti<sup>\*2</sup>

<sup>1,2</sup>Program Studi Teknik Industri, Fakultas Teknologi Industri, Universitas Ahmad Dahlan

Jl. Ringroad Selatan, Krangilan, Tamanan, Bantul, Daerah Istimewa Yogyakarta 55191

Email: wuriisdianto@gmail.com<sup>1</sup>, utaminingsih.linarti@ie.uad.ac.id<sup>2</sup>

## Abstrak

Penyiraman tanaman merupakan salah satu aktivitas yang harus dilakukan oleh DLH Kota Yogyakarta untuk menjaga dan memelihara kelestarian lingkungan hidup. Pemilihan rute penyiraman harus optimal, agar menghemat biaya operasional kendaraan. Tujuan penelitian ini yaitu menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta menggunakan *Genetic Algorithm* (GA). Fungsi tujuan pada penelitian ini untuk meminimalkan total jarak dan total waktu tertempuh. Penelitian ini berhasil menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta di Sektor 3 dengan menggunakan GA dan mampu memberikan solusi rute dengan hasil yang lebih baik daripada rute saat ini. Dengan menggunakan rute usulan, diperoleh total jarak tertempuh sejauh 80.220 meter dan total waktu penyelesaian sebesar 1.056 menit. Terjadi penurunan total jarak tempuh sebesar 11,94 % dan penurunan total waktu tertempuh sebesar 2,88 % terhadap rute saat ini.

**Kata kunci:** *Genetic Algorithm*, DLH, Penyiraman Tanaman

## Abstract

Watering plants is one of the activities that must be carried out by DLH Yogyakarta City to protect and preserve the environment. The selection of the watering route must be optimal, in order to save vehicle operating costs. The purpose of this research is to solve the problem of watering route for plants in Yogyakarta City using *Genetic Algorithm* (GA). The objective function in this study is to minimize the total distance and total time traveled. This study succeeded in solving the problem of the Yogyakarta City watering route in Sector 3 using GA and was able to provide route solutions with better results than the current route. By using the proposed route, the total distance covered is 80,220 meters and the total completion time is 1,056 minutes. There is a decrease in the total distance traveled by 11.94 % and a decrease in the total time traveled by 2.88% to the current route.

**Keywords:** *Genetic Algorithm*, DLH, Watering Plants

## 1. Pendahuluan

Dinas Lingkungan Hidup Kota Yogyakarta atau yang disingkat DLH Kota Yogyakarta merupakan dinas pemerintahan di sektor lingkungan hidup yang mempunyai tugas membuat program, memelihara, mengelola dan melindungi kelestarian lingkungan hidup di Kota Yogyakarta. Penyiraman tanaman merupakan salah satu aktivitas yang harus dilakukan oleh DLH Kota Yogyakarta untuk menjaga dan memelihara kelestarian lingkungan hidup.

Wilayah penyiraman di Kota Yogyakarta terdiri dari 3 sektor, yaitu sektor 1, sektor 2, dan sektor 3. Pada sektor 1 terdapat 34 taman dengan total luas taman sebesar 22.079 m<sup>2</sup>. Sektor 2 terdapat 43 taman

dengan total luas taman sebesar 25.636 m<sup>2</sup> dan pada sektor 3 terdapat 31 taman dengan total luas taman sebesar 16.701 m<sup>2</sup>. DLH Kota Yogyakarta memiliki delapan truk tangki penyiraman, dengan 2 varian ukuran truk. Enam truk tangki mempunyai kapasitas 5000 liter dan dua truk tangki berkapasitas 2500 liter. Satu truk tangki dengan kapasitas 5000 liter dapat menyiram taman seluas 700 m<sup>2</sup>, sedangkan 1 truk tangki dengan kapasitas 2500 liter dapat menyiram taman seluas 350 m<sup>2</sup>. Tanaman-tanaman di Kota Yogyakarta dilakukan penyiraman setiap 2 hari sekali. Dalam satu hari terdapat 2 shift penyiraman yaitu pada pagi dan sore hari.

---

\* Penulis korespondensi

Setiap truk penyiraman yang telah berisi air ditugaskan untuk melakukan penyiraman tanaman berangkat dari depot menuju ke lokasi-lokasi penyiraman, apabila muatan air di dalam truk habis, maka melakukan pengisian air hingga kapasitas muatan terpenuhi. Setelah truk sudah terisi air maka truk dapat kembali ke depot maupun melakukan penyiraman kembali pada lokasi yang belum terlayani tanpa melanggar batasan waktu yang telah ditetapkan.

Permasalahan penentuan rute distribusi yang optimal dengan berbagai kendala seperti kendaraan dengan kapasitas terbatas, adanya batasan waktu untuk melakukan pelayanan, adanya fasilitas untuk memperbaharui kapasitas merupakan permasalahan *Vehicle Routing Problem*. Menurut Toth & Vigo, (2002) *Vehicle Routing Problem* (VRP) merupakan permasalahan penentuan rute kendaraan dalam mendistribusikan barang dari tempat produksi/depot ke sejumlah *customer* dengan tujuan untuk meminimalkan total jarak tempuh kendaraan.

Model dasar VRP yaitu kendaraan yang ditugaskan adalah homogen dan titik layanan hanya boleh dilayani tepat satu kali dalam satu rute pelayanan dengan fungsi tujuan meminimumkan jarak tempuh yang diasosiasikan dengan biaya (Deng et al., 2006). Penelitian terdahulu yang menerapkan model VRP dasar dilakukan oleh Sarwandi & Anjar, (2004), Indrawati et al., (2016), Pakusadewa et al., (2017).

Model VRP dasar terlalu sederhana untuk permasalahan penyiraman tanaman Kota Yogyakarta. Model VRP dasar tidak mempertimbangkan adanya fasilitas antara atau *intermediate facility*, kendaraan dengan kapasitas berbeda, kondisi yang memungkinkan suatu kendaraan untuk melayani lebih dari satu rute dalam satu periode perencanaan penugasan kendaraan. Penelitian yang dilakukan oleh Sari & Sari, (2017) yaitu melakukan penelitian tentang rute penyiraman tanaman di Kota Yogyakarta dengan menggunakan algoritma *Artificial Imune System*. Penelitian Sari & Sari, (2017) yang dilakukan pada tahun 2017 hanya terdapat satu variansi ukuran truk penyiraman, sedangkan pada tahun 2020, DLH sudah mempunyai dua variansi ukuran truk, sehingga pada penelitian ini mengembangkan model penelitian VRP dengan kapasitas truk yang berbeda.

Model VRP yang mempertimbangkan adanya kendaraan dengan kapasitas berbeda disebut sebagai *Heterogeneous Fleet Vehicle Routing Problem* (HFVRP) umumnya mempunyai fungsi tujuan meminimumkan total jarak ditempuh kendaraan (Prins, 2002). Gendreau et al., (1999), Prins, (2002)

dan Hendrawan, (2007), juga telah melakukan penelitian dengan model *Heterogeneous Fleet Vehicle Routing Problem* (HFVRP).

Kendaraan truk penyiraman tanaman Kota Yogyakarta mempunyai rute lebih dari satu kali dalam melakukan penyiraman, sehingga ini termasuk permasalahan *multiple trips*. Menurut Arvianto et al., (2014) batasan *multiple trips* adalah batasan yang memungkinkan suatu kendaraan untuk memiliki lebih dari satu rute dalam satu periode perencanaan penugasan kendaraan. Penugasan kendaraan dalam melakukan penyiraman tanaman dari depot menuju tempat pengisian air lalu melakukan penyiraman dan kembali melakukan pengisian air maupun kembali ke depot adalah persoalan penentuan rute yang mempertimbangkan adanya tempat pengisian air sebagai *intermediate facility*. Menurut Angelelli & Speranza, (2002), *intermediate facility* adalah fasilitas yang harus dikunjungi oleh suatu kendaraan untuk memperbarui kapasitas kendaraan setiap kali kendaraan akan memulai rute baru. Oleh karena itu, model *Vehicle Routing Problem with Multiple Trips and Intermediate Facility* (VRPMTIF) sesuai dengan permasalahan penentuan rute penyiraman tanaman Kota Yogyakarta.

Adanya batasan waktu (*time windows*) dalam melakukan penyiraman tanaman merupakan perluasan model VRP. Jenis VRP dengan adanya batasan *time windows* dikenal dengan *Vehicle Routing Problem with Time Windows* (VRPTW). Menurut Wahyuningsih et al., (2019) batasan *Time Windows* merupakan waktu yang diperbolehkan untuk melakukan pelayanan pada *customer*. Sehingga model *Vehicle Routing Problem with Time Windows* (VRPTW) sesuai dengan permasalahan rute penyiraman tanaman Kota Yogyakarta yaitu dengan adanya batasan waktu penyiraman. Penyiraman tanaman Kota Yogyakarta dilakukan di pagi hari pukul 05.00-10.00 WIB dan di sore hari pukul 15.30-18.30 WIB. Beberapa penelitian yang menggunakan permasalahan VRPTW diantaranya Salaki, (2009), pada persoalan penentuan rute pendistribusian sari roti dan Amri et al., (2014), melakukan penelitian tentang rute pendistribusian minuman ringan dengan menggunakan metode *Nearest Neighbor*.

Beberapa lokasi penyiraman tanaman Kota Yogyakarta dilakukan lebih dari satu kali penugasan kendaraan, karena permintaan air lebih besar dari pada kapasitas truk penyiraman. Permasalahan penugasan kendaraan ke suatu konsumen lebih dari satu kali merupakan permasalahan *split delivery*. Menurut

Archetti et al., (2006) *split delivery* merupakan kondisi dimana kendaraan dapat melayani konsumen lebih dari satu kali kunjungan, karena permintaan konsumen lebih besar dari kapasitas kendaraan.

Penelitian terdahulu dengan permasalahan VRP yang hampir mirip permasalahan penelitian ini yaitu pada penelitian (Armandi et al., 2019). Jenis VRP yang terdapat pada penelitian Armandi et al., (2019), yaitu *Heterogeneous Fleet, Intermediate Facility, Multiple Trips* dan *Split Delivery*. Pada penelitian tersebut, tidak terdapat batasan waktu/ *time windows*, sedangkan pada penelitian ini terdapat batasan waktu penyiraman/ *time windows*.

*Genetic algorithm* adalah algoritma optimasi yang terinspirasi oleh teori Darwin dengan judul "*Theory of Natural Selection*". Pada Teori tersebut dijelaskan bahwa individu-individu yang mempunyai kualitas yang bagus akan mempunyai peluang besar untuk bertahan hidup dan bereproduksi sehingga mewariskan karakteristiknya kepada keturunan-keturunannya. Sedangkan individu-individu yang mempunyai kualitas kurang bagus, maka akan tersingkir dari populasi secara perlahan (Hendrawan, 2007).

Penelitian tentang permasalahan VRP yang diselesaikan dengan *Genetic algorithm* telah banyak dilakukan. Sarwandi & Anjar, (2004) melakukan penelitian permasalahan VRP dasar menggunakan *Genetic algorithm* yang dibandingkan dengan metode *Saving*. Penelitian Sarwandi & Anjar, (2004) diperoleh hasil bahwa *Genetic algorithm* mempunyai solusi yang lebih baik untuk jumlah titik yang kecil dibandingkan dengan metode *Saving*. Surekha & Sumathi, (2011), melakukan penelitian permasalahan VRP multi depot yang diselesaikan dengan *genetic algorithm*. Hasil yang diperoleh pada penelitian Surekha & Sumathi, (2011), bahwa *genetic algorithm* sangat baik dan mempunyai waktu komputasi yang cepat untuk permasalahan VRP multi depot. Hendrawan, (2007), juga menggunakan *genetic algorithm* untuk permasalahan VRP *heterogeneous fleet*, diperoleh hasil bahwa *genetic algorithm* cukup efektif untuk menyelesaikan permasalahan VRP dengan skala besar.

Penelitian yang dilakukan Sari & Sari, (2017), dengan menggunakan model VRP yaitu *Capacitated Vehicle Routing Problem with Time Windows (CVRPTW)* yang diselesaikan dengan Algoritma *Artificial Immune System*. Asumsi yang dipakai penelitian Sari & Sari, (2017), yaitu kendaraan homogen. Penelitian yang dilakukan didasarkan dari

penelitian Sari & Sari, (2017) dengan mengembangkan model VRP rute penyiraman tanaman yang sesuai dengan kondisi saat ini, yaitu VRPHMTIFTWSD (*Vehicle Routing Problem Heterogeneous Fleet, Multiple Trips, Intermediate Facility, Time Windows and Split Delivery*). Tujuan dari penelitian ini yaitu untuk menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta menggunakan *Genetic Algorithm*. Penelitian ini juga membandingkan kinerja GA menggunakan *Nearest Neighbor* pada pembentukan solusi awal dan tanpa *Nearest Neighbor* pada pembentukan solusi awal.

## 2. Metode Penelitian

### 2.1 Karakteristik Sistem

Penelitian ini dilakukan pada sektor 3 penyiraman tanaman Kota Yogyakarta. Sektor 3 terletak di Kecamatan Umbulharjo, Kecamatan Pakualaman dan Kecamatan Kotagede. Pada sektor 3 *shift* pagi terdapat 25 lokasi penyiraman/konsumen dan terdapat 4 lokasi pengisian air. Untuk lokasi pengisian airnya ada di Jembatan Kalimambu, sungai belakang XT Square, selokan depan Kampus UST F. Pertanian dan selokan Jalan Veteran. Tanaman kota Yogyakarta disiram setiap 2 hari sekali, sehingga 25 lokasi penyiraman tersebut dibagi menjadi 2 bagian yaitu penyiraman hari pertama dan penyiraman hari kedua. Penyiraman hari pertama dilaksanakan setiap hari senin, rabu dan jumat, sedangkan penyiraman hari kedua dilaksanakan setiap hari selasa, Kamis dan Sabtu. Dapat dilihat pada Tabel 1 dan Tabel 2 tentang pembagian wilayah penyiraman. Pada sektor ini terdapat 2 variasi ukuran truk penyiraman yang digunakan yaitu 1 truk berkapasitas 5000 liter dan 2 truk berkapasitas 2500 liter.

Waktu penyiraman tanaman yaitu dari pukul 05.00 WIB sampai 10.00 WIB atau sama dengan 300 menit. Rentang Waktu tersebut merupakan batasan waktu untuk melakukan penyiraman tanaman. Truk penyiraman berangkat dari kantor DLH dengan truk yang sudah berisi air menuju lokasi-lokasi penyiraman. Apabila air dalam truk habis, maka truk akan mengisi air di tempat pengisian air atau yang disebut dengan *intermediate facility*. Setelah truk terisi air maka truk kembali melakukan penyiraman sampai lokasi tanaman tersiram semua dengan tidak melanggar batasan waktu penyiraman.

**Tabel 1.** Data Lokasi Penyiraman pada Hari Pertama

No	Lokasi Penyiraman	Kebutuhan Air (liter)	Kode Lokasi
1	Taman Jl. Kusuma Negara (bagian timur)	5936	E
2	Taman Jl. Kerto	8571	G
3	Taman Muja-Muju	929	H
4	Taman Jl. Veteran	10086	I
5	Taman Jl. Gambiran	3036	J
6	Taman Jl. Segitiga Veteran	1857	K
7	Taman Jl. Cendana	14214	Q
8	Taman Jl. Cendana II	1500	R
9	Taman Jl. Kenari (depan SMK 6)	1286	S
10	Taman Jl. Sukonandi	3771	X
11	Taman Jl. Sukonandi II	1429	Y
12	Jembatan Kalimambu	-	I1
13	Sungai Belakang XT Square	-	I2
14	Selokan depan Kampus UST F. Pertanian	-	I3
15	Selokan Jl. Veteran	-	I4

**Tabel 2.** Data Lokasi Penyiraman pada Hari Kedua

No	Lokasi Penyiraman	Kebutuhan Air (liter)	Kode Lokasi
1	Taman Jl. Sultan Agung	5357	A
2	Taman depan Pakualaman	900	B
3	Taman Ki Mangun Sarkoro	71	C
4	Taman Jl. Kusuma Negara (bagian barat)	5936	D
5	Taman Jl. Soga	2500	F
6	Taman Jl. Karang	4286	L
7	Taman Jl. Tegalturi	857	M
8	Taman Jl. Median Terminal Giwangan	929	N
9	Taman Jl. Lowano	1071	O
10	Taman Jl. Kapas	7714	P
11	Taman Jl. Soka	5000	T
12	Taman Jl. Tunjung	1786	U
13	Taman Jl. Mawar	714	V
14	Taman Jl. Cantel	571	W
15	Jembatan Kalimambu	-	I1
16	Sungai Belakang XT Square	-	I2
17	Selokan depan Kampus UST F. Pertanian	-	I3
18	Selokan Jl. Veteran	-	I4

## 2.2 Membuat Model VRP

Model VRP yang digunakan pada penelitian ini yaitu model yang sesuai dengan kondisi yang ditemui di lapangan. Model VRP penelitian ini mengadopsi penelitian Armandi et al., (2019), dengan fungsi tujuan untuk meminimalkan total jarak tempuh dan total waktu penyiraman. Jenis VRP yang digunakan pada penelitian ini yaitu:

- Heterogeneous Fleet*, yaitu kendaraan yang digunakan dalam melakukan penyiraman mempunyai kapasitas yang berbeda dan tidak

mempertimbangkan biaya tetap maupun biaya variabel.

- Multiple Trips*, yaitu setiap kendaraan yang digunakan untuk melakukan penyiraman memungkinkan melayani lebih dari satu rute dalam suatu periode perencanaan penugasan kendaraan.
- Intermediate Facility*, yaitu adanya fasilitas yang digunakan untuk melakukan pengisian air sebelum memulai perjalanan baru selama tidak melebihi batasan waktu yang diijinkan/ *time windows*.
- Time windows*, yaitu batasan waktu yang harus digunakan untuk melakukan penyiraman tanaman. Waktu yang digunakan untuk melakukan pelayanan yaitu dari pukul 05.00-10.00 WIB, atau selama 300 menit.
- Split delivery*, yaitu suatu kondisi dimana lokasi penyiraman tanaman memungkinkan untuk dilayani lebih dari satu kali oleh kendaraan yang sama atau kendaraan yang berbeda. Hal ini karena permintaan/*demand* lebih besar daripada kapasitas truk penyiraman.

## 2.3 Model Matematis

Berdasarkan permasalahan rute penyiraman tanaman di sektor 3 Kota Yogyakarta, maka model matematis yang digunakan pada penelitian ini yaitu:

Himpunan Indeks:

$V = \{1,2,\dots,n\}$  himpunan seluruh titik layanan dengan  $i = 1$  sebagai depot

$F = \{1,2,\dots,o\}$  himpunan *intermediate facility*  $f$  dengan jumlah  $o$

$K = \{1,2,\dots,m\}$  himpunan kendaraan  $k$  dengan kapasitas  $Q_k$ ,

$L = \{1,2,\dots,l\}$  = himpunan rute  $l$ ,

Parameter:

$V_1$  = depot

$f$  = *intermediate facility*

$d_i$  = permintaan air pada titik  $i$  ( $i \in V$ )

$t_{ij}$  = waktu tempuh (menit)

$c_{ij}$  = jarak tempuh (meter)

$p$  = waktu pengisian air/ *loading* (liter/menit)

$q$  = waktu pelayanan (liter/menit)

$m$  = jumlah kendaraan (unit)

$n$  = jumlah titik layanan ( $n \in V$ )

$Q_k$  = kapasitas kendaraan  $k$  (liter) ( $k \in V$ )

$T_{max}$  = waktu penyelesaian maksimal (menit)

$S_k^l$  = waktu pelayanan oleh kendaraan  $k$  pada rute  $l$  (menit) ( $l \in L, k \in K$ )

$CT_k$  = waktu penyelesaian oleh kendaraan  $k$  (menit)

Variable keputusan

$X_{ijk}^l$  = bernilai 1 jika kendaraan  $k$  melakukan perjalanan dari titik  $i$  ke titik  $j$  pada rute  $l$ , bernilai 0 jika sebaliknya ( $(i,j) \in V, k \in K, i \in L$ )

$Y_{ik}^l$  = bernilai 1 jika ada permintaan pada titik  $i$  yang tersiram oleh kendaraan  $k$  pada rute  $l$ , bernilai 0 jika sebaliknya ( $i \in V, k \in K, i \in L$ ).

#### 2.4 Fungsi Tujuan:

Fungsi tujuan dari model VRP pada penelitian ini adalah *bi-objective function* (mempunyai dua fungsi tujuan) yang bertujuan untuk meminimalkan total jarak yang dilalui (*total cost, TC*) dan total waktu penyelesaian perjalanan (*total completion time, TCT*), sehingga perlu membentuk jumlah tertimbang (*weight sum*) fungsi tujuan seperti pada persamaan (1).

$$\text{Min } f(\theta) = \omega_{TC} TC(\theta) + \omega_{TCT} TCT(\theta) \quad (1)$$

Dimana  $\theta$  merupakan solusi, sedangkan  $\omega_{TC}$  dan  $\omega_{TCT}$  adalah bobot untuk masing-masing fungsi tujuan  $TC$  persamaan (2) dan  $TCT$  persamaan (3). Bobot yang digunakan untuk meminimalkan total jarak dan meminimalkan total waktu penyelesaian perjalanan adalah sama yaitu sebesar 0,5.

$$TC = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} X_{ijk}^l \quad (2)$$

$$TCT = \sum_{k \in K} CT_k \quad (3)$$

Persamaan (1) merupakan total jarak yang ditempuh oleh kendaraan dengan bobot  $\omega_{TC}$  yang dijumlahkan dengan total waktu penyelesaian perjalanan dengan bobot  $\omega_{TCT}$ ; dimana total waktu penyelesaian perjalanan adalah penjumlahan dari total waktu perjalanan persamaan (4) dan total waktu pelayanan pada persamaan (5) yang dihitung berdasarkan waktu pelayanan dan waktu pengisian air/ *loading*.

$$S_k^l = (p + q) \cdot \sum_{i \in V} d_i Y_{ik}^l \quad (4)$$

$$CT_k = \sum_{i \in V} \sum_{j \in V} t_{ijk} X_{ijk}^l + \sum_{i \in L} S_k^l, \quad \forall k \in K \quad (5)$$

#### 2.5 Batasan model

a. Memastikan bahwa titik  $i$  minimal dilayani sebanyak 1 kali selama rute perjalanan.

$$\sum_{i \in V} \sum_{k \in K} X_{ijk}^l \geq \sum_{k \in K} Y_{ik}^l, \quad \forall j \in V \quad (6)$$

b. Memastikan bahwa kendaraan akan tiba dan meninggalkan titik layanan yang telah dilayani.

$$\sum_{i \in V} X_{ijk}^l - \sum_{j \in V} X_{ijk}^l = 0, \quad \forall k \in K, \forall l \in L \quad (7)$$

c. Setiap kendaraan berawal dan berakhir di depot

$$\sum_{j \in V} X_{1jk}^l = 1, \quad \sum_{i \in V} X_{i1k}^l = 1, \quad \forall k \in K, \forall i, j \in V \quad (8)$$

d. Memastikan bahwa muatan rute kendaraan  $k$  tidak melebihi kapasitas kendaraan  $Q_k$ .

$$\sum_{i \in V} d_i \sum_{j \in V} X_{ijk}^l \leq Q_k, \quad \forall k \in K, \forall l \in L \quad (9)$$

e. Truk menuju ke tempat pengisian air untuk melakukan *loading* muatan

$$\sum_{i \in V} X_{i,f,k}^l = 1 \quad \forall l \in L, \forall k \in K \quad (10)$$

f. Memperbolehkan truk memulai rute baru dari *intermediate facility* selama batasan waktu masih mencukupi.

$$\sum_{j \in V} X_{f,j,k}^l = 1 \quad \forall l \in L, \forall k \in K \quad (11)$$

g. Batasan (12) dan (13) memungkinkan satu titik penyiraman dapat dikunjungi lebih dari satu kali oleh kendaraan dengan rute berbeda.

$$X_{iok} = X_{oik}, \quad \forall i \in V, \forall k \in K \quad (12)$$

$$\sum_{i \in V'} X_{iek} = \sum_{j \in V'} X_{ejk}, \quad \forall e \in V', \forall k \in K; \\ i, j \neq e \dots \quad (13)$$

h. Memastikan bahwa waktu penyelesaian kendaraan  $k$  tidak melebihi batasan waktu maksimal.

$$CT_k = T_{max} \quad (14)$$

i. Memastikan muatan kendaraan  $k$  tidak negatif.

$$\sum_{i \in V} d_i Y_{ik}^l \geq 0, \quad \forall i \in V, \forall k \in K \quad (15)$$

j. Batasan (16) dan (17) merupakan variabel keputusan bilangan biner.

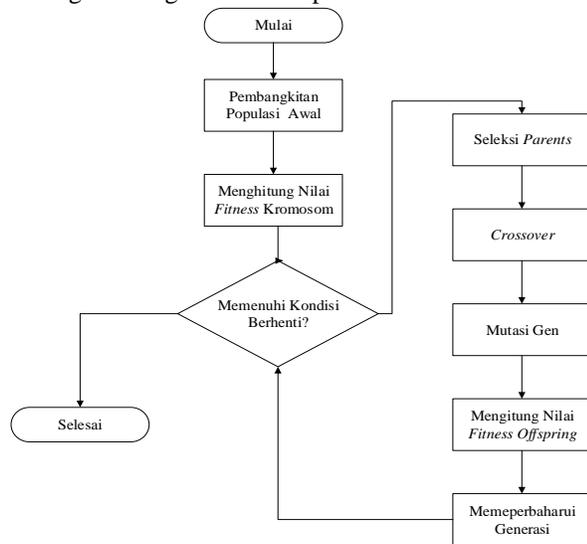
$$X_{ijk}^l \in \{0,1\}, \quad \forall ij \in V, \forall k \in K \quad (16)$$

$$Y_{ik}^l \in \{0,1\}, \forall ij \in V, \forall k \in K \quad (17)$$

## 2.6 Genetic Algorithm (GA)

*Genetic algorithm* adalah algoritma optimasi yang terinspirasi oleh teori Darwin dengan judul “*Theory of Natural Selection*”. Pada Teori tersebut dijelaskan bahwa individu-individu yang mempunyai kualitas yang bagus akan mempunyai peluang besar untuk bertahan hidup dan bereproduksi sehingga mewariskan karakteristiknya kepada keturunan-keturunannya. Sedangkan individu-individu yang mempunyai kualitas kurang bagus, maka akan tersingkir dari populasi secara perlahan (Hendrawan, 2007).

Tahapan - tahapan *Genetic Algorithm* untuk menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta secara umum ada 5 tahap yaitu, pembangkitan populasi awal, menghitung *fitness*, seleksi, kawin silang/*crossover* dan mutasi. Untuk selengkapnya dapat dilihat di Gambar 1. mengenai diagram alir tahapan GA.



Gambar 1. Diagram Alir Tahapan GA

Penjelasan terkait tahapan-tahapan GA seperti pada Gambar 1 adalah sebagai berikut:

### a. Pembangkitan Populasi Awal

Pada tahap ini yaitu membangkitkan sejumlah solusi secara acak. Representasi setiap solusi tersebut disebut dengan kromosom. Ukuran populasi yang digunakan pada penelitian ini berjumlah 40 populasi. Menurut Wibisono, (2018) ukuran populasi yang dapat meningkatkan kinerja GA yaitu berukuran antara 25 sampai 50 populasi. Pada penelitian ini pencarian solusi awal tidak murni menggunakan 100 % acak. Ada satu

solusi yang menggunakan *Nearest Neighbor*. Setelah ada satu solusi yang menggunakan *Nearest Neighbor*, maka solusi selanjutnya diperoleh secara random sampai ukuran populasi yang diinginkan terpenuhi.

### b. Menghitung nilai *fitness* kromosom

Setelah populasi awal terbentuk, maka terbentuk suatu rute besar/*giant tour* TSP. Rute besar TSP tersebut dipecah sehingga menghasilkan rute-rute kecil dengan syarat tanpa melanggar batasan kapasitas dan waktu penyiraman. Setelah diperoleh rute-rute kecil maka dihitung nilai *fitness*. Nilai *fitness* pada penelitian ini merupakan penjumlahan antara nilai total jarak ditempuh dengan nilai dari total waktu penyelesaian.

### c. Seleksi *Parents*

Proses seleksi digunakan untuk memilih kromosom-kromosom yang nantinya akan melakukan kawin silang. Pada penelitian ini menggunakan metode seleksi turnamen. Dengan teknik ini sejumlah kromosom dipilih secara acak dari populasi yang akan dibandingkan nilai *fitness*-nya. Setiap kromosom ditandingkan, kromosom dengan nilai *fitness* terbaik akan menjadi *parent* pertama. Dengan cara yang sama juga digunakan untuk mencari *parent* yang kedua.

### d. Kawin silang/ *Crossover*

Agar keanekaragaman kromosom dalam populasi bertambah, maka digunakan proses *crossover*. Cara kerja *crossover* yaitu 2 *parents* disilangkan sehingga menghasilkan kromosom baru. Proses *crossover* diharapkan agar mendapatkan keturunan yang lebih baik atau nilai *fitness* yang lebih baik. Pada penelitian ini menggunakan *multipoint crossover*. Prosedur *multipoint crossover* yaitu 2 titik pada *parent* pilih secara acak, 2 titik tersebut tidak boleh sama. Setelah diperoleh 2 titik, bagian di antara 2 titik tersebut ditukarkan dengan *parent* pasangannya.

### e. Proses Mutasi

Proses mutasi dilakukan agar menghasilkan individu-individu baru yang berbeda sama sekali dengan individu yang sudah ada. Dengan adanya proses mutasi akan menghasilkan solusi yang terhindar dari lokal optimal. Pada penelitian ini menggunakan teknik *scramble mutation* yang mengadopsi penelitian (Armandi et al., 2019). Cara kerja *scramble mutation* yaitu memilih 2 titik potong T1 dan T2 pada suatu kromosom

secara random untuk menghasilkan segmen kromosom, kemudian lakukan pengacakan posisi pada semua gen yang berada pada segmen tersebut.

f. Menghitung Nilai *Fitness Offspring*

Setelah proses mutasi maka akan menghasilkan keturunan baru, atau yang disebut dengan *offspring*. Kemudian *offspring* ini dihitung nilai *fitness*nya tanpa melanggar batasan kapasitas dan waktu. Prosedur menghitung nilai *fitness offspring* sama seperti menghitung nilai *fitness* kromosom pada populasi awal.

g. Memperbaharui Generasi

Setelah *offspring* dihitung nilai *fitness*nya, maka langkah selanjutnya memperbaharui generasi. Prosedur generasi pada penelitian ini mengadopsi Wibisono, (2018), yaitu memilih secara acak 2 kromosom dari setengah populasi terburuk. Dua kromosom tersebut dibandingkan dengan kromosom *offspring*. Apabila kromosom *offspring* mempunyai *fitness* yang lebih baik, maka kromosom *offspring* akan masuk ke dalam populasi dan menggantikan kromosom yang lama.

Setelah proses generasi maka akan kembali ke proses awal GA. Satu proses generasi merupakan 1

proses iterasi, sehingga apabila iterasi maksimal 2000, maka proses GA tersebut telah melakukan 2000 generasi. Iterasi akan berhenti apabila kondisi pemberhentian telah tercapai. Pada penelitian ini iterasi akan berhenti apabila kondisi iterasi sudah mencapai 15.000.

### 3. Hasil dan Pembahasan

Sebelum menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta, maka terlebih dahulu melakukan studi parameter. Studi parameter dilakukan untuk menentukan parameter apakah yang terbaik untuk GA, yang nantinya parameter tersebut digunakan untuk menyelesaikan kasus rute penyiraman tanaman Kota Yogyakarta.

Kasus permasalahan yang digunakan untuk melakukan studi parameter adalah kasus penyiraman tanaman di hari kedua. Hal ini karena jumlah titik penyiraman di hari kedua lebih banyak. Penyiraman tanaman di hari kedua mempunyai 14 titik penyiraman, sedangkan jumlah titik penyiraman di hari pertama berjumlah 11 titik penyiraman. Untuk jumlah *intermediate facility* adalah sama yaitu terdapat 4 lokasi. Terdapat 4 konfigurasi parameter yang akan dicoba untuk mengetahui hasil yang terbaik, seperti pada Tabel 3. dibawah ini.

**Tabel 3.** Konfigurasi Parameter

Parameter	Konfigurasi I	Konfigurasi II	Konfigurasi III	Konfigurasi IV
Ukuran populasi	49	49	30	40
Probabilitas <i>Crossover</i>	0.9	0.8	0.9	0.9
Probabilitas Mutasi	0.01	0.1	0.2	0.2
Iterasi Maksimum	5000	5000	10000	15000

Dapat dilihat pada Tabel 3, konfigurasi I pada penelitian ini menggunakan konfigurasi seperti pada penelitian Mulloorakam & Mathew Nidhiry (2019). Pada konfigurasi I, ukuran populasi yang digunakan yaitu 49 dengan jumlah iterasi maksimum sebanyak 5000. Untuk probabilitas *crossover* yaitu 0,9 dan probabilitas mutasi sebesar 0,01. Pada konfigurasi II, menggunakan ukuran populasi dan iterasi maksimum yang sama dengan konfigurasi I. Untuk probabilitas *crossover* yang digunakan pada konfigurasi II yaitu 0,8, sedangkan untuk probabilitas mutasi sebesar 0,1.

Pada konfigurasi III menggunakan probabilitas *crossover* yang sama dengan konfigurasi I yaitu 0,9. Untuk ukuran populasi dan probabilitas mutasi sesuai dengan penelitian Prins (dalam Wibisono, (2018)) yaitu ukuran populasi 30 dan probabilitas mutasi 0,2. Jumlah iterasi maksimum pada konfigurasi 3 yaitu 10000. Pada konfigurasi IV menggunakan probabilitas

*crossover* dan probabilitas mutasi yang sama dengan konfigurasi III. Ukuran populasinya menjadi lebih besar yaitu 40 dengan jumlah iterasi maksimum yang lebih besar yaitu 15000.

Masing-masing konfigurasi dijalankan sebanyak 10 kali. Setelah semua konfigurasi dijalankan maka akan diketahui konfigurasi mana yang terbaik. Pada Tabel 4. Dibawah ini merupakan hasil rekap studi parameter.

**Tabel 4.** Rekap Hasil Studi Parameter

Konfigurasi	Rata-rata Nilai <i>Fitness</i>	Rata-rata Waktu Komputasi (detik)
I	20479,5	9,11
II	20236,5	8,49
III	20221,5	16,78
IV	20206,5	24,25

Dapat dilihat pada Tabel 4 bahwa konfigurasi IV mempunyai nilai *fitness* terbaik dengan 20206,5.

Sedangkan untuk waktu komputasi terbaik diperoleh pada konfigurasi II, dengan lama komputasi 8,49 detik. Dari hasil studi parameter ini dapat disimpulkan bahwa semakin besar jumlah iterasi maka waktu komputasinya semakin lama. Jumlah iterasi yang terlalu kecil juga dapat menyebabkan solusi yang dihasilkan belum optimal. Walaupun konfigurasi IV membutuhkan waktu komputasi yang paling lama daripada konfigurasi lainnya, tetapi waktu komputasi konfigurasi IV masih dapat diterima. Berdasarkan hal tersebut, maka konfigurasi terbaik yang akan digunakan untuk menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta yaitu konfigurasi IV. Hal ini karena kualitas solusi yang dihasilkan paling baik dan stabil dengan waktu komputasi yang masih dapat diterima.

Program GA diimplementasikan kedalam perangkat lunak Matlab R2018a dan dijalankan sebanyak 5 kali pada komputer dengan spesifikasi Windows 10 (64-bit) Prosesor Intel Core i5-7200U @ 3,16Ghz dan RAM 4GB. Setelah program dijalankan sebanyak 5 kali, maka dipilih satu solusi dengan nilai *fitness* yang paling minimum. Parameter yang digunakan untuk menjalankan GA yaitu menggunakan konfigurasi parameter IV. Pada Tabel 5. merupakan hasil dari program Matlab setelah dijalankan dengan menggunakan *nearest neighbor* pada solusi awal. Sedangkan pada Tabel 6. merupakan program GA tanpa *Nearest Neighbor* pada solusi awal dengan menggunakan data penyiraman hari kedua.

**Tabel 5.** Hasil Penyelesaian Menggunakan GA

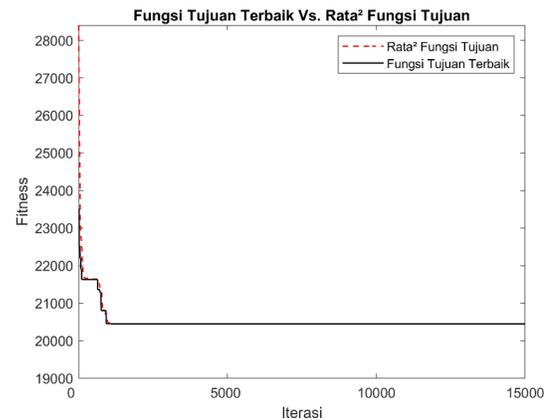
Dijalankan ke-	Penyiraman Hari Pertama (Senin, Rabu, Jumat)		Penyiraman Hari Kedua (Selasa, Kamis, Sabtu)	
	Nilai <i>Fitness</i>	Waktu Komputasi (detik)	Nilai <i>Fitness</i>	Waktu Komputasi (detik)
1	<b>20446,5</b>	27,85	20266,5	28,51
2	21052,5	28,54	<b>20191,5</b>	26,73
3	<b>20446,5</b>	27,93	20266,5	26,65
4	<b>20446,5</b>	28,25	20266,5	26,53
5	<b>20446,5</b>	27,64	<b>20191,5</b>	26,94

**Tabel 6.** Hasil Penyelesaian Menggunakan GA tanpa *Nearest Neighbor*

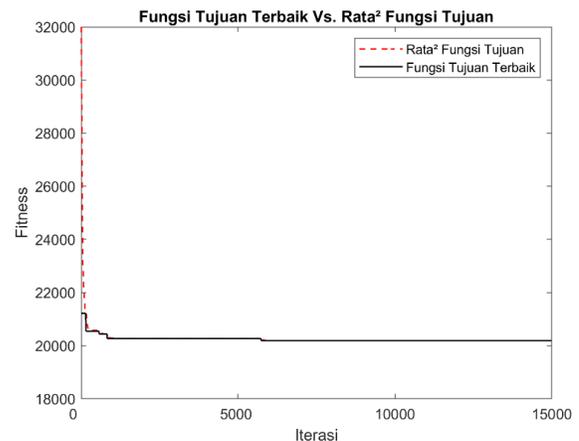
Dijalankan ke-	Penyiraman Hari Kedua (Selasa, Kamis, Sabtu)	
	Nilai <i>Fitness</i>	Waktu Komputasi (detik)
1	21810	26,67
2	21519,5	24,60
3	21519,5	24,39
4	20818,5	25,16
5	<b>20191,5</b>	24,32

Nilai *fitness* yang dicetak tebal pada Tabel 5 merupakan nilai *fitness* dengan hasil terbaik untuk setiap hari penyiraman. Berdasarkan hasil yang didapatkan pada Tabel 5, dapat dilihat pada Gambar 2 merupakan grafik nilai fungsi tujuan dan nilai rata-rata fungsi tujuan ketika GA dijalankan ke-1, untuk penyiraman hari pertama. Untuk penyiraman hari kedua dapat dilihat pada Gambar 3, ketika GA dijalankan ke-2.

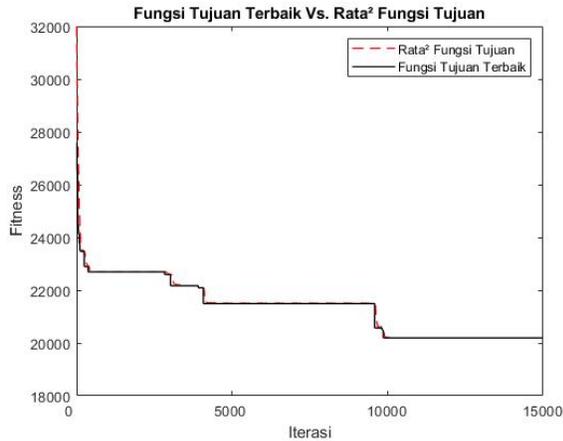
Pada Tabel 6 merupakan hasil GA tanpa adanya *Nearest Neighbor* pada solusi awal. Nilai *fitness* yang diperoleh setiap kali GA dijalankan nilainya tidak stabil. Dari segi waktu komputasinya, GA tanpa *Nearest Neighbor* diperoleh hasil yang lebih cepat, hal ini terjadi karena tidak adanya subprogram *Nearest Neighbor* pada program GA. Hasil GA dengan solusi awal menggunakan *Nearest Neighbor* lebih baik karena nilai/solusi yang dihasilkan stabil.



**Gambar 2.** Grafik Fungsi Tujuan ketika GA Dijalankan ke-1 pada Penyiraman Hari Pertama



**Gambar 3.** Grafik Fungsi Tujuan ketika GA Dijalankan ke-2 pada Penyiraman Hari Kedua



**Gambar 4.** Grafik Fungsi Tujuan ketika GA Dijalankan tanpa Menggunakan *Nearest Neighbor*

Dapat dilihat pada Gambar 2 bahwa nilai fungsi tujuan sudah mencapai konvergen ketika memasuki iterasi ke 1000. Sedangkan pada Gambar 3, nilai fungsi tujuan sudah mencapai konvergen ketika

memasuki iterasi ke 6000. Pada Gambar 4 merupakan grafik fungsi tujuan ketika GA dijalankan ke-5 tanpa menggunakan *Nearest Neighbor* pada solusi awal. Nilai fungsi tujuan pada Gambar 4 sudah mencapai konvergen ketika memasuki iterasi ke 10000.

Dengan menggunakan *Nearest Neighbor* pada solusi awal, maka akan ada solusi yang sudah mendekati optimal pada solusi awal. Dengan demikian dapat mempercepat GA untuk menemukan solusi terbaik/solusi optimal. Apabila tanpa menggunakan *Nearest Neighbor* pada solusi awal, maka GA akan cenderung mencapai nilai sub optimal setiap kali GA dijalankan.

Solusi dengan nilai *fitness* terbaik akan digunakan untuk menentukan rute penyiraman tanaman Kota Yogyakarta. Pada penyiraman hari pertama diambil ketika dijalankan ke-1 dan untuk penyiraman hari kedua diambil ketika program dijalankan ke-2. Pada Tabel 7 disajikan rute-rute penyiraman beserta jarak tempuh dan waktu penyelesaian.

Tabel 7. Rute Usulan Hasil Optimasi pada Kasus Penyiraman Tanaman Kota Yogyakarta Sektor 3

Hari Penyiraman	Kendaraan/ Volume	Rute	Total Jarak Tempuh (m)	Total Waktu Tempuh (menit)
Hari Pertama (Senin, Rabu, Jumat)	AB 8003 UA (2500 liter)	Depot - R - I - I4 - I - I4 - I - I4 - I - I4 - I - K - J - I4 - J - I4 - J - I4 - Depot	15.600	182
	AB 8004 UA (2500 liter)	Depot - Q - II - Q - X - II - X - II - X - S - Y - II - Y - R - II - Depot	9.100	114
	AB 8236 UA (5000 liter)	Depot - H - G - II - G - E - II - E - II - E - Q - II - Q - II - Depot	15.590	277
Hari Kedua (Selasa, Kamis, Sabtu)	AB 8003 UA (2500 liter)	Depot - V - U - II - F - II - P - II - P - II - Depot	7.400	115
	AB 8004 UA (2500 liter)	Depot - P - II - P - D - II - D - II - D - W - T - II - Depot	10.400	120
	AB 8236 UA (5000 liter)	Depot - T - C - B - II - B - A - II - A - O - M - N - L - I4 - L - I4 - Depot	22.130	218

Pada Tabel 7. merupakan hasil optimasi rute penyiraman Kota Yogyakarta di Sektor 3. Rute-rute yang dihasilkan dan total jarak yang ditempuh pada hasil optimasi berbeda dengan rute kondisi saat ini. Pada penyiraman hari pertama, kendaraan yang menempuh total jarak paling jauh diperoleh kendaraan dengan plat nomor AB 8003 UA yang berkapasitas 2500 liter. Waktu tempuh kendaraan tersebut sebesar 182 menit. Waktu tempuh terlama pada penyiraman hari pertama terjadi pada kendaraan dengan plat nomor AB 8236 UA. Waktu yang dibutuhkan kendaraan tersebut yaitu 277 menit.

Pada penyiraman hari kedua, kendaraan yang menempuh total jarak terjauh sekaligus memperoleh total waktu penyelesaian terlama terjadi pada kendaraan berplat nomor AB 8236 UA. Total jarak

tertempuh kendaraan tersebut sejauh 22.130 meter dengan lama total waktu penyelesaian sebesar 218 menit. Pada penyiraman hari pertama maupun penyiraman hari kedua, waktu penyelesaian terlama diperoleh kendaraan dengan kapasitas 5000 liter. Hal ini terjadi karena semakin besar kapasitas kendaraan maka akan dibutuhkan waktu yang lebih lama untuk proses pelayanan maupun proses pengisian air. Pada Tabel 8. merupakan perbandingan total jarak tempuh dan total waktu penyelesaian antara rute saat ini dan rute usulan hasil optimasi.

**Tabel 8.** Perbandingan Total Jarak dan Total Waktu antara Rute Saat Ini dengan Rute Usulan

Hari Penyiraman	Rute Saat Ini		Rute Usulan	
	Total Jarak (m)	Total Waktu (menit)	Total Jarak (m)	Total Waktu (menit)
Hari Pertama	47.990	618,8	40.290	603
Hari Kedua	43.110	468,5	39.930	453
Total	91.100	1.087,3	80.220	1.056

Pada Tabel 8. dapat dilihat bahwa total jarak tertempuh rute usulan lebih pendek dari pada total jarak yang dipakai DLH Kota Yogyakarta saat ini. Total waktu penyelesaian juga lebih cepat dari pada total waktu pada kondisi rute saat ini. Total jarak tempuh pada kondisi saat ini yaitu sebesar 91.100 meter, sedangkan total jarak tempuh rute usulan sebesar 80.220 meter. Terjadi penurunan jarak tempuh yaitu sebesar 10.880 meter atau sebesar 11,94 %. Total waktu penyelesaian pada rute saat ini yaitu 1.087,3 menit, sedangkan total waktu penyelesaian rute usulan diperoleh sebesar 1.056 menit. Terjadi penurunan total waktu sebesar 31,3 menit, atau sebesar 2,88 %.

#### 4. Kesimpulan

Penelitian ini berhasil menyelesaikan permasalahan rute penyiraman tanaman Kota Yogyakarta di Sektor 3. Jenis VRP yang sesuai dengan permasalahan penyiraman tanaman Kota Yogyakarta di Sektor 3 yaitu *heterogeneous fleet, multiple trips, intermediate facility, time windows* dan *split delivery*. Dengan menggunakan GA, mampu memberikan solusi rute dengan dengan hasil yang lebih baik daripada rute saat ini. Dengan menggunakan rute usulan, diperoleh total jarak tertempuh sejauh 80.220 meter dan total waktu penyelesaian sebesar 1.056 menit. Terjadi penurunan total jarak tempuh sebesar 11,94 % dan penurunan total waktu tertempuh sebesar 2,88 % terhadap rute saat ini.

#### Daftar Pustaka

- Amri, M., Rahman, A., & Yuniarti, R. (2014). Penyelesaian Vehicle Routing Problem dengan Menggunakan Metode Nearest Neighbour ( Studi Kasus : MTP Nganjuk Distributor PT . Coca Cola ). *Jurnal Rekayasa Dan Manajemen Sistem Industri*, 2(1), 36–45.
- Angelelli, E., & Speranza, M. G. (2002). *The periodic vehicle routing problem with intermediate facilities*. 137.
- Archetti, C., Speranza, M. G., & Hertz, A. (2006). A tabu search algorithm for the split delivery vehicle routing problem. *Transportation Science*, 40(1), 64–73. <https://doi.org/10.1287/trsc.1040.0103>
- Armandi, E., Purwani, A., & Linarti, U. (2019). Optimasi Rute Pengangkutan Sampah Kota Yogyakarta Menggunakan Hybrid Genetic Algorithm. *Jurnal Ilmiah Teknik Industri*, 18(2), 236–244. <https://doi.org/10.23917/jiti.v18i2.8744>
- Arvianto, A., Setiawan, A. H., & Saptadi, S. (2014). Model Vehicle Routing Problem dengan Karakteristik Rute Majemuk, Multiple Time Windows, Multiple Products dan Heterogeneous Fleet untuk Depot Tunggal. *Jurnal Teknik Industri*, 16(2), 85–96. <https://doi.org/10.9744/jti.16.2.83-94>
- Deng, M., Xiao, H. J., & Yang, F. M. (2006). Genetic algorithm for the open vehicle routing problem. *Beijing Huagong Daxue Xuebao (Ziran Kexueban)/Journal of Beijing University of Chemical Technology (Natural Science Edition)*, 33(4), 84–87.
- Gendreau, M., Laporte, G., Musaraganyi, C., & Taillard, É. D. (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers and Operations Research*, 26(12), 1153–1173. [https://doi.org/10.1016/S0305-0548\(98\)00100-2](https://doi.org/10.1016/S0305-0548(98)00100-2)
- Hendrawan, B. E. (2007). *Implementasi Algoritma Paralel Genetic Algorithm untuk Penyelesaian Heterogeneous Fleet Vehicle Routing Problem*.
- Indrawati, I., Eliyati, N., & Lukowi, A. (2016). Penentuan Rute Optimal pada Pengangkutan Sampah di Kota Palembang dengan Menggunakan Metode Saving Matrix. *Jurnal Penelitian Sains*, 18(3), 168493.
- Mulloorakam, A. T., & Mathew Nidhiry, N. (2019). Combined Objective Optimization for Vehicle Routing Using Genetic Algorithm. *Materials Today: Proceedings*, 11, 891–902. <https://doi.org/10.1016/j.matpr.2018.12.016>
- Pakusadewa, P. G., Dewi, C., & Wihandika, R. C. (2017). Penerapan Hibridisasi Algoritme Genetika dan Simulated Annealing untuk Optimasi Vehicle Routing Problem pada Kasus Pen .... *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 2(February), 3215–3223.
- Prins, C. (2002). Efficient Heuristics for the Heterogeneous Fleet Multitrip VRP with Application to a Large-Scale Real Case. *Journal of Mathematical Modelling and Algorithms*, 1(2), 135–150. <https://doi.org/10.1023/A:1016516326823>

- Salaki, D. T. (2009). *Penyelesaian Vehicle Routing Problem Menggunakan Beberapa Metode Heuristik*.
- Sari, V. A., & Sari, E. R. (2017). *Tanaman Menggunakan Algoritma Artificial Immune System ( Ais ) Di Kota Yogyakarta*. 281–288.
- Sureka, P., & Sumathi, S. (2011). Solution To Multi-Depot Vehicle Routing Problem Using Genetic Algorithms. *World Applied Programming, 13*, 118–131. [www.waprogramming.com](http://www.waprogramming.com)
- Toth, P., & Vigo, D. (2002). 1. An Overview of Vehicle Routing Problems. *The Vehicle Routing Problem*, 1–26. <https://doi.org/10.1137/1.9780898718515.ch1>
- Wahyuningsih, S., Satyananda, D., Octoviana, L. T., Jurusan, R. N., Fmipa, M., & Malang, U. N. (n.d.). *Vehicle Routing Problem with Time Windows Variants and its Application in Distribution Optimization*. 24–35.
- Wibisono, E. (2018). *Logika Logistik* (1st ed.). GRAHA ILMU.

Halaman kosong