

Optimasi Jarak dan Emisi Karbon pada *Vehicle Routing Problem with Time Windows* (VRPTW) Menggunakan Algoritma *Particle Swarm Optimization* (PSO): Studi Kasus PT. Pos Logistik Indonesia

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata-1
Program Studi Teknik Industri - Fakultas Teknologi Industri
Universitas Islam Indonesia**



Nama : Latifah Fitri Nuraida Wibowo
No. Mahasiswa : 21522145

**PROGRAM STUDI TEKNIK INDUSTRI PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2025**

PERNYATAAN KEASLIAN

Saya mengakui bahwa tugas akhir ini adalah hasil karya saya sendiri kecuali kutipan dan ringkasan yang seluruhnya sudah saya jelaskan sumbernya. Jika dikemudian hari ternyata terbukti pengakuan saya ini tidak benar dan melanggar peraturan yang sah maka saya bersedia ijazah yang telah saya terima ditarik kembali oleh Universitas Islam Indonesia.

Yogyakarta, 04 Juli 2025



(Latifah Fitri Nuraida Wibowo)
21522145

SURAT BUKTI PENELITIAN



**FAKULTAS
TEKNOLOGI INDUSTRI**

Gedung KH. Mas Mansur
Kampus Terpadu Universitas Islam Indonesia
Jl. Kaliurang km 14,5 Yogyakarta 55584
T. (0274) 898444 ext. 4110, 4100
F. (0274) 895007
E. fti@uii.ac.id
W. fti.uii.ac.id

SURAT KETERANGAN PENELITIAN

Nomor : 113/Ka.lab SIMANTI/20/Lab.SIMANTI/VI/2025

Assalamu'alaikum Warohmatullahi Wabarokaatuh

Dengan hormat,

Yang bertanda tangan dibawah ini, menerangkan bahwa:

Nama : Latifah Fitri Nuraida Wibowo

Nim : 21522145

Jurusan : Teknik Industri

Dosen Pembimbing : Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM

Menyatakan bahwa mahasiswa tersebut diatas telah melaksanakan penelitian tugas akhir dengan judul ” **Optimasi Jarak dan Emisi Karbon pada *Vehicle Routing Problem with Time windows* (VRPTW) Menggunakan Algoritma *Particle Swarm Optimization* (PSO): Studi Kasus PT. Pos Logistik Indonesia**”. Mulai pelaksanaan penelitian 3 Desember 2024 sampai 12 Juni 2025.

Demikian surat keterangan penelitian ini kami buat. Atas perhatiannya dan kerja samanya kami mengucapkan terima kasih.

Wassalamu'alaikum Warohmatullahi Wabarokaatuh

Yogyakarta, 12 Juni 2025

Kepala Laboratorium

Sistem Manufaktur Terintegrasi


 Putri Dwi Annisa, S.T., M.Sc.

LEMBAR PENGESAHAN PEMBIMBING

**Optimasi Jarak dan Emisi Karbon pada *Vehicle Routing Problem with Time Windows* (VRPTW) Menggunakan Algoritma *Particle Swarm Optimization* (PSO):
Studi Kasus PT. Pos Logistik Indonesia**

TUGAS AKHIR

Disusun Oleh :

Nama : Latifah Fitri Nuraida Wibowo
No. Mahasiswa : 21522145

Yogyakarta, 10 Juli 2025

Dosen Pembimbing



(Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM)

LEMBAR PENGESAHAN DOSEN PENGUJI

**Optimasi Jarak dan Emisi Karbon pada *Vehicle Routing Problem with Time Windows (VRPTW)* Menggunakan Algoritma *Particle Swarm Optimization (PSO)*:
Studi Kasus PT. Pos Logistik Indonesia**

TUGAS AKHIR

Disusun Oleh :

Nama : Latifah Fitri Nuraida Wibowo
No. Mahasiswa : 21 522 145

**Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk
memperoleh gelar Sarjana Strata-1 Teknik Industri Fakultas Teknologi Industri
Universitas Islam Indonesia**

Yogyakarta, 05 – Juli - 2025

Tim Penguji

Ir. Muhammad Ridwan Andi Purnomo,
S.T., M.Sc., Ph.D., IPM
Ketua

Bambang Suratno, S.T., M.T., Ph.D
Anggota I

Ir. Vembri Noor Helia, S.T., M.T., IPM
Anggota II



Mengetahui,
Ketua Program Studi Teknik Industri Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia



Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM
NIK: 015220101

HALAMAN PERSEMBAHAN

Alhamdulillahillāhi rabbil ālamīn, puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, taufik, dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini sebagai salah satu langkah awal dalam perjalanan panjang meraih gelar Sarjana Teknik Industri di Universitas Islam Indonesia. Karya ini penulis persembahkan dengan sepenuh hati kepada diri sendiri, sebagai wujud penghargaan atas setiap proses, perjuangan, dan usaha yang telah dilalui, meskipun penuh tantangan dan hambatan. Tak lupa, persembahan ini juga ditujukan kepada kedua orang tua tercinta Bapak Tri Jaka Gatot Wibawa dan Ibu Adelina Darmawati Lubis atas segala doa, cinta, dan pengorbanan yang tak ternilai, serta kepada saudara-saudara tercinta Andika Nur Dzaky Wibowo dan Rosyida Rizky Nurshabrina Wibowo yang senantiasa menjadi sumber semangat dan kasih sayang. Semoga karya sederhana ini dapat menjadi bagian kecil dari kebanggaan dari seorang anak dan saudara yang masih terus belajar dan akan terus berusaha menjadi lebih baik.

MOTTO

"Sesungguhnya Allah tidak akan mengubah keadaan suatu kaum, sebelum mereka mengubah keadaan diri mereka sendiri." (Ar-Rad:11)

*"Pengetahuan yang baik adalah yang memberikan manfaat, bukan hanya diingat."
(Imam Syafi'i)*

"Cobaan hidupmu bukanlah untuk menguji kekuatan dirimu. Tapi menakar seberapa besar kesungguhan dalam memohon pertolongan kepada Allah." (Ibnu Qoyyim)

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakaatuh

Alhamdulillahirabbil'alamiin, segala puji dan syukur penulis sampaikan kepada Allah Subhanahu Wa Ta'Ala atas berkat rahmat serta nikmat-Nya sehingga dan penyusunan laporan Kerja Praktik dengan judul “Optimasi Biaya dan Emisi Karbon pada Vehicle Routing Problem with *Time windows* (VRPTW) Menggunakan Algoritma *Particle Swarm Optimization* (PSO): Studi Kasus PT. Pos Logistik Indonesia”, sebagai salah satu persyaratan untuk mendapat gelar sarjana. Tidak lupa shalawat dan salam semoga tercurah keapada *Nabi Muhammad Shalallahu'alaihi Wasallam* yang telah membimbing manusia dari zaman kebodohan sampai zaman yang penuh dengan ilmu ini.

Tugas Akhir ini disusun sebagai salah satu syarat yang harus dipenuhi oleh setiap mahasiswa yang mengampu pendidikan di Prgram Studi Teknik Industri untuk menyelesaikan studi Strata-1 pada Fakultas Teknologi Industri, Universitas Islam Indonesia. Pada kesempatan kali ini, penulis ingin mengucapkan terima kasih dan penghargaan yang setinggi-tingginya kepada semua pihak yang telah berjasa memberikan bimbingan dan dukungan dalam menyelesaikan laporan ini. Maka dari itu, penulis ingin mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Ir. Hari Purnomo, M.T., IPU., ASEAN.Eng. selaku Dekan Fakultas Teknologi Industri, Universitas Islam Indonesia.
2. Bapak Dr. Drs. Imam Djati Widodo, M.Eng.Sc. selaku Ketua Jurusan Teknik Industri, Universitas Islam Indonesia.
3. Bapak Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM. selaku Ketua Program Studi Teknik Industri Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia.
4. Bapak Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D. selaku dosen pembimbing laporan Tugas Akhir yang telah memberikan bimbingan serta arahan, motivasi, dan ilmu sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini.
5. Kedua orang tua penulis yaitu Bapak Tri Jaka Gatot Wibawa dan Ibu Adelina Darmawati Lubis serta saudara Andika Nur Dzaky dan Rosyida Rizky Nur Shabrina yang selalu memberikan doa, motivasi, dan dukungan moril maupun materil kepada penulis.
6. Bapak Wahyu Kurniawan selaku *Business Development and Marketing Manager* di PT. Pos Logistik Indonesia yang telah meluangkan waktunya untuk memberi arahan, bimbingan, dan membantu dalam hal pengumpulan data yang diperlukan pada laporan ini.
7. Ibu Putri Dwi Annisa, S.T., M.Sc dan Pak Heri Susilo selaku Kepala dan Staff Laboratorium Sistem Manufaktur yang telah membimbing dan memotivasi selama menjadi asisten labrotorium dan penyusunan Tugas Akhir ini.
8. Baiq Putri Rizka Aulia, Nabila Lintang Kirana, Rifki Nurul Mukarim, Nabila Aulia Azizah, Zaki Abdurrahman, Zhafif Radithya Nugroho, dan Dhinar Elma Arifa selaku teman asisten di Laboratorium Sistem Manufaktur (SIMAN) yang

selalu membantu dan memberikan semangat selama proses penyusunan Laporan Tugas Akhir.

9. Moch. Adib Heramunida, Indrastata Maheswari, Kharnisya Alya, Khalisa Aurelia, Nur Indah, Shelma Hannah, Ayusta Qadrina, Fatimah Azzahra, Putri Yogiana, dan segenap keluarga Teknik Industri angkatan 2021 yang tidak bisa saya sebutkan satu per satu karena telah kebersamai selama masa kuliah dan membantu penyusunan tugas akhir ini.

Penulis juga mengucapkan terima kasih kepada semua pihak yang terkait dan turut membantu dalam penyelesaian laporan ini. Semoga kebaikan yang diberikan oleh semua pihak kepada penulis menjadi amal sholeh yang senantiasa mendapat balasan dan kebaikan yang berlipat dari *Allah Subhanhu Wa Ta'ala*. Amin.

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan model optimasi rute distribusi barang menggunakan algoritma *Particle Swarm Optimization* (PSO) pada kasus *Vehicle Routing Problem with Time Windows* (VRPTW) di PT Pos Logistik Indonesia. Permasalahan utama yang dihadapi perusahaan adalah ketidaksesuaian antara jarak tempuh kendaraan dan waktu pengiriman, serta tingginya emisi karbon akibat rute distribusi yang belum optimal. Penelitian ini mempertimbangkan berbagai variabel seperti kapasitas kendaraan, jadwal *time windows*, dan fluktuasi *demand* harian. Dengan menerapkan PSO, penelitian berhasil meminimalkan total jarak distribusi dari 1982,85 km menjadi 1359,89 km, menurunkan emisi karbon sebesar 34,76%, dan menghemat biaya operasional sebesar 14,73%. Hasil penelitian menunjukkan bahwa PSO efektif sebagai metode multi-objektif dalam menyelesaikan permasalahan VRPTW, serta berkontribusi pada efisiensi operasional dan keberlanjutan lingkungan.

Kata Kunci: VRPTW, PSO, Jarak, Emisi Karbon, Logistik.

DAFTAR ISI

PERNYATAAN KEASLIAN	ii
SURAT BUKTI PENELITIAN	iii
LEMBAR PENGESAHAN PEMBIMBING	iv
LEMBAR PENGESAHAN DOSEN PENGUJI	v
HALAMAN PERSEMBAHAN	vi
MOTTO	vii
KATA PENGANTAR.....	viii
ABSTRAK	x
DAFTAR ISI	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	5
1.3 Tujuan Penelitian	5
1.4 Manfaat Penelitian	6
1.5 Batasan Penelitian.....	6
1.6 Sistematika Penulisan	7
BAB II TINJAUAN PUSTAKA	9
2.1 Kajian Literatur	9
2.2 Landasan Teoritis.....	18
2.2.4 Transportasi	18
2.2.5 Logistik	19
2.2.6 Optimasi.....	19
2.2.7 Emisi Karbon (CO ₂).....	19
2.2.8 Biaya Operasional.....	20
2.2.9 <i>Vehicle Routing Problem</i> (VRP).....	20
2.2.10 <i>Particle Swarm Optimization</i> (PSO).....	21
BAB III METODE PENELITIAN	22
3.1 Objek Penelitian.....	22
3.2 Data Penelitian	22
3.3 Perancangan Penyelesaian VRPTW	23
3.4 Model Matematis	24
3.5 Alur Penelitian	26
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	28
4.1 Profil Perusahaan	28
4.1.1 Visi Misi.....	28
4.1.2 Sejarah Perusahaan.....	29
4.1.3 Layanan Perusahaan.....	30
4.1.4 Lokasi Perusahaan.....	31
4.2 Pengumpulan Data	32

4.2.1 Data Koordinat <i>Main Hub</i> dan <i>Customer</i>	32
4.2.2 Data Matriks Jarak	33
4.2.3 Data <i>Demand Delivery</i>	38
4.2.5 Tipe Kendaraan	39
4.2.6 Data Pendukung	40
4.3 Pengolahan Data	41
4.3.1 Rute Awalan	41
4.3.2 Pengembangan Algoritma PSO pada VRPTW	50
4.3.3 Penyelesaian dengan <i>Particle Swarm Optimization</i> (PSO)	55
BAB V HASIL DAN PEMBAHASAN	71
5.1 Hasil dari Pengembangan Algoritma PSO	71
5.2 Perbandingan Jarak	83
5.3 Perbandingan Emisi Karbon	84
5.4 Perbandingan Biaya Pendistribusian	86
BAB VI PENUTUP	90
6.1 Kesimpulan	90
6.2 Saran	91
LAMPIRAN	98

DAFTAR TABEL

Tabel 2. 1 Kajian Induktif.....	16
Tabel 4. 1 Koordinat Latitude dan Longitude	32
Tabel 4. 2 Matriks Jarak	36
Tabel 4. 3 <i>Delivery Demand</i>	38
Tabel 4. 4 Data <i>Time windows</i>	39
Tabel 4. 5 Spesifikasi Transportasi	40
Tabel 4. 6 Biaya Tetap.....	40
Tabel 4. 7 Biaya Variabel	41
Tabel 4. 8 Rute Awalan	42
Tabel 4. 9 Rincian Rute Awal Kendaraan 1	43
Tabel 4. 10 Rincian Rute Awal Kendaraan 2	45
Tabel 4. 11 Rincian Rute Awal Kendaraan 3	47
Tabel 4. 12 Rincian Rute Awal Kendaraan 4.....	49
Tabel 4. 13 <i>Pseudocode</i> Inisialisasi Parameter	56
Tabel 4. 14 <i>Pseudocode</i> Input Data.....	57
Tabel 4. 15 <i>Pseudocode</i> Emisi Karbon	59
Tabel 4. 16 <i>Pseudocode</i> Evaluasi Solusi	60
Tabel 4. 17 <i>Pseudocode</i> <i>Nearest Neighbor</i>	63
Tabel 4. 18 <i>Pseudocode</i> PSO	64
Tabel 4. 19 <i>Pseudocode</i> <i>Output Execution</i>	67
Tabel 4. 20 Hasil Rute berdasarkan Algoritma PSO	69
Tabel 5. 1 Rincian Rute Optimal Kendaraan 1.....	72
Tabel 5. 2 Rincian Rute Optimal Kendaraan 2.....	75
Tabel 5. 3 Rincian Rute Optimal Kendaraan 3.....	78
Tabel 5. 4 Rincian Rute Optimal Kendaraan 4.....	81
Tabel 5. 5 Perbandingan Jarak Rute Awalan dan Rute Optimal	83
Tabel 5. 6 Perbandingan Emisi Karbon Rute Awalan dan Rute Optimal	85
Tabel 5. 7 Biaya Rute Awalan.....	86
Tabel 5. 8 Biaya Rute Optimasi.....	87

DAFTAR GAMBAR

Gambar 3. 1 Alur Penelitian	27
Gambar 4. 1 Logo PT Pos Logistik Indonesia	28
Gambar 4. 2 <i>Timeline</i> Sejarah PT Pos Logistik Indonesia	29
Gambar 4. 3 Lokasi <i>Head Office</i> PT. Pos Logistik Indonesia.....	32
Gambar 4. 4 Alur Pengembangan PSO	55
Gambar 5. 1 Visualisasi Rute Kendaraan 1	71
Gambar 5. 2 Visualisasi Rute Kendaraan 2	74
Gambar 5. 3 Visualisasi Rute Kendaraan 3	77
Gambar 5. 4 Visualisasi Rute Kendaraan 4	80
Gambar 5. 5 Perbandingan Jarak	84
Gambar 5. 6 Perbandingan Emisi Karbon	85

BAB I

PENDAHULUAN

1.1 Latar Belakang

Salah satu faktor yang mendorong pertumbuhan ekonomi global di abad ini adalah globalisasi perdagangan. Perusahaan-perusahaan perlu menggandeng perusahaan logistik multinasional yang memiliki keahlian lokal saat berusaha menyesuaikan rantai pasok dengan cepat untuk meminimalkan biaya operasional dan meningkatkan kepercayaan konsumen (Sudiantini, et al., 2023). Logistik merupakan proses mendapatkan barang yang tepat, pada waktu yang tepat, dalam jumlah yang sesuai, dalam kondisi yang baik, dengan biaya yang terjangkau, sambil tetap memberikan kontribusi keuntungan bagi penyedia jasa logistik (Kasengkang, Nangoy, & Sumarauw, 2016). Dalam dunia logistik di Indonesia menghadapi berbagai tantangan, antara lain infrastruktur yang belum memadai, biaya transportasi yang tinggi, dan keterbatasan teknologi. Serta sumber daya manusia yang terbatas dan risiko keamanan seperti pencurian serta kerusakan barang selama pengiriman menjadi hambatan. Perusahaan seperti PT Pos Logistik Indonesia berperan penting dalam menghadapi tantangan ini dan memanfaatkan peluang yang ada dalam logistik global yang terus berkembang.

PT Pos Logistik Indonesia merupakan anak perusahaan dari PT Pos Indonesia yang bergerak dalam bidang logistik di Indonesia. PT Pos Logistik Indonesia telah memanfaatkan jaringan Pos Indonesia yang telah terbangun di seluruh Indonesia, yang mencapai 4.367 jaringan dalam Pos Indonesia Group dan 33.000 titik penjualan (Pos Logistics Indonesia, 2025). Kunci utama dalam kegiatan logistik salah satunya adalah transportasi. Transportasi memiliki peran yang signifikan dalam menentukan tingkat kepuasan pelanggan dan kepentingan bisnis perusahaan. Perusahaan manufaktur maupun jasa menjadi pengguna utama fasilitas infrastruktur, karena tingginya aktivitas transportasi logistik yang dilakukan, baik dalam pendistribusian barang antar pelaku bisnis maupun layanan ekspedisi (Andriyani & Resi, 2019). Dalam kegiatan transportasi, penentuan rute merupakan masalah yang berkaitan dengan pengaturan urutan pelanggan

yang akan dikunjungi, dimulai dan diakhiri di depot (Ferdiansyah, et al., 2021). Oleh karena itu perusahaan perlu mempertimbangkan penjadwalan dan penentuan rute yang tepat agar meminimalkan biaya yang dikeluarkan, serta meminimalkan emisi karbon dalam penggunaan transportasi.

PT Pos Logistik Indonesia sebagai salah satu perusahaan penyedia jasa logistik di Indonesia, memiliki *main hub* yang strategis berlokasi di Jakarta. *Main hub* berperan sebagai pusat koordinasi dan pengelolaan operasi logistik yang menyatukan berbagai hub yang tersebar di berbagai wilayah untuk memastikan proses distribusi barang dari dan ke *customer* berjalan dengan baik. *Main hub* akan mengintegrasikan suatu pengiriman barang ke *customer* terdekat dengan estimasi waktu yang telah ditentukan sehingga optimalisasi rute berjalan dengan baik, dapat meminimasi waktu pengiriman, dan penurunan biaya operasional, serta emisi karbon dari kendaraan. *Main hub* juga berfungsi sebagai titik awal pendistribusian barang ke *customer* dan tempat penyimpanan barang yang akan di *delivery* sesuai estimasi waktu yang telah ditentukan.

Pada PT. Pos Logistik Indonesia terdapat tantangan yang dihadapi dimana penentuan rute yang belum optimal untuk mendistribusikan barang dari *main hub* ke *customer* atau suatu permasalahan yang biasa dikenal sebagai *Vehicle Routing Problem* (VRP). Penentuan rute yang dilakukan bersifat konvensional yang terdapat beberapa kekurangan seperti pencarian solusi yang cukup lama, kurang fleksibel terhadap kendala seperti *time windows*, *Demand delivery*, serta kapasitas kendaraan. Dalam wawancara yang dilakukan dijelaskan bahwa evaluasi penentuan rute dalam perusahaan masih belum optimal dan subjektif dari supir yang bertugas mengantar barang. Hal tersebut dinilai masih belum efektif dalam pengambilan keputusan harian dimana *Demand* yang bersifat fluktuatif. Dampak dari permasalahan yang terjadi perusahaan akan mengalami kumulatif *cost* yang tidak stabil, jarak yang kurang optimal pada estimasi waktu yang terbatas, dan emisi karbon yang tinggi akan berdampak pada lingkungan dalam rute pendistribusian barang.

Selain berdampak pada biaya operasional, penggunaan kendaraan logistik secara luas juga berkontribusi besar terhadap peningkatan emisi karbon, terutama di wilayah perkotaan yang padat seperti Jakarta. Emisi karbon dari sektor transportasi menyumbang proporsi signifikan terhadap total emisi gas rumah kaca nasional, di mana kendaraan

berbasis bahan bakar minyak menjadi salah satu penyebab utamanya. PT Pos Logistik Indonesia sebagai salah satu pelaku utama dalam distribusi barang, memiliki armada yang beroperasi secara intensif setiap hari, sehingga akumulasi emisi karbon menjadi isu penting yang perlu diatasi. Dalam konteks pembangunan berkelanjutan dan upaya pemerintah Indonesia untuk menurunkan emisi karbon sebesar 31,89% secara mandiri dan hingga 43,20% dengan bantuan internasional pada tahun 2030 (KLHK, 2022), optimalisasi rute kendaraan menjadi salah satu strategi yang tidak hanya menguntungkan secara ekonomi, tetapi juga mendukung komitmen terhadap kelestarian lingkungan. Dengan demikian, integrasi pendekatan yang mempertimbangkan aspek lingkungan, seperti pengurangan emisi karbon, menjadi semakin relevan dan mendesak dalam sistem manajemen logistik modern. Dalam tantangan yang dihadapi perusahaan diperlukan adanya algoritma yang dapat membantu menentukan rute optimal agar *cost* yang lebih minimal, efisiensi operasional, dan pelayanan yang berkualitas secara berkelanjutan.

Model VRP sendiri pertama kali diperkenalkan pada tahun 1959 oleh Dantzig dan Ramzer dimana masalah pengiriman kendaraan ini telah menjadi masalah selama bertahun-tahun dan sangat penting dalam manajemen logistik (Heitasari & Ghifari, 2022). VRP merupakan model yang bertujuan untuk menemukan rute berbiaya paling rendah untuk mengirimkan produk ke sejumlah besar pelanggan di lokasi berbeda dengan menggunakan kendaraan, dan menggambarkan masalah transportasi dalam bentuk model grafik (Sonna Kristina & Wijaya, 2020). Terdapat berbagai jenis permasalahan VRP dalam bidang logistik, dan yang berkaitan dengan studi kasus ini pada PT. Pos Logistik Indonesia adalah *Vehicle Routing Problem with Time windows* (VRPTW). *Time windows* mengacu pada pengiriman produk yang dilakukan dengan kendaraan dengan memperhatikan batas waktu yang telah ditentukan oleh konsumen (Hidayat, Winarno, & Wahyudin, 2024).

Permasalahan *Vehicle Routing Problem with Time Windows* (VRPTW) tergolong dalam kategori *NP-hard*, yang artinya kompleksitasnya meningkat secara eksponensial seiring bertambahnya jumlah pelanggan, kendaraan, dan batasan operasional seperti kapasitas dan *time windows*. Oleh karena itu, penyelesaian VRPTW menggunakan pendekatan metaheuristik yaitu metode pencarian solusi mendekati optimal dengan waktu

komputasi yang efisien. Beberapa algoritma metaheuristik yang sering digunakan dalam menyelesaikan VRPTW yaitu adalah *Genetic Algorithm* (GA) dan *Particle Swarm Optimization* (PSO). GA bekerja berdasarkan prinsip evolusi genetik dengan mekanisme seleksi, *crossover*, dan mutasi yang memungkinkan eksplorasi ruang solusi yang luas dan PSO meniru perilaku sosial kawanan seperti burung atau ikan dalam menemukan posisi terbaik secara kolektif. Berdasarkan penelitian (Andriyansah & Linarti, 2022), PSO terbukti menghasilkan solusi rute dengan jarak tempuh yang lebih pendek dibandingkan algoritma *Genetic Algorithm* (GA) dalam studi kasus distribusi irigasi di Kota Yogyakarta. Meskipun waktu komputasi GA lebih cepat, PSO memberikan hasil yang lebih optimal dari segi efisiensi rute, dengan penghematan jarak tempuh hingga 345 meter. Selain itu, pada studi benchmark internasional menggunakan dataset Solomon, algoritma PSO versi diskret (S-PSO-VRPTW) berhasil mencatatkan enam solusi terbaik baru dan mengungguli algoritma lain seperti Hybrid Genetic Algorithm (HGA). Hasil ini menunjukkan bahwa PSO tidak hanya kompetitif, tetapi juga lebih efisien dalam menghasilkan solusi berkualitas tinggi untuk permasalahan VRPTW yang kompleks. Keunggulan PSO terletak pada kemampuannya menyeimbangkan eksplorasi dan eksploitasi ruang solusi, serta konvergensi yang stabil menuju solusi optimal.

Dalam kasus VRPTW, PSO dapat digunakan karena kemampuannya dalam eksplorasi dan eksploitasi solusi secara efektif di ruang pencarian yang kompleks. VRPTW adalah masalah optimisasi dengan banyak variabel dan kendala, seperti jadwal waktu, pengambilan, dan pengantaran. PSO akan menganalisis dengan mengoptimalkan solusi menggunakan partikel-partikel yang berbagi informasi dan bergerak menuju solusi terbaik. Implementasi algoritma ini bertujuan agar rute optimal yang dapat menyelesaikan masalah lainnya, yaitu seperti batas waktu pengambilan dan pengantaran, kapasitas kendaraan, dan emisi karbon. Algoritma ini diharapkan menghasilkan rute yang meminimalkan total jarak tempuh atau waktu perjalanan sambil memastikan bahwa semua permintaan pelanggan terpenuhi sesuai dengan batas waktu yang telah ditentukan.

1.2 Rumusan Masalah

Permasalahan dalam penelitian ini dipengaruhi oleh *Demand delivery*, waktu pendistribusian, biaya operasional serta emisi karbon kendaraan dalam mendistribusikan barang-barang. Berikut rumusan masalah dalam penelitian pada PT Pos Logistik Indonesia.

1. Bagaimana menyusun penentuan rute distribusi barang dari *main hub* ke *customer* pada PT Pos Logistik Indonesia yang selama ini masih dilakukan secara konvensional dan belum mempertimbangkan efisiensi jarak serta ketepatan waktu pengiriman, sehingga menghasilkan rute yang lebih optimal dan sesuai dengan batasan waktu layanan (*time windows*)?
2. Bagaimana pengaruh perubahan kapasitas permintaan harian dan jarak tempuh terhadap emisi karbon kendaraan distribusi di PT Pos Logistik Indonesia karena selama ini aspek lingkungan belum menjadi variabel dalam perencanaan rute dalam pendekatan pendekatan *Vehicle Routing Problem with Time Windows* (VRPTW)?
3. Bagaimana penerapan algoritma *Particle Swarm Optimization* (PSO) dapat menjadi solusi atas permasalahan biaya operasional yang belum stabil di PT Pos Logistik Indonesia akibat penentuan rute distribusi yang belum optimal?

1.3 Tujuan Penelitian

Berikut tujuan penelitian pada PT. Pos Logistik Indonesia berdasarkan rumusan masalah diatas:

1. Menyusun model optimasi rute distribusi barang dari *main hub* ke *customer* pada PT Pos Logistik Indonesia dengan mempertimbangkan efisiensi jarak tempuh dan ketepatan waktu pengiriman (*time windows*) untuk menghasilkan rute yang lebih optimal dibandingkan metode konvensional.
2. Menganalisis dampak perubahan kapasitas permintaan harian dan jarak tempuh terhadap emisi karbon yang dihasilkan oleh kendaraan distribusi, serta mengintegrasikan aspek lingkungan ke dalam perencanaan rute menggunakan pendekatan *Vehicle Routing Problem with Time Windows* (VRPTW).

3. Menerapkan algoritma *Particle Swarm Optimization* (PSO) untuk menyelesaikan masalah penentuan rute distribusi secara efisien, sehingga dapat mengurangi biaya operasional dan meningkatkan performa logistik di PT Pos Logistik Indonesia.

1.4 Manfaat Penelitian

Berikut merupakan manfaat dari penelitian yang dilakukan dan diharapkan dapat bermanfaat bagi pihak-pihak terkait:

1. Bagi perusahaan, bermanfaat bagi PT Pos Logistik Indonesia dalam meningkatkan efisiensi operasional melalui perencanaan rute distribusi yang lebih optimal dari segi jarak tempuh dan ketepatan waktu pengiriman. Dengan penerapan algoritma *Particle Swarm Optimization* (PSO), perusahaan dapat menekan biaya operasional akibat rute yang tidak efisien. Selain itu, integrasi perhitungan emisi karbon membantu perusahaan mulai memperhatikan dampak lingkungan dalam aktivitas logistiknya.
2. Bagi mahasiswa, belajar untuk menerapkan konsep optimisasi dan algoritma metaheuristik seperti PSO dalam menyelesaikan permasalahan bidang logistik serta memperdalam wawasan dan keterampilan di bidang logistik dan teknologi optimisasi yang bermanfaat dalam pengembangan karir.

1.5 Batasan Penelitian

Agar masalah tidak meluas dan mempermudah memahami permasalahan yang akan dibahas maka perlu adanya batasan masalah yaitu sebagai berikut.

1. Penelitian ini secara khusus untuk menentukan rute optimal di PT. Pos Logistik Indonesia
2. Masalah yang diselesaikan yaitu *Vehicle Routing Problem with Time windows* (VRPTW) dengan mempertimbangkan titik antar pendistribusiannya untuk meminimalkan jarak dan meminimasi emisi karbon kendaraan.
3. Metode yang digunakan untuk menyelesaikan masalah tersebut adalah *Particle Swarm Optimization* (PSO)

4. Penelitian berfokus pada *main hub* dari PT. Pos Logistik yang berada di Jakarta dan pendistribusiannya kepada *customer* di sekitar Jakarta, Bogor, Depok, Tangerang, dan Bekasi.

1.6 Sistematika Penulisan

Penelitian ini terdiri dari beberapa bab yang akan menjadi bahan referensi untuk membantu penulisan kajian ini secara sistematis. Berikut merupakan pendekatan sistematis yang digunakan untuk menulis penelitian ini:

BAB I PENDAHULUAN

Dalam bab I menjelaskan latar belakang permasalahan dalam proses distribusi barang PT Pos Logistik Indonesia, rumusan masalah, batasan penelitian, tujuan dan manfaat penelitian, serta sistematika penulisan. Fokus utama adalah permasalahan dalam penentuan rute distribusi, emisi karbon, dan efisiensi biaya operasional.

BAB II TINJAUAN PUSTAKA

Pada bab II terdapat kajian literatur dan landasan teori yang membahas teori-teori yang mendukung penelitian, seperti konsep dasar *Vehicle Routing Problem with Time Windows* (VRPTW), emisi karbon dalam distribusi logistik, algoritma *Particle Swarm Optimization* (PSO), serta penelitian terdahulu yang relevan.

BAB III METODE PENELITIAN

Pada bab III menjelaskan pendekatan penelitian yang digunakan, objek penelitian, perancangan penyelesaian menggunakan pendekatan VRPTW, model matematis, dan alur penelitian.

BAB IV PENGAMBILAN DAN PENGOLAHAN DATA

Pada bab IV akan menjelaskan proses pengumpulan data, seperti data lokasi (latitude-longitude) *main hub* dan *customer*, *Demand delivery*, batasan time windows, serta parameter kendaraan (kapasitas, kecepatan, emisi). Kemudian mengolah data menggunakan algoritma *Particle Swarm Optimization* (PSO) pada studi kasus VRPTW di PT Pos Logistik Indonesia.

BAB V PEMBAHASAN

Pada bab V berisi pembahasan hasil implementasi algoritma PSO terhadap distribusi PT Pos Logistik Indonesia. Analisis mencakup rute hasil optimasi, total jarak tempuh, waktu kedatangan kendaraan, tingkat kepatuhan terhadap *time windows*, serta total emisi karbon yang dihasilkan. Hasil ini dibandingkan dengan kondisi aktual untuk menilai efektivitas solusi yang diusulkan.

BAB VI KESIMPULAN DAN SARAN

Pada bab kesimpulan akan berisi pernyataan singkat yang ditulis dengan menggunakan urutan angka untuk menjabarkan hasil penelitian yang dilakukan. Kesimpulan akan menjawab rumusan permasalahan dan membuktikan hipotesis yang ada. Kemudian terdapat saran yang berisi beberapa rekomendasi pengembangan penelitian lanjutan dengan menggunakan cara, alat ataupun metode lain dengan tujuan untuk memperluas pengembangan ilmu.

DAFTAR PUSTAKA

Pada bagian akhir ini akan memberikan kesimpulan dari hasil penelitian yang menjawab rumusan masalah, serta saran untuk penerapan algoritma optimasi dalam sistem distribusi logistik perusahaan. Saran juga mencakup pengembangan metode lanjutan atau penerapan teknologi serupa pada skala yang lebih luas dalam rantai distribusi PT Pos Logistik Indonesia.

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Literatur

Kajian literatur merupakan aktivitas dalam mencari sebuah referensi pada karya tulis ilmiah yang dapat dipertanggung jawabkan kebenarannya. Kajian literatur dimaksudkan untuk menelusuri literatur dan penelitian terdahulu (Kurniati & Jailani, 2023). Dalam kajian literatur terdapat *state of art* yang merupakan penelitian sebelumnya untuk menganalisis pembahasan penelitian sebelumnya dalam membedakan dari penelitian yang dilakukan. *State of art* adalah kumpulan jurnal yang digunakan sebagai referensi untuk penelitian (Jeka, Risnita, Jailani, & Asrulla, 2023). Berikut merupakan *state of art* dari penelitian yang membahas topik *Vehicle Routing Problem* dengan berbagai metode metaheuristik untuk melihat hasil optimasi dan menjadi referensi peneliti dalam menyelesaikan studi kasus serupa.

Penelitian yang dilakukan oleh (Daouda & Atila, 2024) berfokus pada penyelesaian *Multi-Depot Vehicle Routing Problem* (MDVRP) dalam konteks distribusi bantuan dan pengiriman e-commerce. Algoritma *Particle Swarm Optimization* (PSO) digunakan karena kemampuannya dalam mengeksplorasi ruang solusi secara luas dan efisien. Hasil penelitian menunjukkan bahwa PSO mampu menghasilkan solusi dengan biaya minimum sebesar 613.00. Namun, setelah dikombinasikan dengan algoritma *Tabu Search* (TS) menjadi *hybrid* PSO-TS, performanya meningkat signifikan dengan biaya terendah mencapai 608.00 dan rata-rata biaya yang lebih stabil. Hal ini menunjukkan bahwa PSO, terutama dalam bentuk hibrida, sangat efektif untuk mengoptimalkan rute kendaraan secara efisien dan andal dalam skenario logistik nyata yang kompleks.

Dalam penelitian (Ibrahim et al., 2021) bertujuan untuk mengoptimalkan rute kendaraan dalam konteks *Vehicle Routing Problem* (VRP) dengan mempertimbangkan waktu pengambilan dan pengiriman. Selain itu juga meningkatkan efisiensi logistik dan mengurangi biaya operasional dengan menentukan rute yang optimal untuk setiap kendaraan. Penyelesaian masalah dilakukan dengan menggunakan pendekatan algoritma

genetik yang dimodifikasi, di mana tahap *crossover* dan mutasi dioptimalkan untuk menghasilkan solusi yang lebih baik. Dengan demikian, penelitian ini berkontribusi pada pengembangan metode yang lebih efektif dalam menangani masalah VRP yang kompleks.

Pada penelitian (Lo, 2022) yaitu bertujuan untuk mengoptimalkan masalah penjadwalan rute kendaraan dengan memperhitungkan operasi *cross-docking* dan pengurangan emisi karbon dalam manajemen logistik. Penelitian ini menggunakan metode *Particle Swarm Optimization* (PSO) karena metode ini efektif dalam mencari solusi optimal untuk masalah kombinatorial yang kompleks, seperti *Vehicle Routing Problem with Cross-Docking* (VRPCD). Hasil penelitian menunjukkan bahwa pendekatan ePSO yang diusulkan menghasilkan biaya total yang lebih rendah dibandingkan dengan metode *Genetic Algorithm* (GA) pada sebagian besar kasus yang diuji. Analisis statistik menunjukkan bahwa ePSO secara signifikan lebih baik dalam meminimalkan biaya transportasi, biaya emisi CO₂, dan biaya tetap dibandingkan dengan GA, membuktikan keunggulan metode ini dalam pengelolaan logistik berkelanjutan.

Tujuan dari penelitian yang dilakukan oleh (Islam et al., 2021) adalah untuk mengembangkan algoritma hibrida yang menggabungkan *Particle Swarm Optimization* (PSO) dan *Variable Neighborhood Search* (VNS) untuk menyelesaikan masalah *Clustered Vehicle Routing Problem* (CluVRP). Penelitian ini menggunakan PSO karena kemampuannya dalam melakukan pencarian solusi yang beragam dan efisien dalam masalah optimisasi kombinatorial, seperti CluVRP yang merupakan masalah NP-hard. Hasil penelitian menunjukkan bahwa algoritma hibrida yang diusulkan mampu menemukan 138 solusi terbaik yang belum diketahui dari 293 instance *benchmark*, dengan waktu CPU rata-rata hanya 6,99 detik. Ini membuktikan efektivitas pendekatan ini dalam memberikan solusi berkualitas tinggi untuk masalah rute kendaraan yang kompleks dalam konteks logistik.

Dalam penelitian (Srijaroon et al., 2021) bertujuan untuk mengembangkan model pemrograman linier campuran (MILP) untuk masalah *green vehicle routing problem with mixed and simultaneous pickup and delivery*, dengan mempertimbangkan jendela waktu dan jenis jalan. Penelitian ini menggunakan algoritma *Particle Swarm Optimization*

(PSO) karena kemampuannya dalam menangani masalah optimisasi kombinatorial yang kompleks dan efisiensi dalam menemukan solusi yang mendekati optimal. Hasil penelitian menunjukkan bahwa algoritma *Self-Adaptive Learning Particle Swarm Optimization* (SAL-PSO) yang diusulkan mampu menghasilkan solusi yang lebih baik dibandingkan dengan PSO standar, dengan perbaikan rata-rata relatif (RI) antara 0,15% hingga 7,31%. SAL-PSO juga menunjukkan kinerja yang lebih baik dalam kasus masalah skala besar dibandingkan dengan solusi optimal yang diperoleh dari pemrograman linier, sehingga membuktikan efektivitasnya dalam konteks logistik berkelanjutan.

Penelitian (Ramadhani & Garside, 2021) menjelaskan bahwa tujuannya yaitu untuk mengoptimalkan rencana rute kendaraan dengan mempertimbangkan minimisasi konsumsi bahan bakar dan emisi, serta memenuhi batasan berat muatan dan waktu perjalanan. Peneliti memilih algoritma *Particle Swarm Optimization* (PSO) karena algoritma ini terbukti efektif dalam menyelesaikan masalah optimisasi kombinatorial yang kompleks, seperti *Vehicle Routing Problem* (VRP), dengan waktu komputasi yang singkat. Hasil penelitian menunjukkan bahwa penerapan algoritma *Particle Swarm Optimization* (PSO) berhasil mengoptimalkan rute kendaraan dengan mengurangi konsumsi bahan bakar hingga 13%. Rute awal yang digunakan perusahaan mengkonsumsi 50.5179 liter bahan bakar, sedangkan rute yang diusulkan setelah penerapan PSO hanya mengkonsumsi 44.0354 liter.

Pada penelitian (Kuo et al., 2023) tujuannya adalah untuk mengembangkan algoritma *Particle Swarm Optimization* (PSO) yang ditingkatkan untuk menyelesaikan masalah *Vehicle Routing Problem with Time windows* (VRPTW) dalam konteks manajemen rantai pasok yang berkelanjutan, dengan mempertimbangkan gangguan yang mungkin terjadi. Penelitian ini menggunakan PSO karena algoritma ini efektif dalam mencari solusi optimal untuk masalah optimisasi multi-objektif yang kompleks, seperti yang melibatkan biaya total rantai pasok dan emisi karbon. Hasil penelitian menunjukkan bahwa algoritma PSO yang ditingkatkan (MOPSO) berhasil memberikan solusi terbaik untuk berbagai ukuran jaringan rantai pasok, dengan kinerja yang lebih baik dibandingkan dengan algoritma lain yang diuji. MOPSO menunjukkan peningkatan rata-

rata relatif (RI) antara 0,15% hingga 7,31% dibandingkan dengan PSO standar, membuktikan efektivitasnya dalam menghadapi skenario gangguan.

Tujuan dari penelitian (Wu et al., 2024) yaitu untuk mengembangkan algoritma *Neighborhood Comprehensive Learning Particle Swarm Optimization* (N-CLPSO) guna menyelesaikan masalah *Vehicle Routing Problem with Time windows* (VRPTW) yang melibatkan pengambilan dan pengiriman barang secara simultan, serta mempertimbangkan gangguan dalam rantai pasok. Penelitian ini menggunakan PSO karena kemampuannya dalam mencari solusi optimal untuk masalah optimisasi multi-objektif yang kompleks, serta kemudahan dalam penerapannya. Hasil penelitian menunjukkan bahwa N-CLPSO berhasil mencapai solusi optimal pada 29 dari 56 *instance benchmark*, dengan kinerja yang lebih baik dibandingkan dengan algoritma PSO lainnya dan algoritma mutakhir lainnya, serta memberikan peningkatan rata-rata relatif (RI) antara 0,15% hingga 7,31%. Ini mengindikasikan efektivitas N-CLPSO dalam menangani masalah VRPTW yang rumit dan dinamis.

Dalam penelitian (Wang et al., 2024) mengembangkan algoritma *Self-Competition Particle Swarm Optimization* (ScPSO) guna menyelesaikan *Vehicle Routing Problem with Time windows* (VRPTW), yang merupakan masalah NP-Hard dalam logistik dan transportasi. Penelitian ini menggunakan PSO karena kemampuannya yang terbukti dalam mencari solusi optimal untuk masalah optimisasi kompleks, serta untuk meningkatkan kecepatan konvergensi dan akurasi solusi. Hasil penelitian menunjukkan bahwa ScPSO berhasil mencapai solusi optimal pada 29 dari 56 *instance benchmark* Solomon, dan menunjukkan peningkatan rata-rata relatif (RI) antara 0,15% hingga 7,31% dibandingkan dengan algoritma PSO standar dan algoritma mutakhir lainnya. Ini membuktikan efektivitas ScPSO dalam menangani masalah VRPTW dengan mempertimbangkan gangguan dalam rantai pasok.

Tujuan dari penelitian (Fan et al., 2022) yaitu untuk mengembangkan algoritma yang efektif dalam menyelesaikan masalah *Vehicle Routing Problem with Fuzzy Demand* (VRPFD) dengan meminimalkan total biaya, termasuk faktor-faktor seperti konsumsi bahan bakar, jenis kendaraan, kecepatan, dan beban. Peneliti menggunakan *Genetic Algorithm* (GA) karena metode ini merupakan teknik pencarian dan optimasi global yang

stokastik, yang terinspirasi oleh mekanisme evolusi biologis. GA mampu melakukan pencarian yang efisien dan paralel, serta dapat secara otomatis mengakumulasi pengetahuan tentang ruang pencarian selama proses pencarian, meskipun memiliki kelemahan dalam terjebak pada optimum lokal.

Berdasarkan penelitian (Prawira & Santosa, 2021), objeknya adalah permasalahan *Vehicle Routing Problem with Drone* (VRPD), yang bertujuan meminimalkan biaya operasional dalam pengantaran barang menggunakan truk dan drone secara bersamaan. Hasil dari penelitian menunjukkan bahwa algoritma *Particle Swarm Optimization* (PSO) mampu menemukan solusi yang mendekati optimal dalam menyelesaikan *Vehicle Routing Problem with Drone* (VRPD). Meskipun demikian, algoritma PSO cenderung menghasilkan solusi yang sedikit kurang baik dibandingkan dengan algoritma *Simulated Annealing* (SA), yang mampu menemukan solusi yang lebih optimal pada berbagai skenario pengujian. Evaluasi terhadap 24 skenario berbeda menunjukkan bahwa PSO efektif dalam menghasilkan rute pengantaran yang efisien, namun tingkat optimalitasnya lebih rendah dibandingkan SA.

Berdasarkan penelitian (Meitasari & Adhi, 2025), objek penelitiannya adalah rute pengiriman untuk UMKM Cripang Bu Fitri yang melibatkan 15 mitra di Kota Semarang, dengan fokus pada optimasi jarak dan waktu pengiriman menggunakan metode *Particle Swarm Optimization* (PSO). Hasil penelitian menunjukkan bahwa penerapan PSO berhasil menghasilkan rute baru yang lebih efisien, dengan peningkatan efisiensi sebesar 13% dibandingkan rute awal, yaitu dari jarak 115.950 meter menjadi 100.550 meter. Rute baru ini tidak hanya mengurangi jarak tempuh tetapi juga berpotensi menghemat waktu pengiriman dan biaya operasional, sehingga meningkatkan daya saing UMKM Cripang Bu Fitri.

Dalam penelitian (Mageswari, 2024) dijelaskan bahwa objek penelitian tersebut adalah *Vehicle Routing Problem* (VRP), sebuah tantangan optimisasi kombinatorial dalam manajemen logistik dan transportasi. Hasil dari penggunaan *Genetic Algorithm* (GA) dalam penelitian ini menunjukkan bahwa GA mampu menghasilkan solusi yang mendekati optimal dalam waktu yang relatif singkat, dengan tingkat penghematan biaya perjalanan hingga 15-20% dibandingkan metode heuristik konvensional. Selain itu, GA

berhasil meminimalkan total jarak tempuh kendaraan sebesar 10-18% pada berbagai studi kasus yang diuji. Penggunaan *Genetic Algorithm* (GA) dalam penelitian ini meliputi ketergantungan terhadap parameter seperti ukuran populasi, tingkat *crossover*, dan mutasi yang harus di-tuning secara tepat agar mendapatkan hasil optimal. Selain itu, proses pencarian solusi dengan GA dapat memakan waktu komputasi yang cukup besar, terutama untuk masalah VRP yang kompleks dan berukuran besar, karena membutuhkan banyak iterasi hingga mencapai solusi yang memuaskan.

Dalam penelitian (Sun & Li, 2025) bertujuan untuk mengoptimalkan rute transportasi kayu dengan fokus pada pengurangan jarak kosong dan emisi karbon, serta meningkatkan efisiensi lingkungan dan biaya operasional perusahaan. Algoritma yang digunakan adalah algoritma *hybrid* yang menggabungkan keunggulan algoritma genetika (GA) dan algoritma *greedy*, serta dilengkapi mekanisme adaptif dan eliminasi untuk mempercepat konvergensi dan menghindari solusi prematur. Hasil penelitian menunjukkan bahwa penggunaan algoritma *hybrid* mampu mengurangi jarak tempuh total sebesar 17,26% dan emisi karbon selama perjalanan kosong sekitar 38,71%. Selain itu, jumlah kendaraan yang digunakan dapat dikurangi sebesar 29,55%, sehingga biaya operasional dan emisi lingkungan dapat diminimalkan secara signifikan. Dalam penerapan algoritma genetika (GA) meliputi konvergensi yang prematur, kecepatan konvergensi yang lambat, dan kemampuan pencarian lokal yang kurang optimal. Hal ini disebabkan oleh desain parameter seperti populasi awal, fungsi kecocokan, serta operasi *crossover* dan mutasi yang tidak optimal, sehingga seringkali GA memerlukan waktu yang lebih lama untuk mencapai solusi terbaik dan berisiko terjebak pada solusi lokal.

Berdasarkan penelitian (Indrianti, Leuveano, Abdul-Rashid, & Ridho, 2025), Objek penelitian tersebut adalah pengoptimalan rute distribusi LPG menggunakan model *Green Vehicle Routing Problem* (GVRP) yang mengintegrasikan faktor emisi dan biaya operasional, dengan penerapan algoritma genetika (GA) untuk meningkatkan efisiensi dan keberlanjutan logistik. Hasil penelitian menunjukkan bahwa penerapan model GVRP dengan algoritma genetika mampu menghasilkan penghematan biaya operasional sebesar sekitar 10-15% dibandingkan metode perencanaan rute konvensional. Selain itu, model ini berhasil mengurangi emisi karbon sebesar sekitar 8-12%, berkat optimasi rute dan

pengelolaan kendaraan yang lebih efisien serta pengintegrasian biaya terkait emisi seperti pajak karbon. Algoritma genetika (GA) pada penerapannya terjadi konvergensi prematur ke solusi lokal, sehingga tidak selalu menemukan solusi optimal global. Selain itu, GA memerlukan waktu komputasi yang cukup lama terutama untuk masalah skala besar, karena proses pencarian solusi melalui iterasi dan populasi yang besar.

Berdasarkan berbagai penelitian diatas *Vehicle Routing Problem* (VRP) dengan algoritma *Particle Swarm Optimization* (PSO) telah menunjukkan performa yang kompetitif bahkan unggul dibandingkan algoritma lain dalam menyelesaikan permasalahan logistik yang kompleks. PSO memiliki keunggulan dalam hal kecepatan konvergensi, kemudahan implementasi, serta kemampuannya dalam mengeksplorasi ruang solusi secara luas untuk menemukan solusi optimal. PSO dan variannya mampu mengatasi VRPTW secara efisien bahkan dalam skenario realistis dengan banyak kendala seperti *time windows*, emisi karbon, dan gangguan rantai pasok. Dalam penelitian (Ramadhani & Garside, Particle Swarm Optimization Algorithm to Solve Vehicle Routing Problem with Fuel Consumption Minimization , 2021) algoritma *Particle Swarm Optimization* (PSO) berhasil meminimalkan konsumsi bahan bakar pada penyelesaian *Vehicle Routing Problem* (VRP), yang secara langsung berdampak pada pengurangan emisi karbon. Dengan mempertimbangkan berat muatan dan waktu tempuh sebagai batasan dalam pembentukan sub-rute, PSO mampu menghasilkan rute distribusi air minum dalam kemasan yang lebih efisien dibandingkan rute awal yang digunakan perusahaan. Hasilnya, terjadi penghematan konsumsi bahan bakar sebesar 6,5 liter atau 13%, dari 50,52 liter menjadi 44,04 liter, meskipun total jarak tempuh sedikit lebih jauh. Keberhasilan ini menunjukkan bahwa PSO efektif dalam mengoptimalkan rute kendaraan dengan mempertimbangkan aspek lingkungan, khususnya emisi karbon. Beberapa penelitian menunjukkan bahwa algoritma seperti GA juga mampu menghasilkan solusi yang baik, namun terdapat kelemahan seperti konvergensi lambat dan sensitivitas terhadap parameter membuat PSO menjadi pilihan yang lebih stabil dan adaptif dalam menyelesaikan VRPTW. Oleh karena itu, PSO menjadi algoritma yang tepat untuk digunakan dalam penelitian ini guna mengoptimalkan rute distribusi pada kasus *Vehicle Routing Problem with Time Windows* (VRPTW), terutama dalam konteks efisiensi jarak

tempuh dan pengurangan emisi karbon seperti yang dibutuhkan oleh PT Pos Logistik Indonesia.

Tabel 2. 1 Kajian Induktif

No.	Penulis	Judul	Metode/Algoritma		
			PSO	GA	VNS
1.	(DAOUDA & ATILA, 2024)	<i>A Hybrid Particle Swarm Optimization with Tabu Search for Optimizing Aid Distribution Route</i>	√		
2.	(Ibrahim, Nurhakiki, Utama, & Rizaki, 2021)	<i>Optimised Genetic Algorithm Crossover and Mutation Stage for Vehicle Routing Problem Pick-Up and Delivery with Time windows</i>		√	
3.	(Lo, 2022)	<i>A Particle Swarm Optimization Approach to Solve the Vehicle Routing Problem with Cross-Docking and Carbon Emissions Reduction in Logistics Management</i>	√		
4.	(Islam, Gajpal, & ElMekkawy, 2021)	<i>Hybrid Particle Swarm Optimization algorithm for solving the clustered vehicle routing problem</i>	√		√
5.	(Srijaroon, Sethanan, Jamrus, & Chien, 2021)	<i>Green Vehicle Routing Problem With Mixed And Simultaneous Pickup And Delivery, Time windows And Road Types Using Self-Adaptive Learning Particle Swarm Optimization</i>	√		
6.	(Ramadhani & Garside, 2021)	<i>Particle Swarm Optimization Algorithm to Solve Vehicle Routing Problem with Fuel Consumption Minimization</i>	√		

No.	Penulis	Judul	Metode/Algoritma		
			PSO	GA	VNS
7.	(Kuo, Luthfiansyah, Masruroh, & Zulvia, 2023)	<i>Application of improved multi-objective Particle Swarm Optimization algorithm to solve disruption for the two-stage vehicle routing problem with time windows</i>	√		
8.	(Wu, et al., 2023)	<i>A Neighborhood Comprehensive Learning Particle Swarm Optimization For The Vehicle Routing Problem With Time windows</i>	√		
9.	(Wang, et al., 2024)	<i>Self-competition Particle Swarm Optimization Algorithm for the Vehicle Routing Problem with Time Window</i>	√		
10.	(Fan, Ren, Zhang, Zhen, & Fan, 2022)	<i>A Chaotic Genetic Algorithm with Variable Neighborhood Search for Solving Time-Dependent Green VRPTW with Fuzzy Demand</i>		√	
11.	(Prawira & Santosa, 2021)	<i>Pengembangan Algoritma Particle Swarm Optimization dan Simulated Annealing untuk Menyelesaikan Vehicle Routing Problem with Drone</i>	√		
12.	(Meitasari & Adhi, 2025)	<i>Optimisasi Rute Pengiriman Produk Usaha Mikro Kecil dan Menengah Menggunakan Metode Metaheuristik Particle Swarm Optimization</i>	√		
13.	(Mageswari, 2024)	<i>Vehicle Routing Problem (VRP) Using Genetic Algorithm</i>		√	

No.	Penulis	Judul	Metode/Algoritma		
			PSO	GA	VNS
14.	(Sun & Li, 2025)	<i>Optimizing Logistics in Forestry Supply Chains: A Vehicle Routing Problem Based on Carbon Emission Reduction</i>		√	
15.	(Indrianti, Leuveano, Abdul-Rashid, & Ridho, 2025)	<i>Green Vehicle Routing Problem Optimization for LPG Distribution: Genetic Algorithms for Complex Constraints and Emission Reduction</i>		√	

2.2 Landasan Teoritis

Landasan teori dalam penelitian adalah kumpulan konsep, definisi, dan teori yang relevan yang digunakan untuk menjelaskan fenomena yang diteliti. Berikut landasan teori yang relevan dalam penelitian *Vehicle Routing Problem with Time Windows* menggunakan algoritma *Particle Swarm Optimization*.

2.2.4 Transportasi

Transportasi merupakan sarana yang digunakan untuk kendaraan dalam mengangkut barang atau orang sesuai kebutuhan. Selain itu, pengangkutan atau transportasi dapat diartikan sebagai pemindahan orang dan/atau barang dari suatu tempat ke tempat lain dengan menggunakan kendaraan dalam suatu kawasan lalu lintas (Anadi, 2021). Transportasi juga sangat penting karena dapat menambah nilai pada produk dan layanan. Hal tersebut dikarenakan pada era bisnis modern ini tidak dapat beroperasi tanpa membuat sistem pengangkutan untuk mengangkut bahan mentah dan barang jadi dari satu lokasi ke proses bisnis berikutnya (Zai, Yulianti, Shirley, Aqmi, & Rahmah, 2022).

2.2.5 Logistik

Manajemen logistik merupakan serangkaian kegiatan yang perencanaan, pengorganisasian, dan pengawasan pada kegiatan pengadaan, pencatatan, pendistribusian, penyimpanan, dan pemeliharaan logistik untuk mendukung efektivitas dan efisiensi upaya dalam mencapai tujuan organisasi. Sistem informasi manajemen mengacu pada berbagai bagian dalam proses membangun sistem informasi untuk memenuhi kebutuhan. Jika informasinya tidak akurat, tidak terperinci, tidak terkini, atau tidak relevan, kesalahan akan terjadi dalam proses (Alacsel, 2024). Faktor lain yang memengaruhi pesatnya pertumbuhan bisnis logistik adalah gaya hidup masyarakat saat berbelanja dan bertransaksi. Dalam beberapa tahun terakhir, masyarakat sudah terbiasa berbelanja melalui berbagai aplikasi belanja dan melakukan pembayaran melalui sistem pembayaran digital (Hidayat, Muhendra, & Nugroho, 2023).

2.2.6 Optimasi

Optimasi atau optimasi adalah proses pencapaian hasil ideal atau nilai efektif yang dapat dicapai. Optimasi diartikan sebagai bentuk mengoptimalkan apa yang sudah ada atau direncanakan sesuatu yang paling optimal. Optimalisasi dalam perusahaan merupakan sistem untuk mencapai biaya terendah untuk mencapai laba terbesar. Batasan optimasi mencakup banyak kemungkinan penyelesaian. Solusi optimal digunakan untuk mengoptimalkan fungsi objektif dalam masalah pengambilan keputusan (Zubedi, Baruwadi, & Indriani, 2021) .

2.2.7 Emisi Karbon (CO₂)

Meskipun berkontribusi signifikan terhadap perekonomian dunia, sektor logistik merupakan salah satu bidang industri yang menyumbang emisi karbon yang dapat merusak lingkungan. Proses pengaturan melibatkan jaringan transportasi yang menghasilkan emisi gas rumah kaca (GRK), terutama karbondioksida (CO₂) dari konsumsi bahan bakar fosil. Emisi CO₂ menyebabkan pemanasan global yang mengakibatkan peningkatan suhu bumi dan penurunan kualitas lingkungan, yang pada

gilirannya menyebabkan krisis iklim. Menurut data yang dikumpulkan *Climate Watch* pada tahun 2023, sektor transportasi bertanggung jawab atas emisi gas rumah kaca terbesar di seluruh dunia, mencapai 36,44 gigaton karbon dioksida ekuivalen (Gt CO₂e), atau sekitar 71,5% dari total emisi. Peningkatan emisi gas rumah kaca menyebabkan banyak kerusakan yang menghambat aktivitas ekonomi. Untuk menjaga stabilitas ekonomi dan kualitas lingkungan, pengurangan emisi karbon di sektor logistik harus menjadi prioritas utama (Isnaini, Rozaki, Wulandari, & Sari, 2025).

2.2.8 Biaya Operasional

Laba adalah salah satu tujuan umum perusahaan, jadi laba adalah faktor penting untuk semua operasi bisnis. Saat menentukan laba, biaya operasional merupakan beban untuk seluruh aktivitas bisnis dan prinsipnya akan ditentukan oleh manajemen perusahaan. Biaya operasi adalah biaya yang memiliki peran aktif dan mempengaruhi keberhasilan perusahaan saat mencapai tujuan utama perusahaan (Mutiar, 2022). Biaya ini secara umum terbagi menjadi dua jenis, yaitu biaya tetap dan biaya variabel. Biaya tetap (*fixed cost*) adalah biaya yang besarnya tidak berubah meskipun terjadi fluktuasi dalam tingkat produksi atau penjualan. Jumlah biaya ini akan tetap sama baik saat aktivitas meningkat maupun menurun. Sementara itu, biaya variabel (*variable cost*) merupakan biaya yang jumlahnya berubah seiring dengan perubahan aktivitas perusahaan, seperti meningkatnya volume penjualan atau produksi (Aimoruk, Rumbino, & Taopan, 2024).

2.2.9 Vehicle Routing Problem (VRP)

Vehicle Routing Problem (VRP) merupakan masalah di mana sebuah depo memiliki banyak kendaraan yang menjalankan beberapa rute dan lebih dari kendaraan dibutuhkan untuk melayani konsumen yang tersebar di berbagai wilayah. Solusi VRP adalah menentukan rute optimal dan meminimalkan biaya transportasi, memecahkan masalah optimasi sistem distribusi dan transportasi (Patmawati & Nugroho, 2022). Terdapat berbagai jenis batasan/kendala dalam VRP, dan yang digunakan dalam penelitian ini adalah *time windows* yang merupakan kendala tambahan yang mencakup

waktu mulai dan berakhirnya penerimaan barang di tempat *customer*, tergantung pada waktu yang ditentukan (Fatma & Manurung, 2020).

2.2.10 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah metode optimasi yang diperkenalkan oleh Kennedy dan Eberhart pada tahun 1995. Metode ini terinspirasi oleh penelitian mengenai perilaku kawanan burung dan ikan. PSO digunakan dalam berbagai konteks masalah berskala besar dengan tujuan untuk menemukan solusi optimal. Pendekatan ini mencapainya dengan menghitung bobot optimal untuk setiap data record, lalu mengolah rata-rata dari seluruh sampel (Irawan, Qisthiano, Syahril, & Jakak, 2023). Algoritma PSO adalah algoritma optimasi berbasis populasi yang terinspirasi dari perilaku sosial kawanan burung atau ikan dalam mencari makanan. Algoritma ini menggunakan sekelompok "partikel" yang mewakili solusi potensial untuk suatu masalah, dan setiap partikel memiliki posisi dan kecepatan di ruang pencarian. Penerapan algoritma PSO pada masalah tertentu melibatkan penentuan posisi partikel dan penentuan metode transformasi yang diperlukan untuk mendapatkan solusi. Selain itu, mengintegrasikan heuristik khusus yang disesuaikan dengan masalah tertentu ke dalam *framework* PSO merupakan strategi umum untuk meningkatkan kinerja algoritma secara keseluruhan (Ai, 2009). Dalam VRP, PSO berperan sebagai metode optimasi untuk menemukan rute kendaraan yang efisien dengan meminimalkan biaya dan deviasi waktu. Dalam PSO, setiap partikel merepresentasikan beberapa solusi berupa urutan rute kendaraan.

BAB III

METODE PENELITIAN

3.1 Objek Penelitian

Penelitian ini dilakukan di PT. Pos Logistik Indonesia yang berada di Jakarta Pusat, Jakarta sebagai *main hub* serta *hub-hub* lainnya yang berada di daerah Jakarta, Bogor, Depok, Tangerang, dan Bekasi. Objek dalam penelitian ini adalah proses pendistribusian barang sesuai permintaan customer, dan berfokus pada penentuan rute terbaik mulai dari *main hub* kemudian *hub* dan *customer*.

3.2 Data Penelitian

Dalam penelitian ini, untuk melakukan implementasi dari algoritma *Particle Swarm Optimization* (PSO) pada *Vehicle Routing Problem with Time windows* (VRPTW) maka penulis akan menggunakan data sekunder. Data sekunder merupakan data yang sumbernya tidak langsung memberikan data kepada pengumpul data, tetapi data mentah yang telah diolah oleh orang lain. Data sekunder yang digunakan pada penelitian ini bersumber dari karyawan di PT. Pos Logistik Indonesia yaitu Pak Wahyu Kurniawan yang relevan dengan permasalahan *Vehicle Routing Problem with Time Window* (VRPTW) dimana merupakan salah satu permasalahan dalam bidang logistik.

Dataset yang diberikan dari PT. Pos Logistik terkait permasalahan VRPTW yang dimana data-data tersebut akan menjadi *input* dalam penyelesaian menggunakan algoritma PSO. Berikut data utama yang akan digunakan dalam penelitian ini.

- Data koordinat *latitude* dan *longitude* dari setiap lokasi
- Data *Demand delivery*
- Data *time windows*
- Biaya operasional distribusi

3.3 Perancangan Penyelesaian VRPTW

Pada penelitian ini menerapkan algoritma *Particle Swarm Optimization* (PSO) yang akan dikembangkan pada sistem. Sistem ini dibangun untuk menyelesaikan masalah dalam bidang distribusi produk, terutama pada sektor transportasi di PT. Pos Logistik Indonesia. Masalah ini berkaitan dengan optimasi, yaitu mencari jalur atau rute terbaik yang termasuk ke dalam bentuk *Vehicle Routing Problem with Time Window* (VRPTW).

VRPTW merupakan salah satu masalah dengan batasan pada *time windows* untuk memenuhi *Demand delivery* terhadap *customer*. Pada *time windows* meliputi, *earliest time*, *latest time*, dan *service time* pada setiap *node*. Secara umum, masalah dengan VRPTW dapat dijelaskan dalam penelitian ini sebagai berikut.

1. Terdapat satu pusat distribusi (*Main Hub*) sebagai depot yang berfungsi sebagai tempat awal kendaraan yang berlokasi di Jakarta.
2. Setiap kendaraan memiliki kapasitas (Q) dan digunakan untuk melayani permintaan pengantaran ke sejumlah node (*customer*).
3. Rute setiap kendaraan harus dimulai dan diakhiri di *Main Hub*. Kendaraan dapat mengantarkan barang langsung ke *customer*.
4. Jumlah total permintaan pada tiap kendaraan tidak boleh melebihi kapasitasnya (Q) baik untuk barang yang akan diantar.
5. Setiap *customer* hanya akan dilayani satu kali dalam satu perjalanan dan sesuai dengan batas waktu yang telah ditentukan (*time windows*).
6. Setiap pelayanan harus sesuai dengan *time window* depot $[e_0, l_0]$, dimana e_0 adalah waktu awal kendaraan berangkat dari *Main Hub* dan l_0 adalah waktu akhir kendaraan harus kembali ke *Main Hub*.
7. Setiap kendaraan diperbolehkan tiba di lokasi sebelum waktu pelayanan dimulai e_i , tetapi harus menunggu hingga batas waktu awal pelayanannya dalam melayani permintaan *node* tersebut sehingga mencegah ketidakseimbangan dalam jadwal pengiriman.
8. Setiap node ke- i maka akan mempunyai permintaan q_i (untuk *delivery*), waktu pelayanan f_i , dan *time window* $[e_i, l_i]$.

9. Tujuan utama adalah meminimalkan total jarak dan biaya pendistribusian dengan tetap mempertimbangkan *time windows*, efisiensi penggunaan kendaraan, serta emisi karbon kendaraan.

3.4 Model Matematis

3.4.1 Model Matematis VRPTW

Berikut ini merupakan model matematis dari permasalahan *Vehicle Routing Problem with Time Window* (VRPTW) mulai dari fungsi tujuan hingga fungsi kendalanya.

Fungsi Tujuan:

Min

$$\min Z = \alpha \sum_{k=1}^K \sum_{i=0}^n \sum_{j=0}^n C_{ij} X_{ijk} + \beta \sum_{k=1}^K \sum_{i=0}^n \sum_{j=0}^n e \cdot l_{ik} \cdot C_{ijk} \cdot x_{ijk}$$

$$X_{ijk} = \begin{cases} 1, & \text{jika kendaraan } k \text{ mengunjungi titik } j \text{ setelah titik } i, i \neq j \\ 0, & \text{untuk yang lain} \end{cases}$$

Dimana:

Z = Total biaya perjalanan (jarak)

α = bobot jarak

β = bobot emisi karbon

C_{ij} = Jarak dari node i ke j

$x_{ijk} = 1$ jika kendaraan k melayani rute dari i ke j , 0 jika tidak

e = emisi karbon per kg per km

C_{ij} = sisa muatan kendaraan k saat berangkat dari titik i

Dengan fungsi kendala:

1. Setiap *customer* dikunjungi sekali oleh satu kendaraan.

$$\sum_{k=1}^K \sum_{i=1}^n X_{ijk} = 1, \quad \forall j = 1, 2, 3, \dots, n$$

2. Total permintaan tidak boleh melebihi kapasitas kendaraan.

$$\sum_{k=1}^K \sum_{i=1}^n q_i X_{ijk} \leq Q, \quad \forall k = 1, 2, 3, \dots, K.$$

3. Jumlah total kendaraan tidak boleh melebihi kendaraan yang ada di *Main hub*.

$$\sum_{i=0}^n X_{i0k} = 1, \quad \forall k = 1, 2, 3, \dots, K.$$

4. Pendistribusian dimulai dari *Main hub* dan kembali ke *mian hub*.

$$\sum_{j=1}^n X_{0jk} = 1, \quad \forall k = 1, 2, 3, \dots, K.$$

5. Kendaraan yang masuk ke node harus keluar juga (flow conservation)

$$\sum_{i=0}^n X_{ijk} = \sum_{j=0}^n X_{ijk}, \quad \forall k = 1, 2, 3, \dots, K; \quad \forall i = 1, \dots, n$$

6. *Time windows* dipenuhi.

$$t_i + f + t_{ij} - M(1 - x_{ijk}) \leq t_j, \quad \forall i, j = 0, \dots, n; \quad i \neq j; \quad \forall k$$

7. Mendefinisikan keputusan variabel sebagai bilangan *binner*.

$$X_{ijk} = \{0, 1\}$$

8. Mendefinisikan bahwa waktu tidak negatif.

$$t_i \geq 0; \quad i = 1, 2, 3, \dots, n; \quad k = 1, 2, 3, \dots, K.$$

9. Batas waktu kedatangan *customer*.

$$e_i \geq t_i \leq l_i \quad i = 1, 2, 3, \dots, n.$$

10. Perhitungan Sisa Beban l_{ik}

$$l_{ik} = Q - \sum_{j \in \text{Predecessor}(i,k)} q_j$$

11. Memasukkan nilai sisa beban ke dalam fungsi emisi

$$\text{Emisi}_{ijk} = e \cdot l_{ik} \cdot c_{ij}$$

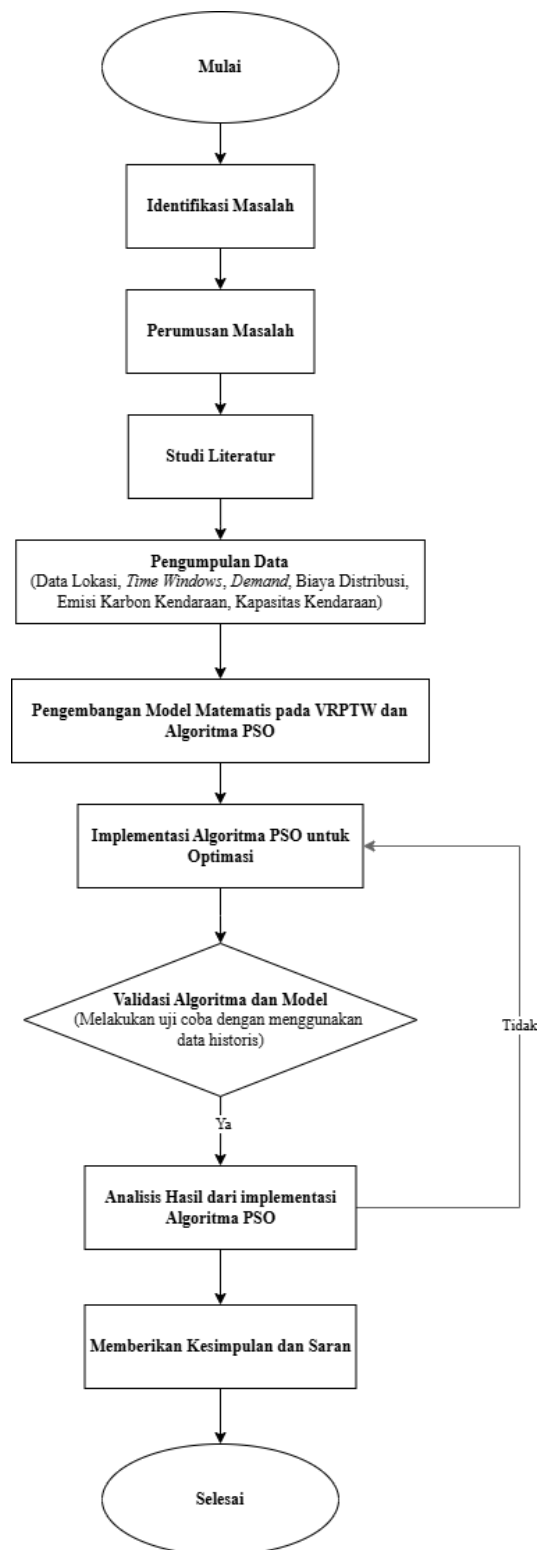
Keterangan dalam persamaan diatas.

- i, j : Jumlah konsumen yang dilayani (jumlah *node*). *Node* 0 sebagai *main hub*.

- k : Jumlah kendaraan
- X_{ijk} : Kendaraan k melayani setiap *customer* j setelah mengunjungi *customer* i
- Q : Kapasitas maksimum kendaraan
- q_i : Permintaan (*Demand*) pada setiap *customer*
- t_{ij} : Waktu tempuh kendaraan dari *customer* i ke *customer* j
- e_i : Waktu paling awal (*earliest time*) untuk melayani *customer* i
- l_i : Waktu paling akhir (*latest time*) untuk melayani *customer* j
- f_i : Waktu pelayanan kepada *customer* i
- t_i : Waktu kedatangan kendaraan pada *customer* i
- K : Banyaknya kendaraan
- n : Banyaknya *customer*
- $Predecessor(i, k)$: himpunan titik yang telah dikunjungi kendaraan k sebelum titik i
- e : faktor emisi dalam satuan gram CO₂ per kilogram per kilometer
- l_{ij} : beban kendaraan saat meninggalkan titik i
- c_{ij} : jarak antara titik i dan titik j

3.5 Alur Penelitian

Berikut merupakan alur penelitian dalam penyelesaian masalah VRPPDPTW menggunakan algoritma PSO.



Gambar 3. 1 Alur Penelitian

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini menjelaskan bagaimana algoritma *Particle Swarm Optimization* (PSO) menyelesaikan masalah *Vehicle Routing Problem with Time windows* (VRPTW). Masalah VRPTW adalah masalah ketika suatu pusat distribusi (*Main Hub*) dengan sejumlah kendaraan dengan kapasitas tertentu menentukan rute kendaraan paling optimal yang dapat mengirim ke *customer*. Penentuan rute dibuat dengan mempertimbangkan batas waktu dan kapasitas maksimum kendaraan.

4.1 Profil Perusahaan



Gambar 4. 1 Logo PT Pos Logistik Indonesia

PT Pos Logistik Indonesia merupakan perusahaan yang berfokus pada penyediaan layanan logistik dengan dukungan tim profesional yang ahli dalam merumuskan strategi dan posisi bisnis. Perusahaan ini beroperasi secara mandiri untuk mengoptimalkan peluang bisnis logistik di Indonesia. Dengan memanfaatkan jaringan luas Pos Indonesia yang tersebar di seluruh wilayah, PT Pos Logistik Indonesia mampu menjangkau berbagai daerah melalui 4.367 jaringan Pos Indonesia Group dan 33.000 titik penjualan.

4.1.1 Visi Misi

A. Visi

Menjadi penyedia solusi logistik terpadu yang terpercaya, terluas, dan terkemuka di Indonesia.

B. Misi

- ✓ Memberikan solusi logistik yang efisien dan terintegrasi bagi pelanggan serta mendukung daya saing *logistic* nasional.
- ✓ Memberikan kontribusi laba yang maksimal dan membangun sinergi usaha dengan PT Pos Indonesia.
- ✓ Membangun kemitraan usaha dengan mitra kerja strategis yang saling menguntungkan.
- ✓ Terus berupaya mengembangkan kompetensi karyawan dan organisasi agar memiliki daya saing nasional.

4.1.2 Sejarah Perusahaan



Gambar 4. 2 *Timeline* Sejarah PT Pos Logistik Indonesia

PT Pos Logistik Indonesia bermula dari sebuah proyek bisnis logistik yang dimulai pada tahun 2004 oleh PT Pos Indonesia (Persero). Pada tahun 2007, proyek ini berkembang menjadi Unit Bisnis Strategis yang fokus pada layanan logistik. Kemudian, pada tanggal 17 Februari 2012, PT Pos Logistik Indonesia resmi didirikan sebagai anak perusahaan dari PT Pos Indonesia (Persero). Sebagai anak perusahaan, PT Pos Logistik Indonesia didukung oleh tim profesional yang merumuskan strategi dan *positioning* bisnis agar dapat beroperasi secara independen. Tujuannya adalah untuk memaksimalkan peluang bisnis logistik di Indonesia dengan memanfaatkan jaringan luas Pos Indonesia yang sudah tersebar di seluruh Indonesia, yang terdiri dari 4.367 kantor cabang dan 33.000 titik penjualan.

PT Pos Logistik Indonesia berkomitmen menjadi penyedia solusi logistik terbaik di Indonesia (*Indonesia's #1 Logistics Solutions Provider*) dengan layanan yang fleksibel dan inovatif, mulai dari transportasi nasional, *contract logistics*, *freight*

forwarding, hingga layanan *e-commerce*. Selain itu, perusahaan juga menekankan nilai-nilai seperti integritas, profesionalisme, orientasi pada pelanggan, keselamatan, dan inovasi. Pada Mei 2024, PT Pos Logistik Indonesia juga mendapatkan sertifikasi halal untuk jasa penyimpanan dari Badan Penyelenggara Jaminan Produk Halal (BPJH), yang menunjukkan komitmen mereka dalam memberikan layanan yang sesuai dengan standar halal.

4.1.3 Layanan Perusahaan

Dalam perubahan yang terus menerus terjadi di industri logistik baik di dalam negeri maupun internasional, PT Pos Logistik Indonesia menyediakan berbagai layanan yang disesuaikan dengan kebutuhan pelanggan. Berikut adalah layanan yang di tawarkan:

1. *Warehouse*

PT. Pos Logistik Indonesia menyediakan solusi penyimpanan yang aman dan efisien dengan fasilitas modern serta sistem manajemen gudang (*Warehouse Management System/WMS*) yang mendukung pengelolaan stok dan distribusi secara optimal. Layanan ini dirancang untuk memenuhi kebutuhan logistik, baik untuk produk umum maupun produk khusus.

2. *Distribution*

PT. Pos Logistik Indonesia menawarkan solusi distribusi yang dapat diandalkan dan tepat waktu, memastikan produk sampai ke tujuan dengan efisien. Dengan jaringan luas dan teknologi terkini, PT. Pos Logistik Indonesia siap mengelola distribusi ke seluruh wilayah secara optimal.

3. *Heavy Lifting*

PT. Pos Logistik Indonesia menyediakan layanan pengangkatan barang berat dan bernilai tinggi dengan aman. Dilengkapi peralatan modern dan tim ahli, setiap proses pengangkatan dilakukan dengan presisi dan efisien.

4. *Domestic Cargo*

PT. Pos Logistik Indonesia memberikan solusi kargo domestik yang cepat dan terpercaya, menjangkau seluruh wilayah Indonesia. Dengan armada yang

lengkap dan dukungan teknologi terintegrasi seperti *Transport Management System* (TMS), pengiriman dijamin tepat waktu dan aman.

5. *International Cargo*

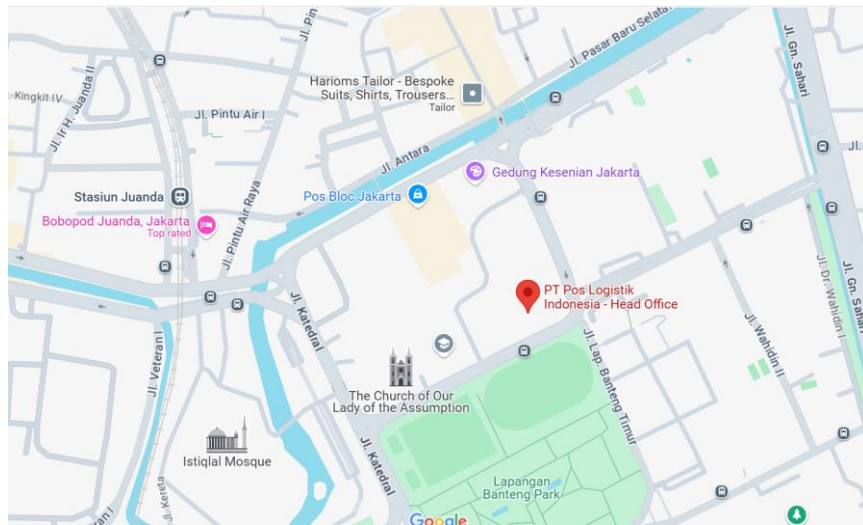
PT. Pos Logistik Indonesia menyediakan layanan kargo internasional yang handal dan efisien, melayani berbagai tujuan di seluruh dunia. Dengan solusi pengiriman terintegrasi, PT. Pos Logistik Indonesia memastikan barang tiba tepat waktu dan dalam kondisi terbaik.

6. *Cold Chain Service*

PT. Pos Logistik Indonesia menawarkan solusi rantai dingin yang menjaga kualitas produk dengan suhu yang dapat disesuaikan selama penyimpanan dan pengiriman. Menggunakan teknologi pendinginan canggih, produk *customer* tetap segar dan aman sampai ke tujuan.

4.1.4 Lokasi Perusahaan

PT Pos Logistik Indonesia beralamat di Gedung Pos Ibukota, yang terletak di Jl. Lapangan Banteng Utara No.1, Pasar Baru, Sawah Besar, Jakarta Pusat, dengan kode pos 10710. Lokasi ini sangat strategis karena berada di pusat ibu kota, memudahkan akses bagi *customer* dari berbagai wilayah. Gedung ini juga dekat dengan berbagai fasilitas transportasi umum, sehingga mendukung kelancaran operasional perusahaan. Dengan posisi yang sentral, PT Pos Logistik Indonesia dapat lebih efektif dalam mengelola jaringan logistiknya di seluruh Indonesia.



Gambar 4. 3 Lokasi *Head Office* PT. Pos Logistik Indonesia

4.2 Pengumpulan Data

4.2.1 Data Koordinat *Main Hub* dan *Customer*

Objek dalam penelitian ini adalah kegiatan distribusi barang di PT. Pos Logistik Indonesia. *Customer* yang menjadi objek penelitian adalah customer yang berada di daerah DKI Jakarta dan sekitarnya berjumlah 110 *customer*. Titik koordinat merupakan representasi suatu lokasi pada permukaan bumi dalam sistem koordinat geografis, di mana nilai *latitude* menunjukkan posisi utara–selatan terhadap garis khatulistiwa, dan nilai *longitude* menunjukkan posisi timur–barat terhadap garis meridian utama. Berikut merupakan data koordinat yang menjadi objek dari penelitian ini.

Tabel 4. 1 Koordinat *Latitude* dan *Longitude*

<i>Customer</i>	<i>Latitude</i>	<i>Longitude</i>
<i>Main hub</i>	-6,17134	106,8371
<i>Customer 1</i>	-6,210823	106,94136
<i>Customer 2</i>	-6,280092	106,835953
<i>Customer 3</i>	-6,155345	106,958512

<i>Customer</i>	<i>Latitude</i>	<i>Longitude</i>
<i>Customer 4</i>	-6,03423	106,669276
<i>Customer 5</i>	-6,122005	106,882981
<i>Customer 6</i>	-6,154297	106,983785
<i>Customer 7</i>	-6,136476	106,894816
<i>Customer 8</i>	-6,110634	106,861882
<i>Customer 9</i>	-6,291452	106,691866
<i>Customer 10</i>	-6,281572	106,887305
<i>Customer 11</i>	-6,225719	106,881188
<i>Customer 12</i>	-6,35444	106,947691
<i>Customer 13</i>	-6,288825	106,919688
<i>Customer 14</i>	-6,139359	106,597719
<i>Customer 15</i>	-6,306816	106,960849
<i>Customer 16</i>	-6,298144	106,674909
<i>Customer 17</i>	-6,257937	107,031237
<i>Customer 18</i>	-6,309047	106,859386
<i>Customer 19</i>	-6,15739	106,583334
<i>Customer 20</i>	-6,045615	106,603521
<i>Customer n</i>	...	...
<i>Customer 110</i>	-6,09556	106,750927

4.2.2 Data Matriks Jarak

Data matriks jarak merupakan representasi tabel yang menunjukkan jarak atau tingkat kemiripan antara setiap pasangan objek dalam suatu kumpulan data. Matriks ini berbentuk persegi dengan ukuran $n \times n$, di mana n adalah jumlah objek yang dianalisis, dan setiap elemen pada matriks menyatakan nilai jarak antara dua objek tertentu. Data lokasi yang tersedia adalah dengan format *latitude* dan *longitude*, maka dibutuhkan perhitungan haversine untuk mengubahnya menjadi matriks jarak. *Formula Haversine* adalah persamaan penting untuk sistem navigasi, dan formula haversine akan menciptakan jarak terpendek antara dua titik, misalnya garis bujur

(*longitude*) dan garis lintang (*latitude*). Formula ini pertama kali ditemukan pada tahun 1805 oleh James Andrew dan pertama kali digunakan pada tahun 1801 pada oleh Josef de Mendoza Yríos (Mahatmia, Hasanuddina, & Umara, 2022). Nilai jarak yang dihitung dari koordinat *latitude* dan *longitude* dikalikan dengan faktor koreksi sebesar 1,3. Nilai konversi ini digunakan untuk memperhitungkan kondisi nyata di lapangan, seperti ketidakaturan jalur, belokan, dan hambatan jalan yang menyebabkan jarak tempuh aktual lebih panjang dibandingkan jarak lurus antar titik. Dengan demikian, matriks jarak yang dihasilkan lebih mendekati kondisi sebenarnya yang dihadapi kendaraan di rute distribusi. Berikut rumus matematis dari formula *haversine*.

$$x = \text{lon}_2 - \text{lon}_1 \cdot \cos \frac{(\text{lat}_1 + \text{lat}_2)}{2}$$

$$y = \text{lat}_2 - \text{lat}_1$$

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

Keterangan:

x = Longitude (Lintang); y = Latitude (Bujur)

d = Jarak

R = Radius Bumi = 6371,8 km

1 derajat = 0,0174532925 radian

❖ Contoh penerapan matematis formula *haversine*:

Lokasi *Main Hub*: $\text{lon}_1 = 106,8371$; $\text{lat}_1 = -6,17134$

Lokasi *Customer 1*: $\text{lon}_2 = 106,94136$; $\text{lat}_2 = -6,210823$

$$\text{lon}_1 = (106,8371) \times (0,01745) = 1,8643 \text{ radian}$$

$$\text{lat}_1 = (-6,17134) \times (0,01745) = -0,1076 \text{ radian}$$

$$\text{lon}_2 = (106,94136) \times (0,01745) = 1,866 \text{ radian}$$

$$\text{lat}_2 = (-6,210823) \times (0,01745) = -0,1084 \text{ radian}$$

$$x = \text{lon}_2 - \text{lon}_1 \cdot \cos \frac{(\text{lat}_1 + \text{lat}_2)}{2}$$

$$x = 1,866 - 1,8643 \cdot \cos \frac{(-0,1076 + (-0,1084))}{2} = 0,00179$$

$$y = lat_2 - lat_1 = -0,1084 - (-0,1076) = 0,0008$$

$$d = \sqrt{((0,03045 \times 0,03045) + (0,0008 \times 0,0008))} \times 6371,8$$

$$d = 12,338 \approx 12,33 \text{ km}$$

$$d = 12,33 \text{ km} \times 1,3 = 16,03 \text{ km}$$

Tabel 4. 2 Matriks Jarak

<i>Distance Matrix</i>	<i>Main Hub</i>	<i>Customer 1</i>	<i>Customer 2</i>	<i>Customer 3</i>	<i>Customer 4</i>	<i>Customer 5</i>	<i>Customer 6</i>	<i>Customer 7</i>	<i>Customer 8</i>	n	<i>Customer 110</i>
Main Hub	0,000	16,034	15,721	17,602	31,219	9,713	21,226	9,707	9,471	...	16,534
Customer 1	16,034	0,000	18,157	8,390	46,701	15,337	10,195	12,659	18,445	...	32,042
Customer 2	15,721	18,157	0,000	25,206	42,859	23,830	27,964	22,417	24,778	...	29,340
Customer 3	17,602	8,390	25,206	0,000	45,110	11,877	3,635	9,552	15,318	...	31,062
Customer 4	31,219	46,701	42,859	45,110	0,000	33,235	48,424	35,629	29,807	...	14,709
Customer 5	9,713	15,337	23,830	11,877	33,235	0,000	15,221	2,696	3,449	...	19,362
Customer 6	21,226	10,195	27,964	3,635	48,424	15,221	0,000	13,044	18,623	...	34,528
Customer 7	9,707	12,659	22,417	9,552	35,629	2,696	13,044	0,000	6,030	...	21,510

<i>Distance Matrix</i>	<i>Main Hub</i>	<i>Customer 1</i>	<i>Customer 2</i>	<i>Customer 3</i>	<i>Customer 4</i>	<i>Customer 5</i>	<i>Customer 6</i>	<i>Customer 7</i>	<i>Customer 8</i>	n	<i>Customer 110</i>
<i>Customer 8</i>	9,471	18,445	24,778	15,318	29,807	3,449	18,623	6,030	0,000	...	16,096
<i>Customer n</i>	...	...	...	...	...	...	...	...	...	...	...
<i>Customer 110</i>	16,534	32,042	29,340	31,062	14,709	19,362	34,528	21,510	16,534	...	0

4.2.3 Data Demand Delivery

Data *Demand delivery customer* merupakan data yang mencakup kebutuhan dan permintaan pelanggan terkait produk atau layanan. Pengumpulan data *Demand delivery customer* sangat penting untuk memastikan bahwa perusahaan dapat memenuhi ekspektasi pelanggan dengan tepat waktu dan akurat, sekaligus mengoptimalkan penggunaan sumber daya seperti armada pengiriman dan stok barang. Berikut *Demand delivery* dari *customer* PT. Pos Logistik Indonesia yang akan digunakan dalam penelitian ini.

Tabel 4. 3 *Delivery Demand*

<i>Customer</i>	<i>Delivery Demand</i>
<i>Main hub</i>	0
<i>Customer 1</i>	12
<i>Customer 2</i>	5
<i>Customer 3</i>	4
<i>Customer 4</i>	11
<i>Customer 5</i>	10
<i>Customer 6</i>	8
<i>Customer 7</i>	9
<i>Customer 8</i>	3
<i>Customer 9</i>	3
<i>Customer 10</i>	4
<i>Customer 11</i>	7
<i>Customer 12</i>	11
<i>Customer 13</i>	15
<i>Customer 14</i>	15
<i>Customer 15</i>	12
<i>Customer 16</i>	9
<i>Customer 17</i>	8

<i>Customer</i>	<i>Delivery Demand</i>
<i>Customer 18</i>	12
<i>Customer 19</i>	12
<i>Customer 20</i>	7
<i>Customer n</i>	...
<i>Customer 110</i>	19

4.2.4 Data *Time windows*

Data *time windows* pada PT Pos Logistik Indonesia merupakan rentang waktu yang telah ditetapkan untuk pelaksanaan pengiriman barang kepada *customer* untuk memastikan efisiensi dan ketepatan waktu dalam proses distribusi. Pembagian waktu ini memungkinkan perusahaan untuk mengatur jadwal pengiriman secara sistematis, mengoptimalkan rute distribusi, serta mengelola beban kerja armada secara merata sepanjang hari. Dengan penerapan *time windows* yang terstruktur, PT Pos Logistik Indonesia dapat meminimalkan risiko keterlambatan, meningkatkan kepuasan pelanggan melalui pengiriman tepat waktu. Terdapat tiga periode waktu utama yang digunakan oleh PT. Pos Logistik Indonesia yaitu sebagai berikut.

Tabel 4. 4 Data *Time windows*

<i>Time windows</i>	<i>Earliest Time</i>		<i>Latest Time</i>
1	08.00 WIB	-	13.00 WIB
2	13.01 WIB	-	17.00 WIB
3	17.01 WIB	-	21.00 WIB

4.2.5 Tipe Kendaraan

Data tipe kendaraan PT Pos Logistik Indonesia menggambarkan spesifikasi transportasi pengiriman yang digunakan dalam operasional logistik perusahaan. Kendaraan yang digunakan adalah *blindvan* yang merupakan jenis van niaga yang memiliki desain tanpa jendela di bagian samping belakang. *Blind van* biasanya memiliki kapasitas penyimpanan

yang luas dan kabin belakang yang dapat dimodifikasi sesuai kebutuhan, menjadikannya cocok untuk bisnis yang membutuhkan pengiriman barang secara aman dan efisien. Kendaraan ini memiliki kapasitas angkut sebesar 120 pcs atau setara dengan 1000 kg, yang menunjukkan kemampuan optimal dalam membawa barang dengan volume dan berat tertentu. Kecepatan rata-rata kendaraan mencapai 22 km/jam. Selain itu, rasio konsumsi bahan bakar kendaraan adalah 1liter untuk setiap 11 km, yang artinya efisiensi penggunaan bahan bakar yang berkontribusi pada pengurangan biaya operasional dan dampak lingkungan.

Tabel 4. 5 Spesifikasi Transportasi

Kategori	Keterangan
Kapasitas	120pcs/1000kg
Kecepatan	22 km/jam
Ratio Konsumsi BBM	1:11

4.2.6 Data Pendukung

Dalam penelitian ini, kegiatan operasional pendistribusian akan menggunakan kendaraan blind van yang dimana PT Pos Logistik Indonesia menanggung berbagai komponen yaitu biaya tetap dan biaya operasional harian. Biaya tetap meliputi gaji sopir dan pendamping sopir yang masing-masing sebesar Rp5.600.000 per hari. Selain itu, terdapat biaya sewa kendaraan blind van sebesar Rp4.120.000 per bulan. Di sisi lain, komponen biaya operasional utama adalah konsumsi bahan bakar, yaitu Pertalite, yang dihargai Rp10.000 per liter. Data biaya ini menjadi dasar dalam perhitungan total biaya logistik dan optimalisasi rute distribusi untuk memastikan efisiensi dalam penggunaan kendaraan dan sumber daya.

Tabel 4. 6 Biaya Tetap

Biaya Tetap	
Gaji Supir	Rp 5.600.000 Per Bulan

Biaya Tetap		
Gaji Pendamping Supir	Rp 5.600.000	Per Bulan
Biaya Sewa <i>Blindvan</i>	Rp 4.120.000,00	Per Bulan

Tabel 4. 7 Biaya Variabel

Biaya Variabel		
BBM Peralite	Rp 10.000	Per liter

4.3 Pengolahan Data

4.3.1 Rute Awalan

Pada tahap awal penelitian didapatkan data rute awalan sebagai dasar evaluasi sebelum dilakukan proses optimasi rute menggunakan algoritma *Particle Swarm Optimization* (PSO). Data ini terdapat informasi mengenai urutan pengiriman dari titik asal ke titik tujuan (*origin-destination*), jarak tempuh antar titik dalam satuan kilometer, permintaan barang (*demand*) dalam satuan kilogram, serta estimasi emisi karbon yang dihasilkan dalam satuan kgCO₂. Informasi tersebut disusun untuk empat kendaraan yang menjalankan distribusi pada rute masing-masing.

Data awal ini memiliki peran penting sebagai acuan dasar dalam mengevaluasi performa rute distribusi. Selain sebagai *baseline* untuk membandingkan hasil setelah optimasi, data ini juga digunakan untuk mengidentifikasi potensi inefisiensi dalam alokasi rute, seperti adanya perjalanan jauh dengan *demand* rendah yang kurang optimal dari sisi biaya dan emisi. Dengan demikian, data awal ini menjadi bahan pertimbangan dalam menyusun strategi optimasi rute yang lebih efisien, baik dari sisi operasional maupun lingkungan.

4.3.1.1 Rute Kendaraan

Berikut merupakan rute awal dari data PT. Pos Logistik Indonesia yang didapatkan dari sistem konvensional pada perusahaan dalam satu hari.

Tabel 4. 8 Rute Awalan

Kendaraan	Rute
Kendaraan 1	0 – 77 – 18 – 30 – 78 – 57 – 2 – 32 – 52 – 109 – 98 – 66 – 67 – 26 – 16 – 89 – 103 – 79 – 58 – 21 – 37 – 80 – 1 – 7 – 12 – 25 – 36 – 43 – 49 – 55 – 64 – 0
Kendaraan 2	0 – 108 – 13 – 10 – 88 – 83 – 70 – 51 – 96 – 38 – 47 – 56 – 105 – 62 – 75 – 40 – 63 – 104 – 22 – 74 – 23 – 29 – 3 – 8 – 15 – 27 – 39 – 45 – 50 – 59 – 65 – 0
Kendaraan 3	0 – 34 – 19 – 71 – 44 – 28 – 5 – 9 – 17 – 33 – 41 – 46 – 53 – 60 – 68 – 76 – 85 – 91 – 95 – 101 – 72 – 82 – 87 – 93 – 99 – 107 – 0
Kendaraan 4	0 – 20 – 4 – 102 – 14 – 31 – 6 – 11 – 24 – 35 – 42 – 48 – 54 – 61 – 69 – 81 – 86 – 92 – 97 – 106 – 73 – 84 – 90 – 100 – 110 – 0

4.3.1.2 Jarak, *Demand* dan Emisi Karbon pada Rute Awalan

1. Kendaraan 1

Rute pertama terdiri dari 30 titik perjalanan dengan total jarak tempuh 436,28km dan total beban yang diangkut sebesar 380 kg. Beban dimulai dari titik awal dan berkurang seiring pengantaran ke pelanggan di sepanjang jalur yang relatif padat dan merata. Distribusi *demand* cukup bervariasi, mulai dari titik dengan beban kecil seperti 3 kg hingga yang besar seperti 20 kg.

Emisi karbon terbesar terjadi pada segmen-segmen awal ketika beban kendaraan masih penuh, seperti pada rute dari titik 0 ke 77 dan 18 ke 30. Sementara itu, emisi per segmen semakin menurun seiring pengurangan beban meskipun jarak tetap tinggi pada beberapa bagian. Pola ini sesuai dengan prinsip bahwa emisi transportasi logistik tidak hanya dipengaruhi oleh jarak tetapi juga oleh berat muatan.

Secara keseluruhan, rute ini menunjukkan efisiensi emisi karbon yang cukup baik dengan rata-rata emisi sekitar 0,096 kg CO₂ per segmen. Rute ini juga memanfaatkan beban kendaraan dengan cukup optimal, menjaga emisi pada tingkat yang relatif rendah meskipun rute tergolong panjang. Berikut rincian dari rute awalan pada kendaraan 1.

Tabel 4. 9 Rincian Rute Awal Kendaraan 1

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO₂)
0 - 77	18,47	8	380	0,35
77 - 18	4,15	8	372	0,08
18 - 30	7,14	16	364	0,13
30 - 78	7,77	16	348	0,14
78 - 57	7,18	20	332	0,12
57 - 2	5,98	5	312	0,09
2 - 32	7,36	5	307	0,11
32 - 52	8,82	18	302	0,13
52 - 109	9,74	18	284	0,14
109 - 98	9,78	8	266	0,13
98 - 66	4,32	6	258	0,06
66 - 67	5,46	8	252	0,07
67 - 26	10,39	18	244	0,13
26 - 16	9,57	12	226	0,11
16 - 89	3,63	8	214	0,04
89 - 103	6,85	14	206	0,07
103 - 79	8,54	19	192	0,08
79 - 58	1,23	13	173	0,01
58 - 21	4,02	7	160	0,03

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO₂)
21 - 37	5,86	16	153	0,04
37 - 80	7,10	16	137	0,05
80 - 1	52,32	12	121	0,32
1 - 7	12,66	9	109	0,07
7 - 12	32,41	11	100	0,16
12 - 25	13,22	16	89	0,06
25 - 36	16,11	17	73	0,06
36 - 43	34,06	14	56	0,10
43 - 49	69,89	18	42	0,15
49 - 55	32,27	11	24	0,04
55 - 64	12,85	13	13	0,01
64 - 0	7,13	0	0	0,35
Total	436,28	380,00		2,71

2. Kendaraan 2

Rute kedua memiliki total beban 355 kg yang diangkut sepanjang 391,93km. Rute ini terdiri dari 30 segmen dengan distribusi *demand* yang sedikit lebih rata dibandingkan rute pertama. Sebagian besar permintaan berkisar antara 5–20 kg, dengan beberapa titik awal memiliki *demand* tinggi yang memengaruhi akumulasi beban dan emisi awal.

Emisi tertinggi ditemukan pada segmen awal dari 0 ke 108 dan dari 10 ke 88, yang merupakan hasil dari kombinasi beban penuh dan jarak yang cukup panjang. Sementara pada segmen akhir seperti 59 ke 65 atau 65 ke 0, emisi hampir nol karena beban kendaraan telah habis atau mendekati habis. Hal ini menegaskan bahwa emisi sangat dipengaruhi oleh penurunan muatan.

Dengan total emisi yang lebih rendah dibanding rute pertama meskipun jarak lebih panjang, rute ini mencerminkan manajemen beban yang efisien. Penyebaran pengantaran secara merata membantu mengurangi lonjakan emisi di tengah atau akhir rute. Berikut rincian data dari kendaraan 2 yang digunakan PT. Pos Logistik Indonesia.

Tabel 4. 10 Rincian Rute Awal Kendaraan 2

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO2)
0 - 108	25,01	8	355	0,44
108 - 13	4,31	15	347	0,07
13 - 10	4,77	4	332	0,08
10 - 88	8,96	5	328	0,15
88 - 83	9,75	17	323	0,16
83 - 70	8,60	18	306	0,13
70 - 51	7,36	19	288	0,11
51 - 96	5,96	10	269	0,08
96 - 38	9,68	13	259	0,13
38 - 47	8,39	20	239	0,10
47 - 56	9,72	8	219	0,11
56 - 105	10,31	6	211	0,11
105 - 62	9,63	11	200	0,10
62 - 75	9,33	9	189	0,09
75 - 40	9,86	15	174	0,09
40 - 63	3,73	9	159	0,03
63 - 104	9,25	16	150	0,07
104 - 22	10,11	10	134	0,07
22 - 74	8,92	14	124	0,06

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO₂)
74 - 23	4,90	9	110	0,03
23 - 29	8,99	20	101	0,05
29 - 3	47,16	4	81	0,19
3 - 8	15,32	3	77	0,06
8 - 15	31,73	13	74	0,12
15 - 27	8,57	10	61	0,03
27 - 39	31,60	5	51	0,08
39 - 45	13,96	16	46	0,03
45 - 50	10,69	18	30	0,02
50 - 59	13,59	7	12	0,01
59 - 65	24,32	11	5	0,01
65 - 0	7,46	0	0	0,00
Total	391,93	343,00		2,76

3. Kendaraan 3

Rute ketiga memiliki total beban 304 kg dan menempuh jarak 587,15km. Dengan jumlah segmen sebanyak 26, rute ini memiliki struktur pengiriman yang lebih panjang per segmen dibandingkan dua rute sebelumnya. Selain itu, beberapa titik memiliki permintaan tinggi hingga 20 kg, yang menyebabkan beban tetap tinggi hingga pertengahan rute.

Pola emisi memperlihatkan lonjakan pada segmen dengan jarak panjang dan beban tinggi, seperti pada 28 ke 5 sebesar 51,67 km dan 9 ke 17 sebesar 49,00 km, di mana emisi mencapai 0,64 dan 0,57 kg CO₂. Hal ini menunjukkan perlunya perhatian pada penggabungan titik jarak jauh saat beban kendaraan masih besar. Sebaliknya, segmen akhir memiliki emisi lebih rendah karena muatan berkurang secara signifikan.

Secara umum, meskipun total beban lebih kecil dari rute pertama dan kedua, total emisi rute ini lebih tinggi. Ini mengindikasikan bahwa jarak tempuh dan urutan pengiriman sangat menentukan efisiensi emisi. Berikut rincian data dari kendaraan 3 yang digunakan PT. Pos Logistik Indonesia.

Tabel 4. 11 Rincian Rute Awal Kendaraan 3

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kgCO2)
0 - 34	33,84	14	304	0,51
34 - 19	6,72	12	290	0,10
19 - 71	8,22	6	278	0,11
71 - 44	9,00	20	272	0,12
44 - 28	4,46	6	252	0,06
28 - 5	51,67	10	246	0,64
5 - 9	36,80	3	236	0,43
9 - 17	49,00	9	233	0,57
17 - 33	27,97	16	224	0,31
33 - 41	11,68	15	208	0,12
41 - 46	20,28	16	193	0,20
46 - 53	14,86	7	177	0,13
53 - 60	12,40	9	168	0,10
60 - 68	15,33	13	159	0,12
68 - 76	36,66	6	146	0,27
76 - 85	25,46	19	140	0,18
85 - 91	25,66	12	121	0,16
91 - 95	39,13	10	111	0,22
95 - 101	52,02	19	101	0,26
101 - 72	17,57	14	82	0,07

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kgCO₂)
72 - 82	22,03	16	68	0,07
82 - 87	4,38	18	52	0,01
87 - 93	3,92	6	34	0,01
93 - 99	27,62	10	28	0,04
99 - 107	13,52	18	18	0,01
107 - 0	16,96	0	0	0,00
Total	587,15	304,00		4,83

4. Kendaraan 4

Rute keempat mengangkut beban total 308 kg dengan total jarak 567,49 km melalui 26 titik layanan. Struktur pengiriman mencakup beberapa segmen panjang dengan *Demand* tinggi yang terletak di bagian tengah rute, seperti 35 ke 42 dan 42 ke 48 yang menyumbang emisi signifikan karena muatan masih cukup besar pada titik tersebut.

Distribusi pengiriman yang merata di awal tidak terlalu membebani emisi, namun segmen-segmen dengan jarak >30 km yang terjadi saat muatan masih tinggi menyebabkan total emisi naik. Segmentasi seperti 31 ke 6 sebesar 54,04 km atau 42 ke 48 sebesar 51,53km menghasilkan emisi hampir setara dengan 0,5 kg CO₂ per segmen. Di sisi lain, bagian akhir rute menunjukkan penurunan emisi drastis. Oleh karena itu, efisiensi rute logistik dapat ditingkatkan dengan menyusun urutan pengiriman agar beban berat ditangani lebih dekat dari depot atau setelah sebagian besar beban sudah didistribusikan. Berikut rincian data dari kendaraan 4 yang digunakan PT. Pos Logistik Indonesia.

Tabel 4. 12 Rincian Awal Kendaraan 4

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kgCO2)
0 - 20	38,18	12	308	0,58
20 - 4	9,59	11	296	0,14
4 - 102	9,47	6	285	0,14
102 - 14	8,89	15	279	0,12
14 - 31	1,51	9	264	0,02
31 - 6	54,04	8	255	0,69
6 - 11	18,00	7	247	0,21
11 - 24	14,21	14	233	0,17
24 - 35	22,14	10	219	0,24
35 - 42	41,08	19	209	0,43
42 - 48	51,53	19	190	0,49
48 - 54	4,72	16	171	0,04
54 - 61	23,94	7	164	0,20
61 - 69	13,79	16	148	0,10
69 - 81	31,76	11	132	0,21
81 - 86	38,39	7	121	0,23
86 - 92	22,11	8	113	0,12
92 - 97	11,42	14	99	0,06
97 - 106	22,27	14	85	0,09
106 - 73	19,11	13	71	0,07
73 - 84	12,49	12	58	0,04
84 - 90	16,95	6	46	0,04
90 - 94	3,12	18	40	0,01
94 - 100	22,01	17	22	0,02
100 - 110	40,21	19	5	0,01

Tujuan (origin- destination)	Jarak (Km)	Demand (Kg)	Sisa Muatan (kg)	Emisi Karbon (kgCO₂)
110 - 0	16,53	0	0	0,00
Total	567,49	308,00		4,46

4.3.2 Pengembangan Algoritma PSO pada VRPTW

4.3.2.1 Penyusunan Algoritma PSO

Pada penelitian ini akan dikembangkan algoritma penyelesaian untuk permasalahan VRPTW berdasarkan algoritma PSO.

1. Langkah pertama adalah mendefinisikan parameter-parameter dasar algoritma dan memuat data masukan. Parameter yang ditentukan antara lain:

- Jumlah customer: n
- Jumlah kendaraan: k
- Kapasitas kendaraan (dalam kg)
- Kecepatan kendaraan (km/jam)
- Jumlah maksimum titik kunjungan per kendaraan (misal: 20–30 titik)
- Waktu operasional: 08.00–21.00 WIB (dalam menit: 480–1260)
- *Time windows* dibagi menjadi beberapa rentang (misalnya: TW1, TW2, TW3)

Data masukan utama terdiri atas:

- Matriks jarak antar titik
 - Permintaan pengantaran masing-masing titik
 - Jendela waktu pelayanan (*time windows*)
2. Setiap partikel dalam PSO mewakili sebuah solusi berupa urutan kunjungan seluruh *customer*. Representasi ini berbasis permutasi dari titik 1 hingga n (tidak termasuk depot). Solusi tersebut kemudian dibagi ke dalam rute-rute kendaraan berdasarkan:
 - Batasan kapasitas kendaraan

- Batas jumlah titik per kendaraan

Setiap partikel juga memiliki atribut posisi (solusi saat ini), kecepatan (berupa perubahan urutan/permutasi), solusi terbaik individu (*pBest*), dan solusi terbaik global (*gBest*).

3. Evaluasi kualitas (*fitness*) setiap solusi dilakukan dengan mempertimbangkan beberapa komponen:

- Total jarak tempuh: dijumlahkan dari semua rute kendaraan berdasarkan matriks jarak.
- Kesesuaian *time window*: dihitung berdasarkan waktu kedatangan di setiap titik. Jika waktu kedatangan melebihi jendela waktu maksimum, dikenakan penalti keterlambatan.
- Emisi karbon: dihitung berdasarkan jarak dan bobot barang yang dibawa.
- Validasi kapasitas kendaraan: rute yang melampaui kapasitas dianggap tidak valid atau dikenakan penalti besar.

Fungsi objektif dapat dirumuskan sebagai:

$$Fitness = Total\ Jarak + \alpha \times Penalti\ keterlambatan \\ + \beta \times Estimasi\ Emisi$$

Dimana α dan β merupakan koefisien bobot yang ditentukan.

4. *Update* partikel PSO diperbarui menggunakan persamaan *velocity update*. Namun karena solusi berupa permutasi (bukan vektor kontinu), maka update posisi diterjemahkan ke dalam operasi *swap* atau *insertion* pada urutan titik kunjungan.
5. Proses iterasi dilakukan hingga jumlah iterasi maksimum tercapai atau perubahan *gBest* tidak signifikan. Pada setiap iterasi:
 - Evaluasi solusi baru partikel
 - Jika solusi lebih baik dari *pBest*, diperbarui
 - Jika lebih baik dari *gBest*, *gBest* juga diperbarui
6. Solusi terbaik *gBest* dikembalikan dalam bentuk:

- Rute yang dilalui setiap kendaraan
- Waktu kedatangan di setiap titik dan status ketepatan waktu
- Total jarak dan emisi karbon per kendaraan

Hasil akhir dari algoritma PSO ini berupa rute optimal untuk masing-masing kendaraan, total jarak tempuh, serta estimasi emisi karbon yang dihasilkan

4.3.2.2 Model Matematis PSO

Dalam kasus VRPTW, PSO dapat dengan mudah diperbarui dengan menyesuaikan kecepatan dan posisi partikel berdasarkan informasi terbaru. Oleh karena itu, dalam penelitian ini digunakan algoritma PSO karena kemampuannya dalam melihat setiap solusi optimal, dapat beradaptasi terhadap batasan waktu, efisiensi komputasi, serta fleksibilitas dalam menangani berbagai kriteria optimasi yang diperlukan dalam permasalahan logistik di PT Pos Logistik Indonesia. Berikut adalah model matematis yang menggambarkan posisi dan kecepatan partikel pada suatu ruang.

1) Dasar Algoritma PSO

$$x_i^{(t)} = (x_{i1}^{(t)}, x_{i2}^{(t)}, x_{i3}^{(t)}, \dots, x_{id}^{(t)})^T \in X$$

$$v_i^{(t)} = (v_{i1}^{(t)}, v_{i2}^{(t)}, v_{i3}^{(t)}, \dots, v_{id}^{(t)})^T \in X$$

Dengan

$$i = 1, 2, 3, \dots, n \quad d = 1, 2, 3, \dots, n$$

Keterangan:

x = posisi partikel

v = kecepatan partikel

i = index partikel

t = iterasi ke t

d = dimensi ruang

Pada pemodelan tersebut menjelaskan posisi saat ini pada sebuah partikel, dalam x pada iterasi ke $-t$. Istilah partikel digunakan untuk menjelaskan hanya posisi pada sekarang yang dinotasikan dengan x_i .

2) *Personal Best Particle Swarm Optimization (PSO)*

Personal Best Particle Swarm Optimization merupakan suatu vektor PSO yang menentukan posisi terbaik untuk partikel ke- i sampai dengan iterasi ke- t berdasarkan nilai *fitness* terbaik.

$$p_i^{(t)} = (p_{i1}^{(t)}, p_{i2}^{(t)}, p_{i3}^{(t)}, \dots, p_{id}^{(t)})^T \in X$$

Dengan

$$i = 1, 2, 3, \dots, n \quad d = 1, 2, 3, \dots, n$$

Keterangan:

p = posisi partikel terbaik

3) *Global Best Particle Swarm Optimization (PSO)*

Global best Particle Swarm Optimization (PSO) adalah partikel terbaik global yang menggabungkan kumpulan posisi terbaik dari keseluruhan *personal best* pada iterasi ke- t . Posisi terbaik global dipengaruhi oleh partikel paling cocok di seluruh kawanan, menggunakan informasi dari semua partikel di kawanan.

$$P_{best,i}^{(t+1)} = \begin{cases} P_{best,i}^{(t+1)}, & \text{jika } f(x_i^{(t+1)}) > P_{best,i}^{(t+1)} \\ x_i^{(t+1)}, & \text{jika } f(x_i^{(t+1)}) \leq P_{best,i}^{(t+1)} \end{cases}$$

Dengan f yang merupakan fungsi *fitness*. Global posisi terbaik (G_{best}) pada saat langkah t akan dihitung sebagai berikut.

$$G_{best} = \min P_{best,i} \text{ dimana } i \in 1, 2, 3, \dots, n \text{ dan } n > 1.$$

Oleh karena itu, posisi terbaik individu adalah posisi terbaik yang ditemukan oleh individu partikel sejak langkah pertama. Di sisi lain, posisi terbaik global adalah posisi terbaik yang ditemukan oleh individu partikel di seluruh kawanan.

4) *Update Velocity (Kecepatan)*

Velocity digunakan untuk menghitung kecepatan partikel yang baru kecepatan sebelumnya, jarak antara posisi saat ini dengan posisi terbaik

partikel (*local best*), dan jarak antara posisi saat ini dengan posisi terbaik kawanannya (*global best*).

$$v_{ij}^{(t+1)} = v_{ij}^{(t)} + c_1 r_1^{(t)} (P_{best,i}^{(t+1)} - X_{ij}^{(t)}) + c_2 r_2^{(t)} (G_{best,i} - X_{ij}^{(t)})$$

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + v_{ij}^{(t)}$$

Keterangan:

$v_{ij}^{(t)}$ = vektor kecepatan partikel ke- i dalam dimensi j pada waktu t

$X_{ij}^{(t)}$ = vektor posisi partikel ke- i dalam dimensi j pada waktu t

$P_{best,i}^{(t)}$ = posisi terbaik *personal* partikel ke- i dalam dimensi j yang ditemukan dari inisialisasi pada waktu t

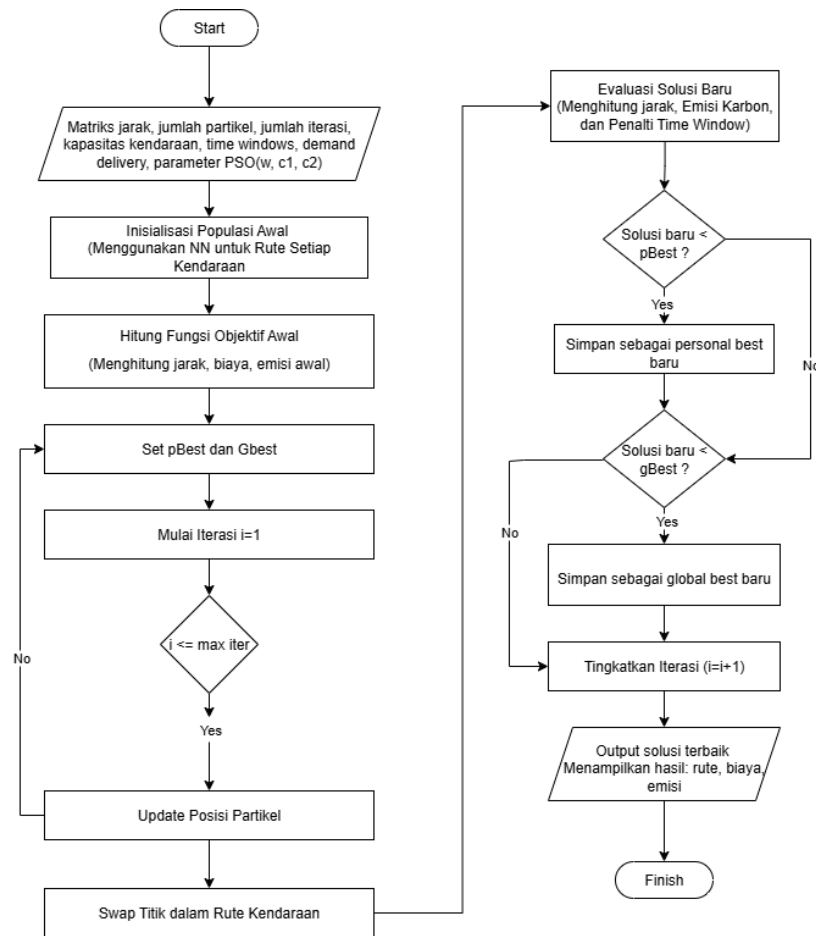
G_{best} = posisi terbaik dari *global* partikel ke- i dalam dimensi j yang ditemukan dari inisialisasi pada waktu t

c_1, c_2 = parameter learning rate untuk mengontrol pengaruh *personal best* dan *global best*

$r_1^{(t)}, r_2^{(t)}$ = angka acak dari distribusi *uniform continue* dari 0 sampai 1 pada iterasi ke t

4.3.2.3 Alur Pengembangan PSO

Berikut merupakan alur dalam pengembangan *Particle Swarm Optimization* pada studi kasus *Vehicle Routing Problem with Time windows* pada studi kasus PT. Pos Logistik Indonesia.



Gambar 4. 4 Alur Pengembangan PSO

4.3.3 Penyelesaian dengan *Particle Swarm Optimization* (PSO)

4.3.3.1 Inisialisasi Parameter

Pemrograman diawali dengan inisialisasi parameter penting dalam model VRPTW, seperti jumlah kendaraan, kapasitas kendaraan (dalam kilogram), kecepatan rata-rata kendaraan, serta jumlah *customer* yang harus dikunjungi. kapasitas maksimum satu kendaraan *blindvan* milik PT Pos Logistik dalam sekali pengantaran setiap kendaraan adalah 1000kg. Waktu operasional dibagi ke dalam tiga kategori *time window*, yakni:

- TW1: pukul 08.00 – 13.00 (480–780 menit),
- TW2: pukul 13.01 – 17.00 (781–1020 menit),

- TW3: pukul 17.01 – 21.00 (1021–1260 menit).

Berikut merupakan parameter yang digunakan dalam pengembangan algoritma PSO dalam penelitian ini.

N (Jumlah *customer*) = 110

K (Jumlah kendaraan) = 4

Q (Kapasitas maksimum kendaraan) = 1000kg

$L_{\max}$ (Jumlah maksimum *node* pada rute) = 30

$L_{\min}$ (Jumlah minimum *node* pada rute) = 20

d_0 (Node depot (*main hub*)) = node 0

v_0 (Kecepatan rata-rata (km/jam)) = 22km/jam

λ (Penalti keterlambatan) = Rp 10.000

Berikut merupakan penjelasan *pseudocode* tahapan inisialisasi parameter yang digunakan.

Tabel 4. 13 *Pseudocode* Inisialisasi Parameter

<i>Pseudocode</i>	Fungsi
<pre> # ----- # PARAMETER DASAR # ----- NUM_CUSTOMERS = 110 NUM_VEHICLES = 4 DEPOT = 0 VEHICLE_SPEED = 22 # km/h TW_RANGES = { 1: (480, 780), 2: (781, 1020), 3: (1021, 1260) } </pre>	<p>Fungsi utama dari bagian ini adalah mengatur parameter utama simulasi, seperti jumlah pelanggan, jumlah kendaraan, batasan rute, kecepatan kendaraan, dan penalti jika kendaraan mengantar melewati <i>time window</i>.</p>

4.3.3.2 Input Data

Data yang digunakan dalam pemodelan masalah *Vehicle Routing Problem with Time windows* (VRPTW) terdiri dari beberapa data input utama, yaitu matriks jarak antar titik (*distance matrix*), data permintaan (*delivery Demand*), dan *time window* setiap pelanggan (*time windows*). Data ini menjadi dasar bagi algoritma

PSO dalam melakukan evaluasi solusi, validasi constraint, dan optimasi rute pengiriman. Berikut merupakan penjelasan *pseudocode* dari tahapan *input* data.

Tabel 4. 14 *Pseudocode Input Data*

Pseudocode	Fungsi
<pre>distance_matrix = [[0.00000, 12.334002, 12.403480, 13.540331, 24.014871, 7.471809, 16.127170, 7.466813, 7.285180, 20.882691, 11.455414, 7.766787, 21.747888, 1 [12.334002, 0.00000, 13.540331, 6.451701, 25.523405, 11.793992, 7.842151, 9.737505, 14.188506, 28.598276, 9.478701, 6.854585, 15.504805, 8.1 [12.403480, 13.540331, 0.00000, 19.389512, 12.508829, 18.131892, 21.130562, 17.244117, 19.490613, 15.375441, 5.478151, 7.885511, 14.803800, [13.540331, 6.451701, 19.389512, 0.00000, 14.690805, 9.136441, 2.796468, 7.347887, 11.781438, 31.131405, 10.092751, 11.588848, 22.170641, 17 [14.014871, 25.523405, 12.508829, 14.690805, 0.00000, 25.565366, 17.244087, 17.400380, 22.528579, 20.728884, 36.570751, 11.653862, 47.064057 [7.471809, 11.793992, 18.131892, 9.136441, 25.565366, 0.00000, 11.788880, 2.073958, 2.453377, 28.387848, 17.740478, 11.534174, 26.817138, 11 [16.127170, 7.842151, 21.130562, 2.796468, 17.244087, 11.788880, 0.00000, 10.813677, 14.125108, 35.011157, 17.728897, 11.845128, 22.609080, [7.466813, 9.737505, 17.244117, 7.347887, 27.480888, 2.873558, 10.813677, 0.00000, 4.638611, 20.288925, 18.153289, 10.470879, 24.531227, 17 [7.285180, 14.188506, 19.490613, 11.781438, 22.528579, 2.453377, 14.125108, 4.638611, 0.00000, 27.523129, 19.214884, 12.973638, 28.721485, 1 [20.882691, 28.598276, 25.574541, 21.131405, 18.728884, 28.387848, 26.611157, 10.288525, 17.523129, 0.00000, 21.629051, 22.165564, 29.121793 [11.455414, 9.478701, 5.478151, 16.892251, 36.570751, 17.740478, 17.728897, 10.155289, 19.214884, 21.629051, 0.00000, 6.247266, 10.487238, 1 [7.766787, 6.854585, 7.885511, 11.588848, 11.653862, 11.134174, 11.845128, 10.017879, 12.973638, 22.165561, 6.247266, 0.00000, 18.001122, 8 [11.747888, 15.504805, 14.803800, 21.170641, 47.064057, 26.817138, 29.686680, 24.531227, 28.721485, 29.127799, 18.457238, 16.001232, 0.00000 [15.519717, 8.988151, 9.385759, 15.450287, 29.595348, 18.988105, 16.551874, 17.382096, 20.818881, 25.181838, 3.648921, 8.208414, 7.925312, 0 [16.703678, 18.818251, 20.818251, 20.818251, 14.125109, 11.571257, 47.714081, 12.847863, 29.388065, 10.817579, 10.780129, 12.775299, 43.47919 [20.346497, 18.891356, 14.126126, 16.844788, 44.246017, 22.279544, 17.147797, 20.298742, 24.483999, 29.778816, 8.599452, 12.681343, 5.491581 [12.521313, 11.214349, 11.748889, 11.951887, 47.118137, 22.290811, 12.646880, 20.244371, 24.875481, 17.688918, 16.114397, 16.968278, 14.15005 [15.509422, 14.591189, 4.112098, 20.381688, 17.087888, 20.961887, 22.820782, 19.584087, 22.864284, 18.617957, 4.342884, 9.573888, 10.987182, [28.096118, 40.012481, 11.879189, 41.477822, 16.608711, 11.580468, 40.237789, 16.114319, 11.215166, 10.153888, 10.127147, 11.718481, 43.84679 [20.365786, 41.624356, 16.608486, 41.102455, 7.388419, 12.045769, 43.746354, 13.755146, 29.467599, 29.028188, 40.698958, 16.851122, 11.25158 [13.286828, 40.020580, 26.147843, 46.848155, 18.912442, 40.711891, 47.258714, 40.598555, 40.078564, 12.596883, 11.107872, 13.428751, 16.87557 [17.499621, 28.748889, 16.567951, 36.838805, 17.731838, 17.566471, 51.628463, 16.138327, 21.881511, 11.080013, 26.725184, 22.383155, 18.4884 [25.582778, 17.415486, 28.161419, 29.028286, 16.404342, 11.068829, 41.822489, 21.147489, 29.033480, 16.749494, 11.618987, 11.132081, 43.11950 [17.983418, 6.175714, 16.718175, 11.374869, 41.811416, 17.051887, 11.183466, 11.888248, 20.386749, 11.644295, 18.236678, 10.519791, 11.53112 [11.044547, 12.154211, 11.821931, 16.438081, 47.722587, 21.880838, 15.688104, 21.887826, 26.128759, 19.789747, 14.156781, 16.525561, 10.18848 [16.649591, 21.784444, 18.531783, 28.133841, 29.493777, 24.079138, 10.617905, 21.880884, 23.638782, 5.189634, 16.218171, 16.915795, 24.173128 [25.109084, 17.386648, 16.277228, 11.489388, 48.578319, 28.248209, 21.732714, 16.366884, 10.199186, 40.388111, 11.818205, 17.044111, 1.520085 [15.454813, 47.018887, 39.713811, 48.397653, 17.748133, 39.747829, 11.158757, 41.178781, 17.385318, 28.493562, 48.831084, 41.889489, 54.58795]</pre>	Memasukkan data distance matrix yang berfungsi menentukan jarak tempuh antar titik dalam perhitungan waktu perjalanan, serta digunakan untuk menghitung total jarak rute kendaraan.
<pre>delivery_demand = [0, 12, 5, 4, 11, 10, 8, 9, 3, 3, 4, 7, 11, 15, 15, 13, 12, 9,</pre>	Memasukkan data <i>Demand delivery</i> (dalam kg) yang harus dikirimkan ke masing-masing <i>customer</i> . Formatnya berupa list satu dimensi, dengan urutan sesuai dengan indeks <i>customer</i> .
<pre>time_windows = [1, 2, 2, 1, 1, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 2, 2,</pre>	Memasukkan data <i>time windows</i> yang merepresentasikan batas waktu layanan untuk setiap <i>customer</i> dalam satuan menit (misalnya 480–780 menit = 08:00–13:00 WIB). Time window disimpan dalam bentuk <i>dictionary</i> yang memetakan indeks <i>node</i> ke <i>tuple</i> waktu mulai dan akhir layanan.

Pseucode	Fungsi
<pre> # TW Label ke range menit (dalam satuan menit dari jam 0) TW_RANGES = { 1: (8 * 60, 13 * 60), # 08:00-13:00 (480-780) 2: (13 * 60 + 1, 17 * 60), # 13:01-17:00 (781-1020) 3: (17 * 60 + 1, 21 * 60) # 17:01-21:00 (1021-1260) } # Waktu service default jika tidak ada TW (misal node tidak terdaftar) DEFAULT_TW = (0, 1000) # Ubah list time_windows ke dict time_windows_actual def map_time_windows(labels, tw_ranges, default_tw=(0, 1000)): time_windows_dict = {0: default_tw} # depot = node 0 for idx, label in enumerate(labels, start=1): time_windows_dict[idx] = tw_ranges.get(label, default_tw) return time_windows_dict # Gunakan fungsi time_windows_actual = map_time_windows </pre>	<p>berfungsi untuk mengelola dan memetakan <i>Time Window</i> (TW) dalam bentuk label menjadi rentang waktu sebenarnya (dalam satuan menit). Hasilnya akan berguna untuk pengecekan <i>constraint time window</i> dalam evaluasi <i>fitness</i> PSO seperti keterlambatan.</p>

4.3.3.3 Menghitung Emisi Karbon

Perhitungan emisi karbon dalam model PSO-VRPTW dilakukan menggunakan pendekatan berbasis ton-kilometer. Emisi dihitung dengan mengalikan jarak tempuh kendaraan dengan bobot barang yang dibawa pada segmen tersebut, serta faktor emisi sebesar 0,065kg CO₂/ton-km. Pendekatan ini lebih akurat karena memperhitungkan dinamika muatan kendaraan selama perjalanan distribusi, bukan hanya total jarak tempuh saja. Nilai faktor emisi 0,065 yang didapatkan dari nilai dasar pada standar DEFRA, studi logistik Eropa, dan asumsi kendaraan logistik ringan.

Emisi karbon dari kendaraan yang memindahkan muatan q_{ij} kg sejauh d_{ij} meter dihitung dengan model:

$$E_{ij} = \left(\frac{d_{ij} \cdot q_{ij}}{1000} \right) \times \gamma$$

Keterangan:

- d_{ij} = jarak dari titik i ke j (km)
- q_{ij} = sisa berat muatan dari titik i ke j (kg)
- γ = emisi karbon (kg CO₂ per km per ton)

Contoh perhitungannya adalah menggunakan jarak dari *main hub* ke *customer* 1 sebagai berikut.

$$E_{ij} = \left(\frac{12,334 \text{ km} \times 300 \text{ kg}}{1000} \right) \times 0,065 = 0,2405 \text{ kgCO}_2$$

Dari hasil perhitungan tersebut emisi karbon yang dihasilkan dari main hub ke customer 1 sebesar 12,33km dengan membawa 300kg atau 0,3 ton muatan adalah 0,2405 kgCO₂. Berikut merupakan penjelasan *pseudocode* dari perhitungan emisi karbon.

Tabel 4. 15 *Pseudocode* Emisi Karbon

<i>Pseudocode</i>	Fungsi
<pre>def hitung_emisi_per_point(jarak, sisa_beban_kg, emisi_per_km_per_ton=0.065): berat_ton = sisa_beban_kg / 1000 return jarak * berat_ton * emisi_per_km_per_ton</pre>	<p>Menghitung estimasi emisi karbon yang dihasilkan oleh kendaraan logistik berdasarkan jarak tempuh dan berat muatan yang dibawa. Menjadi bagian dari penilaian <i>fitness function</i> dalam optimasi berbasis emisi.</p>

4.3.3.4 Evaluasi Solusi

Evaluasi solusi yang ditampilkan dalam algoritma PSO untuk menyelesaikan permasalahan *Vehicle Routing Problem with Time windows* (VRPTW) pada studi kasus PT Pos Logistik berfungsi untuk menghitung kualitas solusi dari setiap partikel (solusi rute kendaraan). Fungsi ini mengevaluasi beberapa aspek penting, yakni total jarak tempuh, emisi karbon, penalti akibat pelanggaran kapasitas dan *time windows*, serta total biaya operasional.

Pada tahap evaluasi solusi, setiap rute yang dihasilkan akan dianalisis berdasarkan beberapa parameter. Pertama, jarak tempuh antar titik diukur menggunakan matriks jarak dan dihitung waktu tempuh berdasarkan kecepatan kendaraan. Selanjutnya, beban total kendaraan dijumlahkan dan diperiksa apakah melebihi kapasitas. Jika ya, maka diberikan penalti *overload*. *Time window* dari

pelanggan juga diperiksa; jika kendaraan datang terlalu awal atau terlambat, maka penalti tambahan diberikan. Total emisi karbon dihitung berdasarkan jarak tempuh dan beban, sedangkan total biaya dihitung berdasarkan tarif per kilometer. Seluruh nilai tersebut kemudian digunakan untuk menentukan nilai *fitness* akhir dari solusi tersebut. Berikut merupakan penjelasan *pseudocode* tahapan evaluasi solusi yang digunakan.

Tabel 4. 16 *Pseudocode* Evaluasi Solusi

<i>Pseudocode</i>	Fungsi
<pre> def evaluate_solution(solution, distance_matrix, demands, time_windows): total_distance = 0 total_emissions = 0 total_penalty = 0 total_cost = 0 vehicle_cost_per_km = 3000 #rupiah/km </pre>	<p>Potongan kode tersebut merupakan bagian dari fungsi <i>evaluate_solution</i> yang digunakan untuk mengevaluasi solusi dalam permasalahan routing. Fungsi ini menerima input berupa solusi rute, matriks jarak, <i>Demand</i>, <i>time windows</i>, dan kapasitas kendaraan maksimum. Di awal, tiga variabel diinisialisasi ke nol untuk menyimpan total jarak tempuh, emisi, dan penalti.</p>
<pre> # Hitung total beban berdasarkan demand customer dalam rute vehicle_load = sum(demand[cust] for cust in route if cust != DEPOT) sisa_beban = vehicle_load # Cek jika muatan melebihi kapasitas maksimal (jika parameter diberikan) if max_vehicle_capacity is not None and vehicle_load > max_vehicle_capacity: penalty += (vehicle_load - max_vehicle_capacity) * 1000 # penalti overload </pre>	<p>menghitung total beban kendaraan dalam satu rute berdasarkan <i>Demand</i> dari setiap <i>customer</i>, dengan mengabaikan node DEPOT. Nilai total beban disimpan</p>

Pseudocode	Fungsi
	<p>dalam <code>vehicle_load</code>, yang juga digunakan sebagai <code>sisa_beban</code>. Setelah itu, dilakukan pengecekan apakah beban tersebut melebihi kapasitas kendaraan, jika nilai kapasitas diberikan. Jika terjadi <i>overload</i>, maka dihitung penalti sebesar selisih beban yang melebihi kapasitas dikalikan 1000, yang kemudian ditambahkan ke variabel penalti.</p>
<pre> for i in range(1, len(route)): next_node = route[i] dist = distance_matrix[current_node][next_node] travel_time = (dist / VEHICLE_SPEED) * 60 time += travel_time if next_node != DEPOT: tw_start, tw_end = TW_RANGES[time_windows[next_node]] if time < tw_start: time = tw_start elif time > tw_end: penalty += (time - tw_end) total_emisi += hitung_emisi_per_point(dist, sisa_beban) sisa_beban -= demand[next_node] total_distance += dist current_node = next_node return total_distance + penalty + total_emisi * 10 # total biaya (jarak + penalti + emisi berbot) </pre>	<p>Potongan kode ini menghitung total jarak, emisi, dan penalti waktu untuk satu rute. Dalam setiap iterasi, ditentukan <i>next_node</i> sebagai tujuan berikutnya, lalu dihitung jarak dan waktu tempuh dari <i>current_node</i>. Jika <i>next_node</i> bukan DEPOT, dicek apakah waktu kedatangan sesuai dengan jendela waktu (<i>time_windows</i>). Jika terlalu awal, waktu disesuaikan dengan <i>tw_start</i>; jika terlambat, penalti diberikan sesuai jumlah keterlambatan.</p>

<i>Pseudocode</i>	Fungsi
	Emisi karbon dihitung berdasarkan jarak dan sisa beban barang, kemudian beban diperbarui setelah pengantaran.

4.3.3.5 Inisialisasi *Nearest Neighbor*

Proses inisialisasi ini dimulai dengan pemetaan jarak antar lokasi menggunakan matriks jarak yang merepresentasikan jarak fisik antara titik-titik *customer* dan depot. Langkah pertama adalah memilih depot sebagai titik awal kendaraan. Kemudian, kendaraan akan diarahkan menuju lokasi *customer* terdekat yang belum dikunjungi berdasarkan jarak terpendek yang tercatat dalam matriks tersebut. Setelah kendaraan mengunjungi *customer* tersebut, permintaan *customer* akan dikurangi dari kapasitas kendaraan yang tersedia. Proses ini diulang secara *greedy*, yaitu selalu memilih *customer* terdekat berikutnya yang masih memenuhi kapasitas kendaraan dan waktu layanan, sehingga tidak ada lagi *customer* yang dapat dilayani oleh kendaraan tersebut. Jika masih terdapat *customer* yang belum terlayani, kendaraan lain akan digunakan dengan cara yang sama. Dengan demikian, pemilihan rute berdasarkan jarak terdekat, pembaruan kapasitas kendaraan setelah setiap kunjungan, serta pengelolaan daftar *customer* yang belum dikunjungi hingga seluruh permintaan terpenuhi. Pendekatan ini memberikan solusi awal yang efisien sebagai dasar untuk optimasi lebih lanjut menggunakan algoritma PSO. Berikut merupakan penjelasan *pseudocode* tahapan *nearest neighbor* dalam PSO.

Tabel 4. 17 *Pseudocode Nearest Neighbor*

<i>Pseudocode</i>	<i>Fungsi</i>
<pre> def nearest_neighbor_route(start, customers, distance_matrix): route = [] current = start unvisited = customers[:] while unvisited: nearest = min(unvisited, key=lambda x: distance_matrix[current][x]) route.append(nearest) unvisited.remove(nearest) current = nearest return route </pre>	<p>Fungsi dimulai dari titik <i>start</i>, lalu membuat daftar <i>unvisited</i> berisi seluruh <i>customer</i> yang belum dikunjungi. Selama masih ada <i>customer</i> yang belum dikunjungi, fungsi akan memilih <i>customer</i> terdekat dari titik saat ini berdasarkan nilai terkecil pada <i>distance matrix</i>. Titik terdekat tersebut ditambahkan ke rute, dihapus dari daftar <i>unvisited</i>, dan dijadikan titik <i>current</i> selanjutnya.</p>
<pre> def generate_initial_solution(distance_matrix): customers = list(range(1, NUM_CUSTOMERS+1)) random.shuffle(customers) k, m = divmod(len(customers), NUM_VEHICLES) solution = [customers[i*k+min(i, m):(i+1)*k+min(i+1, m)] for i in range(NUM_VEHICLES)] return [nearest_neighbor_route(DEPOT, r, distance_matrix) for r in solution] </pre>	<p>Berfungsi untuk menghasilkan solusi awal dalam permasalahan <i>Vehicle Routing Problem</i> (VRP). Pertama, daftar <i>customer</i> dibuat dari nomor 1 hingga <i>num customers</i>, lalu diacak secara acak menggunakan <i>random.shuffle</i>. Kemudian, daftar <i>customer</i> tersebut dibagi rata ke dalam sejumlah kendaraan menggunakan fungsi <i>divmod</i>, yang membagi</p>

<i>Pseudocode</i>	Fungsi
	jumlah <i>customer</i> secara proporsional.

4.3.3.6 Mendefinisikan Partikel dan PSO

Setiap partikel memiliki posisi yang mencerminkan rute kendaraan dan nilai fitness yang menunjukkan kualitas solusi tersebut. Algoritma *Particle Swarm Optimization* (PSO) dimodifikasi untuk beroperasi dalam ruang diskret, dengan mengganti operasi kecepatan klasik dengan operator kombinatorial, seperti pertukaran titik (*swap*) dan *crossover* rute. Dalam solusi VRPTW pada penelitian ini berbentuk kombinatorial, implementasi PSO dalam penelitian ini disesuaikan. Posisi partikel direpresentasikan sebagai daftar rute, dan mutasi dilakukan secara eksplisit melalui penukaran dua node secara acak (*swap mutation*) dalam satu rute. Pendekatan ini mempertahankan prinsip dasar PSO—eksploitasi dan eksplorasi—dengan menyesuaikan strategi pembaruan untuk konteks diskret.

Fungsi *pso* menginisialisasi sejumlah partikel berdasarkan hasil algoritma *nearest neighbor* yang memenuhi batas maksimal rute dan kapasitas kendaraan. Kemudian, algoritma PSO dijalankan dalam sejumlah iterasi untuk memperbaharui posisi setiap partikel berdasarkan solusi terbaik individu dan global. Berikut penjelasan *pseudocode* dari tahapan PSO dan *Particle*.

Tabel 4. 18 *Pseudocode* PSO

<i>Pseudocode</i>	Fungsi
<pre>NUM_PARTICLES = 30 MAX_ITER = 1000</pre>	mendefinisikan dua parameter utama dalam algoritma <i>Particle Swarm Optimization</i> (PSO). Dua parameter ini mengatur skala pencarian solusi dan durasi algoritma dijalankan.

Pseudocode	Fungsi
<pre>def PSO(distance_matrix, demand, time_windows): particles = [generate_initial_solution(distance_matrix) for _ in range(NUM_PARTICLES)] pbest = copy.deepcopy(particles) pbest_scores = [evaluate_solution(p, distance_matrix, demand, time_windows) for p in pbest] gbest = copy.deepcopy(pbest[np.argmin(pbest_scores)])</pre>	<p>Fungsi PSO membangkitkan sejumlah solusi awal (partikel), menyimpan salinannya sebagai posisi terbaik individu (pbest), mengevaluasi kualitas tiap solusi, lalu menentukan solusi terbaik global (gbest) berdasarkan skor terbaik dari seluruh partikel.</p>
<pre>for iteration in range(num_iterations): for particle in particles: fitness, _, _ = evaluate_solution(particle.position, distance_matrix, demands, time_wi if fitness < particle.best_fitness: particle.best_fitness = fitness particle.best_position = copy.deepcopy(particle.position) if fitness < global_best_fitness: global_best_fitness = fitness global_best_position = copy.deepcopy(particle.position)</pre>	<p>Nilai <i>fitness</i> ini kemungkinan besar mencerminkan kualitas rute, misalnya, total jarak atau biaya. Setelah evaluasi, <i>fitness</i> partikel saat ini dibandingkan dengan <i>fitness</i> terbaik yang pernah dicapai oleh partikel tersebut sebelumnya. Jika <i>fitness</i> lebih baik (lebih kecil, karena umumnya masalah optimasi mencari nilai minimum), maka <i>particle.best_fitness</i> dan <i>particle.best_position</i> diperbarui. Selanjutnya, <i>fitness</i> partikel saat ini juga dibandingkan dengan <i>global_best_fitness</i>.</p>

<i>Pseudocode</i>	Fungsi
<pre> for _ in range(MAX_ITER): for i in range(NUM_PARTICLES): new_particle = copy.deepcopy(particles[i]) for route in new_particle: if len(route) > 1: a, b = random.sample(range(len(route)), 2) route[a], route[b] = route[b], route[a] if random.random() < 0.3: a, b = random.sample(range(NUM_VEHICLES), 2) if new_particle[a]: move = new_particle[a].pop() new_particle[b].append(move) score = evaluate_solution(new_particle, distance_matrix, demand, time_windows) if score < pbest_scores[i]: pbest[i] = new_particle pbest_scores[i] = score best_idx = np.argmin(pbest_scores) gbest = copy.deepcopy(pbest[best_idx]) return gbest </pre>	<p>untuk setiap iterasi hingga batas <i>max_iter</i>, setiap partikel dimodifikasi melalui pertukaran titik dalam rute dan kemungkinan perpindahan pelanggan antar kendaraan. Solusi baru dievaluasi, dan jika lebih baik dari solusi terbaik individu sebelumnya (<i>pbest</i>), maka <i>pbest</i> dan skornya diperbarui. Setelah seluruh partikel diperbarui, solusi terbaik global (<i>gbest</i>) diperoleh dari <i>pbest</i> dengan skor terbaik. Hasil akhir <i>gbest</i> dikembalikan sebagai solusi terbaik.</p>

4.3.3.7 Output Execution

Proses *output execution* yaitu bagaimana hasil akhir dari proses optimasi rute ditampilkan secara informatif. Fungsi ini juga memperhitungkan waktu tiba dan validasi terhadap *time window* yang ditentukan pada masing-masing node. Jika kendaraan tiba lebih awal dari batas waktu layanan, maka akan dilakukan simulasi waktu tunggu hingga waktu layanan dimulai. Status kedatangan dicetak sebagai tiga kemungkinan: "terlalu cepat", "terlambat", atau "tepat waktu", yang ditentukan berdasarkan perbandingan antara waktu tiba dan *time window service*. Informasi ini penting untuk memastikan bahwa solusi rute tidak hanya optimal dari sisi jarak dan emisi, tetapi juga memenuhi batasan *time window customer*. Pada akhir eksekusi, total jarak, total *Demand*, dan total emisi dari seluruh rute kendaraan ditampilkan sebagai ringkasan performa solusi. Berikut merupakan *pseudocode* tahapan *output execution* algoritma PSO.

Tabel 4. 19 Pseudocode Output Execution

Pseudocode	Fungsi
<pre>def print_solution(solution, distance_matrix, demand, time_windows): for v, route in enumerate(solution): print(f"\nBlindvan {v+1}") if not route: print(" Tidak ada pengantaran.") continue full_route = [DEPOT] + route + [DEPOT] total_dist = 0 total_emisi = 0 current_time = 480 # Hitung total beban aktual dari demand pelanggan yang dikunjungi vehicle_load = sum(demand[cust] for cust in route) sisa_beban = vehicle_load print(f" Total load awal: {vehicle_load} kg") for i in range(len(full_route) - 1): current_node = full_route[i] next_node = full_route[i + 1] dist = distance_matrix[current_node][next_node] travel_time = (dist / VEHICLE_SPEED) * 60 current_time += travel_time if next_node == DEPOT: break # kembali ke hub</pre>	<p>Jika suatu kendaraan tidak memiliki pengantaran, maka dilewati. Untuk rute yang ada, total jarak, emisi, dan waktu mulai dihitung. Rute diperluas dengan menambahkan depot di awal dan akhir, lalu dihitung total beban berdasarkan permintaan pelanggan. Selanjutnya, iterasi dilakukan untuk menghitung waktu tempuh antar titik berdasarkan jarak dan kecepatan kendaraan. Jika kendaraan kembali ke depot, perhitungan dihentikan.</p>
<pre># Klasifikasi waktu kembali ke hub arrival_status = "" if current_time < 480: arrival_status = "Terlalu cepat" elif current_time <= 1260: arrival_status = "Tepat waktu" else: arrival_status = "Terlambat"</pre>	<p>Berfungsi mengklasifikasikan status waktu kedatangan kendaraan kembali ke hub berdasarkan nilai <i>current_time</i> (dalam satuan menit). Jika waktu kedatangan kurang dari 480 menit (sebelum pukul 08.00), maka statusnya adalah "Terlalu cepat". Jika waktu kedatangan berada di rentang 480 hingga 1260 menit (antara pukul 08.00 hingga</p>

<i>Pseudocode</i>	<i>Fungsi</i>
	21.00), maka statusnya "Tepat waktu". Sedangkan jika melebihi 1260 menit (setelah pukul 21.00), maka statusnya dikategorikan sebagai "Terlambat".
<pre>emisi_titik = hitung_emisi_per_point(dist, sisa_beban) total_emisi += emisi_titik</pre>	<p>Menghitung emisi karbon pada satu titik perjalanan dengan memanggil fungsi <code>hitung_emisi_per_point</code> berdasarkan jarak tempuh dan sisa beban kendaraan, lalu menambahkan hasilnya ke total emisi keseluruhan.</p>
<pre>total_dist += dist print(f"Kembali ke hub, total jarak: {total_dist:.2f} km, " f"Total emisi karbon: {total_emisi:.2f} kgCO2, " f"Sisa muatan: {sisa_beban} kg (harus 0)")</pre>	<p>Mencetak ringkasan hasil perjalanan kendaraan, berupa total jarak tempuh, total emisi karbon yang dihasilkan, dan sisa muatan kendaraan yang idealnya nol saat kembali ke hub.</p>
<pre># Jalankan PSO best_solution = PSO(distance_matrix, delivery_demand, time_windows) # Cetak hasil print_solution(best_solution, distance_matrix, delivery_demand, time_windows)</pre>	<p>Fungsi untuk pemanggilan solusi terbaik dari Algoritma PSO serta mencetak hasil <i>output</i> dari algoritma yang dijalankan.</p>

Berdasarkan algoritma PSO yang dikembangkan, berikut merupakan hasil rute optimal yang dihasilkan pada penelitian ini.

Tabel 4. 20 Hasil Rute berdasarkan Algoritma PSO

Kendaraan	Rute
Kendaraan 1	0 – 33 – 48 – 87 – 56 – 62 – 91 – 35 – 10 – 107 – 24 – 61 – 25 – 95 – 12 – 36 – 32 – 88 – 18 – 109 – 9 – 89 – 58 – 29 – 23 – 49 – 64 – 4 – 6 – 105 – 0
Kendaraan 2	0 – 44 – 39 – 5 – 41 – 69 – 50 – 1 – 46 – 65 – 47 – 101 – 97 – 63 – 22 – 72 – 31 – 14 – 71 – 60 – 37 – 21 – 80 – 81 – 51 – 52 – 83 – 27 – 0
Kendaraan 3	0 – 106 – 11 – 45 – 98 – 57 – 78 – 70 – 96 – 26 – 55 – 75 – 73 – 74 – 19 – 28 – 16 – 76 – 103 – 77 – 108 – 68 – 3 – 94 – 90 – 8 – 86 – 43 – 0
Kendaraan 4	0 – 93 – 54 – 7 – 99 – 53 – 100 – 59 – 17 – 15 – 13 – 66 – 2 – 30 – 67 – 92 – 84 – 82 – 104 – 40 – 110 – 102 – 34 – 20 – 42 – 79 – 38 – 85 – 0

Hasil pengolahan data menunjukkan bahwa penerapan algoritma *Particle Swarm Optimization* (PSO) mampu menghasilkan rute usulan yang lebih optimal dibandingkan dengan rute awal perusahaan. Pada rute awal, distribusi titik pelayanan antar kendaraan cenderung tidak merata, dengan urutan kunjungan yang belum terstruktur secara efisien. Sementara itu, pada rute usulan, urutan kunjungan pelanggan pada masing-masing kendaraan lebih teratur, dengan jumlah titik pelayanan yang lebih seimbang antar kendaraan. Sebagai contoh, kendaraan 1 pada rute usulan melayani titik 0 – 33 – 48 – 87 – ... – 0 dengan pola yang lebih sistematis dibandingkan dengan rute awal yang panjang dan acak. Hal ini menunjukkan bahwa PSO mampu memperhitungkan batasan kapasitas kendaraan, jumlah armada, serta *time windows* untuk memperoleh solusi yang lebih baik secara operasional. Namun demikian, rute usulan ini masih disusun dengan asumsi kecepatan kendaraan konstan, serta jumlah armada dan kapasitas

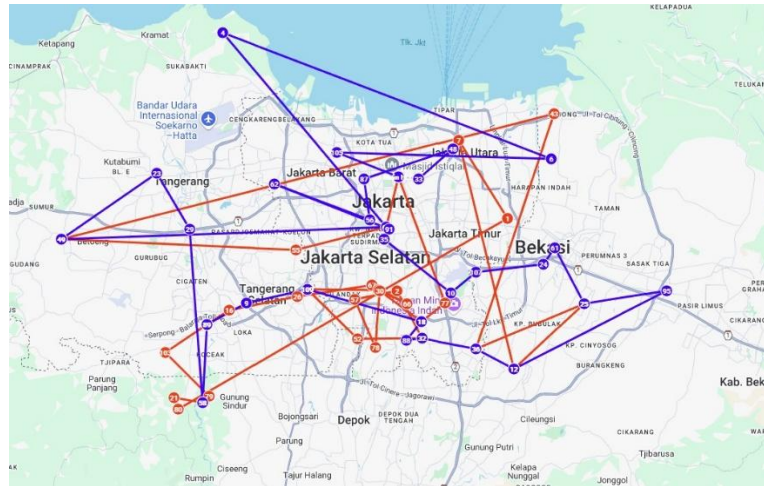
yang tetap, sehingga belum sepenuhnya mencerminkan kondisi aktual di lapangan.

BAB V

HASIL DAN PEMBAHASAN

5.1 Hasil dari Pengembangan Algoritma PSO

5.1.1 Analisis Rute & Jarak Kendaraan 1



Gambar 5. 1 Visualisasi Rute Kendaraan 1

Berdasarkan hasil visualisasi rute kendaraan 1, rute awalan (merah) melayani sebanyak 30 titik *customer* dengan pola distribusi yang kurang efisien, ditandai oleh lintasan yang berulang, tumpang tindih, serta terkonsentrasi pada wilayah utara dan timur Jakarta. Sementara itu, rute usulan (biru) melayani 29 titik *customer* dengan distribusi yang lebih merata ke arah utara, barat, dan selatan, serta jalur yang lebih linier. Perbandingan ini menunjukkan bahwa meskipun jumlah titik *customer* pada rute usulan sedikit lebih rendah, pola distribusinya lebih optimal, sehingga berpotensi menurunkan total jarak tempuh dan meningkatkan efisiensi operasional distribusi dibandingkan dengan rute awalan. Berikut rincian data hasil dari rute kendaraan *blindvan* 1.

Tabel 5. 1 Rincian Rute Optimal Kendaraan 1

Tujuan (origin- destination)	Jarak (km)	<i>Demand</i> (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO2)
0 - 33	2,78	16	336,00	0,05
33 - 48	6,25	19	320,00	0,1
48 - 87	12,91	18	301,00	0,19
87 - 56	5,62	8	283,00	0,08
56 - 62	13,95	11	275,00	0,19
62 - 91	16,93	12	264,00	0,22
91 - 35	1,60	10	252,00	0,02
35 - 10	11,73	4	242,00	0,14
10 - 107	4,46	18	238,00	0,05
107 - 24	9,28	14	220,00	0,1
24 - 61	2,74	7	206,00	0,03
61 - 25	8,67	16	199,00	0,09
25 - 95	11,52	10	183,00	0,11
95 - 12	23,66	11	173,00	0,2
12 - 36	5,89	17	162,00	0,05
36 - 32	7,58	5	145,00	0,05
32 - 88	2,17	5	140,00	0,02
88 - 18	3,33	8	135,00	0,02
18 - 109	16,32	18	127,00	0,1
109 - 9	8,59	3	109,00	0,05
9 - 89	6,21	8	106,00	0,03
89 - 58	10,69	13	98,00	0,05
58 - 29	23,88	20	85,00	0,1
29 - 23	8,98	9	65,00	0,03
23 - 49	15,98	18	56,00	0,04

Tujuan (<i>origin-destination</i>)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO ₂)
49 - 64	44,75	13	38,00	0,09
64 - 4	34,96	11	25,00	0,04
4 - 6	48,43	8	14,00	0,03
6 - 105	29,55	6	6,00	0,01
105 - 0	9,18	0	0,00	0,05
Total	408,56	336,00		2,28

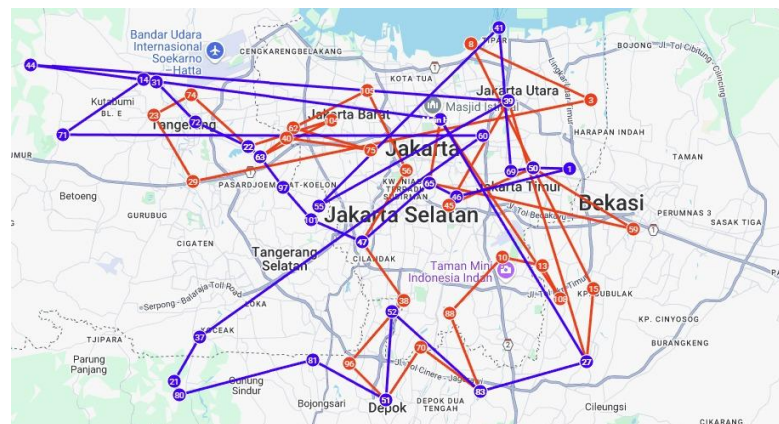
Hasil optimasi rute 1 dengan metode *Particle Swarm Optimization* (PSO) menunjukkan bahwa kendaraan memulai perjalanan dari titik 0 (*main hub*) dan mengunjungi total 29 titik pelanggan sebelum kembali ke titik awal. Jarak tempuh total sepanjang 408,56 km mengindikasikan jangkauan yang cukup luas dan mencakup distribusi ke berbagai wilayah. Dengan kapasitas awal kendaraan sebesar 336 kg, seluruh beban berhasil terdistribusi secara efisien tanpa melebihi batas kapasitas. Hal ini mencerminkan keberhasilan PSO dalam menyusun rute yang mempertimbangkan batasan kapasitas kendaraan secara optimal.

Dari sisi emisi karbon, total emisi yang dihasilkan selama rute ini adalah 2,28 kg CO₂. Nilai ini relatif rendah dikarenakan jumlah titik layanan yang cukup banyak. Ini menunjukkan bahwa PSO tidak hanya memperhatikan aspek jarak dan kapasitas, tetapi juga berhasil meminimalkan pergerakan kendaraan yang tidak efisien, sehingga berdampak positif pada pengurangan emisi. Terutama terlihat pada segmentasi jarak pendek seperti 91–35 sebesar 1,6 km dan 88–18 sebesar 3,33 km yang hanya menghasilkan emisi 0,02 kg CO₂, menunjukkan efisiensi penggabungan titik-titik dengan lokasi berdekatan.

Secara keseluruhan, rute 1 mencerminkan keseimbangan antara penggunaan muatan kendaraan, jarak tempuh, dan pengendalian emisi karbon. PSO berhasil membentuk urutan pengiriman yang memaksimalkan kapasitas

muatan tanpa mengorbankan efisiensi jarak dan emisi. Pengembalian kendaraan ke *hub* dengan sisa muatan 0 kg menunjukkan bahwa distribusi dilakukan secara penuh, tetapi kendaraan kembali ke *main hub* melewati batas waktu operasional, yang mengindikasikan bahwa aspek *time window* belum sepenuhnya terpenuhi dalam rute ini.

5.1.2 Analisis Rute & Jarak Kendaraan 2



Gambar 5. 2 Visualisasi Rute Kendaraan 2

Visualisasi rute kendaraan 2 menunjukkan bahwa rute awalan (merah) melayani 30 titik *customer* dengan pola lintasan yang terpusat di wilayah tengah (Jakarta) dan timur (Bekasi), serta jalur distribusi yang tidak terstruktur dan banyak mengalami persilangan. Sebaliknya, rute usulan (biru) melayani 27 titik *customer* dengan distribusi yang lebih merata ke wilayah barat, selatan, dan timur, serta jalur yang lebih sistematis, linier, dan minim tumpang tindih. Hal ini mengindikasikan bahwa rute usulan lebih efisien dalam meminimalkan jarak tempuh dan lebih efektif dalam mendistribusikan beban pelayanan, meskipun dengan jumlah titik *customer* yang lebih sedikit dibandingkan rute awalan. Berikut rincian data hasil dari rute kendaraan *blindvan* 2.

Tabel 5. 2 Rincian Rute Optimal Kendaraan 2

Tujuan (origin- destination)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO2)
0 - 44	47,98	20,00	362	5,55
44 - 39	55,65	5,00	342	3,54
39 - 5	5,16	10,00	337	4,05
5 - 41	3,59	15,00	327	0,51
41 - 69	16,82	16,00	312	0,83
69 - 50	2,59	18,00	296	2,11
50 - 1	4,20	12,00	278	1,05
1 - 46	13,52	16,00	266	3,59
46 - 65	3,56	11,00	250	2,78
65 - 47	10,32	20,00	239	1,11
47 - 101	6,42	19,00	219	3,51
101 - 97	4,94	14,00	200	7,46
97 - 63	4,47	9,00	186	3,01
63 - 22	1,83	10,00	177	3,04
22 - 72	6,75	14,00	167	3,18
72 - 31	6,47	9,00	153	1,24
31 - 14	1,51	15,00	144	0,5
14 - 71	11,34	6,00	129	0,37
71 - 60	48,74	9,00	123	0,64
60 - 37	40,39	16,00	114	0,51
37 - 21	5,86	7,00	98	1,11
21 - 80	1,73	16,00	91	0,72
80 - 81	15,98	11,00	75	3,25
81 - 51	9,72	19,00	64	2,4
51 - 52	10,28	18,00	45	3,3

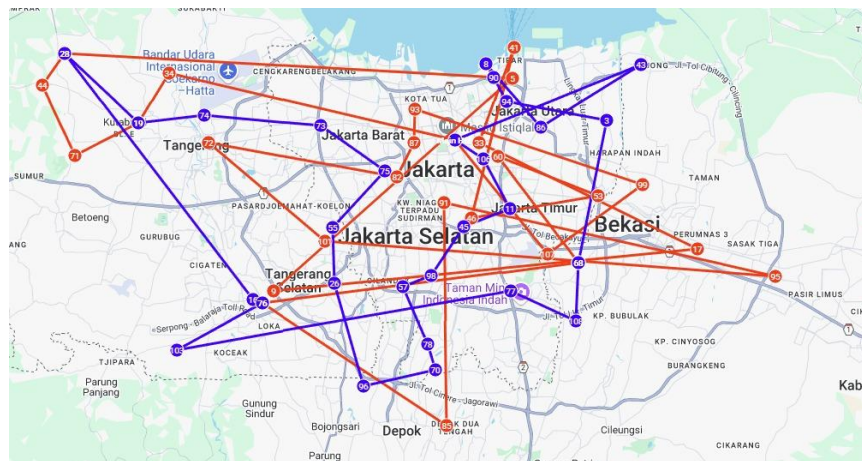
Tujuan (origin- destination)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO₂)
52 - 83	13,79	17,00	27	0,4
83 - 27	12,78	10,00	10	0,8
27 - 0	32,84	0,00	0	1,34
Total	399,24	362,00		3,75

Hasil optimasi rute 2 menunjukkan bahwa kendaraan memulai distribusi dari titik 0 dengan kapasitas awal 362 kg dan berhasil menyelesaikan pengiriman ke 27 titik pelanggan sebelum kembali ke *hub*. Total jarak tempuh mencapai 399,24 km, sedikit lebih pendek dari rute 1, namun dengan total *demand* yang lebih besar. Hal ini mencerminkan performa PSO yang efektif dalam memaksimalkan penggunaan kapasitas kendaraan tanpa melampaui batas, dengan tetap merancang rute yang efisien secara jarak dan distribusi beban. Selain itu, kendaraan berhasil kembali ke *main hub* tepat waktu sesuai dengan batas waktu operasional yang ditentukan, menunjukkan bahwa aspek *time windows* juga terpenuhi secara optimal.

Dari segi emisi karbon, total emisi tercatat sebesar 3,75 kg CO₂. Meskipun lebih tinggi dari rute 1, hal ini dapat dimaklumi karena bobot pengiriman lebih besar dan beberapa segmen jarak jauh seperti 0–44 (47,98 km) dan 60–37 (40,39 km) yang berkontribusi signifikan terhadap emisi. Namun, adanya beberapa segmen pendek dengan emisi tinggi, seperti 97–63 sebesar 3,01 kg CO₂ dan 63–22 sebesar 3,04 kg CO₂, mengindikasikan kemungkinan adanya ketidakseimbangan antara jarak dan muatan atau karakteristik medan yang memengaruhi konsumsi bahan bakar.

Secara keseluruhan, PSO pada rute 2 menunjukkan optimasi yang cukup baik dari segi distribusi beban dan jarak, meskipun masih terdapat ruang untuk perbaikan pada aspek efisiensi emisi karbon. Pengiriman diselesaikan dengan muatan habis, menandakan bahwa kapasitas kendaraan digunakan secara maksimal. Kepulangan kendaraan ke main hub yang tepat waktu menambah validasi bahwa solusi yang dihasilkan memenuhi seluruh kendala sistem logistik, termasuk keterbatasan waktu layanan.

5.1.3 Analisis Rute & Jarak Kendaraan 3



Gambar 5. 3 Visualisasi Rute Kendaraan 3

Pada rute kendaraan 3, rute awalan (merah) melayani 25 titik *customer* dengan pola distribusi yang tidak optimal, ditandai dengan lintasan yang berulang, berpotongan, serta terfokus pada wilayah barat dan timur Jakarta. Sebaliknya, rute usulan (biru) melayani 27 titik *customer* dengan pola distribusi yang lebih merata ke arah barat, utara, dan selatan, serta lintasan yang lebih terstruktur dan linier. Kondisi ini menunjukkan bahwa rute usulan tidak hanya mampu melayani jumlah titik *customer* yang lebih banyak, tetapi juga menyediakan pola distribusi yang lebih efisien dan mendukung peningkatan kinerja distribusi secara keseluruhan. Berikut rincian data hasil dari rute kendaraan *blindvan* 3.

Tabel 5. 3 Rincian Rute Optimal Kendaraan 3

Tujuan (<i>origin- destination</i>)	Jarak (km)	<i>Demand</i> (kg)	Sisa Muatan (kg)	Emisi Karbon (gram CO2)
0 - 106	3,93	14,00	305	0,06
106 - 11	6,47	7,00	291	0,09
11 - 45	5,73	16,00	284	0,08
45 - 98	6,72	8,00	268	0,09
98 - 57	3,56	20,00	260	0,05
57 - 78	7,18	16,00	240	0,09
78 - 70	3,12	18,00	224	0,03
70 - 96	8,52	10,00	206	0,09
96 - 26	12,34	18,00	196	0,12
26 - 55	6,47	11,00	178	0,06
55 - 75	8,84	9,00	167	0,07
75 - 73	9,05	13,00	158	0,07
73 - 74	13,52	14,00	145	0,1
74 - 19	7,59	12,00	131	0,05
19 - 28	11,67	6,00	119	0,07
28 - 16	35,67	12,00	113	0,2
16 - 76	1,21	6,00	101	0,01
76 - 103	11,22	14,00	95	0,05
103 - 77	39,09	8,00	81	0,16
77 - 108	8,22	8,00	73	0,03
108 - 68	6,70	13,00	65	0,02
68 - 3	16,74	4,00	52	0,04
3 - 94	11,71	18,00	48	0,03
94 - 90	3,12	6,00	30	0

Tujuan (origin- destination)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (gram CO2)
90 - 8	1,83	3,00	24	0
8 - 86	9,61	7,00	21	0,01
86 - 43	13,57	14,00	14	0,01
43 - 0	23,06	0,00	0	
Total	296,47	305,00		1,68

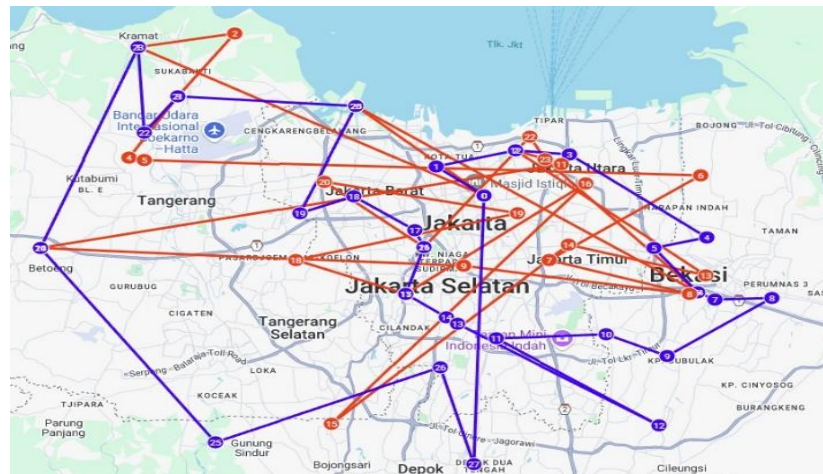
Hasil optimasi rute 3 menunjukkan bahwa kendaraan memulai perjalanan dari main hub dengan total muatan 305 kg dan berhasil mengunjungi 27 titik pelanggan dalam satu siklus distribusi. Total *demand* terdistribusi secara merata, dan seluruh kapasitas kendaraan digunakan secara maksimal hingga muatan tersisa nol saat kembali ke *hub*. Total jarak tempuh rute ini adalah 296,47 km, lebih pendek dibandingkan dua rute sebelumnya, yang menunjukkan efisiensi jarak cukup baik dalam penentuan urutan titik kunjungan. Namun, meskipun jarak relatif lebih pendek, waktu tempuh keseluruhan melebihi batas operasional harian yang telah ditetapkan, sehingga kendaraan terlambat kembali ke *main hub*.

Dari sisi emisi karbon, total emisi tercatat sebesar 1,68 kgCO₂, yang merupakan nilai paling rendah di antara rute-rute sebelumnya. Hal ini didukung oleh distribusi rute dengan segmen-segmen jarak menengah dan pendek yang konsisten, seperti 70–96 sebesar 8,52 km dan 108–68 sebesar 6,70 km, serta efisiensi muatan yang cukup stabil. Beberapa segmen memiliki emisi yang sangat kecil (bahkan nol) seperti pada 90–8 dan 8–86, yang menunjukkan bahwa bobot kendaraan ringan pada akhir rute berkontribusi terhadap penurunan konsumsi bahan bakar. Ini menandakan bahwa optimasi PSO bekerja efektif dalam menjaga keseimbangan antara muatan dan jarak.

Namun, meskipun performa dalam pemanfaatan muatan dan efisiensi emisi cukup baik, keterlambatan kembali ke hub menjadi catatan penting dalam

evaluasi solusi ini. Keterlambatan tersebut berisiko mengganggu operasional harian perusahaan, terutama jika kendaraan harus digunakan kembali untuk rute berikutnya.

5.1.4 Analisis Rute & Jarak Kendaraan 4



Gambar 5. 4 Visualisasi Rute Kendaraan 4

Berdasarkan visualisasi rute pada kendaraan ini, rute awalan (merah) melayani sebanyak 25 titik *customer* dengan pola distribusi yang kurang efisien, ditunjukkan oleh jalur yang berulang, tumpang tindih, dan terkonsentrasi di wilayah pusat kota, utara, serta timur Jakarta. Pola lintasan yang demikian berpotensi meningkatkan jarak tempuh dan waktu pelayanan. Sebaliknya, rute usulan (biru) melayani 24 titik *customer* dengan distribusi yang lebih merata ke wilayah barat, selatan, dan timur, serta lintasan yang lebih linier dan sistematis. Meskipun jumlah titik *customer* pada rute usulan sedikit lebih rendah, pola distribusi dan struktur lintasan yang lebih optimal ini berimplikasi pada peningkatan efisiensi operasional dan penurunan jarak tempuh dibandingkan dengan rute awalan. Berikut rincian data hasil dari rute kendaraan *blindvan* 4.

Tabel 5. 4 Rincian Rute Optimal Kendaraan 4

Tujuan (origin- destination)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO2)
0 - 93	5,84	6,00	332	0,1
93 - 54	7,93	16,00	326	0,13
54 - 7	5,23	9,00	310	0,08
7 - 99	16,76	10,00	301	0,25
99 - 53	5,25	7,00	291	0,08
53 - 100	6,88	17,00	284	0,1
100 - 59	1,89	7,00	267	0,03
59 - 17	5,47	9,00	260	0,07
17 - 15	12,34	13,00	251	0,15
15 - 13	6,46	15,00	238	0,08
13 - 66	10,73	6,00	223	0,12
66 - 2	2,22	5,00	217	0,02
2 - 30	2,41	16,00	212	0,03
30 - 67	1,35	8,00	196	0,01
67 - 92	4,86	8,00	188	0,05
92 - 84	6,02	12,00	180	0,05
84 - 82	2,29	16,00	168	0,02
82 - 104	7,22	16,00	152	0,05
104 - 40	5,59	15,00	136	0,04
40 - 110	14,18	19,00	121	0,09
110 - 102	17,36	6,00	102	0,09
102 - 34	5,49	14,00	96	0,03
34 - 20	10,45	12,00	82	0,04
20 - 42	26,35	19,00	70	0,09
42 - 79	29,17	19,00	51	0,07

Tujuan (origin- destination)	Jarak (km)	Demand (kg)	Sisa Muatan (kg)	Emisi Karbon (kg CO₂)
79 - 38	23,65	13,00	32	0,04
38 - 85	12,26	19,00	19,00	0,01
Total	255,62	332,00		1,92

Hasil optimasi rute 4 menunjukkan bahwa kendaraan memulai pengantaran dari main hub dengan total muatan 332 kg dan berhasil mendistribusikan 332 kg barang ke 27 titik tujuan. Rute ini memiliki total jarak tempuh sebesar 255,62 km, menjadikannya yang paling pendek dibanding rute-rute sebelumnya. Kendaraan secara bertahap mengurangi beban muatannya, dengan sisa muatan hanya 19 kg di titik akhir. Perencanaan distribusi melalui pendekatan PSO tampak cukup efektif dalam meminimalkan jarak dan menjaga stabilitas muatan selama proses pengantaran.

Dari sisi emisi karbon, rute ini menghasilkan total emisi sebesar 1,92 kg CO₂, yang relatif rendah dan efisien mengingat jumlah titik yang dikunjungi dan total beban yang dibawa. Efisiensi ini didukung oleh pola rute yang menghindari segmen-segmen berjarak sangat jauh secara berurutan. Beberapa bagian rute seperti 30–67, 67–92, dan 82–104 memperlihatkan penghematan emisi signifikan, yang memperkuat bukti bahwa algoritma PSO mampu menyusun urutan titik kunjungan yang hemat energi.

Kendaraan pada rute ini berhasil kembali ke *main hub* tepat waktu, menunjukkan bahwa tidak hanya aspek jarak dan emisi yang dioptimalkan, tetapi juga waktu operasional layanan. Kepulangan yang sesuai jadwal ini mencerminkan kepatuhan terhadap *constraint time window* dan efisiensi pergerakan kendaraan. Dengan demikian, rute 4 merupakan salah satu rute yang sangat seimbang dari sisi jarak, emisi, muatan, serta ketepatan waktu dalam sistem distribusi logistik berbasis PSO.

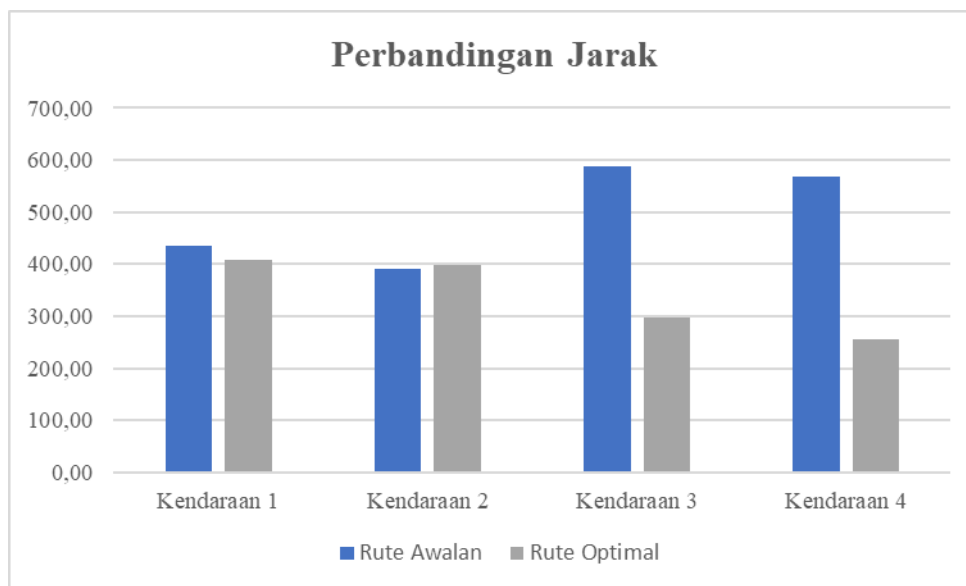
5.2 Perbandingan Jarak

Hasil perbandingan jarak tempuh antara rute awalan dan rute hasil optimasi menggunakan algoritma *Particle Swarm Optimization* (PSO), terdapat penurunan total jarak tempuh yang cukup signifikan. Jarak tempuh total dari keempat kendaraan pada rute awal adalah sebesar 1982,85km, sementara setelah dilakukan optimasi menjadi 1359,89km. Hal ini menunjukkan adanya efisiensi jarak tempuh sebesar 622,96 km atau sekitar 31,42%. Penurunan ini mengindikasikan bahwa algoritma PSO mampu menghasilkan solusi rute yang lebih optimal dan efisien dalam hal jarak tempuh.

Kendaraan 1 mengalami sedikit penurunan jarak dari 436,28 km menjadi 408,56 km, sedangkan kendaraan 2 justru mengalami peningkatan jarak dari 391,93 km menjadi 399,24 km. Namun, kendaraan 3 dan kendaraan 4 menunjukkan perbaikan yang sangat signifikan, dengan penurunan jarak dari 587,15 km menjadi 296,47 km dan dari 567,49 km menjadi 255,62 km. Hal ini menunjukkan bahwa PSO tidak hanya mengurangi total jarak tempuh, tetapi juga berhasil melakukan redistribusi beban rute antar kendaraan secara lebih merata dan efisien. Pengurangan jarak tempuh ini dapat berdampak langsung terhadap penghematan bahan bakar, waktu tempuh, dan penurunan emisi karbon, serta meningkatkan produktivitas armada secara keseluruhan. Berikut data perbandingan jarak dari rute awalan dan rute dari penerapan algoritma PSO.

Tabel 5. 5 Perbandingan Jarak Rute Awalan dan Rute Optimal

Perbandingan Jarak (km)		
	Rute Awalan	Rute Optimal
Kendaraan 1	436,28	408,56
Kendaraan 2	391,93	399,24
Kendaraan 3	587,15	296,47
Kendaraan 4	567,49	255,62
Total	1982,85	1359,89



Gambar 5. 5 Perbandingan Jarak

5.3 Perbandingan Emisi Karbon

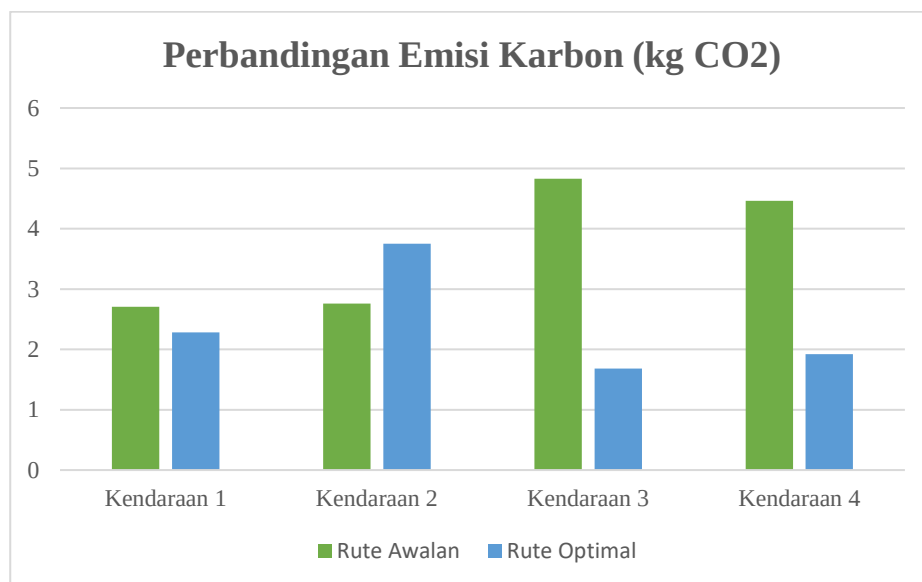
Berdasarkan hasil perbandingan emisi karbon antara rute awalan dan rute hasil optimasi menggunakan algoritma *Particle Swarm Optimization* (PSO), diketahui bahwa terjadi penurunan total emisi karbon yang cukup signifikan. Emisi karbon total dari seluruh kendaraan pada rute awal adalah sebesar 14,76 kg CO₂, sedangkan setelah dilakukan optimasi menjadi 9,63 kg CO₂. Dengan demikian, terjadi pengurangan emisi sebesar 5,13 kg CO₂ atau sekitar 34,76%, yang menunjukkan bahwa optimasi rute tidak hanya berdampak pada efisiensi jarak, tetapi juga berkontribusi terhadap aspek keberlanjutan dan ramah lingkungan.

Kendaraan 1 mengalami penurunan emisi dari 2,71 kg CO₂ menjadi 2,28 kg CO₂, dan Kendaraan 3 serta Kendaraan 4 menunjukkan penurunan yang lebih signifikan, masing-masing dari 4,83 kg CO₂ menjadi 1,68 kg CO₂ (turun 65,22%) dan dari 4,46 kg CO₂ menjadi 1,92 kg CO₂ (turun 56,96%). Kendaraan 2 mengalami peningkatan emisi dari 2,76 kg CO₂ menjadi 3,75 kg CO₂, yang kemungkinan disebabkan oleh redistribusi rute dan beban yang membuat kendaraan ini menempuh jarak lebih jauh

dibandingkan sebelumnya. Meskipun terdapat peningkatan pada salah satu kendaraan, hasil akhir menunjukkan bahwa PSO secara keseluruhan berhasil mengurangi emisi karbon dalam sistem distribusi. Penurunan emisi ini sejalan dengan tujuan keberlanjutan logistik modern yang tidak hanya menekankan efisiensi biaya dan waktu, tetapi juga mengurangi dampak lingkungan dari aktivitas transportasi. Berikut perbandingan dari emisi karbon dalam penelitian ini.

Tabel 5. 6 Perbandingan Emisi Karbon Rute Awalan dan Rute Optimal

Perbandingan Emisi Karbon (kg CO₂)		
Kendaraan	Rute Awalan	Rute Optimal
Kendaraan 1	2,71	2,28
Kendaraan 2	2,76	3,75
Kendaraan 3	4,83	1,68
Kendaraan 4	4,46	1,92
Total	14,76	9,63



Gambar 5. 6 Perbandingan Emisi Karbon

5.4 Perbandingan Biaya Pendistribusian

Salah satu aspek penting dalam manajemen distribusi logistik adalah pengendalian biaya operasional, yang terdiri dari biaya bahan bakar dan biaya tetap kendaraan. Biaya operasional sangat dipengaruhi oleh efisiensi rute yang dilalui kendaraan distribusi, di mana semakin jauh jarak tempuh, maka semakin besar pula konsumsi bahan bakar dan waktu operasional yang dibutuhkan. Oleh karena itu, pengoptimalan rute menjadi langkah strategis untuk menekan pengeluaran dan meningkatkan kinerja logistik secara keseluruhan. Berikut data perhitungan biaya pendistribusian rute awalan dan rute optimal dari algoritma PSO.

Tabel 5. 7 Biaya Rute Awalan

Rute Awalan				
Kendaraan	Jarak Tempuh (km)	BBM/km	Biaya Tetap	Total Biaya (Rp)
1	436,28	396.622	510.667	907.289
2	391,93	356.303	510.667	866.970
3	587,15	533.771	510.667	1.044.438
4	567,49	515.897	510.667	1.026.563
Total	1982,85	1.802.593		3.845.260

Pada rute awalan, total biaya operasional yang dikeluarkan untuk empat kendaraan mencapai Rp3.845.260. Biaya tersebut terdiri dari Rp1.802.593 untuk bahan bakar dan Rp2.042.667 untuk biaya tetap seluruh kendaraan dengan masing-masing kendaraan dikenakan biaya tetap Rp510.667. Kendaraan 3 mencatat biaya tertinggi sebesar Rp1.044.438, disusul oleh kendaraan 4 sebesar Rp1.026.563. Biaya yang tinggi pada kendaraan 3 dan 4 disebabkan oleh panjangnya jarak tempuh masing-masing, yaitu 587,15 km dan 567,49 km. Hal ini menunjukkan bahwa pada rute awalan, beban distribusi belum merata, dan terdapat ketidakseimbangan dalam pembagian wilayah pengantaran antar kendaraan. Biaya variabel pada sistem ini

dihitung berdasarkan tarif operasional sebesar Rp909 per kilometer, yang meliputi konsumsi BBM, penyusutan kendaraan, dan beban operasional lainnya.

Tabel 5. 8 Biaya Rute Optimasi

Rute Optimal				
Kendaraan	Jarak Tempuh (km)	BBM/km	Biaya Tetap	Total Biaya (Rp)
1	408,56	371.422	510.667	882.088
2	399,24	362.948	510.667	873.615
3	296,47	269.514	510.667	780.180
4	255,62	232.381	510.667	743.048
Total	1359,89	1.236.265		3.278.931

Rute optimal merupakan hasil dari penerapan algoritma PSO yang bertujuan untuk meminimalkan total biaya distribusi melalui pengaturan ulang urutan dan alokasi pelanggan ke masing-masing kendaraan. Setelah dilakukan optimasi rute terjadi penurunan total biaya operasional menjadi Rp3.278.931. Komponen biaya bahan bakar turun signifikan menjadi Rp1.236.265, sementara biaya tetap kendaraan tetap konstan. Kendaraan 3 dan 4 mengalami penurunan biaya operasional yang paling besar, masing-masing menjadi Rp780.180 dan Rp743.048, seiring dengan penurunan jarak tempuhnya secara drastis. Penyesuaian rute ini memperlihatkan bahwa algoritma PSO mampu mengatur distribusi rute lebih seimbang antar kendaraan, sehingga mengurangi konsumsi bahan bakar yang tinggi pada kendaraan yang sebelumnya menanggung rute terjauh.

Perbandingan antara rute awalan dan rute hasil optimasi menunjukkan adanya efisiensi biaya operasional sebesar Rp566.328 atau sekitar 14,73% dari total biaya awal. Penghematan ini terutama berasal dari penurunan konsumsi bahan bakar, yang merupakan komponen biaya variabel terbesar dalam operasi distribusi. Hasil ini membuktikan bahwa penggunaan algoritma PSO tidak hanya mengoptimalkan rute

secara spasial, tetapi juga memberikan dampak positif dalam pengurangan beban biaya logistik.

Analisis terhadap hasil optimasi menunjukkan bahwa rute usulan yang dihasilkan oleh algoritma PSO memberikan perbaikan signifikan dibandingkan dengan rute awal, baik dari segi jarak tempuh, biaya bahan bakar, total biaya distribusi, jumlah titik kunjungan, maupun kesesuaian terhadap *time windows*. Total jarak tempuh pada rute usulan mencapai 1359,89 km, lebih rendah sebesar 622,96 km dibandingkan dengan rute awal yang mencapai 1982,85km. Penurunan jarak ini berdampak pada pengurangan konsumsi bahan bakar dari Rp1.802.593 pada rute awal menjadi Rp1.236.265 pada rute usulan, sehingga terjadi penghematan biaya bahan bakar sebesar Rp566.328. Total biaya distribusi yang mencakup biaya tetap dan bahan bakar juga menurun dari Rp3.845.260 pada rute awal menjadi Rp3.278.931 pada rute usulan, menunjukkan efisiensi biaya sekitar 14,73%. Selain itu, jumlah node yang dilayani pada kendaraan 2 dan 4 pada rute usulan lebih seimbang dan lebih sedikit dibandingkan rute awal, sehingga distribusi beban antar kendaraan menjadi lebih merata. Sebagai contoh, kendaraan 2 pada rute awal melayani 32 titik sedangkan pada rute usulan menjadi 28 titik, dengan urutan kunjungan yang lebih sistematis dan jarak lebih pendek. Hal serupa juga terlihat pada kendaraan 4, di mana jumlah node berkurang dari 26 titik pada rute awal menjadi 29 titik pada usulan dengan jarak yang lebih efisien. Dari sisi *time windows*, kendaraan 2 dan 4 berhasil kembali tepat waktu ke *main hub* sesuai batasan operasional, sedangkan kendaraan 1 dan 3 masih mengalami keterlambatan. Temuan ini menunjukkan bahwa PSO mampu mengoptimalkan jarak, biaya, dan emisi secara signifikan, namun penyesuaian lebih lanjut pada alokasi pelanggan per kendaraan masih diperlukan untuk meningkatkan ketepatan waktu seluruh kendaraan.

Hasil Penelitian ini dengan penelitian (Ramadhani & Garside, 2021) sama-sama menunjukkan bahwa penerapan algoritma *Particle Swarm Optimization* (PSO) efektif untuk mengoptimalkan rute distribusi dengan mempertimbangkan aspek lingkungan. Penelitian ini menerapkan PSO pada permasalahan *Vehicle Routing Problem with Time Windows* (VRPTW) di PT Pos Logistik Indonesia, dengan

mempertimbangkan jarak tempuh, batasan waktu pengiriman, kapasitas kendaraan, dan emisi karbon. Hasilnya, emisi karbon berhasil ditekan dari 14,76 kg CO₂ menjadi 9,63 kg CO₂, atau terjadi penghematan sekitar 34,76%. Sementara itu, penelitian Ramadhani & Garside (2021) juga menggunakan PSO untuk meminimalkan konsumsi bahan bakar sebagai proksi emisi karbon, pada permasalahan VRP dengan pembatasan kapasitas dan waktu tempuh maksimum (CVRP + T_{max}) untuk distribusi air minum kemasan di Lombok. Hasilnya menunjukkan konsumsi bahan bakar turun dari 50,52 liter menjadi 44,04 liter, atau penghematan sebesar 13%. Hasil analisis dari penelitian ini terjadi penghematan emisi yang lebih besar, serta secara eksplisit mengintegrasikan batasan *time windows* selain kapasitas dan waktu tempuh. Perbedaan konteks kasus, jumlah pelanggan, dan indikator lingkungan yang digunakan turut mempengaruhi besar kecilnya penghematan yang diperoleh dalam masing-masing studi. Akan tetapi, hasil analisis masih berbasis data permintaan dan kondisi operasional yang bersifat estimatif, serta belum mempertimbangkan faktor dinamis seperti lalu lintas, cuaca, dan fluktuasi permintaan yang dapat mempengaruhi realisasi jarak tempuh, biaya, maupun ketepatan waktu di lapangan. Oleh karena itu, hasil ini masih memerlukan validasi lebih lanjut dengan data empiris untuk memastikan keandalannya dalam implementasi nyata.

BAB VI

PENUTUP

6.1 Kesimpulan

Penelitian ini bertujuan untuk mengoptimalkan biaya operasional dan emisi karbon dalam permasalahan *Vehicle Routing Problem with Time windows* (VRPTW) pada PT Pos Logistik Indonesia menggunakan algoritma *Particle Swarm Optimization* (PSO). Berdasarkan hasil pengolahan data, analisis, dan evaluasi performa solusi, maka diperoleh kesimpulan sebagai berikut:

1. *Vehicle Routing Problem with Time Windows* (VRPTW) merupakan permasalahan logistik yang mempertimbangkan batasan waktu pengantaran ke setiap pelanggan, serta batas kapasitas kendaraan dalam proses distribusi. Untuk mengatasi ketidaksesuaian antara jarak tempuh kendaraan dan waktu pengiriman di PT Pos Logistik Indonesia, algoritma *Particle Swarm Optimization* (PSO) diterapkan sebagai pendekatan optimasi. PSO berhasil menghasilkan rute distribusi yang lebih efisien dibandingkan metode konvensional, dengan total jarak tempuh yang semula 1982,85 km berhasil diminimalkan menjadi 1359,89 km atau berkurang sekitar 31,42%. PSO juga memastikan seluruh 110 pelanggan terlayani tepat waktu dalam rentang time windows yang sudah ditentukan, dan kendaraan tidak melebihi kapasitas maksimum 1.000 kg. Setiap kendaraan melayani hingga maksimal 30 titik distribusi, dengan pola rute yang dimulai dan diakhiri di main hub Jakarta. Hal ini menunjukkan PSO mampu menghasilkan rute yang optimal dengan mempertahankan ketepatan waktu pengiriman.
2. Emisi karbon (CO₂) dari kendaraan distribusi merupakan indikator penting dalam upaya logistik berkelanjutan, terutama di wilayah kota seperti Jabodetabek yang memiliki intensitas pengiriman tinggi. Dalam penelitian ini, dampak peningkatan Demand dan jarak distribusi terhadap besaran emisi karbon dianalisis secara kuantitatif. Emisi dihitung berdasarkan jarak antar titik dan sisa beban kendaraan menggunakan rumus emisi per kg per km. Hasilnya menunjukkan bahwa total

emisi karbon pada rute awal adalah 14,76 kg CO₂, sedangkan setelah optimasi dengan PSO, emisi turun menjadi 9,63 kg CO₂, mengalami penurunan sebesar 5,13 kg CO₂ atau sekitar 34,76%. Penurunan ini dicapai karena algoritma PSO mampu mengatur urutan pengiriman dengan menempatkan titik-titik dengan Demand besar pada posisi awal, sehingga kendaraan mengangkut beban berat untuk jarak yang lebih pendek. Strategi ini tidak hanya menghemat energi, tetapi juga mendukung target keberlanjutan perusahaan dan nasional dalam penurunan emisi gas rumah kaca hingga 43,20% pada tahun 2030.

3. *Particle Swarm Optimization* (PSO) sebagai metode optimasi metaheuristik terbukti efektif dalam menyelesaikan VRPTW secara multi-objektif, yaitu meminimalkan biaya, jarak, dan emisi secara bersamaan. PSO bekerja dengan cara mengadaptasi posisi dan kecepatan partikel berdasarkan pengalaman terbaiknya sendiri maupun kelompok, sehingga sesuai untuk permasalahan kompleks logistik seperti di PT Pos Logistik Indonesia. Dalam penelitian ini, PSO mampu menurunkan total biaya operasional distribusi dari Rp 3.845.260 menjadi Rp 2.278.931, atau terjadi penghematan biaya sebesar Rp566.328 atau sekitar 14,73%. Waktu pemrosesan algoritma juga efisien, yakni sekitar 3–5 menit untuk 1000 iterasi, dengan hasil rute yang memenuhi semua batasan. Hasil ini menunjukkan bahwa PSO merupakan solusi strategis untuk mengoptimalkan distribusi barang secara efisien, hemat biaya, dan ramah lingkungan.

6.2 Saran

Berikut merupakan saran dari penulis untuk penelitian selanjutnya dalam implementasi *Particle Swarm Optimization* pada *Vehicle Routing Problem with Time windows*.

1. Penelitian selanjutnya dapat menggunakan data *demand* aktual harian secara *real-time* yang terhubung langsung dengan sistem manajemen logistik perusahaan. Hal ini akan meningkatkan akurasi model dalam menangani fluktuasi permintaan dan menjadikan optimasi rute lebih adaptif terhadap kondisi operasional yang dinamis.

2. Pengembangan selanjutnya dapat mempertimbangkan penggunaan berbagai jenis kendaraan seperti motor, *pick-up*, atau truk dengan kapasitas dan kecepatan berbeda. Maka model akan lebih fleksibel dan dapat merepresentasikan kenyataan operasional yang lebih kompleks, seperti kombinasi pengiriman jarak jauh dan pendek dalam satu sistem rute.
3. Penyempurnaan model sebaiknya memasukkan faktor eksternal seperti kondisi lalu lintas, waktu tempuh aktual antar titik, serta potensi *delay* pada jam sibuk. Dengan demikian, solusi rute yang dihasilkan tidak hanya optimal secara matematis, tetapi juga realistis untuk diimplementasikan di lapangan.
4. Untuk memperluas wawasan dan validasi hasil, penelitian lanjutan disarankan untuk melakukan perbandingan dengan algoritma lain seperti *Genetic Algorithm* (GA), *Ant Colony Optimization* (ACO), atau *Hybrid PSO*. Hal ini berguna untuk mengetahui algoritma mana yang paling unggul dalam konteks VRPTW di PT Pos Logistik Indonesia.
5. hasil optimasi PSO dapat dikembangkan ke dalam bentuk aplikasi interaktif yang dapat digunakan oleh operator logistik. Sistem ini bisa dilengkapi fitur *input* data otomatis, pemantauan rute, serta simulasi perubahan skenario distribusi secara cepat.

DAFTAR PUSTAKA

- Ai, T. J. (2009). A Particle Swarm Optimisation for Vehicle Routing Problem with Time Windows . *Int. J. Operational Research*, 519-537.
- Aimoruk, S. P., Rumbino, Y., & Taopan, H. V. (2024). Analisis Tingkat Produksi Terhadap Biaya Penambangan Dan Penentuan Harga Jual Sirtu Di Cv.Fiorretti, Kelurahan Tenukiik, Kecamatan Kota Atambua, Kabupaten Belu. *Jurnal Teknologi*, 47-54.
- Alacsel, S. (2024). Penerapan Sistem Informasi Manajemen pada PT. Prima Indonesia Logistik. *AKADEMIK: Jurnal Mahasiswa Humanis*, 122-131.
- Alaia, E. B., Harbaoui, I., Borne, P., & Bouchriha, H. (2018, February). A Comparative Study of the PSO and GA for the m-MDPDPTW. *International Journal of Computers Communications & Control*, 13(1), 8-23.
- Amini, S. (2011). A Novel PSO For Solving The VRPTW With Real Case Study. *International Conference on Industrial Engineering and Operations Management*, 562-567.
- Amini, S., Javanshir, H., & Tavakkoli-Moghaddam. (2010). A PSO Approach For Solving VRPTW With Real Case Study. *International Journal of Recent Research and Applied Studies (IJRRAS)*, 339-347.
- Anadi, Y. R. (2021). Perlindungan Hukum Bagi Pengemudi Transportasi Akibat Pembatalan Sepihak Oleh Konsumen. *Jatiswara*, 36(1), 115-125.
- Andriyani, & Resi, N. (2019). *Penentuan Rute Pendistribusian Produk Cat Menggunakan Capacitated Vehicle Routing Problem (CVRP) dengan Mempertimbangkan Kapasitas Alat Angkut Berdasarkan Dimensi Kubikasi*. Bandung: Universitas Pasundan.
- Andriyansah, R., & Linarti, U. (2022). Optimization of Vehicle Routing Problem Using Particle Swarm Optimization: A Case Study Watering Plants in Yogyakarta City. *Jurnal Ilmiah Teknik Industri*, 311-320.
- Atiqoh, A. N. (2020). *Analisis Inertia Weight Pada Algoritma Particle Swarm Optimization (PSO) Untuk Optimalisasi Dan Pemodelan Sistem Terhadap Persoalan Vehicle Routing Problem With Time Window (VRPTW)*. Surabaya: UIN Sunan Ampel.
- DAOUDA, A. S., & ATİLA, Ü. (2024). A Hybrid Particle Swarm Optimization with Tabu Search for Optimizing Aid Distribution Route. *Artificial Intelligence Studies*, 10-27.
- Dridi, I. H., Alaia, E. B., & Bouchriha, H. (2019). Optimisation of the multi-depots pick-up and delivery problems with time windows and multi-vehicles using PSO algorithm. *International Journal of Production Research*, 58(14), 4201-4214.
- Fallah, M., & Tavakkoli-Moghaddam, R. (2019). A robust approach for a green periodic competitive VRP under uncertainty: DE and PSO algorithms. *Journal of Intelligent and Fuzzy Systems*, 5213-5225.

- Fan, H., Ren, X., Zhang, Y., Zhen, Z., & Fan, H. (2022). A Chaotic Genetic Algorithm with Variable Neighborhood Search for Solving Time-Dependent Green VRPTW with Fuzzy Demand. *symmetry*, 1-18.
- Fatimah, Z. M. (2016). *Penerapan Algoritma Particle Swarm Optimization untuk Vehicle Routing Problem pada Kasus Pendistribusian Barang*. Jember: Universitas Jember.
- Fatma, E., & Manurung, S. (2020). Optimasi Biaya Transportasi Komponen dengan Batasan Jendela Waktu Layanan Sempit dan Kapasitas Kendaraan Beragam. *Jurnal Rekayasa Sistem Dan Industri*, 31-35.
- Fei, W. (2014). Study on VRPTW Based on Improved Particle Swarm Optimization. *Telkomnika: Indonesian Journal of Electrical Engineering*, 4543-4549.
- Fei, W. (2014). Study on VRPTW Based on Improved Particle Swarm Optimization. *Telkomnika Indonesian Journal of Electrical Engineering*, 4543-4549.
- Ferdiansyah, A., Sholihah, S. A., Rifni, M., Grets, E. S., Situmorang, J. K., & Oktaviany, I. (2021). Analisis Perencanaan Rute Pengiriman Barang Menggunakan Metode Vehicle Routing Problem (VRP). *Jurnal Sistem Transportasi dan Logistik*, 1-6.
- Heitasari, D. N., & Ghifari, M. K. (2022). Perbandingan Metode Round Trip Time & Vehicle Routing Problem Time Windows Dalam Pemilihan Supply Point Pada Proses Distribusi Pertashop . *SNTEM*, 924-936.
- Hidayat, H., Muhendra, R., & Nugroho, O. W. (2023). Damage package analysis and product quality improvement in logistics services using the PDCA and FMEA methods (Case Study: PT Ninja Xpress). *Jurnal Terapan Teknik Industri*, 284-295.
- Hidayat, P., Winarno, & Wahyudin. (2024). Penentuan Rute Distribusi Feasible untuk Vehicle Routing Problem with Time Windows dengan Metode Sequential Insertion Pada CV. Hagia Mitra Mandiri. *Serambi Engineering*, 7853-7863.
- Ibrahim, M. F., Nurhakiki, F. R., Utama, D. M., & Rizaki, A. A. (2021). Optimised Genetic Algorithm Crossover and Mutation Stage for Vehicle Routing Problem Pick-Up and Delivery with Time Windows. *IOP Conference Series: Materials Science and Engineering*, 1-7.
- Ibrahim, M. F., Putri, M. M., Farista, D., & Utama, D. M. (2021). An Improved Genetic Algorithm for Vehicle Routing Problem Pick-up and Delivery with Time Windows . *Jurnal Teknik Industri*, 1-17.
- Immanuel, R. M. (2015). *Algoritma Particle Swarm Optimization Untuk Menyelesaikan Multi Depot Vehicle Routing Problem Dengan Variabel Travel Time*. Surabaya: Institut Teknologi Sepuluh Nopember.
- Indrianti, N., Leuveano, R. A., Abdul-Rashid, S. H., & Ridho, M. I. (2025). Green Vehicle Routing Problem Optimization for LPG Distribution: Genetic Algorithms for Complex Constraints and Emission Reduction. *Sustainability*, 1-25.
- Irawan, I., Qisthiano, M. R., Syahril, M., & Jakak, P. M. (2023). Optimasi Prediksi Kelulusan Tepat Waktu: Studi Perbandingan Algoritma Random Forest dan Algoritma K-NN Berbasis PSO. *Jurnal Pengembangan Sistem Informasi dan Informatika*, 26-36.

- Islam, M. A., Gajpal, Y., & ElMekkawy, T. Y. (2021). Hybrid particle swarm optimization algorithm for solving the clustered vehicle routing problem. *Applied Soft Computing* , 1-13.
- Isnaini, N., Rozaki, Z., Wulandari, R., & Sari, L. M. (2025). Pengurangan Emisi Karbon pada Industri Logistik dengan Penggunaan Teknologi Artificial Intelligence. *ISSN Prosiding*, 28-31.
- Jeka, F., Risnita, Jailani, M. S., & Asrulla. (2023). Kajian Literatur dalam Menyusun Referensi Kunci, State Of The Art, dan Keterbaharuan Penelitian (Novelty). *Jurnal Pendidikan Tambusai*, 26466-26474.
- Kasengkang, R. A., Nangoy, S., & Sumarauw, J. (2016). Analisis Logistik (Studi Kasus Pada PT. Remenia Satori Tepas-Kota Manado). *Jurnal Berkala Ilmiah Efisiensi*, 750-759.
- Kunnappadeelert, S., Johnson, J. V., & Phalinonkiat, P. (2022). Green last-mile route planning for efficient e-commerce distribution. *Engineering Management in Production and Services*, 1-12.
- Kuo, R. J., Luthfiansyah, M. F., Masrurroh, N. A., & Zulvia, F. E. (2023). Application of improved multi-objective particle swarm optimization algorithm to solve disruption for the two-stage vehicle routing problem with time windows. *Expert Systems With Applications* , 1-13.
- Kurniati, D., & Jailani, M. (2023). Kajian Literatur : Referensi Kunci, State Of Art, Keterbaharuan Penelitian (Novelty). *Jurnal Pendidikan, Sosial, & Humaniora*, 1-6.
- Lo, S.-C. (2022). A Particle Swarm Optimization Approach to Solve the Vehicle Routing Problem with Cross-Docking and Carbon Emissions Reduction in Logistics Management. *Logistics*, 1-15.
- Mageswari, M. N. (2024). Vehicle Routing Problem (VRP) Using Genetic Algorithm. *International Research Journal of Advanced Engineering and Science*, 1-3.
- Mahatmia, M. F., Hasanuddina, T., & Umara, F. (2022). Implementasi Metode Haversine Formula untuk Menentukan Jarak Terdekat pada Pengantaran Air Galon Depot Anantama Berbasis Android. *Buletin Sistem Informasi dan Teknologi Islam*, 69-78.
- Meitasari, N. A., & Adhi, A. (2025). Optimisasi Rute Pengiriman Produk Usaha Mikro Kecil dan Menengah Menggunakan Metode Metaheuristik Particle Swarm Optimization. *Jurnal Serambi Engineering*, 12684-12690.
- Mutiara, P. (2022). Pengaruh Pendapatan dan Biaya Operasional terhadap Laba Bersih. *Jurnal Manajemen dan Sains*, 244-249.
- Patmawati, H., & Nugroho, Y. A. (2022). Optimalisasi Rute Distribusi Matras Pada Penyelesaian Capacitated Vehicle Routing Problem Dengan Metode Algoritma Genetika. *Jurnal Cakrawala Ilmiah*, 2745-2756.
- Pos Logistics Indonesia. (2025). Retrieved from Pos Logistik: <https://poslogistics.co.id/artikel/>
- Prawira, H. A., & Santosa, B. (2021). Development of Particle Swarm Optimization and Simulated Annealing Algorithms to Solve Vehicle Routing Problems With Drones. *Productivity, Optimization, and Manufacturing System*, 1-12.

- Prawira, H. A., & Santosa, B. (2021). Pengembangan Algoritma Particle Swarm Optimization dan Simulated Annealing untuk Menyelesaikan Vehicle Routing Problem with Drone. *Productivity, Optimization, and Manufacturing System*, 1-12.
- Rahayu, D. A., & Djakman, C. D. (2023). Evaluasi Distribusi Manajemen Rantai Pasok Komoditas Bahan Baku Industri Semen (Studi Kasus Pada PT X). *Journal of Economics and Business UBS*, 2575-2595.
- Ramadhani, B. N., & Garside, A. K. (2021). Particle Swarm Optimization Algorithm to Solve Vehicle Routing Problem with Fuel Consumption Minimization. *Jurnal Optimasi Sistem Industri*, 1-10.
- Ramadhani, B. N., & Garside, A. K. (2021). Particle Swarm Optimization Algorithm to Solve Vehicle Routing Problem with Fuel Consumption Minimization . *Jurnal Optimasi Sistem Industri*, 1-10.
- Shimizu, Y., & Ikeda, M. (2010). A Parallel Hybrid Binary PSO for Capacitated Logistics Network Optimization. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 617-626.
- Shimizu, Yoshiaki, Ikeda, & Masashi. (2010). A Parallel Hybrid Binary PSO For Capacitated Logistics Network Optimization. *Journal of Advanced Mechanical Design Systems and Manufacturing*, 616-626.
- Sonna Kristina, R. S., & Wijaya, V. J. (2020). Pengembangan Algoritma Ant Colony System Pada Heterogeneous Vehicle Routing Problem with Soft Time Window. *Journal of Integrated System*, 85-102.
- Srijaroon, N., Sethanan, K., Jamrus, T., & Chien, C.-F. (2021). Green vehicle routing problem with mixed and simultaneous pickup and delivery, time windows and road types using self-adaptive learning particle swarm optimization. *Engineering and Applied Science Research*, 657-669.
- Sriwana, I. K., S, N. H., Suwandi, A., & Rasjedin, R. (2021). Pengukuran Kinerja Rantai Pasok Menggunakan Supply Chain Operations Reference (SCOR) Di UD. Ananda. *JISI: Jurnal Integrasi Sistem Industri*, 13-24.
- Sudiantini, Wulandari, D., Devianti, I., Firli, F., Muzakki, S., Alifiyah, G. N., & Ayu, D. (2023). Analisis Pengaruh Globalisasi Terhadap Strategi Logistik Pada Perusahaan Multinasional (PT KAI Logistik). *Neraca Manajemen, Akuntansi Ekonomi*, 1-10.
- Sun, G., & Li, T. (2025). Optimizing Logistics in Forestry Supply Chains: A Vehicle Routing Problem Based on Carbon Emission Reduction. *Forests*, 1-21.
- Wahyudi, M. F., Setyawidayat, S., & Fachrudin. (2021). Metode Particle Swarm Optimization Untuk Menentukan Daya Optimal Turbin Gas PLTGU Grati Berdasarkan Heat Rate. *JASEE*, 37-46.
- Wang, Y., Xin, C., Shuang, Z., Ying, Z., Ke, C., & Chunyu, X. (2024). Self-Competition Particle Swarm Optimization Algorithm for the Vehicle Routing Problem With Time Window. *IEEE Access*, 127470-127488.
- Wisittipanit, N., Baisukhan, A., & Srisuwannapa, C. (2021). Comparisons of VRP Optimization Algorithmic Methods for the Optimal Routing of Multiple Delivery

- Vehicles with Time Constraint . *International Journal of Engineering Science*, 131-140.
- Wu, Q., X. X., Song, H., Zeng, H., Xu, X., & Zhang, Y. (2023). A neighborhood comprehensive learning particle swarm optimization for the vehicle routing problem with time windows. *Swarm and Evolutionary Computation* , 1-16.
- Xu, S.-H., Liu, J.-P., Zhang, F.-H., Wang, L., & Sun, L.-J. (2015). A Combination of Genetic Algorithm and Particle Swarm Optimization for Vehicle Routing Problem with Time Windows. *Sensors 2015*, 21033-21053.
- Xu, S.-H., Liu, J.-P., Zhang, F.-H., Wang, L., & Sun, L.-J. (2015). A Combination of Genetic Algorithm and Particle Swarm Optimization for Vehicle Routing Problem with Time Windows. *Sensors*, 21034-21053.
- Zai, I., Yulianti, S. F., Aqmi, A. L., & Rahmah, A. F. (2022). Analisis Pengaruh Peningkatan Kinerja, Incoterms, Transportasi, Distribusi, Keterlibatan TPL dan Manajemen Risiko Terhadap Aktivitas Logistik. *Jurnal Sosial dan Teknologi (SOSTECH)*, 225-238.
- Zhang, W., Yang, D., Zhang, G., & Gen, M. (2020). Hybrid multiobjective evolutionary algorithm with fast sampling strategy-based global search and route sequence difference-based local search for VRPTW. *Expert System with Application*.
- Zubedi, F. F., Baruwadi, M. H., & Indriani, R. (2021). Optimalisasi Industri Rumah Tangga (Studi Pada Industri Kecil Menengah Kinamang Desa Helumo Kecamatan Suwawa). *Agrinesia*, 126-130.

LAMPIRAN

```

# VRPTW dengan 4 kendaraan menggunakan PSO dan emisi karbon dinamis
import numpy as np
import random
import copy
from sklearn.metrics import pairwise_distances

# PARAMETER DASAR
# -----
NUM_CUSTOMERS = 110
NUM_VEHICLES = 4
DEPOT = 0
VEHICLE_SPEED = 22 # km/h

TM_RANGES = {
    1: (480, 780),
    2: (781, 1020),
    3: (1021, 1260)
}

# Fungsi Utilitas
# -----
def format_time(minutes):
    jam = int(minutes // 60)
    menit = int(minutes % 60)
    return f"{jam:02d}:{menit:02d} WIB"

def hitung_emisi_per_point(jarak, sisa_beban_kg, emisi_per_km_per_ton=0.065):
    berat_ton = sisa_beban_kg / 1000
    return jarak * berat_ton * emisi_per_km_per_ton

```

```

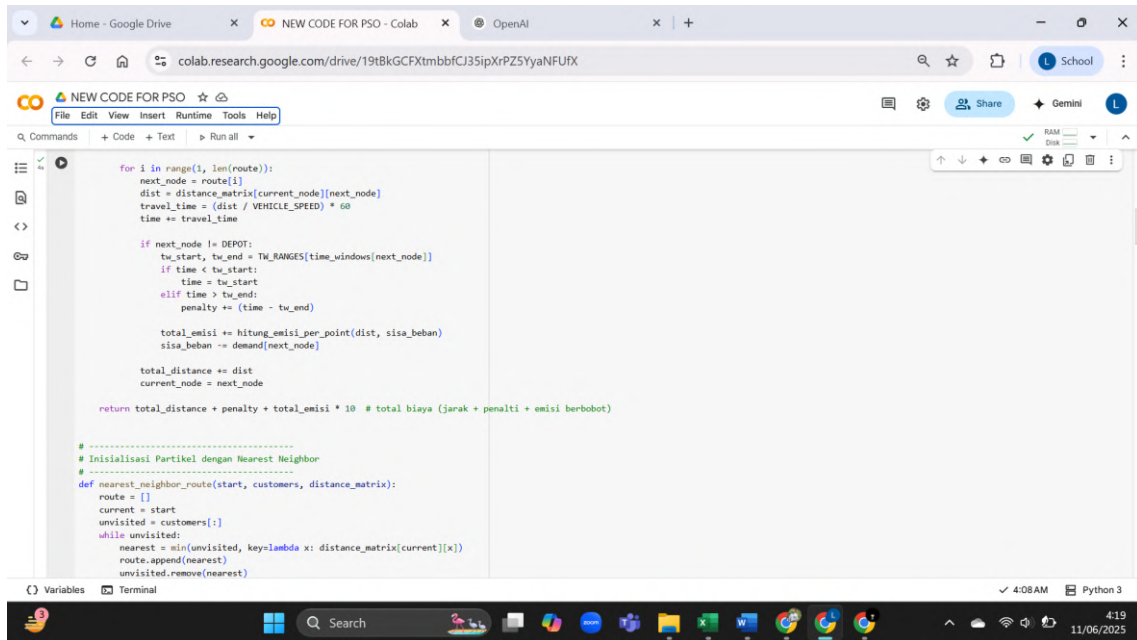
# Fungsi Evaluasi Solusi
# -----
def evaluate_solution(solution, distance_matrix, demand, time_windows, max_vehicle_capacity=None):
    total_distance = 0
    total_emisi = 0
    penalty = 0

    for route in solution:
        if not route:
            continue
        route = [DEPOT] + route + [DEPOT]
        time = 480
        current_node = DEPOT

        # Hitung total beban berdasarkan demand customer dalam rute
        vehicle_load = sum(demand[cust] for cust in route if cust != DEPOT)
        sisa_beban = vehicle_load

        # Cek jika muatan melebihi kapasitas maksimal (jika parameter diberikan)
        if max_vehicle_capacity is not None and vehicle_load > max_vehicle_capacity:
            penalty += (vehicle_load - max_vehicle_capacity) * 1000 # penalti overload

```



```

for i in range(1, len(route)):
    next_node = route[i]
    dist = distance_matrix[current_node][next_node]
    travel_time = (dist / VEHICLE_SPEED) * 60
    time += travel_time

    if next_node != DEPOT:
        tw_start, tw_end = TW_RANGES[time_windows[next_node]]
        if time < tw_start:
            time = tw_start
        elif time > tw_end:
            penalty += (time - tw_end)

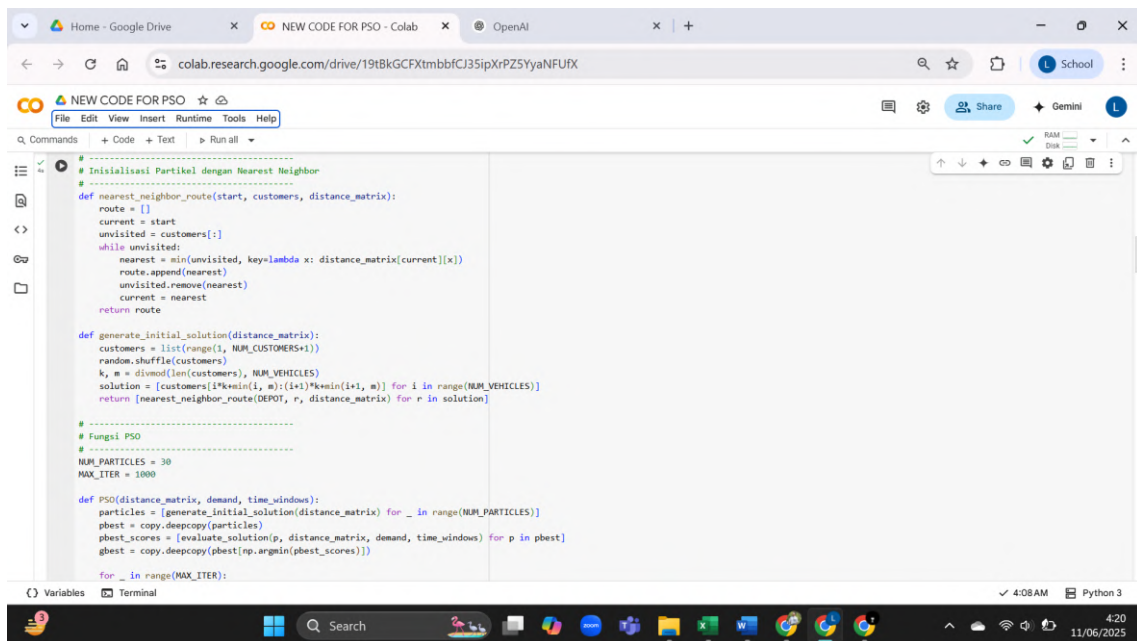
    total_emisi += hitung_emisi_per_point(dist, sisa_beban)
    sisa_beban -= demand[next_node]

    total_distance += dist
    current_node = next_node

return total_distance + penalty + total_emisi * 10 # total biaya (jarak + penalti + emisi berbot)

# -----
# Inisialisasi Partikel dengan Nearest Neighbor
# -----
def nearest_neighbor_route(start, customers, distance_matrix):
    route = []
    current = start
    unvisited = customers[:]
    while unvisited:
        nearest = min(unvisited, key=lambda x: distance_matrix[current][x])
        route.append(nearest)
        unvisited.remove(nearest)
    return route

```



```

# -----
# Inisialisasi Partikel dengan Nearest Neighbor
# -----
def nearest_neighbor_route(start, customers, distance_matrix):
    route = []
    current = start
    unvisited = customers[:]
    while unvisited:
        nearest = min(unvisited, key=lambda x: distance_matrix[current][x])
        route.append(nearest)
        unvisited.remove(nearest)
        current = nearest
    return route

def generate_initial_solution(distance_matrix):
    customers = list(range(1, NUM_CUSTOMERS+1))
    random.shuffle(customers)
    k, m = divmod(len(customers), NUM_VEHICLES)
    solution = [customers[i*k:min(i, m):i+1]*k+min(i+1, m)] for i in range(NUM_VEHICLES)
    return [nearest_neighbor_route(DEPOT, r, distance_matrix) for r in solution]

# -----
# Fungsi PSO
# -----
NUM_PARTICLES = 30
MAX_ITER = 1000

def PSO(distance_matrix, demand, time_windows):
    particles = [generate_initial_solution(distance_matrix) for _ in range(NUM_PARTICLES)]
    pbest = copy.deepcopy(particles)
    pbest_scores = [evaluate_solution(p, distance_matrix, demand, time_windows) for p in pbest]
    gbest = copy.deepcopy(pbest[np.argmax(pbest_scores)])

    for _ in range(MAX_ITER):

```

```

# Fungsi PSO
NUM_PARTICLES = 30
MAX_ITER = 1000

def PSO(distance_matrix, demand, time_windows):
    particles = [generate_initial_solution(distance_matrix) for _ in range(NUM_PARTICLES)]
    pbest = copy.deepcopy(particles)
    pbest_scores = [evaluate_solution(p, distance_matrix, demand, time_windows) for p in pbest]
    gbest = copy.deepcopy(pbest[np.argmin(pbest_scores)])

    for _ in range(MAX_ITER):
        for i in range(NUM_PARTICLES):
            new_particle = copy.deepcopy(particles[i])
            for route in new_particle:
                if len(route) > 1:
                    a, b = random.sample(range(len(route)), 2)
                    route[a], route[b] = route[b], route[a]

            if random.random() < 0.3:
                a, b = random.sample(range(NUM_VEHICLES), 2)
                if new_particle[a]:
                    move = new_particle[a].pop()
                    new_particle[b].append(move)

            score = evaluate_solution(new_particle, distance_matrix, demand, time_windows)
            if score < pbest_scores[i]:
                pbest[i] = new_particle
                pbest_scores[i] = score

        best_idx = np.argmin(pbest_scores)
        gbest = copy.deepcopy(pbest[best_idx])

    return gbest

# Cetak Solusi
def print_solution(solution, distance_matrix, demand, time_windows):
    for v, route in enumerate(solution):
        print(f"v{v}Blindan {v+1}")
        if not route:
            print(" Tidak ada pengantaran.")
            continue

        full_route = [DEPOT] + route + [DEPOT]
        total_dist = 0
        total_emisi = 0
        current_time = 480

        # Hitung total beban aktual dari demand pelanggan yang dikunjungi
        vehicle_load = sum(demand[cust] for cust in route)
        sisa_beban = vehicle_load

        print(f" Total load awal: {vehicle_load} kg")

        for i in range(len(full_route) - 1):
            current_node = full_route[i]
            next_node = full_route[i + 1]
            dist = distance_matrix[current_node][next_node]
            travel_time = (dist / VEHICLE_SPEED) * 60
            current_time += travel_time
  
```

```

        best_idx = np.argmin(pbest_scores)
        gbest = copy.deepcopy(pbest[best_idx])

    return gbest

# Cetak Solusi
def print_solution(solution, distance_matrix, demand, time_windows):
    for v, route in enumerate(solution):
        print(f"v{v}Blindan {v+1}")
        if not route:
            print(" Tidak ada pengantaran.")
            continue

        full_route = [DEPOT] + route + [DEPOT]
        total_dist = 0
        total_emisi = 0
        current_time = 480

        # Hitung total beban aktual dari demand pelanggan yang dikunjungi
        vehicle_load = sum(demand[cust] for cust in route)
        sisa_beban = vehicle_load

        print(f" Total load awal: {vehicle_load} kg")

        for i in range(len(full_route) - 1):
            current_node = full_route[i]
            next_node = full_route[i + 1]
            dist = distance_matrix[current_node][next_node]
            travel_time = (dist / VEHICLE_SPEED) * 60
            current_time += travel_time
  
```

[illegible]

The screenshot shows a Google Colab notebook interface. The top bar indicates the file is named 'NEW CODE FOR PSO - Colab'. The notebook code cell contains a large array of numerical values, which are the results of a PSO optimization process. The array is a single row with 100 elements, ranging from approximately 0.000000 to 0.999999. The interface includes the Google Drive top bar, the Colab notebook editor with a file explorer on the left, and a terminal at the bottom.

