Strategi Pengelolaan *Vehicle Routing Problem*s dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak
(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 : Baiq Putri Rizka Aulia
No. Mahasiswa : 21522129

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, 23 Juni 2025



(Baiq Putri Rizka Aulia)

21522129

SURAT BUKTI PENELITIAN



FAKULTAS
TEKNOLOGI INDUSTRI
Gedung 04, Ma Nanur
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. ft.uii.ac.id

SURAT KETERANGAN PENELITIAN

Nomor : 111/Ka.lab SIMANTI/20/Lab.SIMANTI/VI/2025

Assalamu'alaikum Warohmatullahi Wabarokaatuh

Dengan hormat,

Yang bertanda tangan dibawah ini, menerangkan bahwa:

Nama : Baiq Putri Rizka Aulia

Nim : 21522129

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 "**Strategi Pengelolaan *Vehicle Routing Problem*s dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak ".** Mulai pelaksanaan penelitian 03 Desember 2024 sampai 09 Juni 2025.

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

Wassalamu'alaikum Warohmatullahi Wabarokaatuh

Yogyakarta, 10 Juni 2025

Kepala Laboratorium

Sistem Manufaktur Terintegrasi

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

LEMBAR PENGESAHAN PEMBIMBING

Strategi Pengelolaan *Vehicle Routing Problem*s dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak (Studi Kasus: PT. Pos Logistik Indonesia)

TUGAS AKHIR

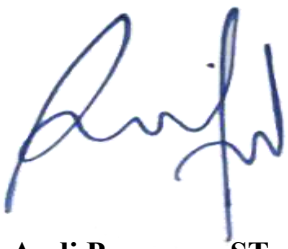
Disusun Oleh :

Nama : Baiq Putri Rizka Aulia

No. Mahasiswa : 21522129

Yogyakarta, 4 Juli 2025

Dosen Pembimbing



(Ir. M. Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM)

LEMBAR PENGESAHAN DOSEN PENGUJI

Strategi Pengelolaan *Vehicle Routing Problems* dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak (Studi Kasus: PT. Pos Logistik Indonesia)

TUGAS AKHIR

Disusun Oleh :

Nama : Baiq Putri Rizka Aulia

No. Mahasiswa : 21522129

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

Yogyakarta, 4 Juli 2025

Tim Penguji

Ir. Muhammad Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM	
Ketua	
Ir. Vembri Noor Helia, S.T., M.T., IPM	
Anggota I	
Nashrullah Setiawan, S.T., M.Sc., Ph.D.	
Anggota II	

Mengetahui,

Ketua Program Studi Teknik Industri Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

Ir. M. Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM

NIK. 01522010

HALAMAN PERSEMBAHAN

Dengan rasa Syukur ke hadirat Allah SWT, Tugas Akhir ini saya persembahkan untuk Ibu saya, Baiq Eka Yuliana, S.E, M.M., atas doa, kasih sayang, dan dukungan tanpa henti yang menjadi sumber semangat dalam menjalankan kuliah hingga menyelesaikan Sarjana. Terima kasih kepada diri saya sendiri, Baiq Putri Rizka Aulia, atas segala usaha, doa, dan perjuangan dalam menjalankan segala urusan di perkuliahan ini hingga selesai. Terima kasih juga kepada teman-teman yang selalu mendukung saya hingga Tugas Akhir ini selesai. Semoga Tugas Akhir ini menjadi pembuka untuk banyaknya hal baik yang akan datang kedepannya dalam hidup saya.

MOTTO

"Sesungguhnya bersama kesulitan ada kemudahan."

(QS. Al-Insyirah (94): 6)

"Dan barang siapa bersungguh-sungguh, maka sungguh usahanya itu untuk dirinya."

(QS. Al-Ankabut (29): 6)

"Dan janganlah kamu berputus asa dari rahmat Allah."

(QS. Az-Zumar (39): 53)

KATA PENGANTAR

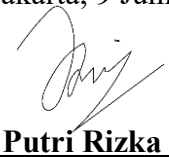
Puji syukur kehadirat Allah SWT atas segala limpahan rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Strategi Pengelolaan *Vehicle Routing Problem*s dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak".

Tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia. Penulis menyadari bahwa tersusunnya tugas akhir ini tidak lepas dari bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih dan penghargaan sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Hari Purnomo, MT. 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 Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Bapak, Ir. M. Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM. selaku Ketua Program Studi Teknik Industri Program Sarjana Universitas Islam Indonesia.
4. Bapak Ir. M. Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM. selaku dosen pembimbing yang telah memberikan bimbingan, arahan, dan motivasi selama proses penyusunan tugas akhir ini.
5. Bapak Wahyu Kurniawan, S.T., M.T., CISCP dan Ibu Putri Dwi Annisa, S.T., M.Sc yang telah memberikan bimbingan serta wadah untuk peneliti sehingga dapat melaksanakan penelitian tugas akhir di PT. Pos Logistik Indonesia dengan baik.
6. Ibu saya, Baiq Eka Yuliana, S.E, M.M. dan keluarga atas doa, kasih sayang, dan dukungan tanpa henti yang menjadi sumber semangat dalam menjalankan kuliah hingga menyelesaikan Sarjana.
7. Zaki Abdurrahman, yang telah memberikan semangat, menemani, membantu, dan sama-sama belajar dalam melaksanakan proses Tugas Akhir ini hingga selesai.
8. Teman-teman Asisten Laboratorium Sistem Manufaktur 2021, Zaki, Latifah, Bila, Iki, Lala, Zhafif, dan Dhinar yang telah sama-sama berjuang dan saling memberikan motivasi untuk tetap semangat dalam menyelesaikan Tugas Akhir ini.
9. Teman-teman dari semester 1, Tata, Shelma, Latifah, Ica, Aurel, Ara, Putyog, Ayusta, dan Indah yang selalu memberikan perhatian dan kasih sayang dalam proses penyusunan Tugas Akhir ini.
10. Bapak Heri Susilo serta Asisten Laboratorium Sistem Manufaktur 2020 dan 2022, yang memberikan semangat dan dukungan kepada penulis selama pengerjaan Tugas Akhir ini.

Penulis menyadari bahwa tugas akhir ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi penyempurnaan karya ini. Akhir kata, semoga Tugas Akhir ini dapat memberikan manfaat bagi pembaca serta menjadi kontribusi kecil dalam pengembangan ilmu pengetahuan, khususnya di bidang Teknik Industri.

Yogyakarta, 9 Juni 2025



Baiq Putri Rizka Aulia
21522129

ABSTRAK

Penelitian ini berjudul *"*Strategi Pengelolaan *Vehicle Routing Problem*s dengan Mempertimbangkan *Overflow Warehouse* Menggunakan *Exact method* Untuk Minimasi Biaya dan Jarak*"*. Latar belakang penelitian ini didasari oleh permasalahan kepadatan barang di *Main Hub* akibat tidak dimanfaatkannya *Hub* sebagai *buffer warehouse*, sehingga menyebabkan ketidakefisienan rute, dan meningkatnya biaya operasional. Permasalahan utama yang diangkat adalah bagaimana mengoptimalkan rute kendaraan untuk mengurangi kepenuhan muatan di *Depot*, menghitung waktu tempuh hasil optimasi, serta menganalisis dampaknya terhadap biaya operasional. Tujuan dari penelitian ini adalah menghasilkan rute distribusi optimal yang efisien secara biaya dan waktu melalui penerapan metode eksak. Metode penelitian ini memformulasikan permasalahan *Vehicle Routing Problem with Pickup and Delivery and Time Windows and Overflow Warehouse* (VRPPDTW-OW) dalam bentuk model *integer linear programming*. Model diimplementasikan menggunakan AMPL dan diselesaikan dengan *solver* Gurobi. Data primer diperoleh melalui wawancara pada proses distribusi aktual, sementara data sekunder didapat dari studi literatur. Analisis dilakukan dengan membandingkan rute awal dan rute hasil optimasi berdasarkan jarak tempuh, waktu tempuh, dan total biaya. Hasil penelitian menunjukkan bahwa metode eksak mampu mengoptimalkan rute distribusi dalam dua tahap yaitu pengiriman awal dari *Depot* ke *Hub* dan *customer* terdekat, serta pengiriman lanjutan dari *Hub* ke *customer* sekitar *Hub*. Strategi ini menghemat waktu distribusi sebesar 42,6% dan menurunkan total biaya operasional harian sebesar Rp197.186 atau 10,47%. Penelitian ini merekomendasikan pemanfaatan *buffer warehouse* sebagai strategi distribusi jangka panjang.

Kata Kunci: *Vehicle Routing Problem*, metode eksak, Optimasi, AMPL, Rute Distribusi

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 ..... **ix**
DAFTAR ISI ..... **x**
DAFTAR TABEL ..... **xii**
DAFTAR GAMBAR ..... **xiii**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.2 Rumusan Masalah 4
1.3 Tujuan Penelitian 4
1.4 Manfaat Penelitian 4
1.5 Batasan Penelitian 5
BAB II TINJAUAN PUSTAKA **8**
2.1 Kajian literatur 8
2.2 Landasan Teori 18
BAB III METODE PENELITIAN **30**
3.1 Objek Penelitian 30
3.2 Instrumen Penelitian 30
3.3 Sumber Data Primer dan Sekunder 31
3.4 Metode Pengumpulan Data 31
3.5 Populasi dan Sampel 32
3.6 Kerangka Konseptual 33
3.7 Alur Penelitian 34
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA **38**
4.1 Pengumpulan Data 38
4.1.1 Profil Perusahaan 38
4.1.2 Formulasi Masalah 39
4.1.3 Data *Main Hub*, *Hub*, dan *Customer* 40
4.1.4 Data *Time window* 41
4.1.5 Data Pendukung 43
4.1.6 Matriks Jarak 43
4.1.7 Rute Awalan Perusahaan 44
4.2 Pengolahan Data 45
4.2.1 Pengembangan Model Matematis 45
4.2.2 Penetapan Notasi dan Parameter Model Matematis 47
4.2.3 Proses Algoritma *Exact* 48
4.2.4 Hasil Optimasi 58

BAB V PEMBAHASAN 60
5.1 Analisis Hasil Optimasi Rute 60
5.1.1 Analisis Hasil Tahap 1 (*Depot*-*Hub*-cust-*Depot*) 61
5.1.2 Analisis Hasil Tahap 2 (*Hub*-cust-*Hub*) 63
5.1.3 Analisis Waktu Tempuh 68
5.2 Analisis Biaya 71
5.3 Perbandingan Rute Awalan dan Usulan 73
5.4 Kekurangan Penelitian 77
BAB VI PENUTUP 79
6.1 Kesimpulan 79
6.2 Saran 80
DAFTAR PUSTAKA 81
LAMPIRAN A-1

DAFTAR TABEL

Tabel 2. 1 Posisi Penelitian 16
Tabel 4. 1 Data *Main hub, Hub*, dan *Customer* 40
Tabel 4. 2 Data *Time Window* 42
Tabel 4. 3 Data *Time Window* setiap *Customer* 42
Tabel 4. 4 Data Pendukung 43
Tabel 4. 5 Matriks Jarak 44
Tabel 4. 6 Hasil Rute Awal Perusahaan 45
Tabel 4. 7 Proses Pemrograman *Python* Tahap 1 48
Tabel 4. 8 Proses Pemrograman *Python* Tahap 2 53
Tabel 4. 9 Hasil Optimasi Rute Tahap 1 59
Tabel 4. 10 Hasil Optimasi Rute Tahap 2 59
Tabel 5. 1 Hasil Rute Tahap 1 61
Tabel 5. 2 Rute per-*node* Tahap 1 62
Tabel 5. 3 Hasil Rute Tahap 2 63
Tabel 5. 4 Rute per-*node Hub* 27 64
Tabel 5. 5 Rute per-*node Hub* 28 65
Tabel 5. 6 Rute per-*node Hub* 29 67
Tabel 5. 7 Waktu tempuh per-*node* Tahap 1 69
Tabel 5. 8 Waktu tempuh per-*node Hub* 27 69
Tabel 5. 9 Waktu tempuh per-*node Hub* 28 69
Tabel 5. 10 Waktu tempuh per-*node Hub* 29 70
Tabel 5. 11 Total Waktu tempuh Usulan 71
Tabel 5. 12 Biaya Sewa Kendaraan 71
Tabel 5. 13 Biaya Tenaga Kerja 71
Tabel 5. 14 Biaya BBM 72
Tabel 5. 15 Total Biaya Rute Awalan PT. Pos Logistik Indonesia 72
Tabel 5. 16 Total Biaya Rute Rute Usulan Optimasi 73
Tabel 5. 17 Perbandingan Rute Awalan dan Usulan 74
Tabel 5. 18 Total Jarak, Waktu Tempuhm Biaya BBM, dan Total Biaya 75
Tabel 5. 19 Komparasi Awalan dan Usulan dengan *workload* pekerja 75

DAFTAR GAMBAR

Gambar 2. 1 Ilustrasi VRP 4 kendaraan 23
Gambar 2. 2 Variasi VRP 24
Gambar 2. 3 Diagram alir VRP dengan *Branch and Bound* 27
Gambar 3. 1 Kerangka Konseptual 33
Gambar 3. 2 Alur Penelitian 35
Gambar 4. 1 Logo Pos Logistik Indonesia 38
Gambar 4. 2 Budaya Perusahaan Pos Logistik Indonesia 39
Gambar 4. 3 Rute Awalan Perusahaan 45
Gambar 4. 4 Hasil Optimasi rute Tahap 1 58
Gambar 4. 5 Hasil Optimasi Rute Tahap 2 59
Gambar 5. 1 Rute Usulan Tahap 1 61
Gambar 5. 2 Rute Usulan Tahap 2 63
Gambar 5. 3 Rute kendaraan *Hub* 27 64
Gambar 5. 4 Rute kendaraan *Hub* 28 65
Gambar 5. 5 Rute kendaraan *Hub* 29 67
Gambar 5. 6 Perbandingan Visualisasi Rute Awalan dan Usulan 74
Gambar 5. 7 Grafik Perbandingan Jarak & Biaya Awalan Perusahaan & Usulan 75

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kepuasan *customer* salah satunya ditandai dengan distribusi yang baik agar bisa sampai ke *customer*. Banyaknya Perusahaan manufaktur yang bekerjasama dengan Perusahaan logistik untuk mendukung pengiriman agar cepat sampai ketangan *customer*. Pendistribusian produk atau barang bisa dilakukan sendiri oleh internal perusahaan dan bisa juga dilakukan dengan menggunakan jasa eksternal perusahaan atau dikenal dengan 3PL – *third party logistics* (Chandra & Setiawan, 2018). Secara umum dapat dikatakan bahwa logistik adalah aliran barang atau jasa mulai dari sumber sampai tujuan. Pengertian logistik yang lebih rinci adalah proses perencanaan, pelaksanaan, dan pengendalian aliran yang efisien dan efektif dari barang atau jasa dan informasi terkait mulai titik asal sampai dari titik penggunaan untuk memenuhi keperluan *customer* (Kasengkang et al., 2016). Logistik juga berperan untuk efisiensi operasional, Lembaga *Supply Chain Indonesia (SCI)*, memproyeksikan bahwa sektor transportasi dan pergudangan akan memberikan kontribusi signifikan terhadap Produk Domestik Bruto (PDB) Indonesia pada tahun 2025, yaitu sebesar Rp 1.623,65 triliun, atau tumbuh sebesar 12,53%. Menurut Badan Pusat Statistik, Rata-Rata Transaksi Perusahaan Ekspedisi Kurir Menurut Regional pada tahun 2023 yaitu sebesar 43,63% di daerah Jawa Bali. Oleh karena itu, logistik merupakan sektor strategis dalam pertumbuhan ekonomi nasional dan diperlukan rute yang optimal agar dapat meminimalisir dampak negatif yang dapat terjadi.

Vehicle Routing Problem (VRP) merupakan metode yang digunakan untuk merancang rute sejumlah kendaraan dengan biaya rendah dimana tiap kendaraan berawal dan berakhir di *Depot*, setiap *customer* hanya dikunjungi tepat sekali, serta total permintaan yang dibawa tidak melebihi kapasitas kendaraan (Rahmania et al., 2023). Fungsi utama VRP dalam logistik adalah untuk meningkatkan efisiensi pengiriman sehingga perusahaan dapat memberikan layanan yang lebih baik kepada *customer*. Selain itu, VRP juga membantu dalam pengambilan keputusan strategis terkait perencanaan rute yang optimal.

PT Pos Logistik Indonesia adalah anak perusahaan dari PT Pos Indonesia yang fokus pada penyediaan layanan logistik dan distribusi. PT Pos Logistik Indonesia menghadapi permasalahan bahwa seluruh barang masih terkonsentrasi di *Main Hub* tanpa memanfaatkan *Hub* cabang lain sebagai *buffer warehouse*, Kapasitas *Main hub* yang dimiliki yaitu 7600 ton, tetapi kondisi nyata yang dialami yaitu barang yang di tampung melebihi kapasitas *Main hub*

yaitu lebih dari 7600 ton barang yang masuk kedalam *Main hub*, sehingga terjadi penumpukan barang di *Depot/ Main hub*. Akibatnya rute kendaraan tidak optimal, proses pengiriman terhambat, dan waktu tunggu *customer* menjadi lebih panjang. *Time window* yang ketat juga dapat mempengaruhi biaya operasional karena kendaraan yang harus menunggu hingga *time window* terbuka atau yang memerlukan penambahan armada untuk memenuhi slot waktu sehingga meningkatkan konsumsi bahan bakar dan biaya tenaga kerja jika terjadi keterlambatan dalam pengiriman. Permasalahan ini berdampak secara langsung terhadap peningkatan biaya operasional serta mengurangi tingkat kepuasan pelanggan.

Untuk mengatasi permasalahan tersebut, PT Pos Logistik perlu segera menerapkan *overflow warehouse* dengan melibatkan *Hub-Hub* cabang sebagai titik distribusi tambahan. Dengan membagi sebagian barang ke *Hub*, penumpukan di *Main Hub* dapat diminimalkan, sehingga rute kendaraan menjadi lebih efisien dan konsumsi bahan bakar menurun tanpa perlu investasi besar untuk gudang baru. *Overflow warehouse* merupakan strategi distribusi yang digunakan untuk mengatasi keterbatasan kapasitas gudang utama dengan memanfaatkan titik penyimpanan tambahan sebagai buffer sementara. Dalam konteks PT Pos Logistik Indonesia, *overflow warehouse* diterapkan dengan menjadikan Hub-Hub cabang sebagai titik distribusi tambahan untuk menampung sebagian barang dari Main Hub yang mengalami kelebihan muatan. Langkah ini memungkinkan proses distribusi dibagi menjadi dua tahap, yakni dari Main Hub ke Hub, dan dilanjutkan dari Hub ke customer terdekat. Strategi ini terbukti efektif karena dapat mengurangi penumpukan barang di pusat distribusi utama, mengoptimalkan jarak tempuh kendaraan, serta menurunkan konsumsi bahan bakar dan biaya operasional harian.

Selain efisiensi biaya dan waktu, penggunaan *overflow warehouse* juga menghindarkan perusahaan dari investasi besar untuk pembangunan gudang baru, karena fasilitas Hub yang sudah ada dapat dioptimalkan. Dengan mendekatkan barang ke area layanan, pengiriman menjadi lebih cepat dan tepat waktu, terutama dalam memenuhi batasan waktu pelayanan atau time windows yang ketat. Strategi ini juga meningkatkan responsivitas dan kepuasan pelanggan karena pengiriman dapat dilakukan lebih fleksibel dan sesuai kebutuhan. Secara keseluruhan, penerapan *overflow warehouse* memberikan solusi logistik yang adaptif dan skalabel, serta memperkuat daya saing PT Pos Logistik Indonesia di tengah tantangan operasional yang semakin kompleks.

Metode eksak adalah pendekatan matematis yang dirancang untuk menemukan solusi optimal dengan memformulasikan seluruh komponen masalah seperti variabel, fungsi tujuan,

dan fungsi kendala ke dalam model pemrograman integer linear atau campuran integer. Keunggulan utama metode eksak yaitu pada keoptimalan hasilnya karena secara sistematis mengeksplorasi ruang solusi misalnya melalui teknik *branch and bound*, *cutting planes*, atau *branch and cut*, sehingga ketika proses komputasi selesai, kita yakin nilai solusi yang diperoleh benar-benar merupakan yang terbaik (Ropke, 2005). Selain itu, karena setiap variabel dan kendala dinyatakan dalam bentuk persamaan matematika, tidak ada risiko pelanggaran kendala baik terkait kapasitas kendaraan, *time window*s, maupun aturan *pickup and delivery* karena *solver* akan memeriksa setiap kombinasi rute yang memungkinkan. Metode eksak juga memudahkan analisis sensitivitas, misalnya mengevaluasi dampak perubahan kapasitas kendaraan atau pelonggaran jendela waktu terhadap total biaya (Baldacci et al., 2004). Berbeda dengan metode heuristik atau metaheuristik yang hanya menghasilkan solusi mendekati optimal tanpa bukti matematis.

Metode lain yang dapat menyelesaikan masalah VRP yaitu seperti *heuristic Nearest Neighbor* atau *metaheuristic* seperti *Simulated Annealing* dan *Tabu Search* telah banyak digunakan pada studi serupa. Meskipun metode tersebut tidak menjamin solusi optimal, namun mampu memberikan hasil yang sangat mendekati optimal dalam waktu komputasi yang jauh lebih cepat. Oleh karena itu, penggunaan metode *exact* dalam penelitian ini tidak hanya menunjukkan potensi solusi optimal, tetapi juga memungkinkan perbandingan yang signifikan terhadap efektivitas dan efisiensi dibandingkan pendekatan lain.

Permasalahan VRPPDTW-OW (*Vehicle Routing Problem with Pickup and Delivery with Time windows and Overflow Warehouse*) bersifat kombinatorial karena setiap penentuan urutan kunjungan pelanggan, alokasi kendaraan ke titik *hub*, pengaturan *pickup delivery*, hingga pemanfaatan *overflow warehouse* akan menghasilkan banyak kemungkinan rute (Baldacci et al., 2012). Dalam permasalahan ini, metode eksak sangat cocok karena dapat memformulasikan seluruh Batasan seperti kapasitas, *time windows*, *pickup delivery*, dan pemanfaatan *buffer warehouse* sebagai persamaan integer linear, lalu menggunakan solver untuk memeriksa kombinasi rute secara menyeluruh dan memangkas solusi yang tidak optimal. Oleh karena itu, PT Pos Logistik Indonesia dapat memperoleh rute kendaraan yang optimal secara matematis, mengurangi biaya bahan bakar, jarak, dan waktu tempuh.

1.2 Rumusan Masalah

Meskipun berbagai penelitian sebelumnya telah mengkaji penyelesaian *Vehicle Routing Problem (VRP)* dengan pendekatan heuristik, metaheuristik, maupun metode *exact*, masih terdapat gap dalam studi yang secara bersamaan mempertimbangkan *pickup delivery*, *time windows,* serta *overflow warehouse* dalam satu model terintegrasi. Sebagian besar studi terdahulu cenderung hanya memfokuskan pada satu atau dua aspek dari kompleksitas ini, seperti optimasi rute dengan batasan waktu tanpa memperhatikan kapasitas *buffer* gudang, atau hanya mempertimbangkan pengiriman tanpa aktivitas penjemputan. Padahal dalam praktik operasional, khususnya di perusahaan logistik seperti PT. Pos Logistik Indonesia, kondisi nyata mencerminkan kombinasi tantangan tersebut. Penelitian ini mencoba menjawab kekosongan tersebut dengan mengembangkan model VRPPDTW yang memasukkan variabel *overflow warehouse* dan menyelesaikannya menggunakan pendekatan *exact method*. Berdasarkan latar belakang tersebut, maka permasalahan utama yang hendak dijawab dalam penelitian ini adalah:

1. Bagaimana cara mengoptimalkan rute kendaraan untuk mengurangi kepenuhan muatan di *Depot* PT. Pos Logistik Indonesia?
2. Berapa waktu tempuh yang dihasilkan untuk melakukan distribusi setelah menggunakan metode *Exact*?
3. Bagaimana dampak optimalisasi rute dan waktu tempuh terhadap biaya operasional?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan penelitian ini adalah:

1. Menghasilkan optimalisasi rute terbaik serta membandingkan rute *existing* Perusahaan dan rute usulan
2. Mengetahui waktu tempuh yang dihasilkan dalam melakukan distribusi barang setelah menggunakan metode *Exact*
3. Mengetahui dampak optimalisasi rute dan waktu tempuh terhadap biaya operasional.

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Manfaat Akademis

   Menambah wawasan dan literatur untuk menjadi referensi penerapan metode optimasi rute kendaraan menggunakan metode VRP salah satunya metode *Exact*.
2. Manfaat Perusahaan

Memberikan solusi bagi PT. Pos Logistik Indonesia dalam meningkatkan efisiensi pengiriman melalui optimasi rute kendaraan dan Menawarkan strategi yang dapat membantu perusahaan dalam mengurangi biaya operasional.

3. Manfaat Teknologi

   Mendorong penerapan teknologi khususnya Perusahaan di bidang logistik dalam optimasi rute untuk meningkatkan efisiensi operasional.

1.5 Batasan Penelitian

Agar penelitian tetap terfokus dan lebih mudah dipahami, diperlukan pembatasan ruang lingkup permasalahan yang akan dibahas, yaitu:

1. Wilayah yang menjadi objek yaitu Pos Logistik dengan cakupan DKI Jakarta.
2. Terdapat satu *Depot* atau *Main Hub*.
3. Waktu layanan yang digunakan mulai pukul 08.00 WIB - 17.00 WIB dan tidak memperhitungkan waktu istiarahat.
4. Penelitian ini tidak mempertimbangkan kapasitas *buffer warehouse*.
5. Penelitian tidak mencangkup kondisi lalu lintas dan cuaca.
6. Kendaraan dianggap hanya 1 tipe saja yaitu *BlindVan*

Sistematika Penulisan

Dalam penelitian ini terdiri dari beberapa bab yang nantinya sebagai acuan supaya sistematis dalam penulisan penelitian ini, adapun sistematika dalam penulisan penelitian ini sebagai berikut:

BAB I PENDAHULUAN

Bab ini menguraikan latar belakang dilakukannya penelitian pada PT Pos Logistik Indonesia, yang menjadi dasar dalam merumuskan permasalahan utama yang ingin diselesaikan. Di dalamnya dijelaskan secara singkat mengenai urgensi dan pentingnya penelitian, serta dirumuskan tujuan penelitian, manfaat yang diharapkan, dan batasan-batasan penelitian agar fokus penelitian tetap terjaga dan terarah.

BAB II TINJAUAN PUSTAKA

Bab ini membahas kajian literatur dan teori-teori yang relevan sebagai landasan dalam pengembangan model dan pendekatan penelitian. Bagian ini mencakup teori mengenai *Vehicle Routing Problem with Pickup and Delivery with Time Windows (VRPPDTW)*, algoritma eksak, serta konsep logistik dan distribusi yang mendukung analisis.

BAB III METODE PENELITIAN

Bab ini menjelaskan metode penelitian yang digunakan untuk mencapai tujuan penelitian. Penjelasan mencakup objek penelitian, instrumen yang digunakan, sumber data baik primer maupun sekunder, teknik pengumpulan data, serta populasi dan sampel yang terlibat dalam penelitian. Alur atau tahapan pelaksanaan penelitian juga dijelaskan secara sistematis untuk menunjukkan proses yang dilakukan dari awal hingga akhir.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini menyajikan hasil pengumpulan data dari PT Pos Logistik Indonesia, termasuk profil perusahaan, data titik distribusi (Main Hub, Hub, dan Customer), data time window, serta data operasional lainnya seperti matriks jarak dan rute awal yang digunakan oleh perusahaan. Selanjutnya, data-data tersebut diolah melalui pengembangan model matematis, penetapan notasi dan parameter, hingga pelaksanaan proses optimasi menggunakan algoritma eksak.

BAB V PEMBAHASAN

Bab ini mengkaji hasil dari pengolahan data dan optimasi yang telah dilakukan. Pembahasan dilakukan dengan membedah rute hasil optimasi dalam dua tahap, menganalisis waktu tempuh, serta menghitung dan membandingkan biaya distribusi sebelum dan sesudah dilakukan optimasi. Selain itu, disajikan juga perbandingan antara rute awal perusahaan dan rute hasil usulan sebagai bentuk evaluasi dari penerapan model.

BAB VI PENUTUP

Bab terakhir berisi kesimpulan dari hasil penelitian yang telah dilakukan serta saran untuk pengembangan penelitian di masa mendatang. Kesimpulan ditarik berdasarkan analisis yang telah dilakukan, sedangkan saran diberikan untuk memperbaiki keterbatasan dan membuka peluang eksplorasi lebih lanjut di bidang serupa.

DAFTAR PUSTAKA

Memuat seluruh referensi yang digunakan dalam penelitian, baik berupa buku, jurnal, maupun sumber literatur lainnya yang relevan.

LAMPIRAN

Berisi dokumen pendukung penelitian seperti data mentah, hasil perhitungan, kode program, serta tabel atau grafik tambahan yang mendukung analisis.

BAB II
TINJAUAN PUSTAKA

2.1 Kajian literatur

Penelitian ini merujuk pada berbagai sumber dari studi terdahulu sebagai landasan teoritis dan konseptual. Sumber-sumber tersebut dianalisis dalam kajian induktif, yang merangkum penelitian sebelumnya dengan topik dan metode yang relevan. Kajian ini bertujuan untuk mengidentifikasi pola serta membandingkan hasil penelitian ini dengan studi terkait. Berikut adalah kajian induktif yang telah dikumpulkan dan dianalisis:

Penelitian yang dilakukan oleh (Sugiono, 2022) dengan judul "Model *Vehicle Routing Problem* untuk Penentuan Rute Distribusi Unit Sepeda Motor dengan Metode *Saving Matrix*". Penelitian ini bertujuan untuk menentukan rute distribusi yang efisien guna meminimalkan biaya transportasi dalam pendistribusian sepeda motor. Metode yang digunakan adalah metode *saving matrix*, yang telah terbukti efektif dalam mengoptimalkan rute distribusi. Hasil dari penelitian ini menunjukkan bahwa penggunaan metode *saving matrix* dapat menghasilkan penghematan biaya distribusi yang signifikan. Secara spesifik, penelitian ini berhasil menurunkan biaya distribusi harian dari Rp. 3.530.000,00 menjadi Rp. 3.323.500,00, yang berarti ada penghematan sebesar Rp. 206.500,00 per hari. Metode ini memungkinkan PT. X untuk merencanakan jalur distribusi yang lebih efisien, sehingga proses pendistribusian produk dapat berjalan optimal dengan biaya yang lebih rendah.

Penelitian yang dilakukan oleh (Panjaitan et al., 2024) dengan judul "*Vehicle Routing Problem* as *a Solution for Determining Goods Delivery Routes* PT. Kreasi Beton Nusa Persada". Penelitian ini bertujuan untuk menganalisis permasalahan jalur transportasi dalam pendistribusian produk dari lokasi awal pendistribusian kepada pengguna. Metode yang digunakan adalah *Nearest Neighbor* dan *Local Search* untuk menentukan dan memperbaiki distribusi rute pengiriman. Hasil penelitian menunjukkan bahwa penerapan Model *Vehicle Routing Problem* (VRP) dengan metode tetangga terdekat dan lokal dapat menghasilkan rute pengiriman yang lebih efektif dan efisien. Rute baru yang dihasilkan mampu mengurangi jarak tempuh sebesar 308.6 km, mempercepat waktu penyelesaian sebesar 5,705.11 menit, dan menghemat biaya bahan bakar kendaraan truk sebesar Rp. 635.819 dibandingkan dengan rute awal yang digunakan selama periode distribusi 1 bulan.

Penelitian yang dilakukan oleh (Muladi et al., 2020) dengan judul "Optimasi Rute Kapal untuk Distribusi *Spare parts* Menggunakan *Vehicle Routing Problem* dengan Algoritma *Tabu*

Search" bertujuan untuk mengoptimalkan rute distribusi kapal dalam mendistribusikan *spare parts* ke 11 kilang minyak milik PHE ONWJ yang tersebar di laut utara Pulau Jawa. Metode yang digunakan dalam penelitian ini adalah algoritma *Tabu Search* dan metode eksak. Hasil penelitian menunjukkan bahwa rata-rata total jarak rute yang dihasilkan oleh metode Tabu Search adalah 1008,4 km, sedangkan metode eksak menghasilkan rata-rata jarak 993,1 km. Meskipun metode *Tabu Search* tidak menghasilkan solusi optimal seperti metode eksak, hasilnya cukup baik dan mendekati biaya distribusi minimum. Hal ini menunjukkan bahwa Tabu Search dapat digunakan untuk menghasilkan rute distribusi yang efisien meskipun belum optimal.

Penelitian yang dilakukan oleh (Wulandari, 2020) dengan judul "Penentuan Rute Distribusi Menggunakan Metode *Nearest Neighbors* dan Metode *Branch and Bound* untuk Meminimumkan Biaya Distribusi di PT. X". Penelitian ini bertujuan untuk meminimumkan biaya pengiriman produk sepatu ke seluruh store di Indonesia dari gudang tunggal yang dimiliki PT. X dengan menggunakan metode *Nearest Neighbors* dan *Branch and Bound*. Hasil penelitian menunjukkan bahwa penggunaan metode *Nearest Neighbors* menghasilkan total jarak tempuh sebesar 6604.2 km dengan total biaya sebesar Rp 8,161,015.00. Selanjutnya, perbaikan dengan menggunakan metode *Branch and Bound* menghasilkan total jarak tempuh sebesar 6505.8 km dan total biaya sebesar Rp 8,076,555.00. Dari hasil tersebut, diharapkan PT. X dapat menerapkan metode *Branch and Bound* untuk menentukan jalur terpendek serta biaya paling minimum.

Penelitian yang dilakukan oleh (Arifian & Pulansari, 2023) dengan judul "*Determination of the Shortest Distribution Routes to Minimize Fuel Consumption and CO2 Emission Using Sweep and Clarke & Wright Saving Algorithms*". Penelitian ini bertujuan untuk menentukan rute distribusi yang paling pendek guna meminimalkan konsumsi bahan bakar dan emisi CO2. Metode yang digunakan dalam penelitian ini adalah algoritma *Sweep dan Clarke & Wright Saving*. Hasil dari penelitian ini menunjukkan bahwa penerapan algoritma tersebut dapat mengurangi konsumsi bahan bakar dan emisi CO2 secara signifikan. Algoritma ini membantu dalam menentukan rute distribusi yang lebih efisien dengan mempertimbangkan aspek lingkungan, sehingga dapat mengurangi dampak negatif dari emisi gas rumah kaca yang dihasilkan oleh kendaraan. Salah satu hasil spesifiknya adalah penurunan emisi CO2 yang dihasilkan dari penggunaan bahan bakar fosil dalam kendaraan, yang merupakan salah satu gas rumah kaca dominan yang berdampak negatif pada lingkungan.

Penelitian yang dilakukan oleh (Sumarsa & Widyastiti, 2022) dengan judul "Penerapan *Traveling Traveling* Problem *with Time windows* dalam Pendistribusian Produk". Penelitian ini bertujuan untuk menentukan rute optimal dalam pendistribusian produk sehingga diperoleh jarak terpendek. Metode yang digunakan adalah metode *Branch and Bound* dengan menggunakan model *Traveling Traveling* Problem *with Time windows*. Hasil penelitian menunjukkan bahwa rute pendistribusian yang diperoleh memiliki jarak sebesar 20,904 meter. Rute yang dihasilkan adalah *Depot* 1 - toko 10 - toko 18 - toko 7 - toko 2 - toko 11 - toko 22 - toko 27 - toko 23 - toko 31 - toko 24 - toko 29 - toko 12 - toko 28 - toko 16 - toko 8 - toko 25 - toko 26 - toko 6 - toko 5 - toko 15 - toko 14 - toko 3 - toko 32 - toko 4 - toko 9 - toko 30 - toko 17 - toko 21 - toko 20 - toko 13 - toko 19 - *Depot* 1. Dibandingkan dengan rute tanpa menggunakan metode *Branch and Bound*, rute ini lebih efisien dengan selisih jarak 3,950 meter atau 18.9% lebih pendek. Hal ini menunjukkan bahwa penggunaan metode *Branch and Bound* dapat mengurangi waktu tempuh dan meningkatkan efisiensi distribusi produk.

Penelitian yang dilakukan oleh (Praxedes et al., 2024) dengan judul "*A Unified Branch-Cut-and-Price Algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery*" bertujuan untuk mengembangkan algoritma yang dapat menyelesaikan sepuluh varian dari masalah *routing* kendaraan dengan pengambilan dan pengantaran simultan (VRPSPD). Metode yang digunakan adalah algoritma *branch cut and price* (BCP) yang mencakup fitur-fitur seperti algoritma pelabelan berbasis *bucket graph*, *rank-1* cuts dengan memori terbatas, dan *dynamic sets*. Penelitian ini berhasil melakukan eksperimen komputasi ekstensif pada lebih dari 550 *instance benchmark* dan algoritma yang diusulkan mampu memperoleh sejumlah solusi optimal baru serta meningkatkan batas bawah (*lower bounds*) untuk semua varian yang dibahas. Secara spesifik, penelitian ini memberikan batas bawah valid pertama untuk semua 14 *instance* dari dataset Avc1 yang terkait dengan HVRPSPD, dan 9 di antaranya terbukti optimal. Selain itu, batas atas (*upper bounds*) dari tiga *instance* berhasil ditingkatkan.

Penelitian yang dilakukan oleh (Hester Patmawati1, 2022) dengan judul "Optimalisasi Rute Distribusi Matras pada Penyelesaian *Capacitated Vehicle Routing Problem* dengan Metode Algoritma Genetika". Penelitian ini bertujuan untuk mengoptimalkan rute distribusi matras dengan menggunakan metode Algoritma Genetika. Hasil dari penelitian ini menunjukkan bahwa penggunaan Algoritma Genetika dapat mengurangi jarak tempuh distribusi secara signifikan. Proses pengembangan algoritma dimulai dengan pembentukan kromosom dan inisiasi populasi, yang kemudian dievaluasi untuk menemukan solusi optimal. Pengolahan data

menunjukkan perbandingan antara jarak tempuh awal dan jarak tempuh usulan, yang mengindikasikan peningkatan efisiensi distribusi.

Penelitian yang dilakukan oleh (Annisa et al., 2025) berjudul "Optimasi Rute Pengiriman UMKM Cripang Bu Fitri Menggunakan Metode *Metaheuristik Particle Swarm Optimization* (PSO)" bertujuan untuk mencari dan menentukan rute pengiriman tercepat agar *Traveling* dapat mendistribusikan barang secara lebih efektif dan efisien. Metode yang digunakan dalam penelitian ini adalah *metaheuristik Particle Swarm Optimization* (PSO). Hasil penelitian menunjukkan bahwa metode PSO efektif dalam mengoptimalkan rute pengiriman untuk UMKM Cripang Bu Fitri. Rute baru yang dihasilkan adalah 1 - 3 - 2 - 16 - 5 - 18 - 17 - 8 - 9 - 10 - 11 - 7 - 13 - 17 - 18 - 6 - 12 - 4 - 14 - 15 - 1, yang terbukti lebih efisien dibandingkan rute awal dengan peningkatan efisiensi sebesar 13%. Penggunaan PSO mengurangi jarak tempuh secara signifikan, sehingga dapat menghemat waktu pengiriman dan biaya operasional. PSO juga berhasil mengeksplorasi ruang pencarian secara efisien dan memberikan solusi yang lebih baik dibandingkan pendekatan intuitif, memberikan dampak positif bagi pemilik maupun *Traveling* UMKM Cripang Bu Fitri dengan meningkatkan daya saing di pasar.

Penelitian yang dilakukan (Rizkiani et al., 2023) dengan judul "*A Simulated Annealing for Heterogenous Fleet Vehicle Routing Problem with Multiple Trips and Pickup Delivery*". Penelitian ini bertujuan untuk menyelesaikan masalah rute kendaraan pada perusahaan distribusi LPG yang menghadapi tantangan dalam pengantaran dan pengumpulan tabung gas secara simultan dengan menggunakan jenis kendaraan yang berbeda-beda dan dapat melakukan perjalanan ganda. Metode yang digunakan dalam penelitian ini adalah *Simulated Annealing* (SA) dengan menggunakan *Savings Matrix* dan *Nearest Neighbor* untuk mendapatkan solusi awal, dengan tujuan meminimalkan jarak total. Hasil penelitian menunjukkan bahwa algoritma yang diusulkan memberikan solusi yang lebih baik dalam hal jarak total dibandingkan dengan solusi saat ini dari perusahaan. Secara spesifik, metode ini berhasil mengurangi jarak keseluruhan sebesar 31.22% dan waktu pengiriman kumulatif dalam seminggu berkurang dari 9135.12 menit menjadi 6597.14 menit. Analisis sensitivitas terhadap parameter SA juga dilakukan untuk menilai kinerja algoritma yang diusulkan.

Penelitian yang dilakukan oleh (Ferdiansyah et al., 2020)dengan judul "Analisis Perencanaan Rute Pengiriman Barang Menggunakan Metode *Vehicle Routing Problem* (VRP)" bertujuan untuk menentukan rute optimal pengiriman barang dari Distribution Center Samsung Elektronik ke toko ritel di wilayah Jabodetabek. Penelitian ini menggunakan metode VRP

model *Saving matrix* untuk mengoptimalkan rute pengiriman sehingga dapat mengurangi jarak tempuh, biaya bahan bakar, dan waktu distribusi. Hasil penelitian menunjukkan pengurangan jarak tempuh sebesar 19%, penghematan bahan bakar 20,1%, serta pengurangan waktu pengiriman sebesar 22,6% dibandingkan rute awal. Usulan perbaikan yang diberikan adalah penerapan metode VRP model *Saving matrix* dalam perencanaan rute pengiriman untuk meningkatkan efisiensi distribusi dan menekan biaya transportasi.

Penelitian yang dilakukan oleh (Dumez et al., 2021)dengan judul "*Hybridizing large neighborhood search and exact methods for generalized Vehicle Routing Problems with time windows*" bertujuan untuk mengembangkan metode *matheuristik* berbasis *Large Neighborhood Search* (LNS) yang mampu menangani berbagai varian General *Vehicle Routing Problem*s with *Time windows* (GVRPTW), termasuk varian yang kompleks seperti VRPDO. Penelitian ini memperkenalkan lapisan adaptif yang mengontrol ukuran operasi destroy serta penggunaan dua komponen eksak opsional, yaitu formulasi *set partitioning* (SPP) dan *neighborhood Balas-Simonetti* (BS), yang membantu meningkatkan solusi ketika proses LNS standar mengalami stagnasi. Evaluasi dilakukan dengan membandingkan empat konfigurasi matheuristik pada berbagai benchmark standar, menunjukkan bahwa kombinasi komponen SPP dan BS memberikan hasil solusi berkualitas tinggi dengan efisiensi waktu yang terkontrol.

Penelitian yang dilakukan oleh (Anggraeni et al., 2023) dengan judul " *Optimization of Fuel Distribution Routes for Green Logistics in Multi Compartment Vehicle Routing Problem (MCVRP) using Branch and Bound Algorithm* (Case Study: Boyolali *Fuel* Terminal)" bertujuan untuk mengoptimalkan proses distribusi bahan bakar bensin dan solar di terminal bahan bakar Boyolali yang melayani lebih dari 294 SPBU di Jawa Tengah dan Jawa Timur. Penelitian ini menggunakan metode *clustering P-Median* untuk mengelompokkan SPBU di Zona 3 menjadi 9 klaster dan mengaplikasikan model *Multi Compartment Vehicle Routing Problem* (MCVRP) dengan *split delivery* dan multi produk, serta algoritma *Branch and Bound* untuk meminimalkan total biaya distribusi yang mencakup biaya bahan bakar truk tangki dan biaya emisi karbon. Hasil penelitian menunjukkan bahwa optimasi ini berhasil menurunkan biaya distribusi harian sebesar 28%, dari IDR 12.712.435,20 menjadi IDR 9.179.116,80.

Penelitian yang dilakukan oleh(Zhao et al., 2024) dengan judul "Optimisasi Jaringan Pengiriman Barang di *Urban Express Delivery* Menggunakan Model RVRP dan Algoritma *Branch and Price*". Penelitian ini bertujuan untuk mengoptimalkan jaringan pengiriman barang di layanan pengiriman ekspres perkotaan dengan mempertimbangkan berbagai kendala

operasional seperti *pickup* dan *delivery*, waktu tertentu, *multiple trips*, serta *fleet heterogen*. Penelitian ini memperkenalkan formulasi berbasis trip dan algoritma *branch and price* yang mampu menyelesaikan masalah VRP kompleks secara efisien. Salah satu inovasi yang diusulkan adalah pengenalan diskon *backhaul* yang bertujuan untuk meningkatkan efisiensi biaya pengiriman *round trip*. Selain itu, penelitian ini juga melakukan validasi menggunakan data nyata dari SF *Express* dan menunjukkan bahwa algoritma yang dikembangkan mampu menyelesaikan sebagian besar permasalahan secara optimal dalam waktu kurang dari lima menit, serta memberikan wawasan terkait pengelolaan operasional yang lebih baik.

Penelitian yang dilakukan oleh (James & Nugroho, 2024) dengan judul "Penyelesaian *Vehicles Routing Problem* dalam Meminimumkan Waktu Transportasi PT Petrogas Prima *Services*". Penelitian ini membahas model dan metode dalam perencanaan distribusi dan pengiriman barang, khususnya melalui model transportasi dan *Vehicle Routing Problem* (VRP). Tujuan utama penelitian ini adalah untuk mengoptimalkan pengiriman agar biaya dan jarak tempuh dapat diminimalkan sesuai kendala kapasitas dan permintaan. Salah satu metode yang digunakan adalah algoritma *heuristik* seperti *nearest insert* dan *nearest neighbor* untuk menentukan rute optimal. Hasil penelitian menunjukkan adanya tiga rute utama dengan jarak masing-masing 32,70 km, 26,50 km, dan 54,50 km, yang dihasilkan dari pengelompokan rute berdasarkan metode *saving matrix*. Disarankan agar perusahaan memperhatikan efisiensi kendaraan dan meningkatkan pengelolaan transportasi guna mencapai pengiriman yang lebih efisien.

Penelitian ini diusulkan untuk mengisi gap yang masih jarang dibahas secara bersamaan dalam literatur, yaitu integrasi antara *Vehicle Routing Problem with Pickup and Delivery and Time Windows (VRPPDTW)* dengan strategi pemanfaatan *Overflow Warehouse* menggunakan pendekatan *exact*. Fokus utama penelitian ini adalah memberikan solusi distribusi logistik yang lebih efisien, tidak hanya dari sisi waktu dan jarak, tetapi juga pengurangan beban muatan yang selama ini menumpuk di *Main Hub* akibat tidak dimanfaatkannya *Hub-Hub* cabang sebagai gudang penyangga (*buffer warehouse*). Objek penelitian ini adalah PT. Pos Logistik Indonesia, perusahaan logistik nasional dengan cakupan operasional luas, yang menghadapi masalah nyata berupa ketidakefisienan distribusi, penumpukan barang, keterlambatan pengiriman akibat batasan waktu (*time windows*), dan meningkatnya biaya operasional. Penelitian ini menyelesaikan permasalahan tersebut melalui pengembangan model matematis yang memformulasikan seluruh aspek penting dalam jaringan distribusi, seperti alokasi kendaraan,

urutan kunjungan pelanggan, kapasitas kendaraan, batasan waktu pelayanan (TW), serta relokasi barang dari *Main Hub ke Hub* (*overflow warehouse*). Seluruh variabel dan kendala dimodelkan dalam bentuk *Mixed Integer Linear Programming (MILP)* dan diselesaikan menggunakan *solver* Gurobi melalui bahasa pemodelan AMPL. Pemilihan metode *exact* dilakukan pada kebutuhan akan solusi optimal dan validasi matematis yang kuat, terutama karena keputusan distribusi logistik memiliki dampak finansial dan operasional yang signifikan. Tidak seperti metode heuristik atau metaheuristik yang bersifat tidak pasti, pendekatan *exact* memastikan bahwa solusi yang diperoleh merupakan *feasible* secara teoritis di antara seluruh kombinasi rute yang mungkin.

Kekurangan dan Kelebihan Penelitian terhadap penelitian serupa:

Penelitian yang dilakukan oleh Praxedes et al. (2024) merupakan salah satu Penelitian yang paling mendekati pendekatan penelitian ini karena sama-sama membahas *Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPSPD)* dan menerapkan metode *exact*, yaitu algoritma *Branch Cut and Price* (BCP). Keunggulan dari penelitian tersebut terletak pada algoritma *exact* yang digunakan, yang mampu menghasilkan solusi dengan kompleksitas tinggi. Penelitian ini mencakup teknik seperti *bucket graph labeling*, *rank-1 cuts*, dan *dynamic sets*, yang meningkatkan efisiensi pencarian solusi. Namun demikian, penelitian tersebut belum memasukkan variabel *time windows* dan tidak mempertimbangkan keberadaan *overflow warehouse* sebagai bagian dari skema distribusi. Berbeda dengan penelitian ini yang tidak hanya menyelesaikan masalah VRPSPD, tetapi juga memperluas model dengan mengintegrasikan kendala waktu pelayanan (*time windows*) dan *buffer* lokasi (*overflow warehouse*) sebagai strategi untuk meningkatkan efisiensi distribusi. Oleh karena itu, penelitian ini mendekati kondisi operasional dunia nyata, khususnya pada industri logistik berskala besar seperti PT Pos Logistik Indonesia.

Sementara itu, penelitian oleh Rizkiani et al. (2023) membahas permasalahan *Heterogeneous Fleet Vehicle Routing Problem with Multiple Trips and Pickup Delivery*, yang memiliki fokus pada penggunaan beragam jenis armada (heterogen) dan perjalanan ganda (*multiple trips*) dalam satu hari operasional. Kelebihan utama penelitian ini terletak pada fleksibilitas armadanya dan pemanfaatan metode *Simulated Annealing* untuk mencari solusi efisien. Selain itu, aspek *pickup* dan *delivery* simultan juga diperhatikan, menjadikan penelitian ini cukup komprehensif dalam hal operasional distribusi. Namun demikian, penelitian ini tidak

mempertimbangkan batasan waktu (*time windows*), serta tidak menggunakan konsep *overflow warehouse*, yang menjadi salah satu inovasi penting dalam penelitian ini.

Selanjutnya, penelitian oleh Dumez et al. (2021) memfokuskan kajiannya pada varian *Generalized Vehicle Routing Problem with Time Windows (GVRPTW)* dan mengusulkan metode metaheuristik berbasis *Large Neighborhood Search* (LNS) yang dikombinasikan dengan komponen *exact* seperti *set partitioning* dan *neighborhood*. Penelitian ini memiliki kelebihan karena pendekatan yang digunakan yaitu hybrid yang adaptif dan fleksibel. Penelitian ini juga berhasil memberikan solusi berkualitas tinggi dalam waktu yang relatif efisien, khususnya untuk kasus VRPTW kompleks. Namun, kekurangan yang cukup signifikan dari penelitian ini adalah tidak mempertimbangkan aktivitas *pickup* dan *delivery*, serta tidak ada konsep *overflow warehouse* dalam struktur modelnya. Dibandingkan dengan penelitian ini, penelitian tersebut hanya menyelesaikan distribusi. Penelitian ini justru menyatukan elemen VRP, yaitu *pickup-delivery*, *time window*, dan distribusi dengan pemanfaatan *hub/ overflow warehouse* (OW), yang menjadikannya lebih representatif terhadap tantangan nyata dalam industri logistik saat ini.

Tabel 2. 1 Posisi Penelitian

Penulis	**Obyek (Industri)**			**Metode**			**Solusi Khusus**
	Minyak	**Manufaktur**	***Retail***	***Exact***	***Heuristic***	***Metaheuristic***	***Overflow Warehouse***
(Sugiono, 2022)		V		V			
(Panjaitan et al., 2024)		V			V		
(Muladi et al., 2020)	V			V	V		
(Wulandari, 2020)		V		V	V		
(Arifian & Pulansari, 2023)		V			V		
(Sumarsa & Widyastiti, 2022)		V		V			
(Praxedes et al., 2024)		V		V			
(Hester Patmawati1, 2022)			V			V	
(Annisa et al., 2025)			V			V	
(Rizkiani et al., 2023)			V		V	V	
(Ferdiansyah et al., 2020)			V		V		
(Dumez et al., 2021)			V	V		V	
(Anggraeni et al., 2023)	V			V			
(Zhao et al., 2024)			V	V			

Penulis	Obyek (Industri)			Metode			Solusi Khusus
	Minyak	**Manufaktur**	***Retail***	***Exact***	***Heuristic***	***Metaheuristic***	***Overflow Warehouse***
(James & Nugroho, 2024)		V			V		
Usulan		V		V			Solusi yang dilakukan yaitu melakukan analisis VRPPDTW dengan memanfaatkan *Overflow warehouse*

2.2 Landasan Teori

2.2.1 Optimasi

Optimasi adalah proses menentukan kondisi yang tepat untuk memperoleh hasil yang paling optimal dalam suatu situasi. Pendekatan normatif dalam optimasi memungkinkan identifikasi solusi terbaik suatu permasalahan dengan tujuan memaksimalkan atau meminimalkan hasil berdasarkan fungsi tertentu. Dalam pengambilan keputusan, optimasi digunakan sebagai metode untuk menemukan solusi paling efektif sesuai dengan batasan yang ada. Dengan menerapkan optimasi, perusahaan dapat mencapai hasil terbaik yang tetap selaras dengan kendala yang telah ditetapkan (Suryanto et al., 2019). Optimasi juga merupakan suatu permasalahan yang berkaitan dengan penentuan nilai terbaik dari suatu fungsi dengan beberapa variabel, baik untuk mencapai nilai maksimum maupun minimum, dengan tetap memperhatikan berbagai batasan yang ada. Batasan-batasan tersebut umumnya mencakup sumber daya seperti tenaga kerja, modal, dan material sebagai input, serta faktor waktu dan ruang. Sementara itu, prinsip optimasi dalam produksi berfokus pada pengalokasian sumber daya yang terbatas secara efisien, sehingga dapat menghasilkan output yang maksimal (Purba & Ahyaningsih, 2020)

2.2.2 Distribusi

Distribusi merupakan rangkaian kegiatan yang memungkinkan produk atau layanan dari produsen dapat menjangkau konsumen atau pengguna tepat waktu dan di lokasi yang dibutuhkan. Proses ini melibatkan berbagai pihak, termasuk distributor dan pelaku usaha, yang berperan dalam perpindahan produk secara fisik serta pengalihan kepemilikan guna meningkatkan daya serap pasar. Untuk tetap bersaing di tengah persaingan yang semakin ketat, perusahaan menerapkan kebijakan distribusi yang strategis dan terarah (Ayu Muchlisa & Surianto, 2021). Distribusi dapat diartikan sebagai suatu kegiatan pemasaran yang ditujukan untuk memperlancar dan mempermudah penyaluran produk dari produsen ke konsumen secara tepat sasaran. Oleh karena itu, produk yang didistribusikan harus mendapat perhatian mendetail, terutama ketika perusahaan memiliki beberapa titik pengiriman, sehingga perlu merancang rute distribusi atau penyaluran yang tepat untuk menjamin kepuasan konsumen. Aktivitas distribusi menyerap biaya yang signifikan dalam perusahaan manufaktur, sehingga evaluasi dan perbaikan metode distribusi selalu dilakukan secara berkelanjutan (Kushariyadi et al., 2024).

2.2.3 Supply Chain Management

Di tengah era globalisasi dan persaingan yang semakin ketat, manajemen rantai pasok atau *Supply Chain Management* (SCM) menjadi elemen krusial dalam meningkatkan kinerja operasional perusahaan. SCM tidak hanya berperan dalam mengelola aliran barang, informasi, dan keuangan di sepanjang rantai pasok, tetapi juga berfungsi sebagai strategi untuk meningkatkan efisiensi serta memperkuat daya saing Perusahaan. Manajemen rantai pasok didefinisikan sebagai proses logistik dan produksi dalam suatu jaringan perusahaan di industri tertentu. Salah satu aspek yang membutuhkan biaya besar dalam pemasaran produk adalah manajemen logistik, yang mencakup perancangan produk, peramalan permintaan, pengadaan bahan baku, proses produksi, pengelolaan persediaan, serta penyimpanan barang. Sementara itu, rantai pasok atau *supply chain* merupakan sistem yang memungkinkan suatu organisasi menyalurkan produk dan layanan kepada *customer*. Rantai ini terdiri dari jaringan berbagai organisasi yang saling ter*hub*ung dan bekerja sama dengan tujuan yang sama, yaitu memastikan proses pengadaan barang berjalan secara optimal (Anisa1 et al., 2025). Dengan demikian, rantai pasok dapat dianggap sebagai jaringan yang meng*hub*ungkan unit-unit organisasi yang beroperasi dan berkoordinasi untuk mengelola, mengendalikan, serta meningkatkan aliran materi dan informasi, yang dimulai dari pemasok hingga ke masyarakat (konsumen). Karena adanya kompleksitas dalam proses pengambilan keputusan di rantai pasok, maka kebutuhan terhadap metodologi pemodelan meningkat (Nurdin et al., 2023).

2.2.4 Manajemen Logistik

Manajemen logistik melibatkan rangkaian kegiatan yang mencakup perencanaan, pengorganisasian, serta pengawasan atas seluruh proses, mulai dari pengadaan, pencatatan, pendistribusian, penyimpanan, pemeliharaan, hingga penghapusan *logistic* dengan tujuan mendukung efektivitas dan efisiensi dalam mencapai target organisasi. Bila informasi yang tersedia tidak akurat, kurang detail, tidak tepat waktu, atau tidak relevan, maka proses pengambilan keputusan di perusahaan pun akan mengalami kesalahan. logistik selalu berkutat dalam menemukan keseimbangan untuk 2 hal yang amatlah sulit untuk disinergikan, yaitu menekan biaya serendah-rendahnya tetapi tetap menjaga tingkat kualitas jasa dan kepuasan konsumen. Beberapa hal yang termasuk sebagai masukan logistik antara lain (Alacsel, 2024):

1. Sumber Daya Alam

   Sumber daya alam memainkan peranan yang sangat penting dalam aktivitas ekonomi suatu perusahaan, karena semua kebutuhan dasar bisnis berasal dari alam. Sebagai komponen

utama dalam manajemen logistik, sumber daya alam berfungsi sebagai modal dasar dalam kegiatan produksi. Penggunaan yang optimal dari sumber daya alam menjadi kunci dalam penyediaan bahan baku yang nantinya akan diproses menjadi produk bernilai.

2. Sumber Daya Manusia

   Dalam lingkungan bisnis, sumber daya manusia merupakan aset yang sangat berharga dan sering disebut sebagai penggerak utama perkembangan. Keberhasilan operasional sangat bergantung pada kualitas SDM, yang juga berperan sebagai faktor produksi dalam manajemen logistik. Untuk mencapai sasaran organisasi, pengelolaan SDM harus dilakukan secara menyeluruh. Peningkatan kualitas SDM dapat dicapai melalui strategi pengembangan seperti pendidikan dan pelatihan yang tepat, sistem evaluasi kinerja beserta mekanisme penghargaan, serta proses rekrutmen dan seleksi yang efektif, didukung oleh perencanaan anggaran yang memadai dan komunikasi internal yang baik. Di era teknologi informasi saat ini, kemampuan menguasai teknologi berperan penting dalam menentukan pencapaian target perusahaan, yang tentunya memerlukan SDM yang kompeten.

3. Fasilitas Pendanaan

   Fasilitas pendanaan, yang mencakup aspek financing atau pengeluaran, merujuk pada dana yang disediakan untuk mendukung investasi yang telah direncanakan, baik melalui pendanaan internal maupun eksternal. Dalam konteks manajemen logistik, penyediaan dana yang tepat merupakan aspek penting agar semua aktivitas dapat terlaksana sesuai dengan rencana dan tujuan yang diharapkan.

4. Sistem Informasi

   Keberadaan sistem informasi sangat krusial dalam manajemen logistik, karena berfungsi sebagai dasar bagi pengambilan keputusan. Sistem ini menyajikan data yang diperlukan untuk menentukan langkah-langkah selanjutnya dalam proses logistik. Pengelolaan sistem informasi yang efektif memastikan bahwa data yang dihasilkan akurat, cepat, dan relevan, sehingga keputusan yang diambil dapat berjalan optimal. Sistem informasi mendukung berbagai area dalam manajemen logistik, mulai dari pengelolaan sumber daya manusia, pengendalian penggunaan sumber daya alam sebagai input, hingga pengawasan keuangan dan pergerakan barang, yang semuanya harus menghasilkan informasi yang dapat diandalkan guna mendukung keputusan strategis.

Manajemen logistik merupakan bagian dari proses rantai pasok yang fungsinya mengendalikan efisiensi serta efektivitas arus penyimpanan barang, jasa, dan informasi

pelayanan beserta informasi terkait, mulai dari tempat asal hingga ke tujuan, baik di lokasi produksi maupun konsumsi (Afiya et al., 2022).

2.2.5 Mixed Integer Linear Programing

Pengembangan model matematis pada jaringan distribusi merupakan bagian dari riset operasional yang berkaitan dengan pengambilan keputusan dalam mengalokasikan sumber daya yang terbatas. Penelitian mengenai desain jaringan distribusi telah banyak dilakukan sebelumnya. Salah satu penelitian penting adalah Geoffrion, dkk (1974), di mana masalah multi komoditas diselesaikan melalui pemrograman integer campuran untuk menentukan lokasi dan kapasitas pusat distribusi, permintaan *customer*, serta aliran transportasi, dengan menggunakan metode dekomposisi Benders. Rajkanth, dkk (2017) mengemukakan model matematika optimasi biaya bertingkat menggunakan *Mixed Integer Linear Programming* (MILP), di mana biaya produksi, transportasi, dan penanganan produk dijadikan kriteria optimisasi yang harus dihitung secara menyeluruh dalam sistem distribusi. Selain itu, berbagai model jaringan telah diusulkan untuk mengoptimalkan distribusi yang mencakup beberapa pabrik, produk, pusat distribusi, pengguna akhir, dan periode waktu yang berbeda (Fatma & Manurung, 2021). Metode *Mixed Integer Linear Programming* (MILP) dipandang dapat memberikan kerangka kerja optimisasi matematika untuk merepresentasikan karakteristik persoalan (Nurdin et al., 2023).

2.2.6 *Traveling Traveling* Problem (TSP)

Banyak perusahaan besar maupun yang sedang berkembang harus mendistribusikan barang ke sejumlah konsumen. Proses pendistribusian ini harus diiringi dengan penentuan rute yang tepat agar distribusi berjalan secara optimal. Terdapat banyak pilihan rute yang dapat diambil perusahaan, sehingga jarak tempuh, biaya, dan waktu yang dibutuhkan pun berbeda-beda. Oleh karena itu, diperlukan suatu metode yang mampu menganalisis proses pendistribusian produk agar dapat meminimalkan jarak, biaya, waktu, serta tenaga yang dikeluarkan. Dalam kasus seperti ini, salah satu pendekatan penyelesaian masalah pendistribusian adalah dengan menggunakan *Traveling Traveling* Problem (TSP). Tujuan utama dari TSP adalah meminimalkan jarak total yang ditempuh dalam satu perjalanan (Maya Widyastiti, 2024). Salah satu permasalahan routing kendaraan yang terkenal adalah *Traveling Traveling* Problem (TSP). Dalam TSP, seorang *Traveling* harus mengunjungi sejumlah kota dan kembali ke kota asalnya dengan memilih rute terpendek. *Traveling* tersebut menempuh perjalanan melalui serangkaian kota tertentu, kemudian kembali ke titik awal, sehingga total jarak yang ditempuh minimal

(Setiadi et al., 2024). Model matematis dalam travelling *Traveling* problem, dapat didefinisikan seperti pada Persamaan (1) hingga Persamaan (4).

$$\text{Minimize Z} = \sum_{i=1}^{n}\sum_{j=1}^{n} d\,[i,j] * c\,[i,j] \qquad (1)$$

$$\sum_{j=1}^{n} x\,[i,j] = 1, \qquad i = 1,..n \qquad (2)$$

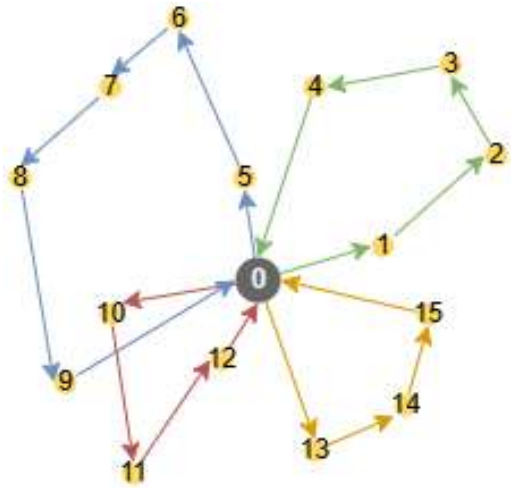
$$\sum_{i=1}^{n} x\,[i,j] = 1, \qquad j = 1,..n \qquad (3)$$

$$x\,[i,j] \in \{\,0,1\,\}, \qquad i,j = 1,..n \qquad (4)$$

Permasalahan TSP akan berusaha mencari rute terpendek, dimana rute yang ditempuh akan mulai dan berakhir pada nodes yang sama, dan mengunjungi nodes lain tepat satu kali. Pada Persamaan (1) merupakan fungsi tujuan dari masalah TSP, dimana akan meminimalkan nilai perkalian antara jarak dengan *cost*, untuk setiap kombinasi *nodes* [i, j], yang sejumlah n. Persamaan (2) dan (3) menyatakan fungsi pembatas, sedemikian hingga setiap kombinasi *nodes* [i, j] hanya boleh dikunjungi satu kali dalam rute, dan untuk Persamaan (4) sebagai bentuk biner dari keputusan untuk mengunjungi *nodes* (bernilai 1) atau tidak mengunjungi *nodes* (bernilai 0) (Tjoea & Halim, 2023).

2.2.7 *Vehicle Routing Problem* (VRP)

Vehicle Routing Problem (VRP) adalah permasalahan dalam pengelolaan distribusi yang melibatkan penentuan rute optimal bagi sejumlah kendaraan yang beroperasi dari satu atau lebih *Depot* untuk mengantarkan barang kepada *customer* yang tersebar di berbagai lokasi. Setiap kendaraan memiliki kapasitas angkut tertentu, sedangkan setiap *customer* memiliki tingkat permintaan yang harus dipenuhi. Tujuan utama dari VRP adalah mengoptimalkan pengiriman agar dapat menjangkau seluruh *customer* dengan biaya seminimal mungkin melalui perencanaan rute kendaraan yang efisien, baik dalam perjalanan menuju maupun kembali ke *Depot* (Kushariyadi & Bambang Sugito, 2022). Berikut merupakan ilustrasi VRP menggunakan 4 kendaraan:



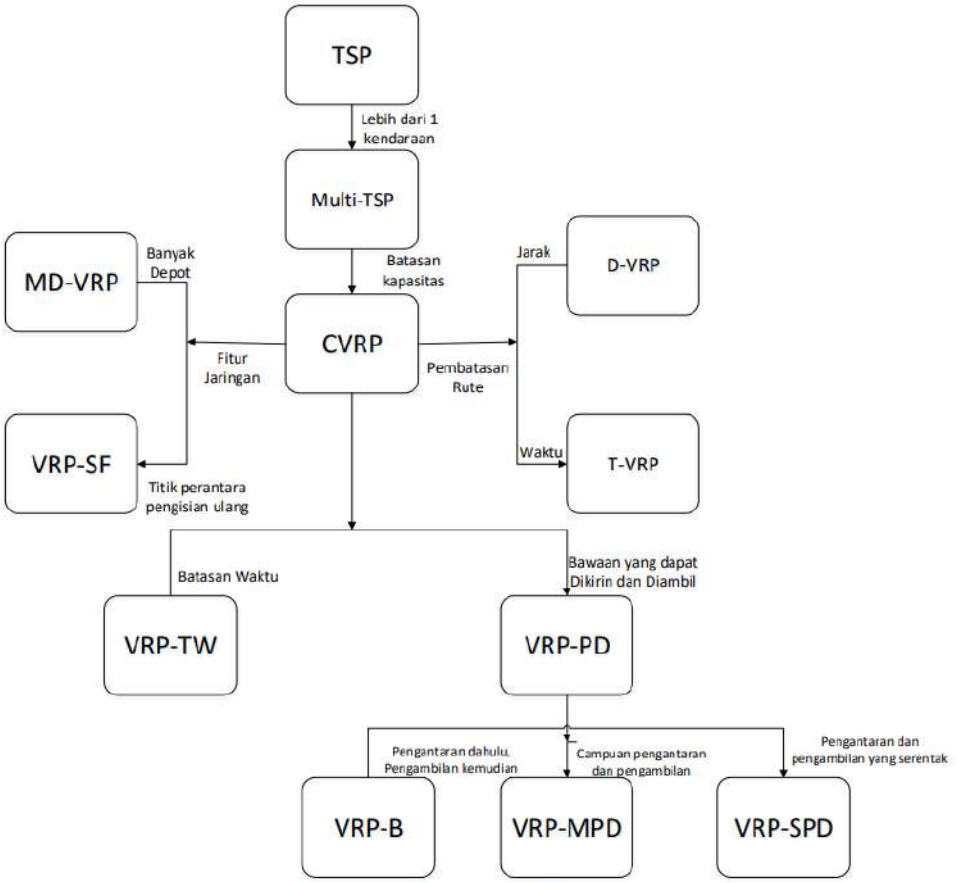
Gambar 2. 1 Ilustrasi VRP 4 kendaraan

Sumber gambar: (Haris & Muhsi, 2023)

Berdasarkan pada gambar diatas dapat diketahui bahwa titik 0 merupakan Lokasi *Depot*, dan diketahui memiliki 4 alur distribusi menggunakan 4 kendaraan yang dimiliki. Kendaraan 1 melakukan rute distribusi dimulai dari *Depot*- 1-2-3-4- *Depot*, kendaraan 2 dimulai dari *Depot*-5-6-7-8-9-*Depot*, kendaraan 3 dimulai dari *Depot*-10-11-12-*Depot*, kendaraan 4 dimulai dari *Depot*-13-14-15-*Depot*.

Vehicle Routing Problem (VRP) juga merupakan suatu metode optimasi dalam sistem distribusi yang bertujuan menentukan rute terbaik bagi sejumlah kendaraan dengan kapasitas tertentu, sehingga dapat memenuhi permintaan *customer* yang telah ditetapkan secara efisien. Setiap kendaraan harus menempuh rute yang dimulai dan berakhir di *Depot* asalnya, dengan tetap memperhatikan kendala operasional yang ada. Rute yang optimal adalah rute yang mampu memenuhi seluruh permintaan *customer* dengan meminimalkan biaya transportasi, serta memiliki total jarak dan waktu tempuh yang paling efisien, sekaligus menggunakan jumlah kendaraan yang terbatas secara optimal (Dewi & Faishal Mashuda, 2022).

Ada beberapa batasan yang tidak boleh diabaikan, antara lain ketidakmampuan armada untuk mengangkut produk melebihi kapasitasnya, ketidakmampuan armada untuk melintasi rute selain yang telah ditentukan, dan pelayanan yang diberikan kepada *customer* harus diselesaikan dalam waktu yang telah ditentukan(Ritonga et al., 2024). Menurut Gonzales-Feliu, klasifikasi VRP di bedakan menjadi beberapa bagian seperti diperlihatkan pada bagan dibawah ini (Hester Patmawati1, 2022):



Gambar 1 Pembagian CRP
Sumber: (Gonzales-Feliu, 2019)

Gambar 2. 2 Variasi VRP

Sumber gambar: (Gonzales-Feliu, 2019)

Berikut merupakan penjelasan klasifikasi VRP (Dewi & Faishal Mashuda, 2022):

1. ***Capacitated Vehicle Routing Problem (CVRP).*** adalah sistem distribusi memiliki satu *Depot* dan hanya memiliki batasan kapasitas kendaraan dengan fungsi tujuan meminimalkan total biaya transportasi. (Toth & Vigo, 2002). *Capacitated* VRP *Capacitated* VRP adalah mencari cara terbaik untuk mengirimkan produk dari satu gudang ke beberapa *customer*, dengan mempertimbangkan kapasitas kendaraan yang terbatas, dan tujuan utamanya adalah meminimalkan jumlah kendaraan yang digunakan misalnya jumlah barang yang dibawa dilarang melebihi kapasitas armada (Ritonga et al., 2024).
2. ***Distance Constrained Vehicle Routing Problem (DCVRP*).** Tipe permasalahan ini merupakan turunan dari permasalahani CVRP, dengan menambahkan batasan total

waktu tempuh dari setiap rute. Tujuan dari permasalahan ini adalah meminimumkan total jarak atau waktu tempuh.

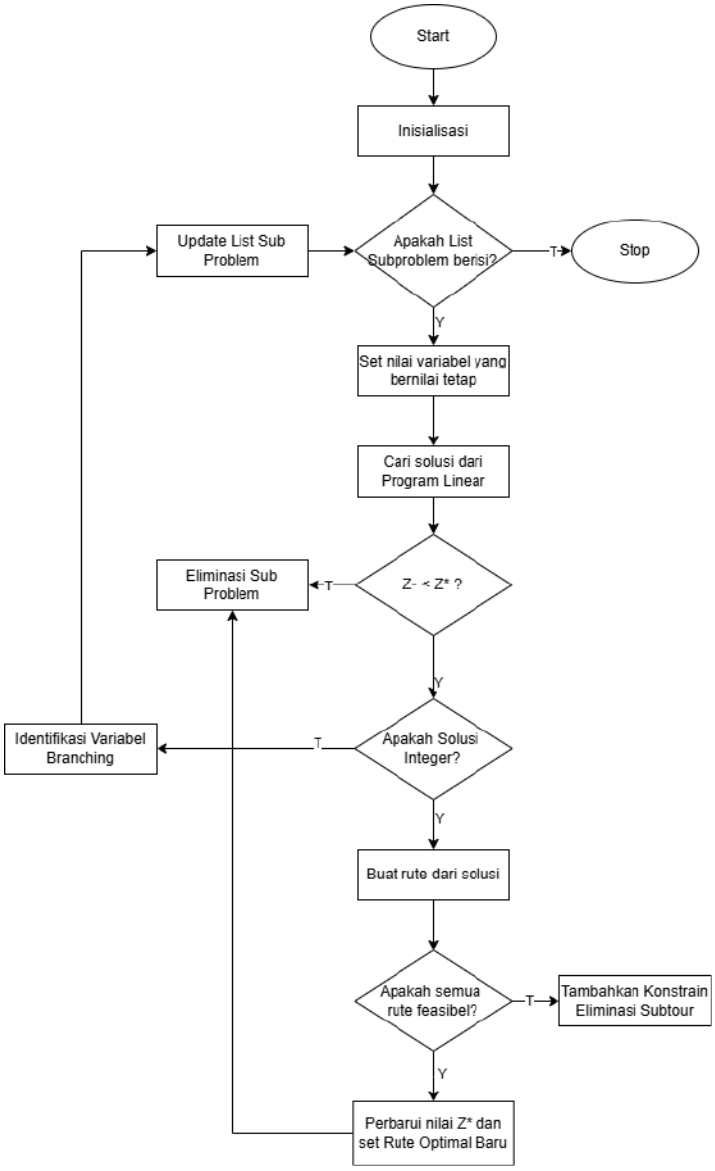
3. **Vehicle Routing Problem with Back Hauls (VRPB).** Ciri khas dari permasalahan ini adalah CVRP yang dibagi dalam 2 tipe konsumen, yaitu konsumen yang meminta layanan angkut (*back haul*) dan konsumen yang meminta layanan antar (*line haul*). Dimana sebelum melakukan pengangkutan maka kendaraan harus memenuhi semua jadwal pengiriman terlebihi dahulu. (Toth & Vigo, 2002). VRP dilengkapi dengan *Backhauls* VRP *backhaul* adalah kategorisasi VRP dimana armada mengantarkan dan mengambil pada rute yang sama (Ritonga et al., 2024).
4. ***Vehicle Routing Problem* with *Time windows* (VRPTW).** VRP yang mempertimbangkan batasan-batasan *time windows* (dimana tiap konsumen memiliki waktu kunjungan tertentu). (Toth & Vigo, 2002). VRP *Time windows* (VRPTW) merupakan jenis atau klasifikasi VRP dengan permasalahan pengiriman yang dilakukan oleh armada harus sesuai dengan *time window* yang telah ditentukan oleh masing-masing *customer*. *Time window* dapat juga disebut waktu jendela yaitu rentang atau selang waktu penerimaan barang oleh *customer* dari perusahaan. Pengiriman barang diluar *time window customer* akan dikenakan penalti tergantung dari masing-masing *customer*. VRPTW membutuhkan pertimbangan berupa *service time, travel time,* dan *waiting time* untuk setiap armada yang mengunjungi lokasi *customer* (Akbar et al., 2021).
5. ***Vehicle Routing Problem* with Pick Up and Delivery (VRPPD).** VRP dimana setiap *customer* memiliki satu lokasi *pick up* (jemput) dan satu lokasi *delivery* (antar). Dengan lokasi *pick up* bisa identik bisa tidak dengan *lokasi delivery*. (Toth & Vigo, 2002). VRP *Pickup and Delivery* adalah kategorisasi VRP yang mensyaratkan barang dimuat pada titik tertentu dan dibongkar di tempat tujuan. Kendaraan yang sama harus digunakan untuk penjemputan dan pengantaran sehingga lokasi penjemputan dan pengantaran berada pada rute yang sama (Ritonga et al., 2024).
6. ***Vehicle Routing Problem with Back Hauls and Time windows* (VRPBTW).** VRPB yang memperhatikan *time windows* (Toth & Vigo, 2002) .
7. ***Vehicle Routing Problem with Pick Up and Delivery with Time windows (VRPPDTW).*** VRPPD dengan ada aturan kunjungan tertentu pada setiap lokasi penjemputan dan pengiriman. (Toth & Vigo, 2002)

8. ***Multiple Depot Vehicle Routing Problem (MDVRP*).** VRP dengan jumlah *Depot* lebih dari satu. (Toth & Vigo, 2002).
9. ***Open Vehicle Routing Problem (OVRP)*** adalah bahwa kendaraan tidak perlu kembali ke tempat yang merupakan pusat distribusi (*Depot*). (Toth & Vigo, 2002).
10. ***Heterogeneous Fleet VRP (HFVRP*)** dapat juga disebut *Mixed Fleet* VRP adalah jenis atau klasifikasi dari VRP yang mempunyai permasalahan kendaraan pengiriman yang digunakan lebih dari satu dan mempunyai karakteristik yang berbeda-beda untuk setiap jenis armada yang digunakan. Karakteristik kendaraan berupa kapasitas, *variabel* dan *fixed cost*, kecepatan, dan akses kepada *customer* yang berbeda untuk setiap armada yang dimiliki Perusahaan (Akbar et al., 2021). *Heterogeneous Fleet* VRP merupakan varian dari VRP dimana kendaraan dalam armada berbeda satu sama lain dalam hal kapasitas, kecepatan, dan biaya bahan bakar. HVRP adalah salah satu jenis VRP dengan konflik bahwa suatu perusahaan mempunyai armada yang bervariasi, umumnya berupa kapasitas angkut setiap armada. VRP ini pula bisa dianggap menjadi VRP armada campuran (Ritonga et al., 2024)

2.2.8 Algoritma Penyelesaian VRP

1. Eksak

   Metode eksak bertujuan untuk menemukan solusi optimal di antara semua kemungkinan solusi yang ada. Beberapa metode eksak yang dikenal meliputi algoritma klasik seperti *dynamic programming* dan keluarga *branch & X algorithm* (termasuk *Branch and Bound*, *branch and cut*, serta *branch and price*)**.** Berikut merupakan Diagram alir proses penyelesaian VRP dengan Algoritma Sekuensial *Branch and Bound*:



Gambar 2. 3 Diagram alir VRP dengan *Branch and Bound*

Sumber gambar: (Arunanto & Hintono, 2011)

Selain itu, metode ini juga mencakup algoritma yang dikembangkan dalam bidang riset operasi, seperti *constraint programming* dan algoritma pencarian dari keluarga *A* (*A* search algorithm). Untuk mempermudah penerapan metode eksak, terdapat berbagai perangkat lunak seperti AMPL, CPLEX, LINDO, MPL, OMP, dan XPRESS, yang dapat membantu menyelesaikan masalah tanpa harus mengkodekan algoritma secara manual. Penggunaan perangkat lunak ini lebih praktis karena lebih sederhana dibandingkan dengan membuat kode dari setiap algoritma secara langsung. Meskipun metode eksak sangat efektif dalam menyelesaikan masalah berskala kecil, metode ini tidak disarankan untuk menangani permasalahan yang lebih kompleks dengan skala besar. Hal ini dikarenakan waktu komputasi yang dibutuhkan akan meningkat secara

signifikan seiring bertambahnya ukuran masalah. Oleh karena itu, dalam menyelesaikan masalah yang lebih kompleks, diperlukan metode yang memiliki waktu komputasi lebih cepat namun tetap mampu menghasilkan solusi dengan kualitas tinggi (Redi A.A.N et al., 2021).

Metode eksak memiliki keunggulan dalam memberikan solusi optimal dengan jaminan keakuratan yang tinggi serta lebih kuat secara teoritis. Namun, metode ini mengalami kesulitan dalam menangani kompleksitas permasalahan VRP di dunia nyata yang semakin meningkat, terutama dalam hal efisiensi komputasi. (Liu et al., n.d.)

2. *Heuristic*

   metode heuristik menjadi solusi yang banyak digunakan karena mampu menghasilkan solusi dengan kualitas tinggi dalam waktu komputasi yang lebih wajar serta memiliki kemampuan generalisasi yang baik. Metode heuristik dalam VRP dapat diklasifikasikan menjadi tiga kategori utama. Pertama, *Constructive heuristic*, yaitu algoritma yang membangun solusi rute dari nol berdasarkan prosedur heuristik tertentu. Metode ini mampu menghasilkan solusi yang layak dengan cepat dan mudah diterapkan pada berbagai varian VRP, namun sering kali masih memiliki kesenjangan terhadap solusi optimal. Kedua, *Improvement heuristic*, yang bekerja dengan meningkatkan solusi rute awal secara iteratif melalui pencarian lokal dalam lingkup tertentu. Meskipun efisien dalam menemukan solusi lokal terbaik, metode ini rentan terjebak dalam *local optimum* dan kualitas akhirnya sangat bergantung pada titik awal pencarian (Liu et al., n.d.)

3. *Metaheuristic*

   Vehicle Routing Problem (VRP) awalnya diselesaikan menggunakan metode eksak berbasis matematika. Namun, karena keterbatasan metode ini dalam menangani kompleksitas VRP yang memiliki banyak variabel, para peneliti mulai beralih ke metode heuristik dan metaheuristik yang lebih efisien. Metaheuristik terbagi menjadi dua jenis utama, yaitu *local search* dan *population search*, dengan *local search* lebih sering digunakan karena lebih cepat dalam menemukan solusi. Beberapa algoritma *local search* yang paling umum digunakan adalah *Tabu Search* (TS) dan *Variable Neighborhood Search* (VNS), sementara algoritma lain seperti *Large Neighborhood Search* (LNS), *Simulated Annealing* (SA), *Iterated Local Search* (ILS), dan *Greedy Randomized Adaptive Search Procedure* (GRASP) digunakan lebih jarang. Adapun dalam kategori *population search*, algoritma yang sering diterapkan adalah *Genetic Algorithm* (GA), *Ant Colony Optimization* (ACO), dan *Particle Swarm Optimization* (PSO) (Elatar et al., 2023).

2.2.9 Solver Gurobi

Gurobi digunakan sebagai alat untuk menyelesaikan model pemrograman linier campuran (MILP) dalam konteks masalah *Multi-Trip Vehicle* Routing (MTDLC-VR). Gurobi *Optimizer* dapat menyelesaikan masalah kecil, tetapi tidak dapat menyelesaikan model MILP untuk instance yang memiliki lebih dari 25 *customer* dalam waktu dua jam karena kompleksitas komputasi yang tinggi dan penggunaan banyak variabel tambahan untuk melinierkan batasan. Untuk menyelesaikan permasalahan, penelitian oleh (Nguyen et al., 2022)menggunakan model *Mixed Integer Linear Programming* (MILP) yang dirancang untuk menangani kasus skala kecil serta kasus adaptasi dari *benchmark* yang telah ada. Model ini diuji menggunakan perangkat lunak *GUROBI Optimizer*, yang merupakan salah satu solver terbaik dalam menyelesaikan masalah optimasi linier dan integer secara efisien. GUROBI mampu menangani permasalahan kompleks dengan jumlah variabel dan kendala yang besar menggunakan teknik seperti *branch and bound*, *cutting planes*, dan *Optimizer search*, sehingga dapat menemukan solusi optimal dalam waktu yang lebih singkat dibandingkan metode konvensional.

BAB III
METODE PENELITIAN

3.1 Objek Penelitian

Objek penelitian merupakan entitas, fenomena, atau sistem nyata yang menjadi fokus utama studi. Objek penelitian mencakup aspek-aspek yang diobservasi dan dianalisis. Objek penelitian ini yaitu optimasi distribusi yang melibatkan armada kendaraan pengiriman, termasuk rute perjalanan, titik-titik *customer* dan juga *Depot* oleh PT. Pos logistik Indonesia. Berikut ini merupakan profil dari PT. Pos Logistik Indonesia:

Nama Perusahaan	: PT. Pos Logistik Indonesia
Alamat	: Gedung Pos Ibukota, Jl. Lap. Banteng Utara No.1, Pasar Baru, Sawah Besar, Central Jakarta City, Jakarta 10710
No. telepon	: +6281297274831
Bidang Usaha	: - *Warehouse* - *Distribution* - *Heavy Lifting* - *International Cargo* - *Domestic Cargo* - *Cold Chain*
Bentuk perusahaan	: Perseroan Terbatas
Tahun Berdiri	: 2012

3.2 Instrumen Penelitian

Dalam studi ini, beberapa peralatan diperlukan untuk mendukung pengumpulan dan analisis data. Berikut adalah instrumen penelitian yang digunakan:

1. *Dataset*, berisikan data yang dibutuhkan dalam pengolahan data seperti koordinat lokasi *customer*, data batas waktu, dan jumlah permintaan setiap *customer*.
2. *Visual Code / python*, Digunakan untuk penulisan kode, pemrosesan data, dan implementasi algoritma
3. AMPL, Merupakan bahasa pemodelan matematika yang digunakan untuk merumuskan permasalahan optimasi secara sistematis kedalam Bahasa pemrograman di *python*. AMPL menyusun model matematis VRP yang melibatkan berbagai kendala dan variable.

4. Gurobi, digunakan untuk mencari solusi optimal dari model yang telah diformulasikan di AMPL dan menyelesaikan *code* dalam python sesuai dengan alur pada metode *Branch and Bound* dan juga simpleks didamnya.

3.3 Sumber Data Primer dan Sekunder

Berikut adalah jenis dan sumber data yang digunakan dalam penelitian ini, yang meliputi data primer dan data sekunder:

1. Data primer, merupakan informasi yang dikumpulkan langsung tanpa perantara. Dalam penelitian ini, data primer didapatkan melalui observasi langsung atau melalui data historis yang mencatat proses distribusi. Data yang dikumpulkan meliputi koordinat titik lokasi, batas waktu pengiriman (*time window*), jumlah permintaan *customer*, jumlah kendaraan yang digunakan, serta kapasitas maksimal kendaraan.
2. Data sekunder diperoleh melalui perantara atau kajian literatur, seperti dokumen organisasi, buku, jurnal, artikel, dan sumber tertulis lainnya. Data ini berfungsi sebagai pendukung dan memperkuat deskripsi kualitatif dalam penelitian, serta membantu memberikan tambahan terhadap data primer yang diperoleh.

3.4 Metode Pengumpulan Data

Pengumpulan data merupakan tahapan penting dalam proses penelitian guna memperoleh informasi yang akurat, relevan, dan mendukung dalam rangka mencapai tujuan penelitian. Dalam penelitian ini, data dikumpulkan melalui dua metode utama, yaitu observasi tidak langsung melalui kajian literatur, serta wawancara, yang dijelaskan sebagai berikut:

1. Wawancara

   Wawancara dilakukan kepada pihak PT Pos Logistik Indonesia yang termasuk praktisi logistik, pengelola gudang, dan akademisi yang relevan. Tujuan dari wawancara ini adalah untuk memperoleh pandangan terkait tantangan operasional dalam permasalahan perusahaan, pengelolaan rute distribusi, khususnya dalam konteks keterbatasan kapasitas gudang dan peran *overflow warehouse* sebagai solusi alternatif. Selain itu, wawancara juga digunakan untuk menggali informasi terkait kebijakan penjadwalan pengiriman, *time windows* pelayanan *customer*, serta biaya eksisting yang dikeluarkan Perusahaan dalam melaukan distribusi.
2. Studi Literatur

Observasi dilakukan secara tidak langsung melalui studi pustaka, kajian literatur, serta mencari berbagai sumber informasi daring yang relevan dengan topik penelitian. Sumber informasi tersebut mencakup jurnal ilmiah, laporan penelitian sebelumnya, artikel industri, data sekunder PT. Pos Logistik Indonesia, serta dokumen teknis yang berkaitan dengan operasional distribusi, peran *overflow warehouse*, dan penerapan algoritma *Exact method* dalam penyelesaian masalah VRP. Penelitian ini ini juga mencakup pengumpulan data yang diperoleh dari contoh kasus atau ilustrasi yang mendekati kondisi nyata seperti parameter-parameter penting lainnya yang diperlukan dalam pemodelan dan pengolahan data.

3.5 Populasi dan Sampel

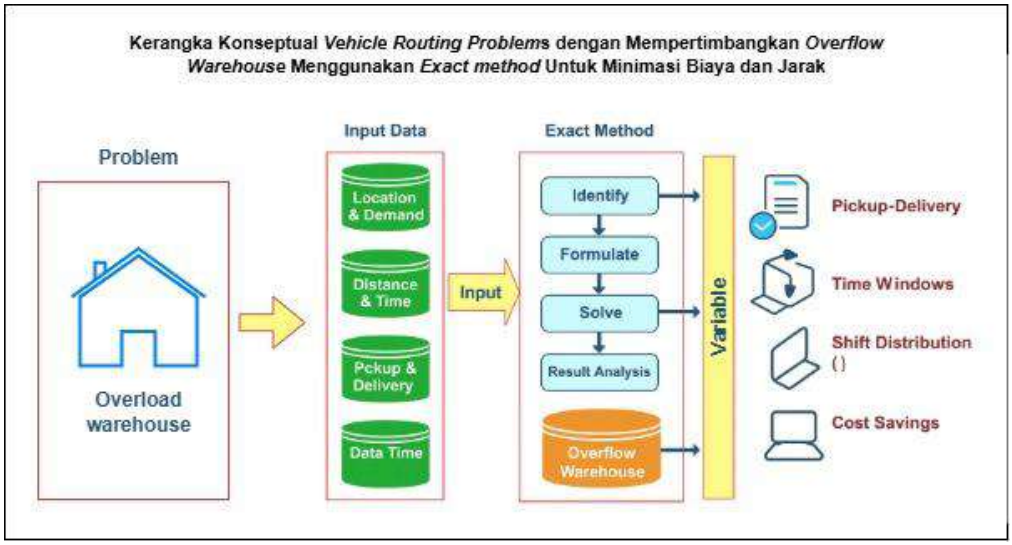
3.5.1 Populasi

Populasi adalah seluruh kelompok atau kumpulan elemen (individu, objek, peristiwa, atau data) yang menjadi objek kajian dalam suatu penelitian. Populasi mencakup semua anggota yang memiliki karakteristik tertentu yang ingin diteliti. Dalam penelitian ini, populasi yang dimaksud adalah jumlah keseluruhan koordinat pengantaran produk PT. Pos Logistik Indonesia.

3.5.2 Sampel

Sampel adalah bagian dari populasi yang dipilih secara representatif untuk dijadikan sumber data dalam penelitian. Karena meneliti seluruh populasi seringkali tidak memungkinkan (karena keterbatasan waktu, biaya, tenaga), maka digunakan sampel yang dianggap cukup mewakili populasi tersebut. Ukuran sampel dalam penelitian ini mencakup titik pengantaran produk di area Jakarta, sebanyak 20-30 titik.

3.6 Kerangka Konseptual



Gambar 3. 1 Kerangka Konseptual

Kerangka konseptual ini menggambarkan alur penyelesaian masalah distribusi logistik yang dihadapi oleh PT Pos Logistik Indonesia, khususnya terkait overload warehouse pada Main Hub. Permasalahan ini diatasi melalui pendekatan Vehicle Routing Problems (VRP) dengan mempertimbangkan *Overflow Warehouse* menggunakan *Exact Method* untuk meminimalkan biaya dan jarak tempuh distribusi.

1. Problem

   Permasalahan utama adalah penumpukan barang di Main Hub karena belum dimanfaatkannya hub cabang sebagai buffer (*overflow warehouse*).

2. Input Data

   Data yang digunakan meliputi:

   1. Lokasi dan permintaan pelanggan *(Location & Demand)*
   2. Matriks jarak dan waktu *(Distance & Time)*
   3. Aktivitas pickup dan delivery *(Pickup & Delivery),*
   4. Batasan waktu layanan *(Data Time).*

3. *Exact Method*

   Proses optimasi dilakukan melalui tahapan:

   1. Identify: mengenali struktur jaringan distribusi dan kendala,
   2. Formulate: memodelkan masalah dalam bentuk *Mixed Integer Linear Programming (MILP)*,
   3. Solve: menyelesaikan model dengan solver (Gurobi),

4. *Result Analysis*: mengevaluasi hasil dan distribusi ulang melalui pemanfaatan Overflow Warehouse.

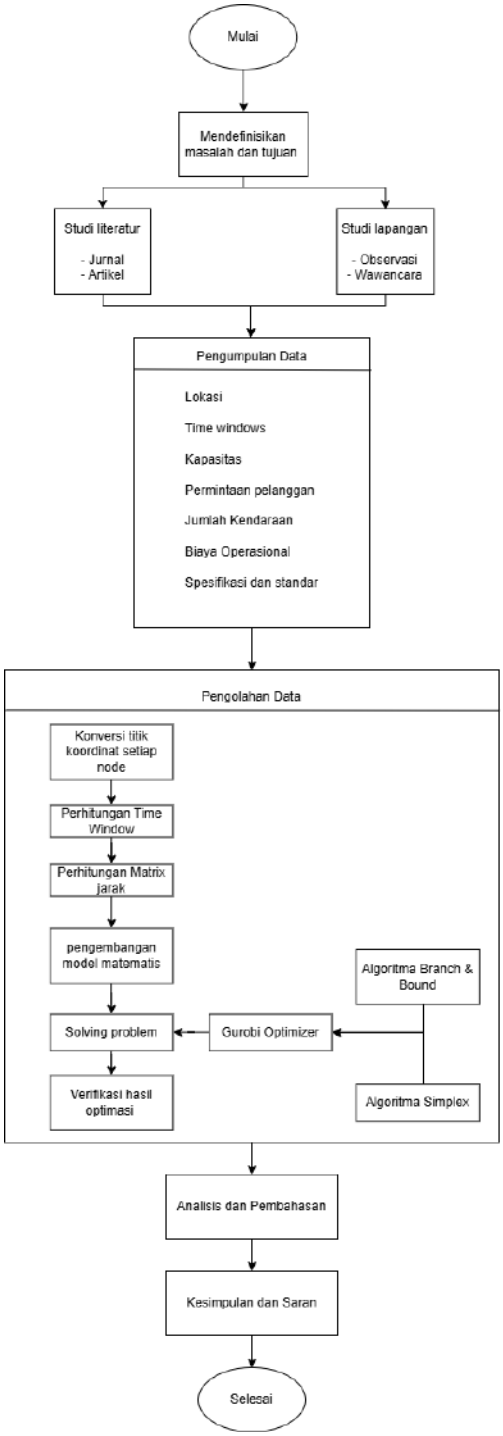
4. Variabel Penting dalam Usulan Rute

   Rute distribusi usulan mempertimbangkan:

   1. Pickup Delivery
   2. Time Windows
   3. Shift Distribution dari Main Hub ke Hub
   4. *Cost Savings* dari total biaya operasional harian

3.7 Alur Penelitian

Alur penelitian bertujuan untuk menjabarkan Langkah-langkah dalam mengembangkan dan merancangan penelitian ini dari awal hingga akhir. Berikut adalah *flowchart* dari alur penelitian ini:



Gambar 3. 2 Alur Penelitian

Penjelasan dari alur penelitian di atas adalah sebagai berikut:

1. Mulai
2. Mendefinisikan masalah dan tujuan

   langkah pertama yaitu mengetahui masalah yang dihadapi serta tujuan yang ingin dicapai. Tahap ini sangat penting karena akan menjadi dasar dari semua tahapan berikutnya.

Dalam proses ini proses yang dilakukan yaitu mengidentifikasi masalah melalui wawancara, menentukan batasan permasalahan, serta tujuan yang ingin dicapai seperti meminimalkan biaya operasional pengiriman.

3. Studi literatur

   Studi literatur dilakukan untuk memperoleh pemahaman konseptual dan teoretis yang kuat. Studi literatur dilakukan menggunakan berbagai jurnal ilmiah, artikel, dan sumber literatur lainnya yang relevan dengan topik. Tujuan dilakukannya yaitu untuk memahami permasalahan perusahaan dan cara menyelesaikan masalah tersebut menggunakan metodologi yang telah digunakan pada penelitian sebelumnya. Selain itu, kajian literatur juga membantu dalam menyusun model matematis dan memilih algoritma yang tepat seperti *Branch & Bound*, Simplex dengan menggunakan Gurobi *Optimizer*.

4. Studi lapangan

   Studi lapangan dilakukan bersamaan dengan melakukan kajian literatur. Untuk mendapatkan data aktual sesuai dengan kondisi nyata, dilakukan studi lapangan. Proses ini melibatkan observasi langsung dan wawancara dengan pihak PT. Pos Logistik Indonesia. Studi lapangan sangat penting untuk melengkapi data serta memastikan bahwa asumsi yang dibuat dalam model sesuai dengan kondisi nyata di lapangan. Informasi dari lapangan juga digunakan untuk memvalidasi relevansi model terhadap proses penyelesaian permasalahan distribusi aktual.

5. Pengumpulan data

   Tahap ini merupakan proses dalam mengumpulkan seluruh data yang dibutuhkan untuk proses pemodelan. Data yang dikumpulkan mencakup lokasi titik *customer* dan *Depot*, batasan waktu pelayanan (*time windows*), kapasitas kendaraan, permintaan setiap *customer*, jumlah kendaraan yang tersedia, biaya operasional, serta spesifikasi dan standar yang harus dipenuhi. Data ini akan menjadi input utama dalam pengembangan model matematis dan proses optimasi rute distribusi.

6. Pengolahan Data

   Setelah data terkumpul, selanjutnya masuk ke tahap pengolahan data. Tahap ini dimulai dengan melakukan konversi titik longitude dan latitude menjadi titik koordinat X dan Y agar menjadi lebih sederhana dan menghitung matrix jarak serta time window setiap node, setelah itu pengembangan model matematis dan penentuan *constraint-constraint* yang harus dipertimbangkan dalam merepresentasikan permasalahan distribusi dalam

bentuk formula dan kendala. Model ini kemudian diselesaikan dengan Bahasa pemrograman *phyton* di *Visual studio code* menggunakan Gurobi *Optimizer*, yaitu *solver* yang dirancang untuk menyelesaikan masalah optimasi. Dalam menyelesaikan model ini, digunakan algoritma *Branch and Bound* dan *Simplex* untuk menemukan solusi optimal. Setelah solusi diperoleh, dilakukan proses verifikasi hasil optimasi untuk mengecek apakah hasil yang diperoleh secara logis dan matematis dapat menyelesaikan permasalahan perusaan atau belum. Jika belum sesuai, maka dilakukan perbaikan terhadap model.

7. Analisis dan pembahasan

   Setelah model tervalidasi, tahapan selanjutnya yaitu melakukan analisis terhadap hasil yang diperoleh. Analisis ini mencakup performa solusi dari berbagai aspek, seperti rute, Waktu tempuh, serta strategi mengurangi biaya operasional. Selain itu juga membandingkan kondisi sebelum dan sesudah optimasi, serta membahas kelebihan dan kekurangan dari metode yang digunakan.

8. Kesimpulan dan Saran

   Dari hasil analisis yang dilakukan, kemudian merumuskan kesimpulan utama dari penelitian. Kesimpulan ini menjawab tujuan awal yang telah dirumuskan dan memberikan gambaran apakah penelitian berhasil menyelesaikan masalah yang diidentifikasi. Selain itu, menyampaikan saran untuk pengembangan penelitian selanjutnya. Saran ini dapat mencakup pengembangan metode, peningkatan kualitas data, atau penerapan pada skenario yang lebih luas.

9. Selesai

BAB IV
PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

4.1.1 Profil Perusahaan

Gambar 4. 1 Logo Pos Logistik Indonesia

Sumber Gambar: *Website* Pos Logistik Indonesia

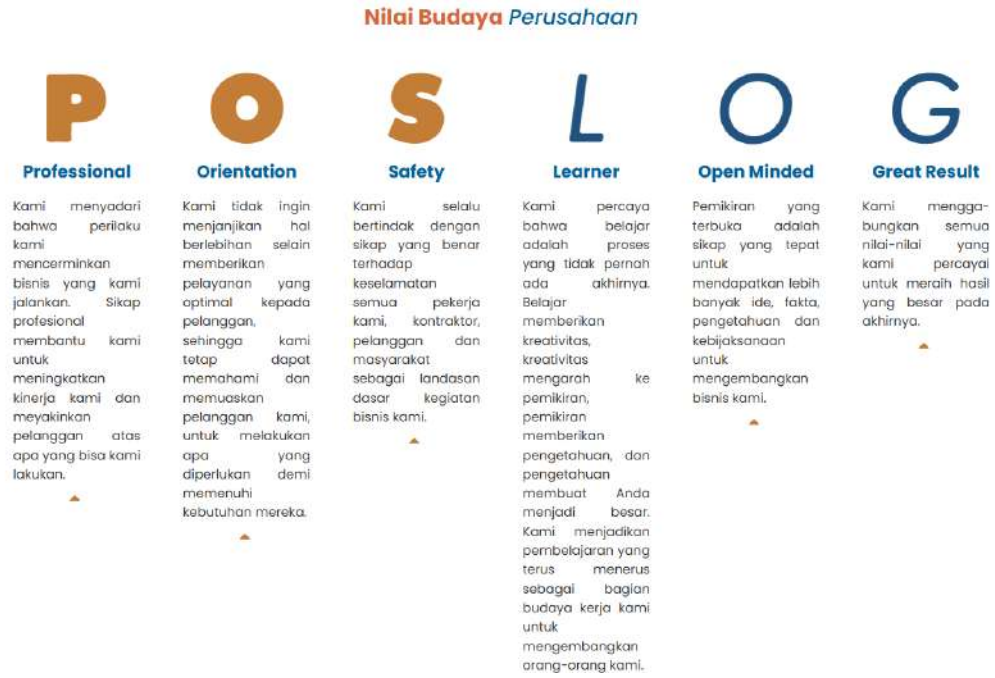
PT Pos Logistik Indonesia (POSLOG) adalah perusahaan yang bergerak di bidang logistik dan merupakan anak usaha dari PT Pos Indonesia (Persero), sebuah Badan Usaha Milik Negara (BUMN) yang telah lama dikenal sebagai penyedia layanan pos nasional. Didirikan pada tanggal 17 Februari 2012 di Gedung Pos Ibukota Lt. 4 Jl. Lapangan Banteng Utara No.1, Kel. Pasar Baru, Kec. Sawah Besar, Kota Jakarta Pusat, DKI Jakarta 10710, POSLOG lahir dari transformasi proyek bisnis yang dimulai pada tahun 2004 dan kemudian berkembang menjadi *Strategic Business Unit* (SBU) pada tahun 2007. Pembentukan perusahaan ini bertujuan untuk merespons pertumbuhan kebutuhan pasar logistik yang kian pesat serta memperkuat posisi PT Pos Indonesia di sektor logistik yang terus berkembang.

Dengan visi menjadi penyedia solusi logistik terbaik di Indonesia, POSLOG berkomitmen untuk memberikan layanan logistik yang menyeluruh, efisien, dan terintegrasi. Misi perusahaan mencakup pemberian solusi bernilai tambah bagi *customer*, peningkatan daya saing logistik nasional, kontribusi terhadap kinerja keuangan grup, pembangunan sinergi internal dengan PT Pos Indonesia, serta pembentukan kemitraan strategis dengan berbagai pihak.

Pos Logistik menawarkan berbagai layanan mulai dari pengiriman ekspres, kurir, kargo, hingga solusi logistik menyeluruh yang tersebar di seluruh penjuru Indonesia. Tak hanya fokus

pada logistik, POSLOG juga telah merambah ke bidang *warehouse, Distribution, Heavy Lifting, Domestic Cargo, International cargo,* dan *Cold Chain service* untuk memperluas portofolio layanan.

Pos Logistik memiliki Nilai Budaya Perusahaan yang dimplementasikan oleh Perusahaan yang sangat penting dalam membentuk identitas, perilaku, dan arah strategis sebuah perusahaan. Berikut beberapa nilai budaya perusahaan:



Gambar 4. 2 Budaya Perusahaan Pos Logistik Indonesia

Sumber Gambar: *Website* Pos Logistik Indonesia

4.1.2 Formulasi Masalah

Penelitian ini difokuskan pada penyelesaian permasalahan penentuan rute distribusi kendaraan di PT Pos Logistik Indonesia. Salah satu tantangan utama yang dihadapi perusahaan adalah belum optimalnya pemanfaatan *Hub* cabang sebagai *buffer warehouse*, sehingga distribusi barang masih terpusat di *Main Hub*. Kondisi ini mengakibatkan terjadinya penumpukan barang, peningkatan waktu tunggu, serta pemborosan sumber daya dalam penggunaan bahan bakar akibat rute kendaraan yang tidak efisien.

Penelitian ini mengkaji distribusi barang dari satu *Main Hub* ke sejumlah *customer* di Jakarta, dengan memanfaatkan beberapa *Hub* cabang sebagai *overflow warehouse/ buffer warehouse* untuk mendukung kelancaran pengiriman. Setiap *customer* dilayani sekali dalam jendela waktu tertentu (*time windows*), dan kendaraan yang digunakan memiliki kapasitas

maksimum 1000 kg. Permasalahan ini termasuk dalam *Vehicle Routing Problem with Pickup Delivery and Time windows with Overflow Warehouse* (VRPPDTW-OW), yang mempertimbangkan aspek *pickup and delivery*, batasan waktu, kapasitas kendaraan, serta pemanfaatan *buffer warehouse*. Tujuan utamanya adalah menghasilkan rute pengiriman optimal dengan jarak tempuh dan waktu pengantaran yang minimal.

Pada penelitian ini, kapasitas hub tidak di pertimbangkan dikarenakan fokus Utama dari penelitian ini yaitu untuk menyelesaikan permasalahan terkati VRP dengan *constraint-constraint* seperti kapasitas kendaraan dan *time windows*nya. dengan tidak mempertimbangkan kapasitas hub, maka model akan lebih di fokuskan kepada penentuan rute yang optimal dengan keberhasilan pemanfaatan *hub hub* dari PT. Pos Logistik Indonesia dengan menggunakan *overflow warehouse* sebagai *buffer.*

Sebagai pendekatan penyelesaian, penelitian ini menerapkan metode *Exact* dengan memanfaatkan *Gurobi solver* untuk menghasilkan solusi optimal. Metode ini digunakan karena mampu menyelesaikan permasalahan dengan kompleksitas tinggi secara matematis dan memberikan hasil yang akurat dalam waktu komputasi yang efisien. Model jaringan distribusi yang digunakan mencakup representasi node dengan $i = 0$ sebagai *Main Hub* (*Depot*), node i = 1, 2, 3, ..., n sebagai *customer*, serta sejumlah node tambahan yang difungsikan sebagai *buffer warehouse*. Melalui pendekatan ini, diharapkan PT Pos Logistik Indonesia dapat meningkatkan efektivitas sistem distribusinya dan mengoptimalkan pemanfaatan aset yang ada.

4.1.3 Data *Main Hub*, *Hub*, dan *Customer*

Dalam melakukan analisis VRPPD menggunakan *Exact method* ini diperlukan data Lokasi *Main Hub*, *Hub*, dan juga *customer*, dalam penelitian ini terdapat 30 titik yang terdiri dari 1 *Main Hub/ Depot*, 26 *Customer*, dan 3 *Hub*. Berikut merupakan data Lokasi *Depot, Hub-Hub*, *Customer*, dan juga *demand* setiap *customer*:

Tabel 4. 1 Data *Main hub, Hub*, dan *Customer*

Node	***ID Number***	***Decimal Degrees***		***UTM***		***Demand***
		Latitude	***Longitude***	***East (X)***	***North (Y)***	***delivery***
Main Hub	0	-6.202303	106.916725	712,08	9314,03	0
Customer 1	1	-6.288459	106.420009	657,08	9304,70	4
Customer 2	2	-6.579977	106.578569	674,53	9272,41	18
Customer 3	3	-6.210823	106.941360	714,81	9313,10	5
Customer 4	4	-6.086256	106.469551	662,63	9327,04	24

Node	*ID Number*	*Decimal Degrees*		*UTM*		*Demand*
		Latitude	*Longitude*	*East (X)*	*North (Y)*	*delivery*
Customer 5	5	-6.462463	106.423838	657,45	9285,45	28
Customer 6	6	-6.625090	106.804284	699,47	9267,33	22
Customer 7	7	-6.778771	106.559070	672,30	9250,43	9
Customer 8	8	-6.280092	106.835953	703,11	9305,48	29
Customer 9	9	-6.623648	106.871413	706,90	9267,47	21
Customer 10	10	-6.152456	106.405199	655,48	9319,74	23
Customer 11	11	-6.122005	106.882981	708,38	9322,94	20
Customer 12	12	-6.154297	106.983785	719,52	9319,33	8
Customer 13	13	-6.371018	107.178493	729,11	9310,48	19
Customer 14	14	-6.136476	106.894816	709,68	9321,34	17
Customer 15	15	-6.110634	106.861882	706,05	9324,21	14
Customer 16	16	-6.236343	106.436660	675,88	9300,59	21
Customer 17	17	-6.617681	106.631510	680,37	9268,22	21
Customer 18	18	-6.736166	106.586233	675,32	9255,13	2
Customer 19	19	-6.719199	106.622379	679,32	9257,00	21
Customer 20	20	-6.291452	106.691866	687,16	9304,27	12
Customer 21	21	-6.503855	106.567606	673,34	9280,83	3
Customer 22	22	-6.343279	106.987197	719,82	9298,43	3
Customer 23	23	-6.281572	106.887305	708,79	9305,29	22
Customer 24	24	-6.349438	106.982359	719,29	9297,75	27
Customer 25	25	-6.669278	106.703564	688,32	9262,49	16
Customer 26	26	-6.008381	106.912000	711,64	9335,50	1
Hub 1	27	-6.168686	106.834678	703,01	9317,80	0
Hub 2	28	-6.197407	106,761562	694,91	9314,65	0
Hub 3	29	-6.223945	106.799311	699,08	9311,70	0

4.1.4 Data *Time window*

Time window adalah rentang waktu tertentu di mana suatu layanan pengiriman atau pengambilan barang harus dilakukan di lokasi *customer*. Apabila Perusahaan melakukan pengiriman diluar *time window* yang sudah ditentukan maka dapat memicu ketidakpuasan *customer,* penambahan jarak tempuh dan bahan bakar. Berikut merupakan *Time window* yang dimiliki Perusahaan yang akan dilakukan dalam pengolahan data:

Tabel 4. 2 Data *Time Window*

Time window	**Waktu**
Time window 1	08.00 WIB – 13.00 WIB
Time window 2	13.01 WIB – 17.00 WIB

Pada Tabel diatas, dapat diketahui Perusahaan memiliki *Time window 1* yang Dimana kendaraan akan melakukan distribususi barang dimulai dari pukul 08.00 pagi hingga 13.00 siang. Sedangkan *Time window 2* dimulai dari pukul 13.01 siang hingga 17.00 sore. Berikut merupakan *Time window* yang dimiliki oleh setiap *customer*:

Tabel 4. 3 Data *Time Window* setiap *Customer*

Node	***ID Number***	***Time windows***	***TW ' earliest***	***TW' latest***	***TW' service***
Main Hub	0	0	0	540	5
Customer 1	1	2	0	300	5
Customer 2	2	1	0	300	5
Customer 3	3	1	301	540	5
Customer 4	4	2	0	300	5
Customer 5	5	2	0	300	5
Customer 6	6	1	301	540	5
Customer 7	7	1	0	300	5
Customer 8	8	2	301	540	5
Customer 9	9	2	301	540	5
Customer 10	10	1	0	300	5
Customer 11	11	1	0	300	5
Customer 12	12	2	301	540	5
Customer 13	13	2	0	300	5
Customer 14	14	1	0	300	5
Customer 15	15	1	0	300	5
Customer 16	16	2	301	540	5
Customer 17	17	2	0	300	5
Customer 18	18	1	0	300	5
Customer 19	19	1	0	300	5

Node	*ID Number*	*Time windows*	*TW ' earliest*	*TW' latest*	*TW' service*
Customer 20	20	2	301	540	5
Customer 21	21	2	301	540	5
Customer 22	22	1	0	300	5
Customer 23	23	1	0	300	5
Customer 24	24	2	0	300	5
Customer 25	25	1	0	300	5
Customer 26	26	1	0	300	5
Hub 1	27	0	0	540	5
Hub 2	28	0	0	540	5
Hub 3	29	0	0	540	5

Dapat dilihat pada tabel diatas, bahwa setiap *customer* memiliki *Time window* diantara *Time window 1* maupun *Time window 2*. Namun untuk node 0 (*Main Hub*/ *Depot*), node 27 hingga 29 (*Hub-Hub*) memiliki *Time window* yang dimulai dari jam 08.00-17.00 WIB.

4.1.5 Data Pendukung

Berikut merupakan data pendukung yang dimiliki Perusahaan untuk melakukan pengolahan data:

Tabel 4. 4 Data Pendukung

Jenis Kendaraan	*BlindVan*
Jumlah Kendaraan	4
Kapasitas Kendaraan	1000kg
Rasio konsumsi BBM	1:11 (1 liter menempuh 11 km)
Harga BBM	Rp10.000,00
Rata-rata kecepatan kendaraan	22 km/jam

4.1.6 Matriks Jarak

Matrix jarak menggambarkan tingkat kedekatan antar titik pada jaringan distribusi, mulai dari *Depot* dan *hub* utama hingga ke setiap pelanggan. Matriks jarak dihitung berdasarkan koordinat x dan y setiap *node* menggunakan rumus *Euclidean*, sehingga setiap elemen matriks (i,j)

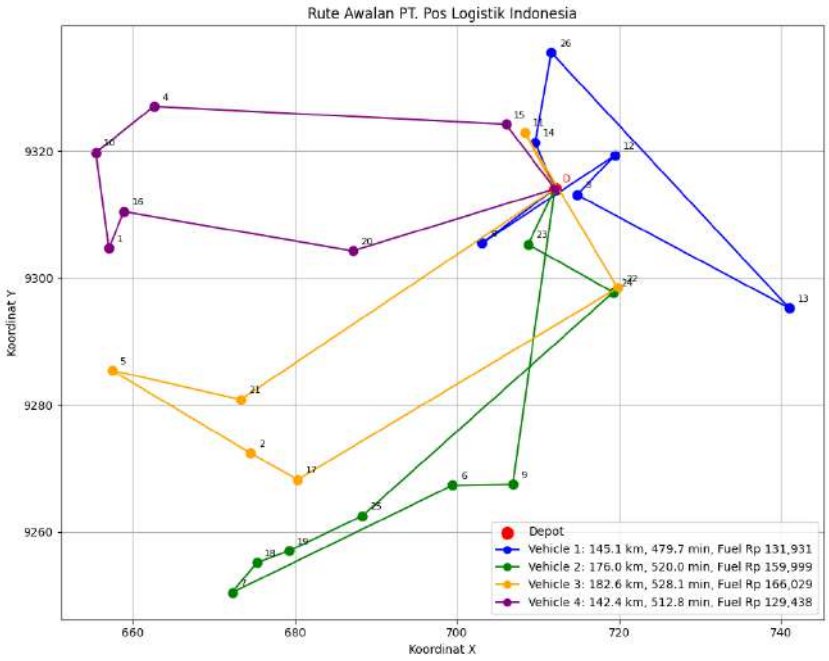
menunjukkan jarak terpendek antara node i dan node j dalam satuan kilometer. Penyajian matriks jarak sangat penting karena menjadi dasar perhitungan biaya, waktu tempuh, dan validasi *time windows* dalam model Ve*hicle Routing Problem with Pickup and Delivery* and *with Overflow Warehouse* (VRPPDTW-OW). Berikut merupakan Matriks jarak yang digunakan pada pengolahan data untuk rute usulan:

Tabel 4. 5 Matriks Jarak

Distance Matrix	0	1	2	3	4	5	6	7	8	9	10	n	27	28	29
0	**0,00**	55,79	56,06	2,88	51,14	61,65	48,37	75,01	12,39	46,85	56,89	...	9,82	17,18	13,21
1	55,79	**0,00**	36,70	58,33	23,02	19,25	56,50	56,36	46,04	62,19	15,13	...	47,76	39,12	42,58
2	56,06	36,70	**0,00**	57,25	55,91	21,49	25,46	22,09	43,71	32,74	51,02	...	53,59	46,90	46,33
3	2,88	58,33	57,25	**0,00**	54,01	63,67	48,26	75,72	13,96	46,31	59,69	...	12,69	19,96	15,79
4	51,14	23,02	55,91	54,01	**0,00**	41,91	70,16	77,22	45,87	74,22	10,21	...	41,43	34,58	39,55
5	61,65	19,25	21,49	63,67	41,91	**0,00**	45,76	38,04	49,86	52,61	34,34	...	55,87	47,49	49,21
6	48,37	56,50	25,46	48,26	70,16	45,76	**0,00**	32,00	38,32	7,43	68,42	...	50,59	47,54	44,37
7	75,01	56,36	22,09	75,72	77,22	38,04	32,00	**0,00**	63,08	38,56	71,32	...	74,04	68,08	66,87
8	12,39	46,04	43,71	13,96	45,87	49,86	38,32	63,08	**0,00**	38,20	49,72	...	12,32	12,30	7,42
9	46,85	62,19	32,74	46,31	74,22	52,61	7,43	38,56	38,20	**0,00**	73,32	...	50,48	48,68	44,92
10	56,89	15,13	51,02	59,69	10,21	34,34	68,42	71,32	49,72	73,32	**0,00**	...	9,82	17,18	13,21
n	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
27	9,82	47,76	53,59	12,69	41,43	55,87	50,59	74,04	12,32	50,48	47,57	...	**0,00**	8,69	7,26
28	17,18	39,12	46,90	19,96	34,58	47,49	47,54	68,08	12,30	48,68	39,75	...	8,69	**0,00**	5,11
29	13,21	42,58	46,33	15,79	39,55	49,21	44,37	66,87	7,42	44,92	44,33	...	7,26	5,11	**0,00**

4.1.7 Rute Awalan Perusahaan

Berikut merupakan Rute Awalan Pos Logistik Indonesia:



Gambar 4. 3 Rute Awalan Perusahaan

Berdasarkan Pengumpulan data dan juga gambar diatas, dapat diketahui bahwa rute awal yang dimiliki dan ditentukan Perusahaan sebagai berikut:

Tabel 4. 6 Hasil Rute Awal Perusahaan

Vehicle	***Route***	***Load***	***Travel Time (minutes)***	***Distance (km)***	***Fuel Used (liters)***	***Fuel Cost***
Vehicle-1	0-14-26-13-3-12-8-0	79	479.66	145.12	13.19	Rp 131,931,00
Vehicle-2	0-23-24-25-19-18-7-6-9-0	140	520.00	176.00	16.00	Rp 159,999,00
Vehicle-3	0-11-22-17-2-5-21-0	93	528.09	182.63	16.60	Rp 166,029,00
Vehicle-4	0-15-4-10-1-16-20-0	98	512.78	142.38	12.94	Rp 129,438,00
	Total		**2040.52**	**646, 13**		**Rp 587,397,00**

4.2 Pengolahan Data

4.2.1 Pengembangan Model Matematis

Dalam proses pengolahan data, Langkah pertama dalam melakukan optimasi rute menggunakan Bahasa pemrograman *python* yaitu dengan melakukan pengembangan model matematis berdasarkan permasalahan Perusahaan dan data yang sudah dikumpulkan. Pengembangan model matematis dilakukan untuk menentukan fungsi tujuan dan juga fungsi kendala. Adapun

model matematis yang sudah dikembangkan berdasarkan permasalahan yang ada yaitu sebagai berikut:

Fungsi Objektif:

$$\min \sum_{k \in v} \sum_{i \in N} \sum_{j \in N} Cij\ Xijk$$

Meminimalkan total biaya dan jarak yang ditempuh oleh seluruh kendaraan.

Constraint 1:

$$\sum_{j \in N \setminus \{start\}} Xstart, j = 1$$

kendaraan pasti meninggalkan node start (*Depot* atau *Hub*) persis satu kali, memastikan rute benar-benar dimulai di sana.

Constraint 2:

$$\sum_{j \in N \setminus \{finish\}} Xi, finish = 1$$

kendaraan pasti kembali dan tiba persis satu kali di *node finish*, memastikan rute diakhiri di titik yang benar.

Constraint 3:

$$\forall\, j\ \in C : \sum_{i \in N \setminus \{j\}} Xij = 1$$

setiap node j dalam himpunan C (*customer* atau *buffer-Hub*) harus dikunjungi tepat sekali oleh suatu kendaraan.

Constraint 4:

$$\forall\, h\ \in N\ \{start\} : \sum_{i \in N \setminus \{h\}} Xi, h = \sum_{j \in N \setminus \{h\}} Xh, j$$

Setiap node (kecuali *start*) yang dikunjungi harus "masuk" dan "keluar" sama jumlahnya, mencegah "rantai" yang terputus.

Constraint 5:

$$u\ start = 0$$

Menetapkan variabel urutan Ui pada node awal menjadi nol, sebagai dasar untuk *subtour-elimination*.

Constraint 6:

$$\forall\, i\,, j\ \in N\ \{start\}, i < j:\ \ ui - uj + |N|\ Xij \leq |N| - 1, uj - ui + |N|\ Xji \leq |N| - 1$$

Mencegah terbentuknya *subtour* (rute tertutup kecil) di luar rute utama, dengan memaksa urutan Ui konsisten sepanjang rute.

Constraint 7:

$$l_{start} = L^{init}\ dengan\ L^{init} = \sum_{i \in N \setminus \{start\}} demand_i$$

Kendaraan memulai dengan beban total berupa jumlah permintaan semua *node* yang akan dikunjungi (*customer* + *drop to Hub*), siap untuk didistribusikan.

Constraint 8:

$$\forall i\ \neq j\ \in N:\ l_j \geq\ l_i - demand_i - M\ (1 - xij)$$

jika xij =1 (kendaraan bergerak dari i ke j), maka

$$l_j = l_i - demand_i$$

mengurangi beban sebesar permintaan i. Jika xij=0, *constraint* dilemahkan oleh M.

Constraint 9:

$$\forall i\ \in N:\ l_i - Q$$

beban pada saat tiba di mana pun tidak boleh melebihi kapasitas maksimum Q.

z

4.2.2 Penetapan Notasi dan Parameter Model Matematis

Berdasarkan model matematis yang sudah dikembangkan dalam model VRPPDTW dengan *Exact method*, berikut merupakan notasi yang digunakan:

C	: Himpunan node yang wajib dikunjungi (*customer* + *buffer Hub*).
$Finish$	: Node akhir rute (biasanya sama dengan *Depot* atau *buffer Hub*).
l_i	: Beban kendaraan saat tiba di *node* i.
M	: "Big-M" (konstanta besar) untuk melemahkan *constraint* beban bila xij=0
N	: Himpunan semua *node* dalam *sub-problem* (*Depot, buffer Hub*, *customer*).
Q	: Kapasitas maksimum kendaraan.

d_{ij} : Jarak (atau biaya) dari *node* i ke node j.

$Demand_i$: Permintaan (*weight*) *customer* di node i.

$Start$: *Node* awal rute (*Depot* atau *buffer Hub*).

u_i : Variabel urutan kunjungan MTZ pada node i (mencegah *subtour*).

x_{ij} : Variabel biner:

$$xij = \begin{cases} 1, & jika\ kendaraan\ bergerak\ langsung\ dari\ i \rightarrow j \\ 0, & lainnya \end{cases}$$

berikut merupakan parameter yang digunakan:

$d_{ij} \geq 0$: jarak (atau biaya) dari node i ke j.

$demand_i \geq 0$: permintaan (berat) pada node i.

$Q > 0$: kapasitas maksimum kendaraan.

$M \gg 0$: "Big-M" untuk linearitas *constraint* beban.

4.2.3 Proses Algoritma *Exact*

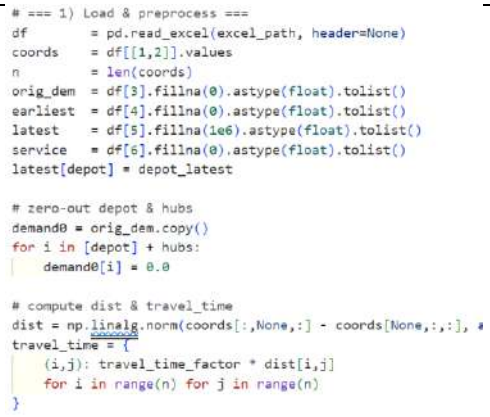
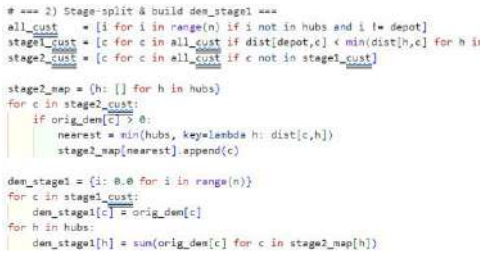
4.2.3.1 Tahap 1 (*Depot-Hub-cust-Depot*)

Tabel 4. 7 Proses Pemrograman *Python* Tahap 1

Tahapan	*Code*	Fungsi

1. *Import Library*	`from amplpy import AMPL, Environmen`
	`import pandas as pd`
	`import numpy as np`
	`import matplotlib.pyplot as plt`
		Langkah pertama yaitu mengimport empat *library* seperti amplpy (kelas AMPL dan *Environment*) digunakan untuk menyelesaikan model optimasi VRP dengan *solver* Gurobi, *pandas* digunakan untuk membaca file *Excel* dan mengubahnya menjadi struktur data yang mudah diolah, *numpy* untuk fungsi vektor dan matriks terutama untuk menghitung jarak *Euclidean* antar koordinat, dan *matplotlib.pyplot* dipakai untuk

Tahapan	*Code*	Fungsi
		menggambar peta, memplot titik-titik node, serta menampilkan rute yang dihasilkan secara grafis.
2. Inisialisasi Parameter & Data		Pada tahap ini kita hanya menyiapkan variabel input seperti *excel_path* menunjukkan file *Excel* berisi koordinat dan permintaan *customer*, *Depot* = 0 dan *Hub*s = [27,28,29] menandai titik *Depot* utama serta *buffer-warehouse*, speed = 22.0 km/jam (dengan *travel_time_factor* = 60/speed) mengonversi jarak menjadi waktu tempuh, dan km_per_liter = 11.0 bersama *fuel_price_per_liter* = 10 000 menetapkan efisiensi serta harga bahan bakar. Semua parameter ini akan dipakai untuk memuat data, menghitung jarak, dan pada akhirnya menghitung konsumsi serta biaya bensin dari rute optimal.

Tahapan	*Code*	Fungsi
3. *Load& Preprocessing Data*		Kode ini bertujuan menyiapkan semua data yang dibutuhkan untuk model VRP seperti membaca lokasi, permintaan, dan *time window* dari *file Excel*, mengosongkan permintaan untuk *Depot* dan *Hub*, lalu menghitung jarak *Euclidean* dan waktu tempuh antar node. Oleh karena itu, model optimasi dapat langsung menggunakan nilai-nilai jarak, waktu tempuh, dan permintaan yang sudah bersih dan terstruktur.
4. Pembagian Tahap dan Pembangunan Permintaan Tahap 1		Pada tahap ini, seluruh *customer* (*node*) yang bukan *Depot* maupun *Hub* dikelompokkan menjadi dua set: *Stage* 1 *customers* (mereka yang lebih dekat ke *Depot* dibandingkan ke *Hub*) dan *Stage 2 customers* (sisanya). Kemudian, setiap *customer Stage* 2 dipetakan ke *Hub* terdekatnya (membentuk *stage2_map*). Kode ini memisahkan *customer* menjadi dua level distribusi yang langsung dilayani dari *Depot* (*Stage 1*) dan yang diturunkan ke *Hub* terlebih dahulu (*Stage 2*) sehingga algoritma VRP dua tahap dapat menangani beban *Hub* secara akurat pada *Stage 1* dan

Tahapan	*Code*	Fungsi
		kemudian merinci rute akhir di *Stage 2*. Dengan demikian, kapasitas dan rute di setiap *Hub* dapat dioptimalkan secara efisien.
5. Definisi Model AMPL		Fungsi ini mendefinisikan dan menjalankan model optimasi VRP dengan *time windows* dan kapasitas di AMPL/Gurobi, Manfaatnya kita secara otomatis mendapatkan solusi rute minimum jarak yang memenuhi semua batasan (kunjungan sekali, kapasitas muatan, dan *time window*), tanpa perlu menulis *solver* dari nol.

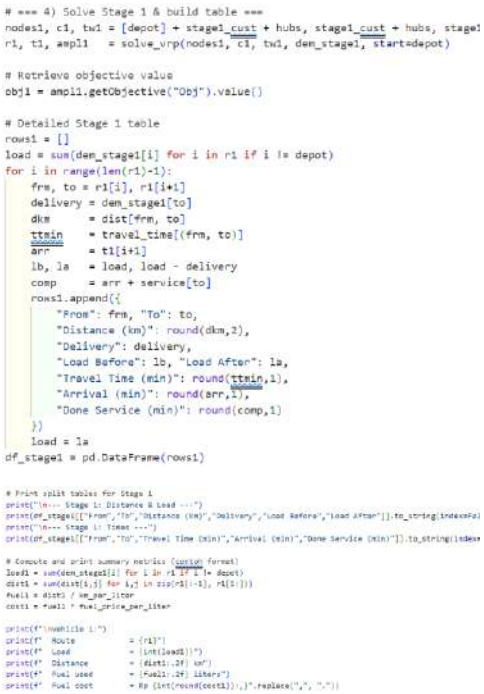
```
# === 3) AMPL model text ===
model_text = """
set N; set C within N; set TW within N;
param startNode integer; param Q>0; param M>0;
param d{i in N,j in N}>=0; param demand{i in N}>=0;
param e{i in N}>=0; param l{i in N}>=0;
param s{i in N}>=0; param tt{i in N,j in N}>=0;
var x{i in N,j in N} binary; var load{i in N}>=0;
var u{i in N}>=0,<=card(N)-1; var t{i in N}>=0;
minimize Obj: sum{i in N,j in N} d[i,j]*x[i,j];
subject to LeaveDepot:  sum{j in N diff {startNode}} x[startNode,j]
subject to ReturnDepot: sum{i in N diff {startNode}} x[i,startNode]
subject to VisitOnce{j in C}: sum{i in N diff {j}} x[i,j]=1;
subject to FlowCons{h in N diff {startNode}}:
    sum{i in N diff {h}} x[i,h] = sum{j in N diff {h}} x[h,j];
subject to UStart: u[startNode]=0;
subject to MTZ1{i in N diff {startNode},j in N diff {startNode}:i<j
    u[i]-u[j]+card(N)*x[i,j] <= card(N)-1;
subject to MTZ2{i in N diff {startNode},j in N diff {startNode}:i<j
    u[j]-u[i]+card(N)*x[j,i] <= card(N)-1;
subject to LoadFwd{i in N,j in N:i!=j}:
    load[j]>=load[i]-demand[i]-M*(1-x[i,j]);
subject to Cap{i in N}: load[i]<=Q;
subject to StartTime: t[startNode]=0;
subject to TW_Early{j in TW}: t[j]>=e[j];
subject to TW_Late {j in TW}: t[j]<=l[j];
subject to TimeFlow{i in N,j in N:i!=j and j!=startNode}:
    t[j]>=t[i]+s[i]+tt[i,j]-M*(1-x[i,j]);
"""

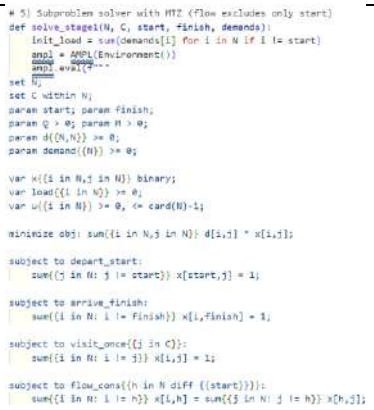
def solve_vrp(nodes, custs, tws, demands, start):
    Q0 = sum(demands[i] for i in nodes if i != start)
    ampl = AMPL(Environment())
    ampl.eval(model_text)
    ampl.getSet("N").setValues(nodes)
    ampl.getSet("C").setValues(custs)
    ampl.getSet("TW").setValues(tws)
    ampl.param["startNode"] = start
    ampl.param["Q"]         = Q0
    ampl.param["M"]         = M_big
    ampl.param["d"]         = {(i,j): dist[i,j]           for i in nodes for j in nodes}
    ampl.param["demand"]    = {i: demands[i]               for i in nodes}
    ampl.param["e"]         = {i: earliest[i]              for i in nodes}
    ampl.param["l"]         = {i: latest[i]                for i in nodes}
    ampl.param["s"]         = {i: service[i]              for i in nodes}
    ampl.param["tt"]        = {(i,j): travel_time[(i,j)]     for i in nodes for j in node
    ampl.setOption("solver","gurobi")
    ampl.setOption("TimeLimit",20)
    ampl.setOption("MIPGap","0.001")
    ampl.solve()

    x = ampl.getVariable("x").getValues().toDict()
    t = ampl.getVariable("t").getValues().toDict()

    route, times = [start], [0.0]
    cur = start
    while True:
        nxt = next((j for (i,j),v in x.items() if i==cur and v>0.5), None)
        if nxt is None or nxt == start:
            break
        route.append(nxt)
        times.append(t[nxt])
        cur = nxt

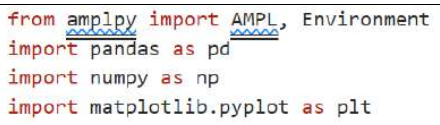
    back = times[-1] + service[cur] + travel_time[(cur, start)]
    if back > depot_latest:
        raise ValueError(f"Return at {back:.1f} > {depot_latest}")
    route.append(start)
    times.append(back)
    return route, times, ampl
```

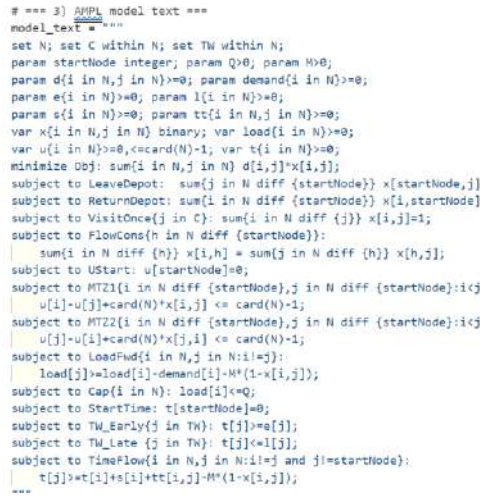
Tahapan	*Code*	Fungsi
6. Pemecahan Tahap 1 dan Pembuatan Tabel Detail		Pada tahap ini, kode memanggil fungsi *solve_vrp* dengan set node, *customer*, dan *time window* untuk Tahap 1 sehingga diperoleh rute optimal (r1) dan waktu kedatangan (t1). Nilai objective (total jarak) diambil dari objek AMPL (obj1). Selanjutnya, dibuat tabel rinci per segmen rute berisi asal tujuan, jarak, jumlah kiriman, muatan sebelum dan sesudah, waktu tempuh, waktu kedatangan, dan waktu selesai layanan. Tabel ini memudahkan analisis setiap langkah perjalanan. Terakhir, dihitung metrik ringkas total muatan, total jarak, bahan bakar terpakai, dan biaya lalu dicetak dalam format singkat seperti rute, muatan, jarak, konsumsi bahan bakar, dan biaya. Manfaatnya, dapat memperoleh baik detail per *node* maupun ringkasan keseluruhan rute Tahap 1 secara otomatis.

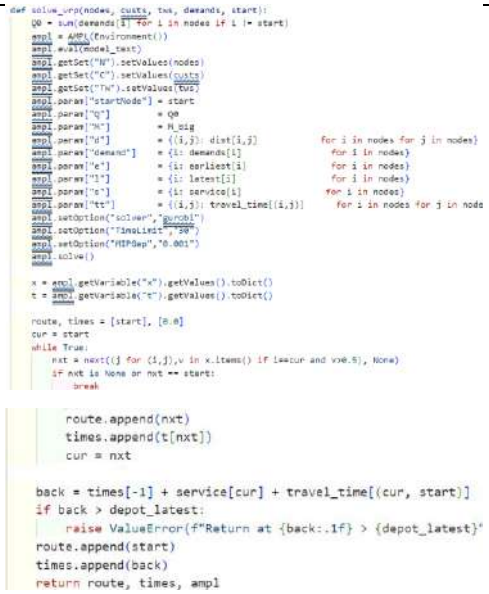
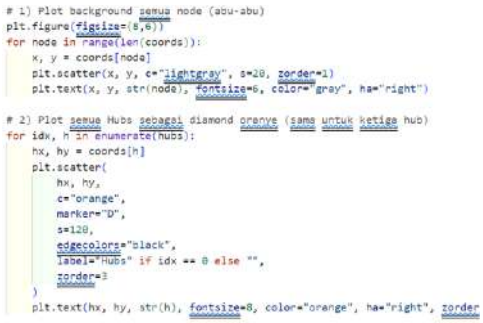
Tahapan	*Code*	Fungsi
7. Visualisasi Tahap 1 (Plot Rute)		Pada langkah ini kita menggunakan *matplotlib* untuk menggambarkan hasil rute hari pertama secara grafis. Manfaatnya adalah kita bisa dengan cepat melihat posisi semua titik, mengidentifikasi mana yang dilayani di Tahap 1, dan mengikuti visual secara langsung jalur perjalanan kendaraan, lengkap dengan label *node*.

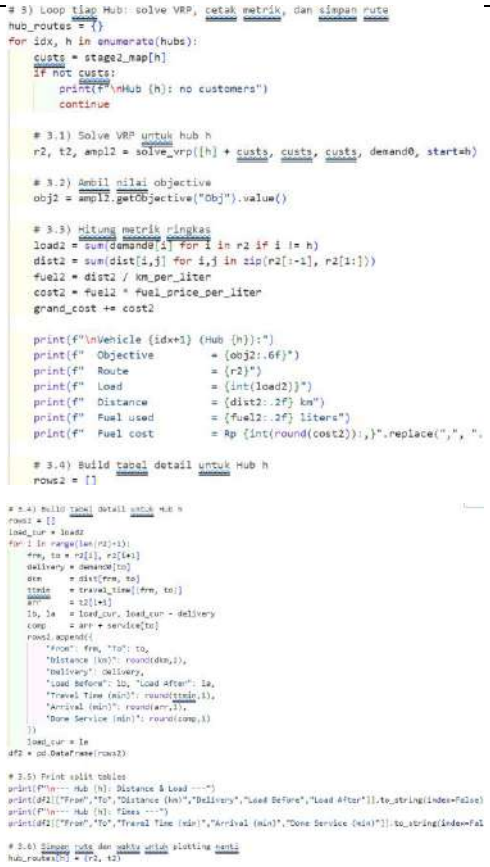
4.2.3.2 Tahap 2 (*Hub-cust-Hub*)

Tabel 4. 8 Proses Pemrograman *Python* Tahap 2

Tahapan	*Code*	Fungsi
1. Import Library		Langkah pertama yaitu mengimpor empat *library* seperti amplpy (kelas AMPL dan *Environment*) digunakan untuk menyelesaikan model optimasi VRP dengan *solver* Gurobi, *pandas* digunakan untuk membaca file *Excel* dan mengubahnya menjadi struktur data yang mudah diolah, *numpy* untuk fungsi vektor dan matriks terutama untuk menghitung jarak *Euclidean* antar koordinat, dan *matplotlib.pyplot* dipakai untuk menggambar peta, memplot titik-titik node, serta menampilkan rute yang dihasilkan secara grafis.

Tahapan	*Code*	Fungsi
2. Inisialisasi Parameter & Data		Pada tahap ini kita hanya menyiapkan variabel input seperti *excel_path* menunjukkan file *Excel* berisi koordinat dan permintaan *customer*, *Depot* = 0 dan *Hub*s = [27,28,29] menandai titik *Depot* utama serta *buffer-warehouse*, speed = 22.0 km/jam (dengan *travel_time_factor* = 60/speed) mengonversi jarak menjadi waktu tempuh, dan km_per_liter = 11.0 bersama *fuel_price_per_liter* = 10 000 menetapkan efisiensi serta harga bahan bakar. Semua parameter ini akan dipakai untuk memuat data, menghitung jarak, dan pada akhirnya menghitung konsumsi serta biaya bensin dari rute optimal.
3. Definisi Model AMPL		Fungsi ini mendefinisikan dan menjalankan model optimasi VRP dengan *time windows* dan kapasitas di AMPL/Gurobi, Manfaatnya kita secara otomatis mendapatkan solusi rute minimum jarak yang memenuhi semua batasan (kunjungan sekali, kapasitas muatan, dan *time window*), tanpa perlu menulis solver dari nol.

Tahapan	*Code*	Fungsi
		
4. Visualisasi *backgound* dan Penandaan *Hub*		Kode ini untuk memberikan *background* yang menampilkan semua titik. dalam warna abu-abu dengan label ID masing-masing, lalu menyorot titik *Hub* dengan bentuk *diamond* oranye dan label ID mereka. Manfaatnya, kita dapat melihat letak geografis semua *node* sekaligus menandai posisi *Hub* secara jelas sebelum menggambar rute membantu memastikan konteks spasial dan memudahkan pemahaman di mana *Hub* berada relatif terhadap *node* lain.

Tahapan	*Code*	Fungsi
5. Pemecahan Tahap 2 per *Hub* dan Pembuatan Tabel Detail		Kode ini menjalankan optimasi VRP terpisah untuk tiap *Hub* Stage 2, mencetak metrik ringkas (nilai *objective*, total jarak, rute, muatan, jarak, bahan bakar, dan biaya), lalu menyusun tabel detail per segmen (asal tujuan, jarak, muatan sebelum/sesudah, waktu tempuh, kedatangan, selesai layanan). Dan yang terakhir, rute dan waktu kedatangan tiap *Hub* disimpan di *Hub_routes* untuk *plotting*. Manfaatnya, kita memperoleh solusi optimal dan semua rincian operasional untuk setiap *Hub* secara terstruktur, sehingga memudahkan analisis dan visualisasi tahap kedua.
6. Visualisasi Gabungan Rute *Stage 2*		Kode ini menggabungkan dan menggambar rute semua *Hub* Stage 2 dalam satu plot. setiap rute digambarkan dengan garis putus-putus dan *marker* “o” menggunakan warna khusus (color_map[h]) untuk membedakan masing-masing *Hub*. Selain itu, setiap titik node di rute diberi label ID di atasnya. Manfaatnya adalah memvisualisasikan secara serentak keseluruhan perjalanan kendaraan dari semua *Hub*, sehingga kita dapat

Tahapan	*Code*	Fungsi
		dengan cepat melihat tumpang tindih, sebaran rute, dan struktur jaringan distribusi *Stage 2* dalam satu tampilan.
7. Menampilkan Total Biaya Bahan Bakar	`# 5) Cetak total biaya` `print(f"\nGrand Total Fuel Cost = Rp {int(round(grand_cost)):,}".replace(",", "."))`	Mencetak jumlah keseluruhan biaya bahan bakar (*Grand Total Fuel Cost*) dari semua rute *Stage 2*. Ini memberikan ringkasan cepat berapa total biaya operasional bahan bakar untuk mengukur efektivitas dan anggaran.

8. Finalisasi *plot* gabungan	`# 6) Finalisasi plot gabungan`
	`plt.title("Stage 2: All Hub Routes")`
	`plt.xlabel("X (km)"); plt.ylabel("Y (km)")`
	`plt.grid(True)`
	`plt.legend(fontsize="small", loc="upper left")`
	`plt.tight_layout()`
	`plt.show()`
		Menambahkan judul, label sumbu, dan *grid*, pada plot gabungan semua rute Stage 2, lalu menampilkan gambar. Dengan begitu, visualisasi rute menjadi jelas dan informatif sebelum ditutup.

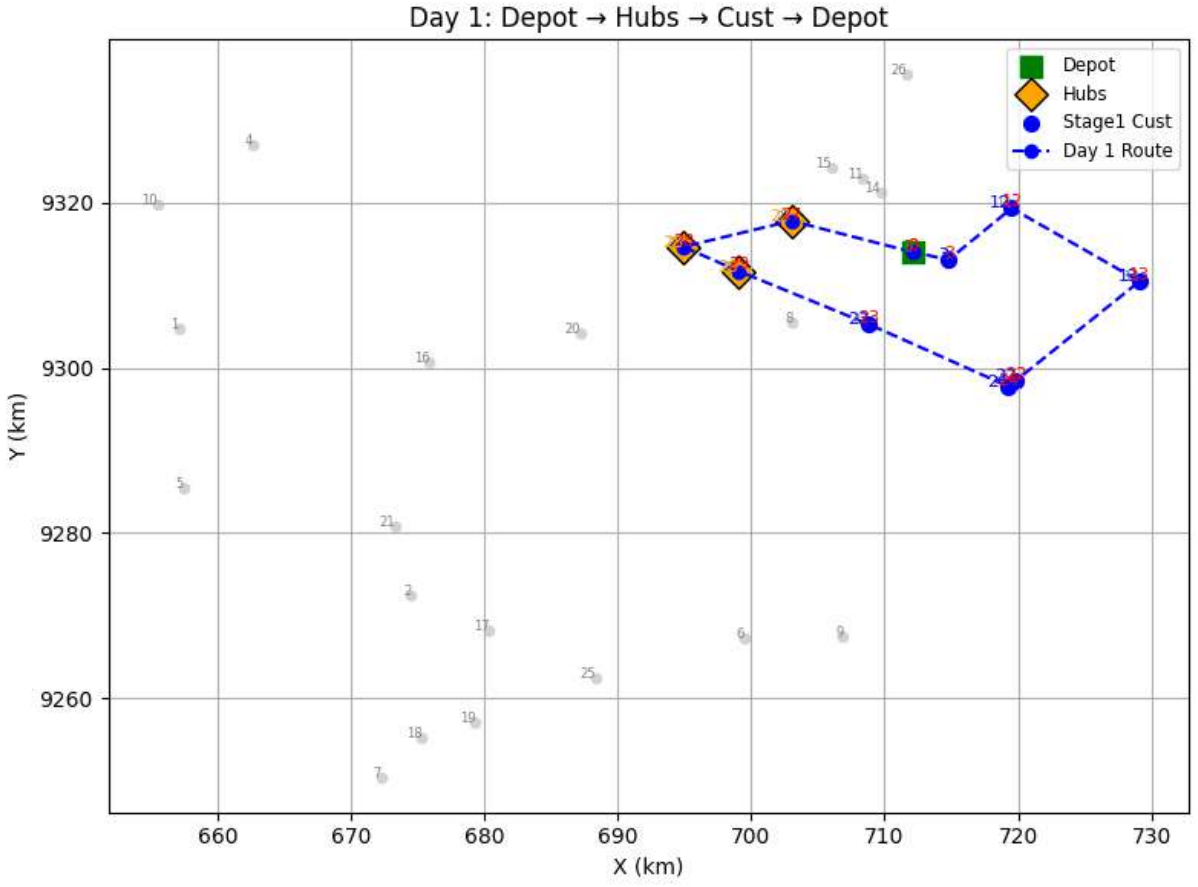
9. Visualisasi Rute *Stage 2* per *Hub*	`# 7) Plot terpisah untuk setiap hub (pakai warna yang sama)`
	`for h in hubs:`
	`  if h not in hub_routes:`
	`  continue`
	`  r2, t2 = hub_routes[h]`
	`  hub_color = color_map[h]`
	`  plt.figure(figsize=(8,6))`
	`  # 7.1) Background semua node`
	`  for node in range(len(coords)):`
	`  x, y = coords[node]`
	`  plt.scatter(x, y, c="lightgray", s=20, zorder=1)`
	`  plt.text(x, y, str(node), fontsize=6, color="gray", ha="right", zorder=2)`
	`  # 7.2) Highlight hub h (diamond oranye)`
	`  hx, hy = coords[h]`
	`  plt.scatter(`
	`  hx, hy,`
	`  c="orange",`
	`  marker="D",`
	`  s=150,`
	`  edgecolors="black",`
	`  label=f"Hub {h}",`
	`  zorder=4`
	`  )`
	`  plt.text(hx, hy, str(h), fontsize=8, color="orange", ha="right", zorder=5)`
	`  # 7.3) Highlight pelanggan Stage 2 untuk hub h (titik biru/oranye/hijau sesuai`
	`  for ci, c in enumerate(stage2_map[h]):`
	`  cx, cy = coords[c]`
		Membuat plot terpisah untuk setiap *Hub*, Dengan menampilkan satu per satu plot seperti ini, kita dapat menganalisis rute masing-masing *Hub* secara terpisah, memastikan setiap jalur sudah benar, sekaligus presentasi visual yang lebih mudah dilihat.

Tahapan	*Code*	Fungsi

	`# 7.3) Highlight pelanggan Stage 2 untuk hub h (titik biru/oranye/hijau sesuai map)`
	`for ci, c in enumerate(stage2_map[h]):`
	`  cx, cy = coords[c]`
	`  plt.scatter(cx, cy, c="blue", s=50, label="Stage2 Cust" if ci == 0 else "", zorder`
	`  plt.text(cx, cy, str(c), fontsize=8, color="blue", ha="right", zorder=5)`
	
	`# 7.4) Gambar rute hub h`
	`pts2 = coords[r2]`
	`plt.plot(`
	`  pts2[:,0], pts2[:,1],`
	`  linestyle="--",`
	`  marker="s",`
	`  color=hub_color,`
	`  label=f"Route Hub {h}",`
	`  zorder=6`
	`)`
	`for node in r2:`
	`  x, y = coords[node]`
	`  plt.text(x, y + 0.2, str(node), fontsize=8, color=hub_color, ha="center", zorder=7`
	
	`# 7.5) Finalisasi plot per-hub`
	`plt.title(f"Stage 2 Route: Hub {h}")`
	`plt.xlabel("X (km)"); plt.ylabel("Y (km)")`
	`plt.grid(True)`
	`plt.legend(fontsize="small", loc="upper left")`
	`plt.tight_layout()`
	`plt.show()`	

4.2.4 Hasil Optimasi

4.2.4.1 Hasil Optimasi Tahap 1

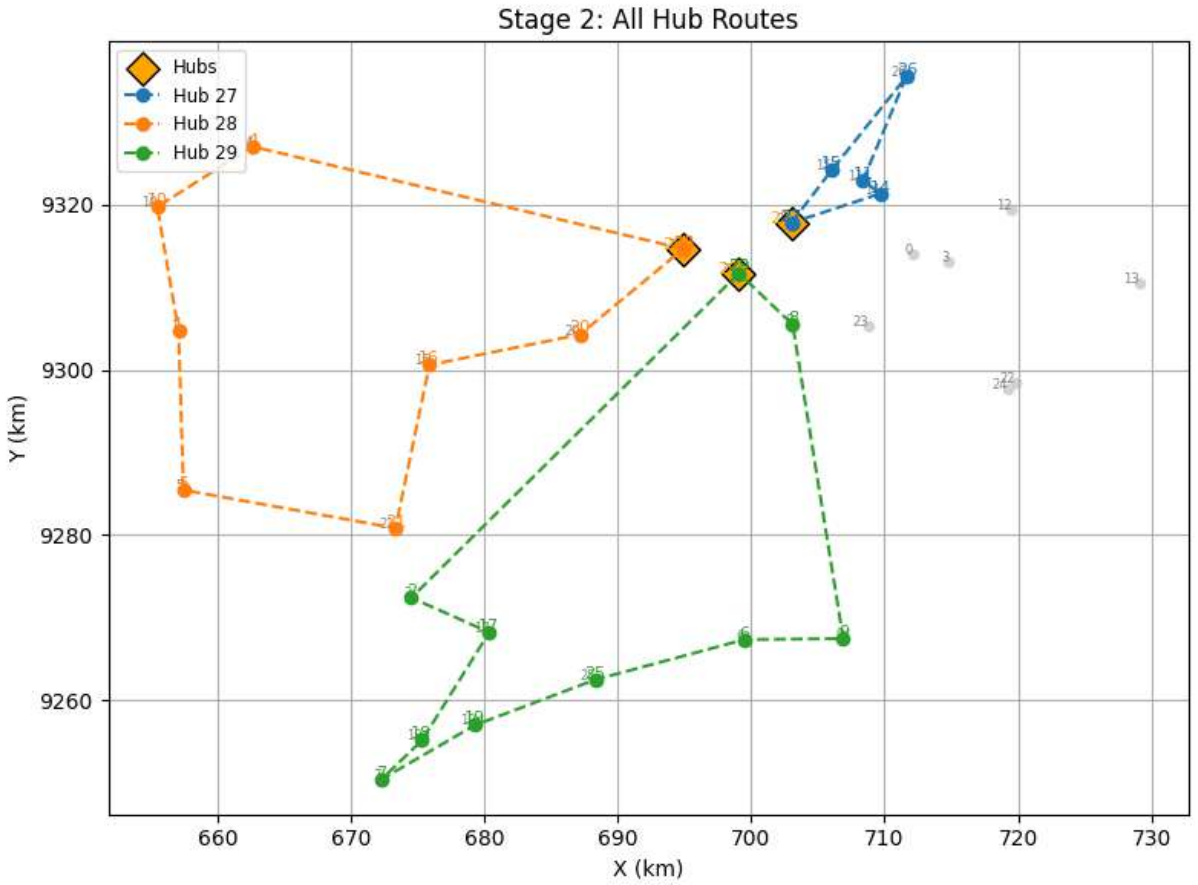


Gambar 4. 4 Hasil Optimasi rute Tahap 1

Tabel 4. 9 Hasil Optimasi Rute Tahap 1

Vehicle	*Route*	*Load*	*Distance*	*Fuel Used*	*Fuel Cost*
Vehicle-1	0-27-28-29-23-24-22-13-12-3-0	410	88.01 km	8.00 liters	Rp 80.010,00
	Grand Total Fuel Cost Day 1				Rp 80.010,00

4.2.4.2 Hasil Optimasi Tahap 2



Gambar 4. 5 Hasil Optimasi Rute Tahap 2

Tabel 4. 10 Hasil Optimasi Rute Tahap 2

Vehicle	*Route*	*Load*	*Distance*	*Fuel Used*	*Fuel Cost*
Vehicle-1	27-15-26-11-14-27	52	42.28 km	3.84 liters	Rp 38.436,00
Vehicle-2	28-4-10-1-5-21-16-20-28	115	140.45 km	12.77 liters	Rp 127.684,00
Vehicle-3	29-2-17-18-7-19-25-6-9-8-29	159	158.49 km	14.41 liters	Rp 144.081,00
	Grand Total Fuel Cost Day 2				**Rp 310.201,00**

BAB V
PEMBAHASAN

5.1 Analisis Hasil Optimasi Rute

Strategi yang direkomendasikan dalam penelitian ini adalah melakukan perubahan dan optimasi rute distribusi menjadi dua hari pengiriman, yang disusun secara sistematis agar lebih efisien namun tetap memenuhi seluruh batasan operasional yang berlaku. Pada hari pertama, rute yang dilalui dimulai dari depot, kemudian menuju ke gudang *buffer* (hub), dilanjutkan dengan melayani pelanggan pada tahap pertama (*customer stage 1*), dan kembali ke depot. Strategi ini memungkinkan pengumpulan barang dari gudang cadangan untuk mempermudah pengiriman dan mengurangi beban langsung dari depot.

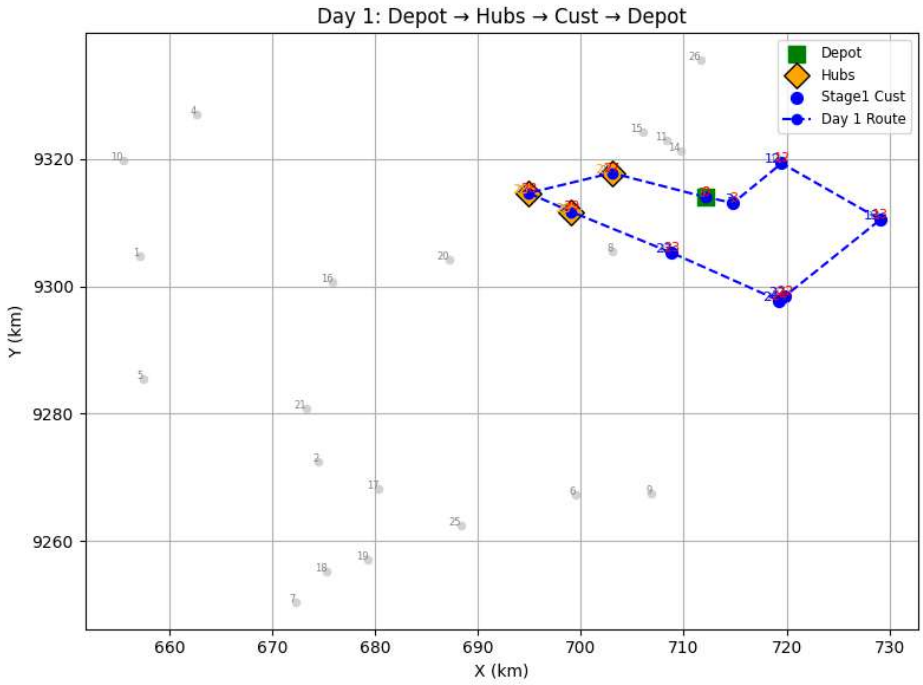
Selanjutnya, pada hari kedua, rute dimulai dari hub, dilanjutkan dengan pengiriman ke pelanggan pada tahap kedua (*customer stage 2*), dan kembali lagi ke hub. Skema dua hari ini dirancang untuk membagi beban pengiriman secara lebih merata dan terstruktur, sekaligus memaksimalkan penggunaan *overflow warehouse* (gudang *buffer*) sebagai titik antara dalam proses distribusi. Pendekatan ini tidak hanya menurunkan total jarak tempuh dan waktu perjalanan, tetapi juga memberikan fleksibilitas lebih dalam pengelolaan kapasitas kendaraan dan pemenuhan *time windows* pelanggan. Dengan pembagian rute menjadi dua hari dan pemanfaatan *hub* sebagai titik transit strategis, perusahaan dapat meningkatkan efisiensi distribusi tanpa mengorbankan kualitas layanan atau melanggar batasan teknis sistem yang sudah ditetapkan.

Strategi dua hari distribusi tidak menyebabkan pengangguran tenaga kerja, melainkan menyeimbangkan beban kerja secara lebih efisien. Dengan membagi pengiriman menjadi dua hari, beban yang sebelumnya sangat padat dalam satu hari dapat disebar ke dua hari operasional. Hal ini mencegah kelelahan pekerja, memperkecil risiko keterlambatan, dan memberikan ruang bagi pekerja untuk menjalankan aktivitas dengan lebih terencana.

Hub dapat berfungsi sebagai *buffer warehouse* yang penting karena dapat untuk menerima dan menyimpan sementara barang dari Main Hub. PT Pos Logistik Indonesia dapat memanfaatkan hub dengan berperan sebagai titik konsolidasi dan transit, menjadikan hub sebagai buffer yang efektif terutama saat volume kiriman tinggi. Penelitian ini memilih 3 Hub karena masing-masing Hub berada di lokasi strategis yang dapat menjangkau area Jakarta Pusat dengan baik. Pemilihan ini disesuaikan dengan pola distribusi dan kebutuhan operasional PT

Pos Logistik Indonesia di wilayah Jakarta Pusat, yang memiliki tingkat kepadatan dan intensitas pengiriman tinggi. Berikut merupakan analisis hasil tahap 1 dan juga tahap 2:

5.1.1 Analisis Hasil Tahap 1 (*Depot*-*Hub*-cust-*Depot*)



Gambar 5. 1 Rute Usulan Tahap 1

Tahap pertama dirancang untuk memecah beban distribusi besar langsung dari *Depot* menjadi dua tugas yaitu menyalurkan sejumlah besar barang ke titik-titik perantara (*buffer warehouse*) dan melayani *customer* yang paling efisien diakses langsung dari *Depot*. Tanpa tahap ini kendaraan harus mengantarkan beban penuh sekaligus ke semua *customer* baik yang dekat maupun yang jauh dari *Depot* yang seringkali menyebabkan rute panjang risiko melampaui kapasitas dan tingginya biaya bahan bakar. Dengan melakukan "*drop to Hub*" ke *Hub* 27, 28, dan 29, *Depot* mentransfer sebagian besar permintaan *customer* yang jauh ke titik-titik perantara tersebut sambil memegang sisanya untuk *customer Stage 1* (day-1) yang paling dekat. Cara seperti ini dapat menghasilkan kapasitas sisa kendaraan yang tidak terjadi *overloading* dan kunjungan ke *customer* Stage 1 memiliki *loop* singkat setelah "*drop to Hub*" meminimalkan jarak bolak-balik ke *Depot*.

Tabel 5. 1 Hasil Rute Tahap 1

Vehicle	***Route***	***Load***	***Distance***	***Fuel Used***	***Fuel Cost***
Vehicle-1	0-27-28-29-23-24-22-13-12-3-0	410	88.01 km	8.00 liters	Rp 80.010,00

Grand Total Fuel Cost Day 1	Rp 80.010,00

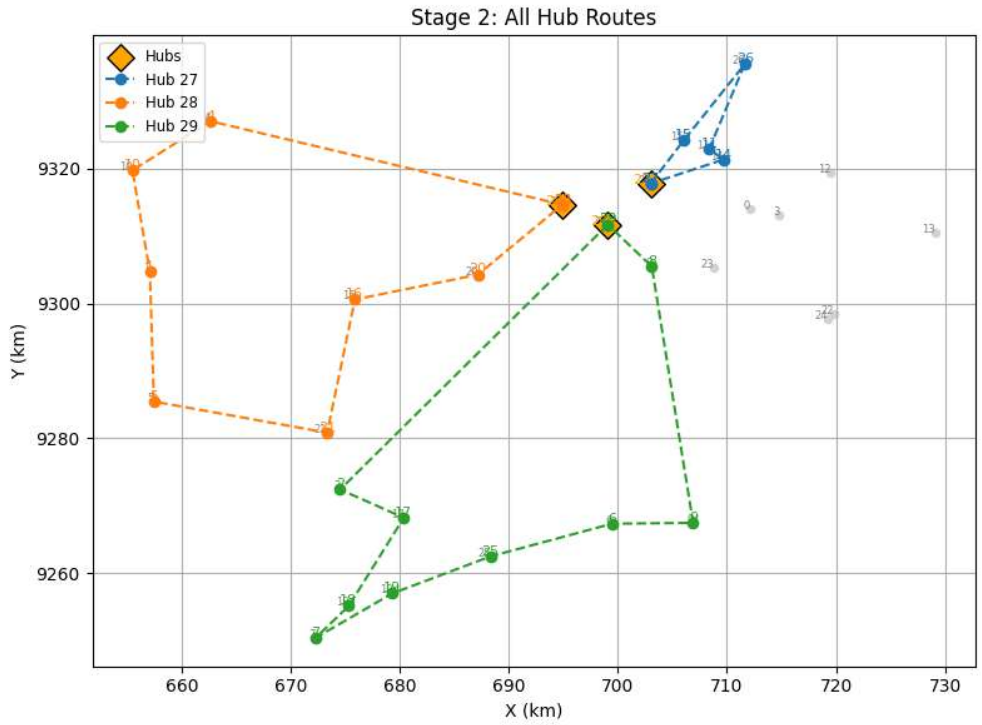
Berdasarkan *output* yang didapatkan, berikut merupakan detail jarak tempuh dan *load* antar *node* di tahap-1:

Tabel 5. 2 Rute per-*node* Tahap 1

From -> To	*Dist (km)*	*Delivery*	*Load before*	*Load After*
0-27	9.82 km	52	410	358
27-28	8.69 km	115	358	243
28-29	5.11 km	159	243	84
29-23	11.64 km	22	84	62
23-24	12.92 km	27	62	35
24-22	0.87 km	3	35	32
22-13	15.22 km	19	32	13
13-12	13.04 km	8	13	5
12-3	7.82 km	5	5	0
3-0	2.88 km	0	0	0

Berdasarkan tabel diatas, Pada Hari 1, diketahui sebagian kapasitas dari *Depot* utama ke tiga *buffer warehouse* (node 27, 28, 29) terlihat sangat efektif. Kendaraan berangkat dari *Depot* membawa total kapasitas 410 kg lalu menyerahkan masing-masing 52 kg, 115 kg, dan 159 kg ke *Hub* 27, 28, dan 29 sebelum menunaikan pengantaran langsung ke beberapa *customer* yang lokasinya paling dekat dengan *Depot* seperti *node* 23, 24, 22, 3, 12 dan 3. Jalur rutenya [0-27-28-29-23-24-22-13-12-3-0] mengoptimalkan urutan kunjungan berdasar jarak dan kapasitas kendaraan menghasilkan total perjalanan 88.01 km dengan menggunakan 8 liters bahan bakar dan menghabiskan biaya Rp 80.010. Dengan menyelesaikan sebagian pengantaran *customer* dekat *Depot* sekaligus melakukan "*drop to Hub*", perusahaan berhasil menekan beban operasional Hari 2 sekaligus memanfaatkan kapasitas kendaraan untuk menjangkau titik-titik *Hub*.

5.1.2 Analisis Hasil Tahap 2 (*Hub*-cust-*Hub*)



Gambar 5. 2 Rute Usulan Tahap 2

Meskipun *Depot* sudah menyalurkan barang ke *buffer warehouse* pada tahap pertama, *customer* yang lebih jauh tetap harus dijangkau. Tahap kedua memfokuskan pengiriman dari setiap *Hub* ke kelompok *customer Stage 2* di sekitarnya. Kendaraan akan menghadapi rute panjang yang menggabungkan *drop to Hub* dan *direct delivery* ke *customer* jauh dalam satu kali perjalanan yang dilakukan untuk menghindari *subtour*, *overcapacity*, dan biaya bahan bakar yang tinggi. Dengan memisahkan keesokan harinya dan memulai rute langsung di masing-masing *Hub* (tanpa perlu kembali ke *Depot*) kendaraan dapat mengambil beban yang lebih kecil dan merutekan dengan efisiensi terbaik.

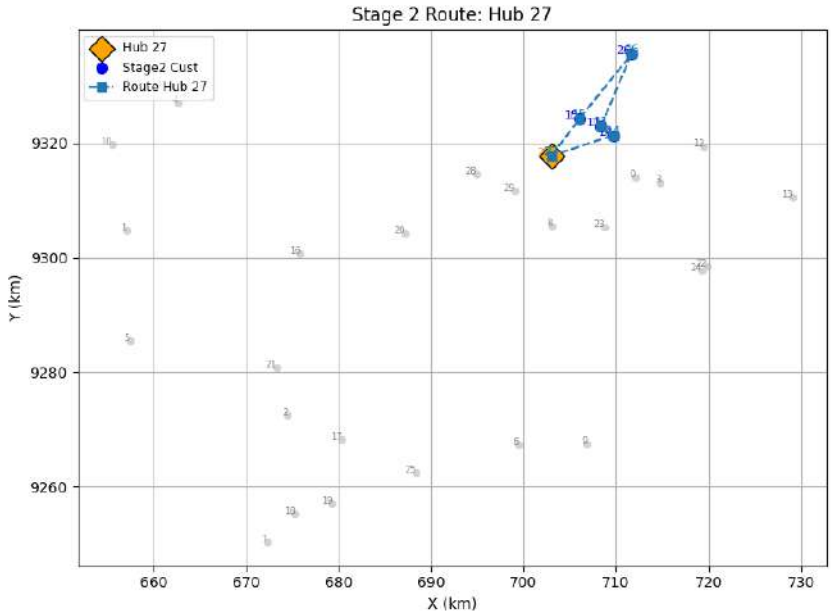
Tabel 5. 3 Hasil Rute Tahap 2

Vehicle	***Route***	***Load***	***Distance***	***Fuel Used***	***Fuel Cost***
Vehicle-1	27-15-26-11-14-27	52	42.28 km	3.84 liters	Rp 38.436,00
Vehicle-2	28-4-10-1-5-21-16-20-28	115	140.45 km	12.77 liters	Rp 127.684,00
Vehicle-3	29-2-17-18-7-19-25-6-9-8-29	159	158.49 km	14.41 liters	Rp 144.081,00
	Grand Total Fuel Cost Day 2				**Rp 310.201,00**

Contohnya *Hub* 27 menempuh 42.28 km dengan biaya Rp 38.436,00, sedangkan *Hub* 28 dan *Hub* 29 masing-masing menempuh 140.45 km dengan biaya Rp 127.684,00 dan 158.49 km

dengan biaya Rp 144.081,00. Strategi dua tahap ini memastikan bahwa setiap kendaraan hanya menangani *cluster local*, mengurangi jarak tempuh gabungan, mencegah *subtour*, dan pada akhirnya menekan total biaya bahan bakar. Berdasarkan *output* yang didapatkan, berikut merupakan detail jarak tempuh dan *load* antar *node* di tahap 2:

1. *Vehicle* 1



Gambar 5. 3 Rute kendaraan *Hub* 27

Tabel 5. 4 Rute per-*node Hub* 27

From -> To	*Dist (km)*	*Delivery*	*Load before*	*Load After*
27-15	7.09 km	14	52	38
15-26	12.60 km	1	38	37
26 -11	12.97 km	20	37	17
11-14	2.07 km	17	17	0
14 -27	7.55 km	0	0	0

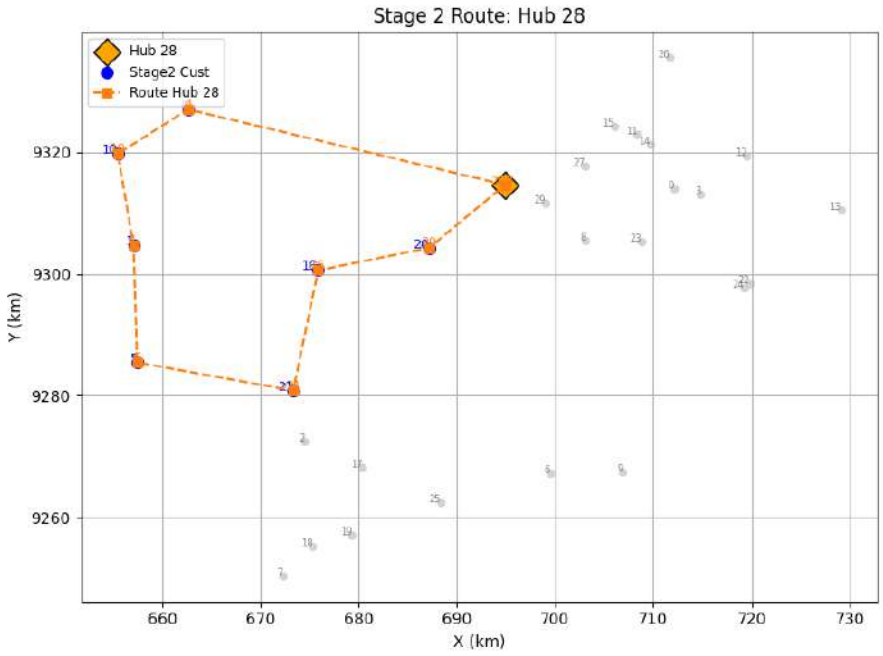
Pada Tahap 2, *Vehicle* 1 memulai perjalanan dari *Buffer warehouse* 27 dengan muatan awal 52 kg dan melakukan urutan pengantaran sebagai berikut:

- 27-15 (7,09 km): Kendaraan menempuh 7,09 km ke *customer* 15, menyerahkan 14 kg. Kapasitas turun dari 52 kg menjadi 38 kg.
- 15-26 (12,60 km): perjalanan sejauh sejauh 12,60 km menuju *customer* 26, menyerahkan 1 kg. Kapasitas berubah dari 38 kg menjadi 37 kg.

- 26-11 (12,97 km): Melanjutkan 12,97 km ke *customer* 11, menyerahkan 20 kg. Kapasitas turun dari 37 kg menjadi 17 kg.
- 11-14 (2,07 km): Menempuh jarak 2,07 km ke *customer* 14, menyerahkan 17 kg. Kapasitas dari 17 kg menjadi 0 kg.
- 14-27 (7,55 km): Akhirnya, kendaraan kembali ke *Hub* 27 sejauh 7,55 km dengan muatan 0 kg.

Secara total, *Vehicle* 1 menempuh 42,28 km dan mengantarkan 52 kg barang. Dengan pola "*drop to Hub*" pada Tahap 1, muatan di Tahap 2 relatif ringan sehingga rute menjadi lebih ringkas, mengurangi waktu tempuh dan konsumsi bahan bakar, sekaligus memastikan setiap *customer* dilayani tepat sekali oleh *Hub* terdekat.

2. *Vehicle* 2



Gambar 5. 4 Rute kendaraan *Hub* 28

Tabel 5. 5 Rute per-*node Hub* 28

From -> To	*Dist (km)*	*Delivery*	*Load before*	*Load After*
28-4	34.58 km	24	115	91
4-10	10.21 km	23	91	68
10-1	15.13 km	4	68	64
1-5	19.25 km	28	64	36
5-21	16.55 km	3	36	33

21-16	19.93 km	21	33	12
16-20	11.86 km	12	12	0
20-28	12.95 km	0	0	0

Pada Tahap 2, *Vehicle* 2 memulai dari *hub* 28 dengan kapasitas awal 115 kg dan menjalani rangkaian pengantaran berikut:

• 28-4 (34,58 km): Kendaraan menempuh 34,58 km untuk mencapai *customer* 4 dan menyerahkan 24 kg. Kapasitas turun dari 115 kg menjadi 91 kg.

• 4-10 (10,21 km): Perjalanan sejauh 10,21 km dilanjutkan ke *customer* 10, di mana 23 kg diserahkan. Kapasitas berkurang dari 91 kg menjadi 68 kg.

• 10-1 (15,13 km): Setelah menempuh 15,13 km, kendaraan sampai di *customer* 1 dan mengirim 4 kg, menurunkan kapasitas dari 68 kg menjadi 64 kg.

• 1-5 (19,25 km): Jarak 19,25 km ditempuh menuju *customer* 5 untuk menyerahkan 28 kg, sehingga kapasitas menurun dari 64 kg menjadi 36 kg.

• 5-21 (16,55 km): Kendaraan menempuh 16,55 km untuk memenuhi permintaan *customer* 21 sebesar 3 kg, menurunkan kapasitas dari 36 kg menjadi 33 kg.

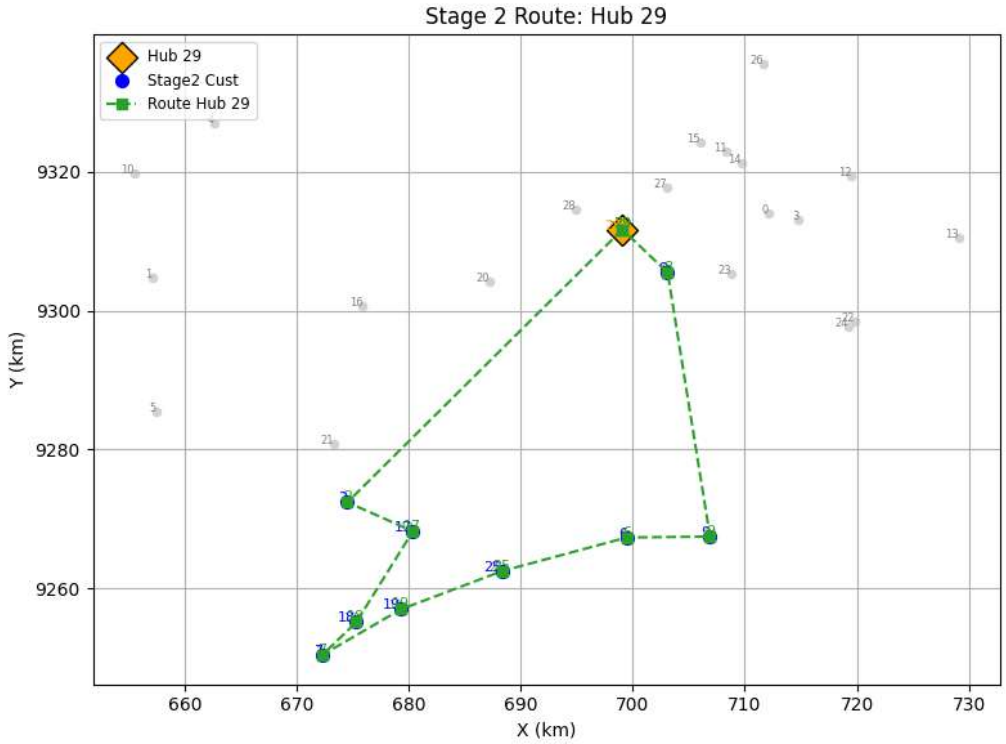
• 21-16 (19,93 km): Perjalanan sejauh 19,93 km menuju *customer* 16, di mana 21 kg di-*drop*, menyisakan 12 kg.

• 16-20 (11,86 km): Setelah menempuh 11,86 km, kendaraan sampai di *customer* 20 dan menyerahkan 12 kg terakhir, membuat kapasitas menjadi 0 kg.

• 20-28 (12,95 km): Terakhir, kendaraan kembali ke *Hub* 28 sejauh 12,95 km tanpa sisa muatan.

Dengan total jarak 136,71 km, *Vehicle* 2 berhasil mengantar 115 kg ke tujuh *customer*. Pola rute ini menunjukkan efisiensi *clustering* dengan memanfaatkan *Hub* sebagai titik awal dan akhir, jarak tempuh keseluruhan dapat diminimalkan sambil menjaga beban kendaraan sesuai kapasitas.

3. *Vehicle* 3



Gambar 5. 5 Rute kendaraan *Hub* 29

Tabel 5. 6 Rute per-*node Hub* 29

From -> To	*Dist (km)*	*Delivery*	*Load before*	*Load After*
29-2	46.33 km	18	159	141
2-17	7.19 km	21	141	120
17-18	14.03 km	2	120	118
18-7	5.59 km	9	118	109
7-19	9.61 km	21	109	88
19-25	10.54 km	16	88	72
25-6	12.16 km	22	72	50
6-9	7.43 km	21	50	29
9-8	38.20 km	29	29	0
8-29	7.42 km	0	0	0

Berikut adalah uraian rinci perjalanan *Vehicle* 3 pada Tahap 2- mulai dan berakhir di *Buffer warehouse* 29:

- 29-2 (46,33 km): Kendaraan menempuh jarak terjauh di rute ini, yakni 46,33 km, untuk mencapai *customer* 2. Dari kapasitas awal 159 kg, sebanyak 18 kg diserahkan sehingga sisa kapasitas turun menjadi 141 kg.
- 2-17 (7,19 km): Perjalanan sejauh 7,19 km dilanjutkan ke *customer* 17, dimana 21 kg diserahkan. Kapasitas berkurang dari 141 kg menjadi 120 kg.
- 17-18 (14,03 km): jarak 14,03 km menuju *customer* 18, kendaraan hanya men-*drop* 2 kg kapasitas menjadi 118 kg.
- 18-7 (5,59 km): Dengan jarak 5,59 km, 9 kg diantarkan ke *customer* 7, mengurangi kapasitas menjadi 109 kg.
- 7-19 (9,61 km): Kendaraan menempuh 9,61 km untuk *customer* 19 dan menyerahkan 21 kg, sehingga sisa kapasitas 88 kg.
- 19-25 (10,54 km): Pada jarak 10,54 km berikutnya, 16 kg terakhir di atas 72 kg diserahkan, menurunkan kapasitas menjadi 72 kg.
- 25-6 (12,16 km): Perjalanan 12,16 km dilanjutkan ke *customer* 6, dimana 22 kg diserahkan, sisa kapasitas yaitu 50 kg.
- 6-9 (7,43 km): Perjalanan sejauh 7,43 km, kendaraan menyerahkan 21 kg ke *customer* 9, menyisakan 29 kg.
- 9-8 (38,20 km): jarak 38,20 km ditempuh menuju *customer* 8, dan seluruh kapasitas sebesar 29 kg diserahkan, kapasitas menjadi 0 kg.
- 8-29 (7,42 km): Dan yang terakhir, kendaraan kembali ke *Hub* 29 sejauh 7,42 km tanpa kapasitas.

Secara keseluruhan, *Vehicle* 3 menempuh 158.49 km untuk melayani sembilan *customer* di sekitar *Buffer warehouse* dan *Hub* 29 menyalurkan 159 kg barang. Rute ini menyeimbangkan *demand delivery customer* dengan urutan *drop-off* dan mengurangi jarak kosong sekaligus memastikan beban kendaraan selalu dalam batas kapasitas.

5.1.3 Analisis Waktu Tempuh

Analisis waktu tempuh setelah melakukan optimasi VRPPDTW menggunakan metode *Exact* sangat berperan sangat penting dalam memastikan bahwa solusi rute yang dihasilkan tidak hanya teroptimal dari segi jarak dan biaya bahan bakar, tetapi juga layak secara operasional. Dengan menghitung durasi perjalanan antara *node* termasuk waktu tempuh, dapat melengkapi optimasi jarak dan biaya dengan aspek keberlanjutan operasional, memastikan rute yang

dihasilkan benar-benar dapat dijalankan di lapangan sesuai target waktu dan regulasi. Berikut merupakan *detail* waktu tempuh yang dilakukan setiap kendaraan:

1. *Vehicle 1* (Tahap 1)

Tabel 5. 7 Waktu tempuh per-*node* Tahap 1

From -> To	*Travel Time (min)*	*Arrival (min)*	*Done Service (min)*
0-27	26.8	31.8	36.8
27-28	23.7	60.5	65.5
28-29	13.9	79.4	84.4
29-23	31.7	116.2	121.2
23-24	35.2	156.4	161.4
24-22	2.4	163.8	168.8
22-13	41.5	210.3	215.3
13-12	35.6	301.0	306.0
12-3	21.3	327.3	332.3
3-0	7.8	340.2	345.2

2. *Vehicle 1* / *Hub* 27 (Tahap 2)

Tabel 5. 8 Waktu tempuh per-*node Hub* 27

From -> To	*Travel Time (min)*	*Arrival (min)*	*Done Service (min)*
27-15	19.3	24.3	29.3
15-26	34.4	63.7	68.7
26 -11	35.4	104.1	109.1
11-14	5.6	114.7	119.7
14 -27	20.6	140.3	145.3

3. *Vehicle 2 / Hub 28* (Tahap 2)

Tabel 5. 9 Waktu tempuh per-*node Hub* 28

From -> To	*Travel Time (min)*	*Arrival (min)*	*Done Service (min)*
28-4	94.3	99.3	104.3
4-10	27.9	132.2	137.2

10-1	41.3	178.4	183.4
1-5	52.5	235.9	240.9
5-21	45.1	301.0	306.0
21-16	54.3	360.3	365.3
16-20	32.4	397.7	402.7
20-28	35.3	438.0	443.0

4. *Vehicle 3* / *Hub* 29 (Tahap 2)

Tabel 5. 10 Waktu tempuh per-*node Hub* 29

From -> To	*Travel Time (min)*	*Arrival (min)*	*Done Service (min)*
29-2	126.4	131.4	136.4
2-17	19.6	156.0	161.0
17-18	38.3	199.2	204.2
18-7	15.2	219.5	224.5
7-19	26.2	250.7	255.7
19-25	28.7	284.4	289.4
25-6	33.2	322.6	327.6
6-9	20.3	347.8	352.8
9-8	104.2	457.0	462.0
8-29	20.2	482.2	487.2

Secara umum, sebagian besar rute seperti Tahap 1 dan tahap 2 mengoptimalkan waktu dengan menerapkan durasi layanan yang sama sekitar 5 menit per *node*, sehingga kendaraan langsung bergerak ke titik berikutnya tanpa jeda. Namun pada *Hub* 28 dan Tahap 1 muncul periode menunggu yang cukup lama yaitu sekitar 15 menit dan 50.1 menit, karena beberapa *node* memiliki *earliest time window* yang lebih lambat daripada perkiraan *travel + service*.

Pada *Hub* 28, periode menunggu sekitar 15 menit terjadi ketika kendaraan tiba di *customer* 21. selesai melayani di *node* 5 pada menit 235,9, lalu *travel time* 45,1 menit menuju *node* 21 membuat perkiraan tiba pukul 301,0, tetapi karena earliest *node* 21 baru di 351,0, kendaraan harus menunggu sekitar 65 menit. Kemudian pada tahap 1 kendaraan yang telah selesai menyelesaikan layanannya pada *node* 13 akan berangkat menuju *node* 12 dengan waktu tempuh 35.6 menit sehingga akan sampai pada waktu 250.9 menit, akan tetapi karena pada node 12

waktu *earliest* pelayanan baru dimulai pada 301, maka kendaraan akan menunggu selama 50.1 menit. Berikut merupakan Total Waktu tempuh untuk rute Usulan:

Tabel 5. 11 Total Waktu tempuh Usulan

Vehicle	***Travel Time***
Vehicle 1 (Tahap 1)	239.9 menit
Vehicle 1 / *Hub* 27 (Tahap 2)	115.3 menit
Vehicle 2 / Hub 28 (Tahap 2)	383.1 menit
Vehicle 3 / *Hub* 29 (Tahap 2)	432.3 menit
Total	**1170.6 menit**

5.2 Analisis Biaya

Analisis biaya distribusi sangat penting karena membantu perusahaan mengidentifikasi dan memahami secara mendetail seluruh komponen pengeluaran yang ada dalam proses pengiriman barang, mulai dari bahan bakar, tenaga kerja, dan *overhead* operasional. Dengan mengetahui biaya, perusahaan dapat menemukan titik-titik yang tidak efisien, misalnya rute yang terlalu panjang atau kendaraan yang kurang optimal, dan mengambil langkah perbaikan seperti merancang ulang rute pengiriman, memilih kendaraan yang lebih hemat bahan bakar, atau menyesuaikan jadwal operasional. Berdasarkan data yang sudah dikumpulkan berikut merupakan biaya-biaya yang digunakan PT. Pos Logistik Indonesia:

A. Biaya Tetap

Berikut merupakan komponen biaya tetap PT. Pos Logistik Indonesia:

Tabel 5. 12 Biaya Sewa Kendaraan

Jenis Biaya	**Jenis Kendaraan**	**Biaya Sewa/Bulan**	**Biaya Sewa/Hari**
Sewa Kendaraan	*BlindVan*	Rp4.120.000,00	Rp137.333,33

Tabel 5. 13 Biaya Tenaga Kerja

Jenis Biaya	**Jumlah Petugas/Rute**	**Gaji/Bulan**	**Gaji/Hari**
Tenaga Kerja	1	Rp5.600.000,00	Rp186.666,67

Biaya tetap dalam operasi distribusi di PT Pos Logistik Indonesia seperti sewa kendaraan dan gaji sopir bersifat independen terhadap seberapa jauh rute yang ditempuh atau berapa banyak *customer* yang dilayani dalam satu hari. Komponen biaya tetap tersebut tetap wajib dibayar penuh. Oleh karena itu, dalam perhitungan total biaya harian, nilai sewa kendaraan dan gaji

sopir harus dialokasikan secara penuh setiap hari, tanpa melihat fluktuasi aktivitas pengiriman. Akibatnya, perusahaan perlu memastikan pemanfaatan rute dan volume pengiriman yang cukup untuk menutupi beban biaya tetap yang tidak bisa dihilangkan dalam jangka pendek. Sehingga Biaya tetap yang dikeluarkan dalam sehari yaitu Biaya sewa kendaraan sebesar Rp137.333,33 dijumlahkan dengan biaya tenaga kerja sebesar Rp186.666,67 yaitu Rp324.000,00.

B. Biaya Variabel

Berikut merupakan biaya Variabel PT. Pos Logistik Indonesia:

Tabel 5. 14 Biaya BBM

Jenis BBM	**Rasio Konsumsi BBM**	**Harga BBM/Liter**	**Harga BBM/km**
Pertalite	1 : 11	Rp10.000,00	Rp909,09

Karena biaya BBM tergantung langsung pada jarak tempuh, maka biaya ini disebut variable semakin jauh rute atau semakin sering kendaraan beroperasi, biaya BBM semakin tinggi.

C. Biaya Tetap dan Variabel Rute Awalan

Berikut Merupakan Perhitungan Total Biaya Rute Awalan PT. Pos Logistik Indonesia:

Tabel 5. 15 Total Biaya Rute Awalan PT. Pos Logistik Indonesia

Vehicle	Jumlah *Node* yang dikunjungi	Jarak Tempuh	Biaya Tetap	Biaya Variabel
1	6	145.12 km	Rp 324.000,00.	Rp 131,931,00
2	8	176.00 km	Rp 324.000,00.	Rp 159,999,00
3	6	182.63 km	Rp 324.000,00.	Rp 166,029,00
4	6	142.38 km	Rp 324.000,00.	Rp 129,438,00
	Sub Total		**Rp 1.296.000,00**	**Rp 587,397,00**
	Total			**Rp 1.883.397**

Pada Tabel diatas, perusahaan hanya mendistribusikan 26 *customer* secara langsung tanpa melewati *Hub-Hub* perantara. Empat unit kendaraan dipakai untuk menjangkau semua *customer* tersebut, dan setiap kendaraan menanggung biaya tetap harian sebesar Rp 324.000, sehingga subtotal biaya tetap menjadi Rp 1.296.000 (4 × Rp 324.000). Rata-rata jarak tempuh tiap kendaraan yaitu 145.12 km, 176 km, 182.63 km, dan 142.38 km yang memicu biaya

variabel dengan total sebesar Rp 587.397. Dengan demikian, total biaya harian rute awal mencapai Rp 1.883.397 (Rp 1.296.000 tetap + Rp 587.397 variabel).

D. Biaya Tetap dan Variabel Rute Usulan

Berikut Merupakan Perhitungan Total Biaya Rute Usulan Optimasi VRPPDTW menggunakan algoritma *Exact*:

Tabel 5. 16 Total Biaya Rute Rute Usulan Optimasi

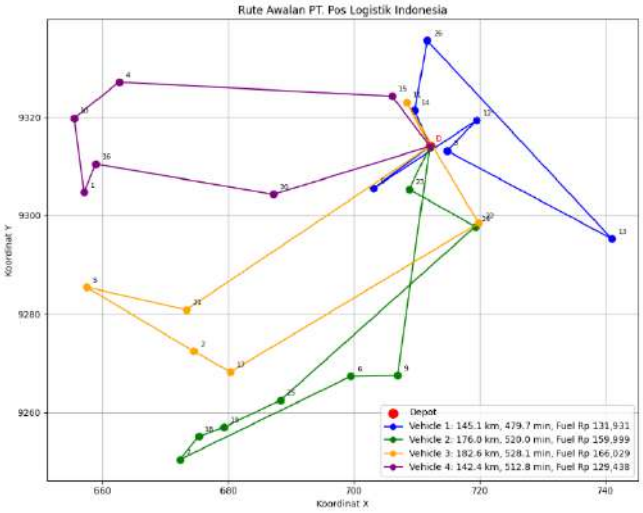
Vehicle	Jumlah *Node* yang dikunjungi	Jarak Tempuh	Biaya Tetap	Biaya Variabel
1 (Tahap 1)	9	88.01 km	Rp 324.000,00.	Rp 80.010,00
1 (Tahap 2)	4	42.28 km	Rp 324.000,00.	Rp 38.436,00
2 (Tahap 2)	7	140.45 km	Rp 324.000,00.	Rp 127.684,00
3 (Tahap 2)	9	158.49 km	Rp 324.000,00.	Rp 144.081,00
	Sub Total		**Rp 1.296.000,00**	**Rp390.211,00**
		Total		**Rp 1.686.211**

Tabel diatass melibatkan penggunaan 3 *Hub* (*Buffer warehouse*) untuk merutekan seluruh *customer*, sehingga total *node* yang dilayani justru bertambah dari 26 *customer* di rute awal menjadi 29 titik kunjungan, 9 node pada Tahap 1, ditambah 4, 7, dan 9 node pada masing-masing Tahap 2. Meskipun demikian, optimasi rute membuat total jarak efektif harian menjadi 88,01 km (Tahap 1) + 42,28 km (Tahap 2 *Hub* 27) + 140,45 km (Tahap 2 *Hub* 28) + 158,49 km (Tahap 2 *Hub* 29) dengan total jarak yaitu 429,23 km. Karena struktur rute baru meminimalkan perjalanan kosong antara *Depot*, *Hub*, dan *customer*, sehingga biaya variabel menjadi turun. Oleh karena itu, total biaya rute usulan hanya Rp 1.686.211. Dengan demikian, meski node bertambah karena perlu melewati tiga *Hub* rute usulan dapat mengurangi biaya variabel, sehingga total biaya harian turun sekitar Rp 197.186 dibandingkan rute awal.

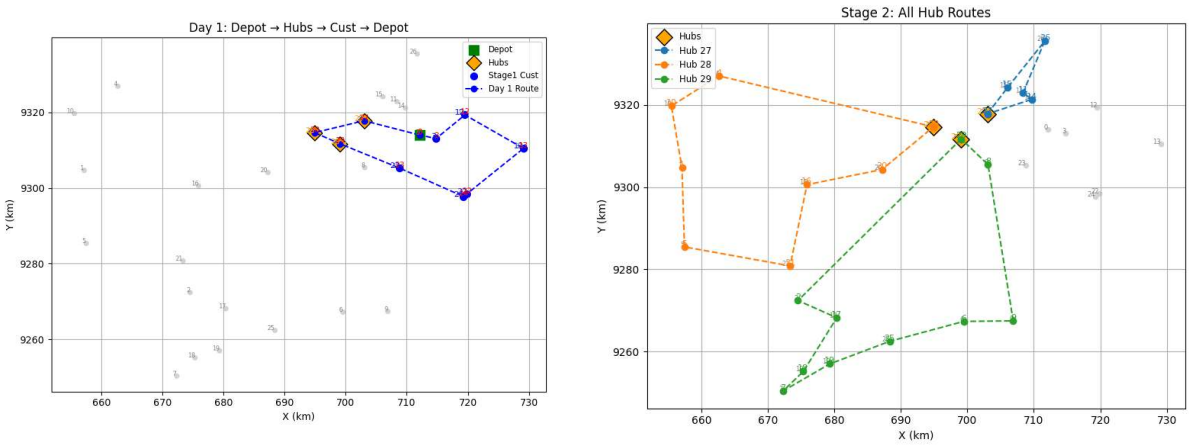
5.3 Perbandingan Rute Awalan dan Usulan

Berikut merupakan perbandingan rute, Jarak, Biaya BBM yang digunakan, dan juga waktu tempuhnya:

Rute Awalan:



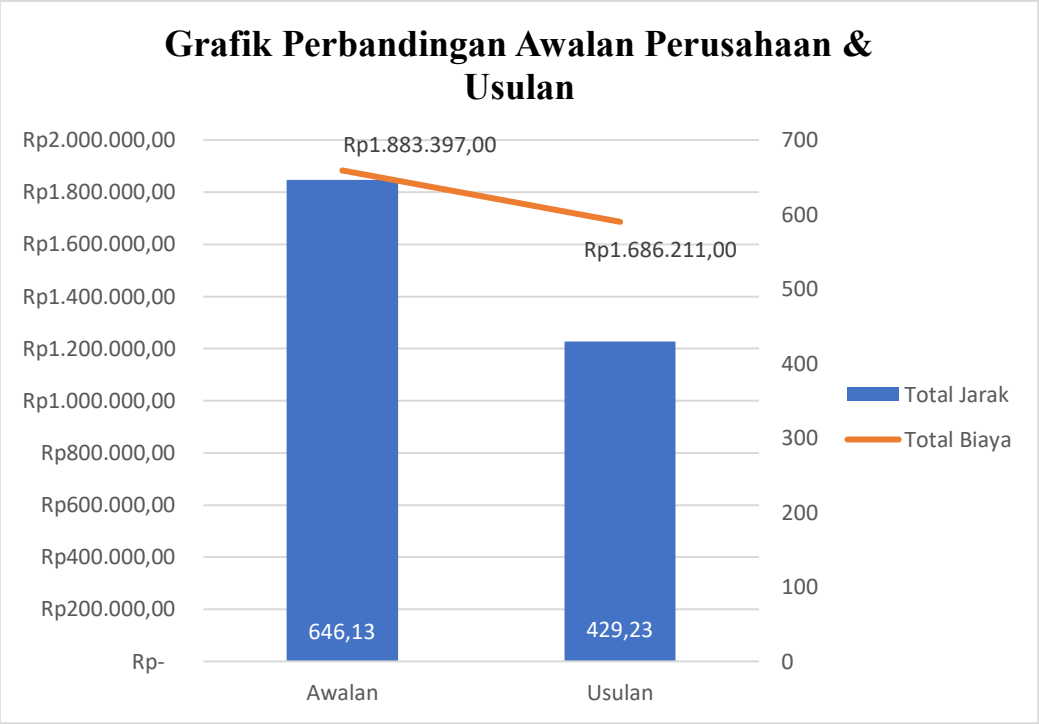
Rute Usulan:



Gambar 5. 6 Perbandingan Visualisasi Rute Awalan dan Usulan

Tabel 5. 17 Perbandingan Rute Awalan dan Usulan

	Rute Awalan	Rute Usulan
Vehicle 1	0-14-26-13-3-12-8-0	0-27-28-29-23-24-22-13-12-3-0
Vehicle 2	0-23-24-25-19-18-7-6-9-0	27-15-26-11-14-27
Vehicle 3	0-11-22-17-2-5-21-0	28-4-10-1-5-21-16-20-28
Vehicle 4	0-15-4-10-1-16-20-0	29-2-17-18-7-19-25-6-9-8-29



Gambar 5. 7 Grafik Perbandingan Jarak & Biaya Awalan Perusahaan & Usulan

Tabel 5. 18 Total Jarak, Waktu Tempuhm Biaya BBM, dan Total Biaya

	Awalan	**Usulan**
Jarak	646.13 km	429.23 km
Waktu tempuh	2040.52 menit	1170.6 menit
Biaya BBM	Rp 587.397,00	Rp390.211,00
Total Biaya	Rp 1.883.397,00	Rp 1.686.211,00

Tabel 5. 19 Komparasi Awalan dan Usulan dengan *workload* pekerja

Aspek	**Kondisi Awal (Satu Hari)**	**Usulan (Dua Hari)**
Durasi kerja harian	Sangat padat (2040,52 menit total, risiko overtime tinggi)	Lebih ringan per hari (1170,6 menit dibagi ke dua hari)
Workload pengemudi	potensi lelah dah stress Tinggi dalam satu hari	Lebih merata, lebih sedikit risiko kelelahan

Efisiensi kendaraan	Berisiko idle di luar jadwal	Lebih terjadwal dan optimal (per hari hanya bawa sesuai kebutuhan)
Produktivitas gudang/hub	Terpusat di satu titik (depot utama)	Terdistribusi: depot + hub, workload lebih seimbang
Kepuasan pelanggan	Potensi keterlambatan jika terjadi overload	Lebih tepat waktu karena beban per hari lebih ringan

Selisih Total = **Rp 1.883.397**– **Rp 1.686.211** = Rp 197.186 per hari

Persentase Penghematan:

$$\frac{\text{Rp } 197.186}{\mathbf{Rp\ 1.883.397}} x\ 100\% = 10.47\%$$

Perhitungan menunjukkan bahwa biaya harian rute awal sebesar Rp 1.883.397 berkurang menjadi Rp 1.686.211 setelah menerapkan rute usulan. Selisih sebesar Rp 197.186 menggambarkan jumlah uang yang dihemat setiap hari, sementara persentase penghematan dihitung dengan membagi Rp 197.186 dengan biaya awal Rp 1.883.397, lalu dikali 100 %, menghasilkan sekitar 10,47 %. Oleh karena itu, pengaturan rute baru mengurangi beban biaya harian perusahaan sekitar 10%, terutama karena berkurangnya pengeluaran variabel seperti bahan bakar akibat optimasi rute dengan pemanfaatan *Buffer warehouse*.

Penurunan biaya sebesar 10,47% yang ditunjukkan dalam perbandingan antara rute awal dan rute usulan memberikan dampak signifikan terhadap efisiensi operasional perusahaan. Dengan berkurangnya total jarak tempuh dan waktu perjalanan, perusahaan dapat mengurangi beban pengeluaran logistik harian. Jika penghematan ini dikalkulasikan selama satu bulan (30 hari), maka perusahaan berpotensi menghemat biaya operasional hingga Rp 5.915.580. Hal ini tentunya memberikan peluang untuk meningkatkan efisiensi alokasi anggaran. Selain efisiensi biaya, dana yang dihemat juga dapat dialokasikan untuk kegiatan yang bersifat strategis, seperti perawatan dan peningkatan armada kendaraan, pengembangan layanan distribusi, hingga ekspansi rute baru untuk menjangkau pelanggan secara lebih luas. Di sisi lain, penurunan waktu tempuh turut berdampak positif terhadap ketepatan waktu pengiriman. Dengan rute yang lebih singkat, pengiriman dapat dilakukan dengan lebih cepat dan dapat diandalkan, sehingga meningkatkan kepuasan pelanggan serta memperkuat kepercayaan terhadap layanan perusahaan.

5.4 Kekurangan Penelitian

Perubahan rute pada penelitian ini tetap memperhatikan sejumlah batasan penting dalam perencanaan operasional agar penelitian tidak terlalu luas, seperti wilayah yang menjadi objek yaitu Pos Logistik dengan cakupan DKI Jakarta, terdapat satu *Depot* atau *Main Hub*, waktu layanan yang digunakan mulai pukul 08.00 WIB – 17.00 WIB dan tidak memperhitungkan waktu istirahat, penelitian ini tidak mempertimbangkan kapasitas *buffer warehouse*, tidak mencakup kondisi lalu lintas dan cuaca, serta kendaraan dianggap hanya satu tipe saja yaitu *BlindVan*. Secara keseluruhan, penurunan biaya yang dicapai melalui optimasi rute ini merupakan bukti bahwa efisiensi operasional dapat dicapai tanpa mengabaikan batasan-batasan sistem yang ada. Meskipun penelitian ini berhasil menghasilkan rancangan rute distribusi yang lebih efisien, terdapat beberapa keterbatasan yang perlu dicatat untuk memberikan gambaran yang lebih realistis terhadap penerapannya.

1. Tidak dimasukkannya faktor lalu lintas (*traffic*) dan kondisi cuaca membuat estimasi waktu tempuh yang dihasilkan cenderung optimis. Dalam praktiknya, kemacetan dan keterlambatan cuaca bisa berdampak besar terhadap keandalan pengiriman, apalagi di wilayah perkotaan padat seperti Jakarta.
2. Berdasarkan hasil penelitian, secara biaya berkurang atau terdapat efisiensi dari segi biaya, jarak, dan waktu tempuh, namun dari segi tenaga kerja terdapat ketidak seimbangan dikarenakan strategi melakukan aktivitas distribusi dalam dua hari
3. penelitian ini juga belum memanfaatkan data ETA (*Estimated Time of Arrival*) atau *real-time tracking* sebagai pendekatan prediktif untuk memperkirakan waktu tiba secara dinamis. Hal ini membatasi tingkat akurasi dalam perencanaan jadwal distribusi harian. Selain itu, pendekatan ini hanya diterapkan pada wilayah terbatas dengan jumlah node yang relatif kecil**,** sehingga belum dapat mencerminkan kompleksitas rute distribusi berskala nasional yang melibatkan puluhan hingga ratusan titik layanan.
4. Dari sisi metodologi, meskipun metode *exact* berbasis MILP memberikan hasil yang optimal secara matematis, namun memiliki keterbatasan skalabilitas karena waktu komputasi akan meningkat drastis seiring bertambahnya jumlah node, kendaraan, dan kendala yang dipertimbangkan. Dalam skala besar, metode ini bisa menjadi tidak praktis, sehingga perlu dikombinasikan dengan pendekatan lain seperti *matheuristic* atau *metaheuristic* agar solusi tetap akurat namun efisien secara waktu.

Penelitian selanjutnya dapat melakukan pengembangan model yang mengakomodasi kapasitas aktual Hub, mempertimbangkan jenis kendaraan lainnya, integrasi data lalu lintas *real time*, serta mengkombinasikan *exact* dengan *heuristic* atau *metaheuristic* dapat menjadi arah lanjutan yang memperkaya pendekatan optimasi distribusi ini. Dengan demikian, model yang dihasilkan akan semakin aplikatif terhadap kebutuhan operasional logistik yang semakin kompleks dan dinamis.

BAB VI
PENUTUP

6.1 Kesimpulan

Berikut meruupakan Kesimpulan penelitian VRPPDTW menggunakan algoritma *Exact* pada PT. Pos Logistik Indonesia:

1. Metode *Exact* untuk VRPPDTW berhasil menemukan skema dua tahap yaitu *Depot* - *Hub* - *Customer* Stage 1 - *Depot* pada Hari 1, kemudian *Hub* - *Customer* Stage 2 - *Hub* pada Hari 2 sebagai rute terbaik. Jika dibandingkan dengan rute existing (empat kendaraan langsung dari *Depot* ke 26 *customer* tanpa *Hub*), rute usulan yang melibatkan *tiga buffer warehouse* (*Hub* 27, 28, 29) menghasilkan total kunjungan 29 node, sementara rute existing hanya menyinggahi 26 node. Meski jumlah node meningkat, jarak efektif rute usulan yaitu 429,23 km jauh lebih pendek daripada jarak rute existing yaitu 646,13 km, sehingga biaya keseluruhan termasuk biaya tetap dan variabel berkurang dari Rp 1.883.397 menjadi Rp 1.686.211 per hari.
2. Dari hasil optimasi, diperoleh waktu tempuh yang jauh lebih terukur dan realistis dibandingkan rute awalan. Pada skema usulan, Tahap 1 menempuh sekitar 239,9 menit kendaraan berangkat dari *Depot*, singgah di *Hub* 27, *Hub* 28, dan *Hub* 29, kemudian melayani *customer Stage* 1, dan akhirnya kembali ke *Depot*. Selain itu, pada tahap 2 setiap *Hub* menjalankan rute lokalnya masing-masing, *Hub* 27 membutuhkan sekitar 115,3 menit, *Hub* 28 sekitar 383,1 menit, dan *Hub* 29 sekitar 432,3 menit. Jika dijumlah, total waktu tempuh sistem usulan adalah sekitar 1.170,6 menit untuk menuntaskan kedua tahap pengiriman. Sementara itu, rute memerlukan sekitar 2.040,5 menit secara keseluruhan. Dengan demikian, rute usulan mengefisienkan durasi operasional hampir setengahnya. Secara keseluruhan, hasil ini menunjukkan bahwa model *Exact* tidak hanya mengurangi jarak dan biaya bahan bakar, tetapi juga secara signifikan menekan total waktu tempuh distribusi dibanding skema rute awalan.
3. Strategi terbaik yang dihasilkan adalah pada hari pertama Langkah yang dilakukan adalah *"Drop to Hub"* pada Tahap 1 dengan Mengirimkan sebagian muatan ke *buffer warehouse* (*Hub* 27, 28, 29) sambil melayani *customer Stage* 1 yang terdekat dari *Depot*. Hal ini mencegah beban penuh langsung ke setiap *customer* dan memecah distribusi menjadi *loop* yang lebih pendek. Untuk hari kedua dilakukan Klasterisasi lokal sehingga Setiap *Hub* kemudian melayani *last mile customer* Stage 2 di sekitarnya, sehingga setiap kendaraan

hanya menjalani *loop* lokal. Penerapan strategi ini menekan biaya variabel hingga Rp 197.186 per hari sehingga melakukan penghematan biaya sekitar 10,47 %.

6.2 Saran

1. Untuk Perusahaan

   Berdasarkan hasil penelitian ini, perusahaan dapat mengoptimalkan penggunaan *buffer warehouse* (*Hub*) sebagai inventaris strategis untuk menekan biaya distribusi. Dengan menempatkan stok yang disesuaikan berdasarkan permintaan lokal di masing-masing *Hub*, beban pengiriman dari *Depot* pusat dapat dipecah menjadi rute yang lebih pendek dan terklaster. Memastikan ketiga *buffer warehouse* (*Hub* 27, 28, 29) memiliki kapasitas penyimpanan, fasilitas bongkar muat, dan tenaga kerja yang memadai. Untuk mengimplementasikan strategi ini, perusahaan sebaiknya menggunakan algoritma *Exact VRPPDTW* atau metode heuristik berbasis *clustering* untuk menentukan jumlah dan lokasi *Hub* optimal, level stok di setiap *Hub*, serta beban muatan yang efisien. Algoritma tersebut mempertimbangkan batasan *time-window* dan kapasitas kendaraan untuk memastikan rute yang dihasilkan dapat langsung dijalankan di lapangan tanpa pelanggaran jadwal ataupun *overload*.

2. Untuk Penelitian selanjutnya

   Untuk penelitian selanjutnya, disarankan memperluas model dengan mempertimbangkan armada berkapasitas beragam (misalnya van kecil, truk sedang, truk besar) agar setiap *cluster customer* dapat dialokasikan kendaraan paling sesuai. mengeksplorasi skema multi *Depot* dan *splitting load* secara dinamis untuk mengurangi jarak kosong antar *Depot* dan *Hub*, serta menambahkan komponen biaya. Penelitian selanjutnya juga disarankan untuk mempertimbangkan variabel kondisi lalu lintas dengan memanfaatkan data *Estimated Time of Arrival* (ETA) sebagai *input*. Hal ini menjadi relevan terutama apabila lokasi penelitian berada di wilayah dengan tingkat kemacetan tinggi, seperti Jakarta, sehingga estimasi waktu dalam proses perencanaan rute dan analisis dapat lebih akurat dan representatif terhadap kondisi nyata di lapangan.

DAFTAR PUSTAKA

Afiya, N., Permadi, Y. W., Rahmatullah, St., & Ningrum, W. A. (2022). Analisis Pengelolaan Manajemen Logistik Obat Di Instalasi Farmasi Rumah Sakit Qim Batang Tahun 2021. *Jurnal Ilmiah Jophus : Journal Of Pharmacy Umus*, *3*(02), 138–145. Https://Doi.Org/10.46772/Jophus.V3i02.521

Akbar, I., Ardiansyah, M. N., Giri, P., & Kusuma, A. (2021). Perancangan Rute Pengiriman Menggunakan Model Integer Linear Programming Untuk Meminimasi Tingkat Keterlambatan Pengiriman Dan Biaya Transportasi (Studi Kasus: Pt. Pos Logistik Indonesia) Designing Delivery Route Using Integer Linear Programming Model To . *Agustus*, *8*(4), 4028.

Alacsel, S. (2024). Penerapan Sistem Informasi Manajemen Pada Pt. Prima Indonesia Logistik. *Akademik: Jurnal Mahasiswa Humanis*, *4*(1), 122–131. Https://Doi.Org/10.37481/Jmh.V4i1.719

Anggraeni, M. S., Tazkiya, O. N., Mawandi, E., & Amarilies, H. S. (2023). *Optimization Of Fuel Distribution Routes For Green Logistics In Multi Compartment Vehicle Routing Problem (Mcvrp) Using Branch And Bound Algorithm (Case Study: Boyolali Fuel Terminal)*. 1132–1144. Https://Doi.Org/10.46254/An13.20230333

Anisa1, L. N., Andawiah2, S., Utama3, D. P., & Ibnu Afan4. (2025). *Implementasi Supply Chain Management Untuk Meningkatkan Kinerja Logistik Perusahaan*. *11*(1), 460–471.

Annisa, N., Meitasari, V., & Adhi, A. (2025). *Optimisasi Rute Pengiriman Produk Usaha Mikro Kecil Dan Menengah Menggunakan Metode Metaheuristik Particle Swarm Optimization*. *X*(1), 12684–12690.

Arifian, R., & Pulansari, F. (2023). Determination Of The Shortest Distribution Routes To Minimize Fuel Consumption And Co2 Emission Using Sweep And Clarke & Wright Saving Algorithms. *Ijiem - Indonesian Journal Of Industrial Engineering And Management*, *4*(2), 146. Https://Doi.Org/10.22441/Ijiem.V4i2.20493

Ayu Muchlisa, N., & Surianto, Moh. A. (2021). Analisis Saluran Distribusi Pada Pt. Panahmas Dwitama Distrindo Jember. *Jurnal Indonesia Sosial Sains*, *2*(12), 2059–2068. Https://Doi.Org/10.36418/Jiss.V2i12.480

Baldacci, R., Hadjiconstantinou, E., & Mingozzi, A. (2004). An Exact Algorithm For The Capacitated Vehicle Routing Problem Based On A Two-Commodity Network Flow Formulation. *Operations Research*, *52*(5), 723–738. Https://Doi.Org/10.1287/Opre.1040.0111

Baldacci, R., Mingozzi, A., & Roberti, R. (2012). Recent Exact Algorithms For Solving The Vehicle Routing Problem Under Capacity And Time Window Constraints. *European Journal Of Operational Research*, *218*(1), 1–6. Https://Doi.Org/10.1016/J.Ejor.2011.07.037

Chandra, A., & Setiawan, B. (2018). Optimasi Jalur Distribusi Dengan Metode Vehicle Routing Problem (Vrp). *Jurnal Manajemen Transportasi & Logistik (Jmtranslog)*, *5*(2), 105. Https://Doi.Org/10.54324/J.Mtl.V5i2.233

Dewi, N. K., & Faishal Mashuda, R. (2022). Analisis Distibusi Paving Blok Dengan Metode Capacitated Vehicle Routing Problem Dan Metode Tabu Search Pada Pt. X. *Jurnal Manajemen Logistik Dan Transportasi*, *8*(3), 141–166.

Dumez, D., Tilk, C., Irnich, S., Lehuédé, F., & Péton, O. (2021). Hybridizing Large Neighborhood Search And Exact Methods For Generalized Vehicle Routing Problems With Time Windows. *Euro Journal On Transportation And Logistics*, *10*(April). Https://Doi.Org/10.1016/J.Ejtl.2021.100040

Elatar, S., Abouelmehdi, K., & Riffi, M. E. (2023). The Vehicle Routing Problem In The Last Decade: Variants, Taxonomy And Metaheuristics. *Procedia Computer Science*, *220*, 398–404. Https://Doi.Org/10.1016/J.Procs.2023.03.051

Fatma, E., & Manurung, S. (2021). Optimasi Biaya Distribusi Multi-Eselon Untuk Multi-Produk Menggunakan Mix Integer Linier Programming. *Jurnal Manajemen Transportasi & Logistik (Jmtranslog)*, *8*(1), 75. Https://Doi.Org/10.54324/J.Mtl.V8i1.418

Ferdiansyah, A., Sholihah, S. A., Rifni, M., Grets, E. S., Situmorang, J. K., & Oktaviany, I. (2020). *Analisis Perencanaan Rute Pengiriman Barang Menggunakan Metode Vehicle Routing Problem (Vrp)*. 4–9.

Hester Patmawati1, Y. A. N. (2022). *Optimalisasi Rute Distribusi Matras Pada Penyelesaian Capacitated Vehicle Routing Problem Dengan Metode Algoritma Genetika*. *20*(1), 105–123.

James, J. J., & Nugroho, A. J. (2024). Penyelesaian Vehicles Routing Problem Dalam Meminimumkan Waktu Transportasi Pt Petrogas Prima Services. *Sci-Tech Journal*, *3*, 60–73.

Kasengkang, R. A., Nangoy, S., & Sumarauw, J. (2016). Analisis Logistik (Studi Kasus Pada Pt. Remenia Satori Tepas-Kota Manado). *Jurnal Berkala Ilmiah Efisiensi*, *16*(01), 750–759.

Kushariyadi, & Bambang Sugito. (2022). Optimasi Distribusi Transportasi Bahan Bakar Minyak (Bbm) Jenis Bio Solar Di Wilayah Jawa Tengah. *Jurnal Pendidikan Dan Konseling*, *4*(5), 1359–1367.

Kushariyadi, Sono, Adi, T. W., Eka Aristantia, S., & Aviciena Taufiqurrahman, M. (2024). Analisis Rute Distribusi Bbm Di Pertashop Menggunakan Metode Saving Matrik. *Jurnal Sistim Informasi Dan Teknologi*, *5*, 51–56. Https://Doi.Org/10.60083/Jsisfotek.V5i4.332

Liu, F., Lu, C., Gui, L., Zhang, Q., Tong, X., & Yuan, M. (N.D.). *Heuristics For Vehicle Routing Problem : A Survey And*. 1–67.

Maya Widyastiti, A. S. (2024). *Penerapan Multiple Traveling Salesman Problem Pada Optimasi Pendistribusian Maya Widyastiti * Program Studi Matematika , Fmipa , Universitas Pakuan Amar Sumarsa Program Studi Matematika , Fmipa , Universitas Pakuan Alfina Yulia Permana Program Studi Matem*. *16*(1), 19–28.

Muladi, J. O., Kristiansyah Dezzano, K., Febrianto, A., Kurniawan, A. C., Ruswandi, N., & Redi, A. A. N. P. (2020). Optimasi Rute Kapal Untuk Distribusi Spare Parts Menggunakan Vehicle Routing Problem Dengan Algoritma Tabu Search. *Go-Integratif : Jurnal Teknik Sistem Dan Industri*, *1*(01), 1–10. Https://Doi.Org/10.35261/Gijtsi.V1i01.4316

Nguyen, V. S., Pham, Q. D., Nguyen, T. H., & Bui, Q. T. (2022). Modeling And Solving A Multi-Trip Multi-Distribution Center Vehicle Routing Problem With Lower-Bound Capacity Constraints. *Computers And Industrial Engineering*, *172*(Pa), 108597. Https://Doi.Org/10.1016/J.Cie.2022.108597

Nurdin, N., Taufiq, T., Bustami, B., Marleni, M., & Khairuni, K. (2023). Optimization Model Of Fishery Products Supply Chain Using Mixed Integer Linear Programming Method. *Journal Of Informatics And Telecommunication Engineering*, *6*(2), 378–392. Https://Doi.Org/10.31289/Jite.V6i2.8186

Panjaitan, D. J., Aprilia, R., & Anjeli, S. (2024). Vehicle Routing Problem As A Solution For Determining Goods Delivery Routes Pt. Kreasi Beton Nusa Persada. *Jst (Jurnal Sains Dan Teknologi)*, *12*(3), 622–632. Https://Doi.Org/10.23887/Jstundiksha.V12i3.67809

Praxedes, R., Bulhões, T., Subramanian, A., & Uchoa, E. (2024). A Unified Exact Approach For A Broad Class Of Vehicle Routing Problems With Simultaneous Pickup And Delivery. *Computers And Operations Research*, *162*(September 2023), 106467. Https://Doi.Org/10.1016/J.Cor.2023.106467

Purba, S. D., & Ahyaningsih, F. (2020). Integer Programming Dengan Metode Branch And Bound Dalam Optimasi Jumlah Produksi Setiap Jenis Roti Pada Pt. Arma Anugerah Abadi. *Jurnal Karismatika*, *6*(03), 20–29.

Rahmania, S. N. A., Wahyuda, W., & Gunawan, S. (2023). Penentuan Rute Distribusi Barang Menggunakan Vehicle Routing Problem (Studi Kasus: Cv. Surya Inti Distrindo). *Intecoms: Journal Of Information Technology And Computer Science*, *6*(2), 578–585. Https://Doi.Org/10.31539/Intecoms.V6i2.6816

Redi A.A.N, P., Zahra, F., Islami, P. A. N., Rachmawati, N. L., Nadlifatin, R., & Supranartha, A. (2021). Simulated Annealing Algorithm For Vehicle Routing Problem With Simultaneous Pick Up And Delivery: A Case Study Of Liquid Petroleum Gas Distribution. *Proceedings Of The Second Asia Pacific International Conference On Industrial Engineering And Operations Management*, 667–676.

Ritonga, G. A., Muttaqin, P. S., & Novitasari, N. (2024). *Penentuan Rute Yang Optimal Pada Cv Uli Menggunakan Integer Linear Programming Untuk Dapat Mencapai Target Jumlah Kunjungan Salesman* . *2*(1), 49–56.

Rizkiani, F. N., Sari, R., & Imran, A. (2023). *A Simulated Annealing For Heterogenous Fleet Vehicle Routing Problem With Multiple Trips And Pickup-Delivery*. *7*(November), 277–288. Https://Doi.Org/10.26760/Jrh.V7i3.277-288

Ropke, S. (2005). Heuristic And Exact Algorithms For Vehicle Routing Problems. *Unpublished Phd Thesis, Computer Science Department, University Of Copenhagen*, *December*, 256. Http://Www.Diku.Dk/~Sropke/Papers/Phdthesis.Pdf

Setiadi, T., Darnis, F., & Ilhami, S. D. (2024). Optimasi Perutean Jalur Kendaraan Terdekat Traveling Salesman Problem Dengan Artificial Bee Colony Algorithm. *Jurnal Komtekinfo*, *10*, 42–48. Https://Doi.Org/10.35134/Komtekinfo.V11i2.502

Sugiono, M. C. (2022). Model Vehicle Routing Problem Untuk Penentuan Rute Distribusi Unit Sepeda Motor Dengan Metode Saving Matrix. *Journal Industrial Servicess*, *7*(2), 230. Https://Doi.Org/10.36055/Jiss.V7i2.14018

Sumarsa, A., & Widyastiti, M. (2022). *Penerapan Traveling Salesman Problem With Time Windows Dalam Pendistribusian Produk*. *November 2022*, 1–6.

Suryanto, Nugroho, E. S., & Putra, R. A. K. (2019). Analisis Optimasi Keuntungan Dalam Produksi Keripik Daun Singkong Dengan Linier Programming Melalui Metode Simpleks. *Jurnal Manajemen*, *11*(2), 226–236.

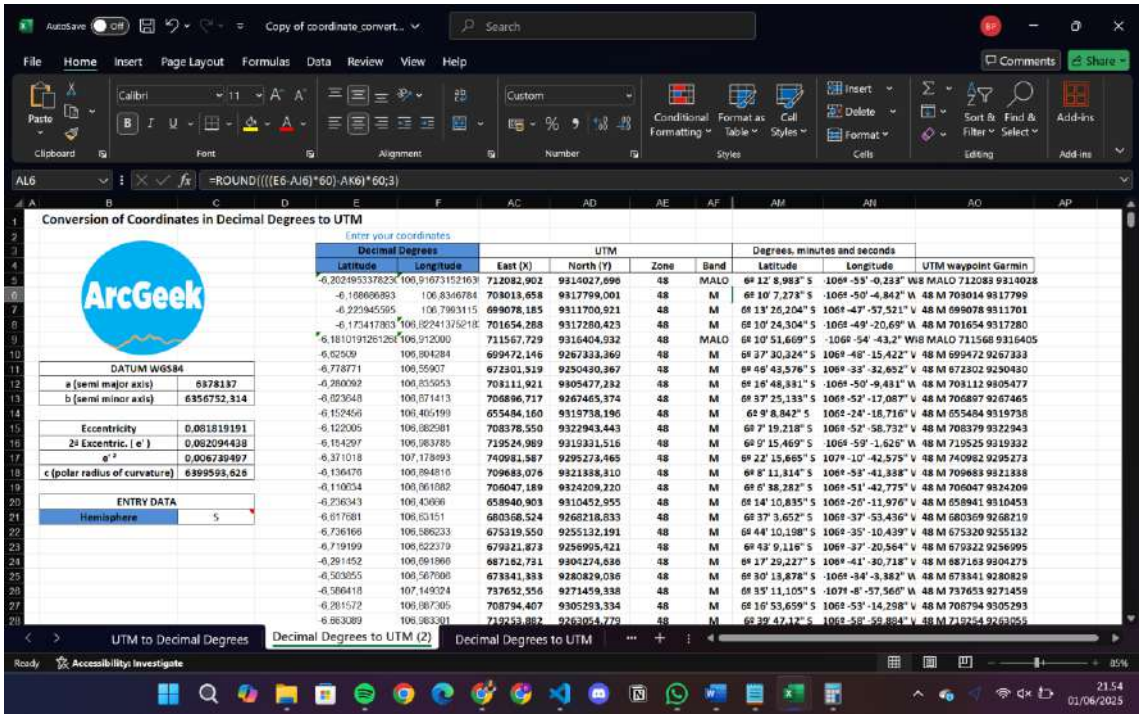
Tjoea, F. J., & Halim, S. (2023). Evaluasi Rute Pelayaran Kapal Dengan Pendekatan Modified Minimum…. *Jurnal Titra*, *11*(2), 265–272.

Wulandari, C. B. K. (2020). Penentuan Rute Distribusi Menggunakan Metode Nearest Neighbors Dan Metode Branch And Bound Untuk Meminimumkan Biaya Distribusi Di Pt. X. *Jurnal Optimasi Teknik Industri (Joti)*, *2*(1), 7. Https://Doi.Org/10.30998/Joti.V2i1.3848

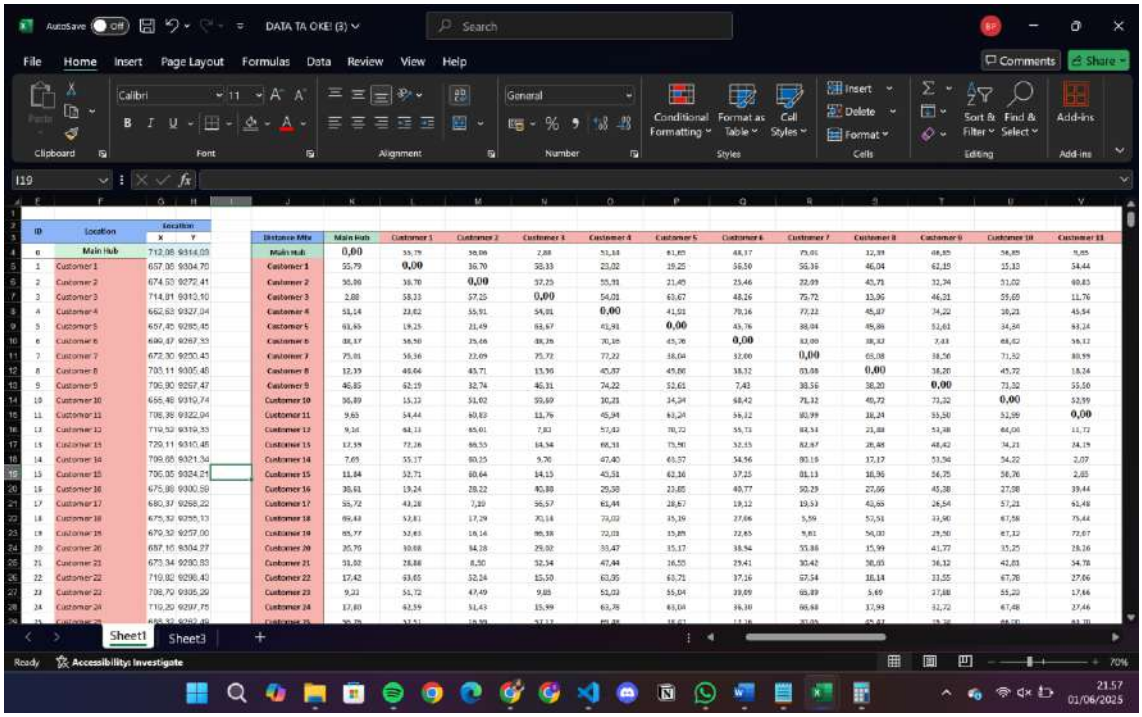
Zhao, J., Liu, Y., Zhang, J., Zhang, J., Huang, Y., Yu, L., & Xie, B. (2024). An Exact Method For Vehicle Routing Problem With Backhaul Discounts In Urban Express Delivery Network. *Cleaner Logistics And Supply Chain*, *11*(May). Https://Doi.Org/10.1016/J.Clscn.2024.100157

LAMPIRAN

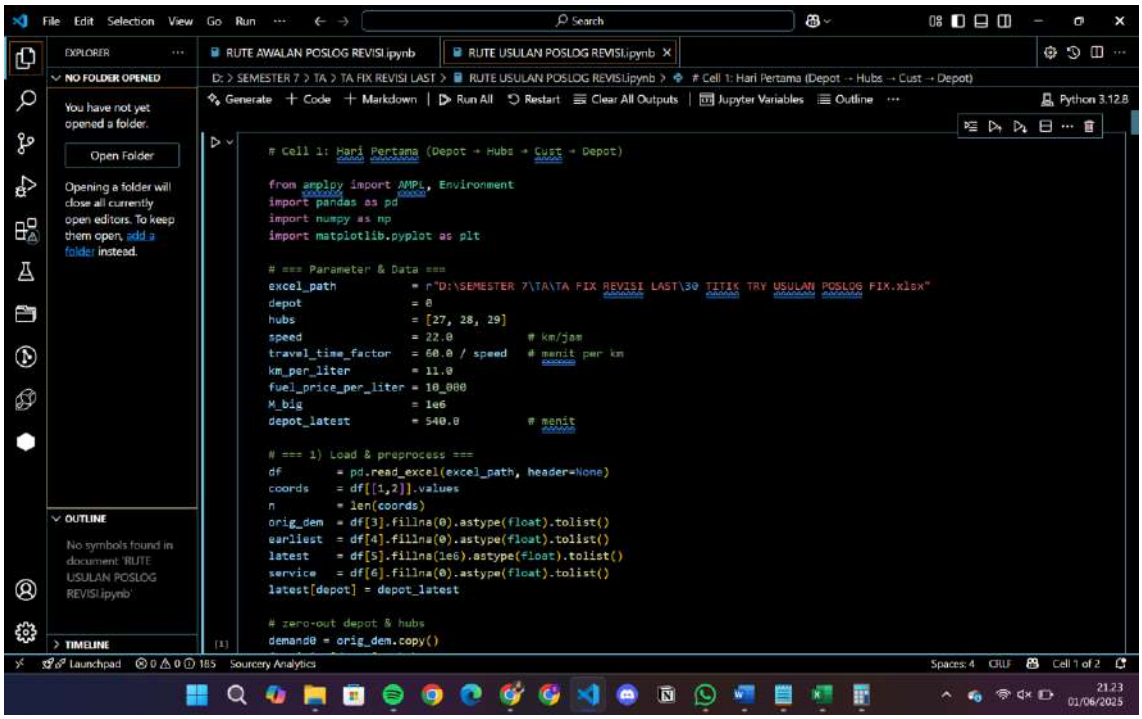
A - Coordinate Converter (Latitude Longitude – UTM)



B - Pengolahan Matriks Jarak



C - Pseudocode Algoritma Exact – VRPPDTW



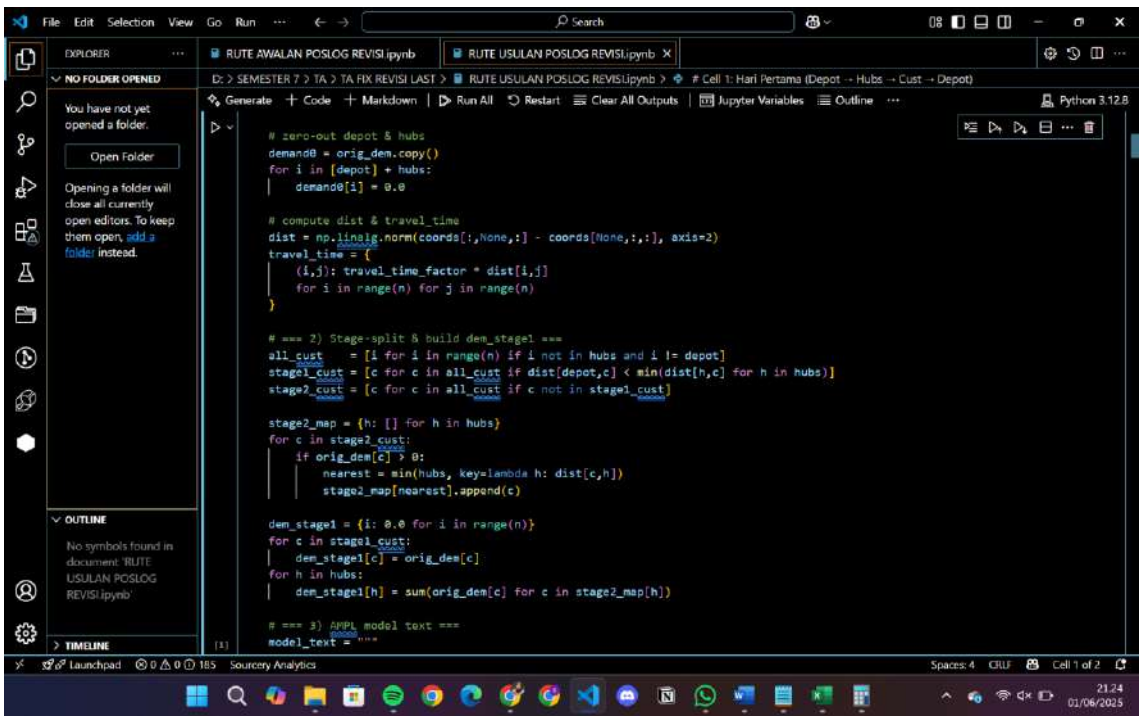
```
# Cell 1: Hari Pertama (Depot → Hubs → Cust → Depot)

from amplpy import AMPL, Environment
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# === Parameter & Data ===
excel_path          = r"D:\SEMESTER 7\TA\TA FIX REVISI LAST\30 TITIK TRY USULAN POSLOG FIX.xlsx"
depot               = 0
hubs                = [27, 28, 29]
speed               = 22.0          # km/jam
travel_time_factor  = 60.0 / speed  # menit per km
km_per_liter        = 11.0
fuel_price_per_liter = 10_000
M_big               = 1e6
depot_latest        = 540.0         # menit

# === 1) Load & preprocess ===
df        = pd.read_excel(excel_path, header=None)
coords    = df[[1,2]].values
n         = len(coords)
orig_dem  = df[3].fillna(0).astype(float).tolist()
earliest  = df[4].fillna(0).astype(float).tolist()
latest    = df[5].fillna(1e6).astype(float).tolist()
service   = df[6].fillna(0).astype(float).tolist()
latest[depot] = depot_latest

# zero-out depot & hubs
demand0 = orig_dem.copy()
```



```
# zero-out depot & hubs
demand0 = orig_dem.copy()
for i in [depot] + hubs:
    demand0[i] = 0.0

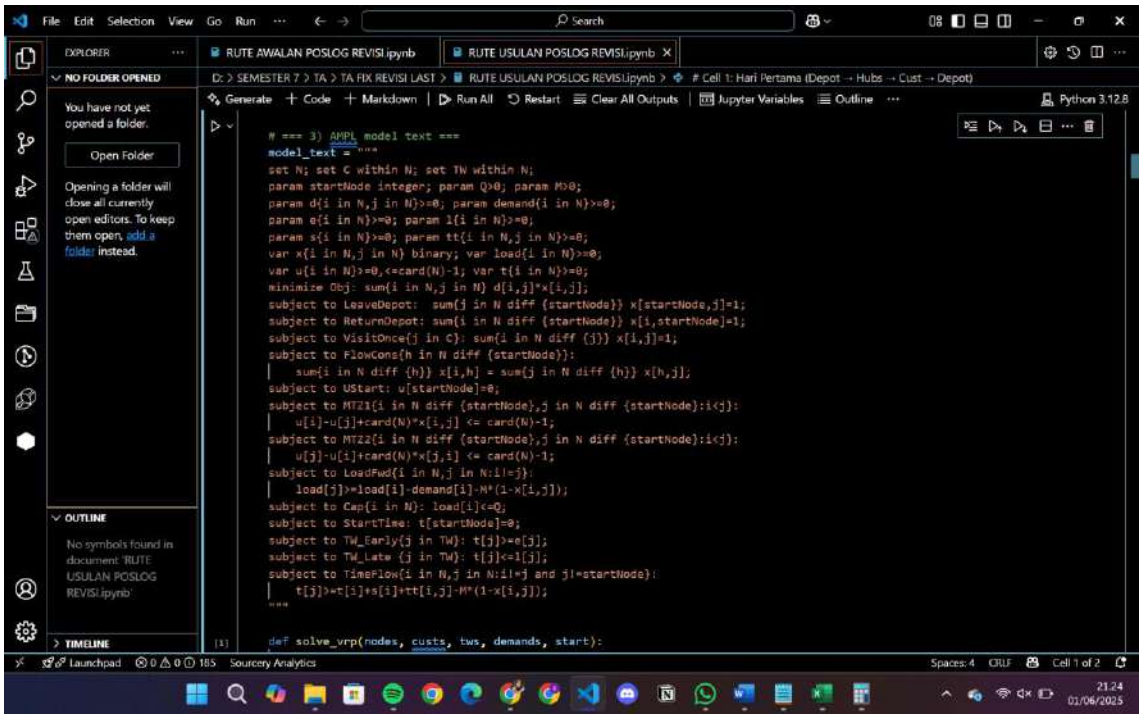
# compute dist & travel_time
dist = np.linalg.norm(coords[:,None,:] - coords[None,:,:], axis=2)
travel_time = {
    (i,j): travel_time_factor * dist[i,j]
    for i in range(n) for j in range(n)
}

# === 2) Stage-split & build dem_stage1 ===
all_cust    = [i for i in range(n) if i not in hubs and i != depot]
stage1_cust = [c for c in all_cust if dist[depot,c] < min(dist[h,c] for h in hubs)]
stage2_cust = [c for c in all_cust if c not in stage1_cust]

stage2_map = {h: [] for h in hubs}
for c in stage2_cust:
    if orig_dem[c] > 0:
        nearest = min(hubs, key=lambda h: dist[c,h])
        stage2_map[nearest].append(c)

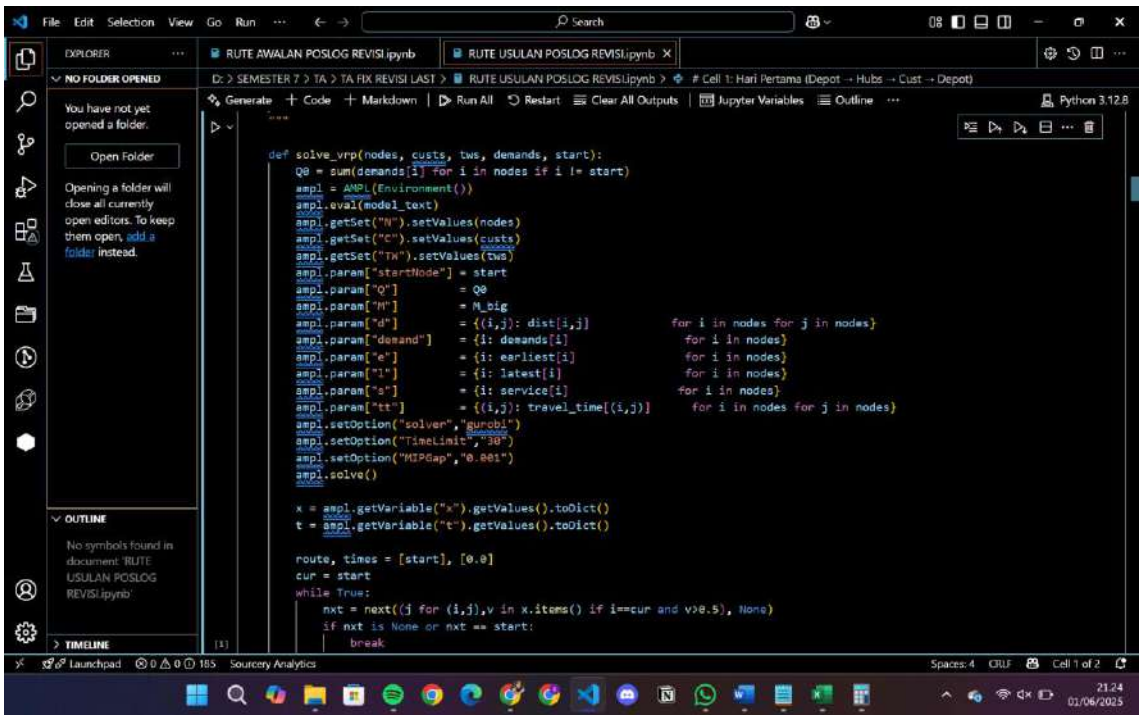
dem_stage1 = {i: 0.0 for i in range(n)}
for c in stage1_cust:
    dem_stage1[c] = orig_dem[c]
for h in hubs:
    dem_stage1[h] = sum(orig_dem[c] for c in stage2_map[h])

# === 3) AMPL model text ===
model_text = """
```



```
# === 3) AMPL model text ===
model_text = """
set N; set C within N; set TW within N;
param startNode integer; param Q>0; param M>0;
param d{i in N,j in N}>=0; param demand{i in N}>=0;
param e{i in N}>=0; param l{i in N}>=0;
param s{i in N}>=0; param tt{i in N,j in N}>=0;
var x{i in N,j in N} binary; var load{i in N}>=0;
var u{i in N}>=0,<=card(N)-1; var t{i in N}>=0;
minimize Obj: sum{i in N,j in N} d[i,j]*x[i,j];
subject to LeaveDepot:  sum{j in N diff {startNode}} x[startNode,j]=1;
subject to ReturnDepot: sum{i in N diff {startNode}} x[i,startNode]=1;
subject to VisitOnce{j in C}: sum{i in N diff {j}} x[i,j]=1;
subject to FlowCons{h in N diff {startNode}}:
   sum{i in N diff {h}} x[i,h] = sum{j in N diff {h}} x[h,j];
subject to UStart: u[startNode]=0;
subject to MTZ1{i in N diff {startNode},j in N diff {startNode}:i<j}:
   u[i]-u[j]+card(N)*x[i,j] <= card(N)-1;
subject to MTZ2{i in N diff {startNode},j in N diff {startNode}:i<j}:
   u[j]-u[i]+card(N)*x[j,i] <= card(N)-1;
subject to LoadFwd{i in N,j in N:i!=j}:
   load[j]>=load[i]-demand[i]-M*(1-x[i,j]);
subject to Cap{i in N}: load[i]<=Q;
subject to StartTime: t[startNode]=0;
subject to TW_Early{j in TW}: t[j]>=e[j];
subject to TW_Late {j in TW}: t[j]<=l[j];
subject to TimeFlow{i in N,j in N:i!=j and j!=startNode}:
   t[j]>=t[i]+s[i]+tt[i,j]-M*(1-x[i,j]);
"""

def solve_vrp(nodes, custs, tws, demands, start):
```

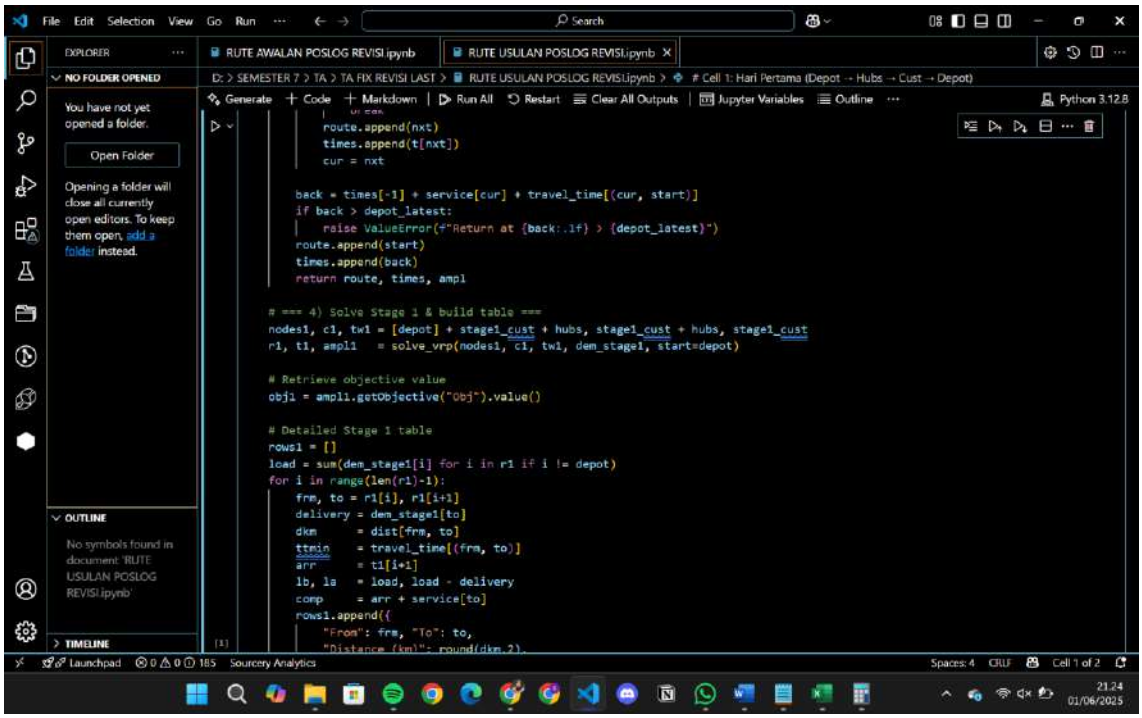


```
"""

def solve_vrp(nodes, custs, tws, demands, start):
    Q0 = sum(demands[i] for i in nodes if i != start)
    ampl = AMPL(Environment())
    ampl.eval(model_text)
    ampl.getSet("N").setValues(nodes)
    ampl.getSet("C").setValues(custs)
    ampl.getSet("TW").setValues(tws)
    ampl.param["startNode"] = start
    ampl.param["Q"]         = Q0
    ampl.param["M"]         = M_big
    ampl.param["d"]         = {(i,j): dist[i,j]          for i in nodes for j in nodes}
    ampl.param["demand"]    = {i: demands[i]               for i in nodes}
    ampl.param["e"]         = {i: earliest[i]              for i in nodes}
    ampl.param["l"]         = {i: latest[i]                for i in nodes}
    ampl.param["s"]         = {i: service[i]              for i in nodes}
    ampl.param["tt"]        = {(i,j): travel_time[(i,j)]    for i in nodes for j in nodes}
    ampl.setOption("solver","gurobi")
    ampl.setOption("TimeLimit","30")
    ampl.setOption("MIPGap","0.001")
    ampl.solve()

    x = ampl.getVariable("x").getValues().toDict()
    t = ampl.getVariable("t").getValues().toDict()

    route, times = [start], [0.0]
    cur = start
    while True:
        nxt = next((j for (i,j),v in x.items() if i==cur and v>0.5), None)
        if nxt is None or nxt == start:
            break
```



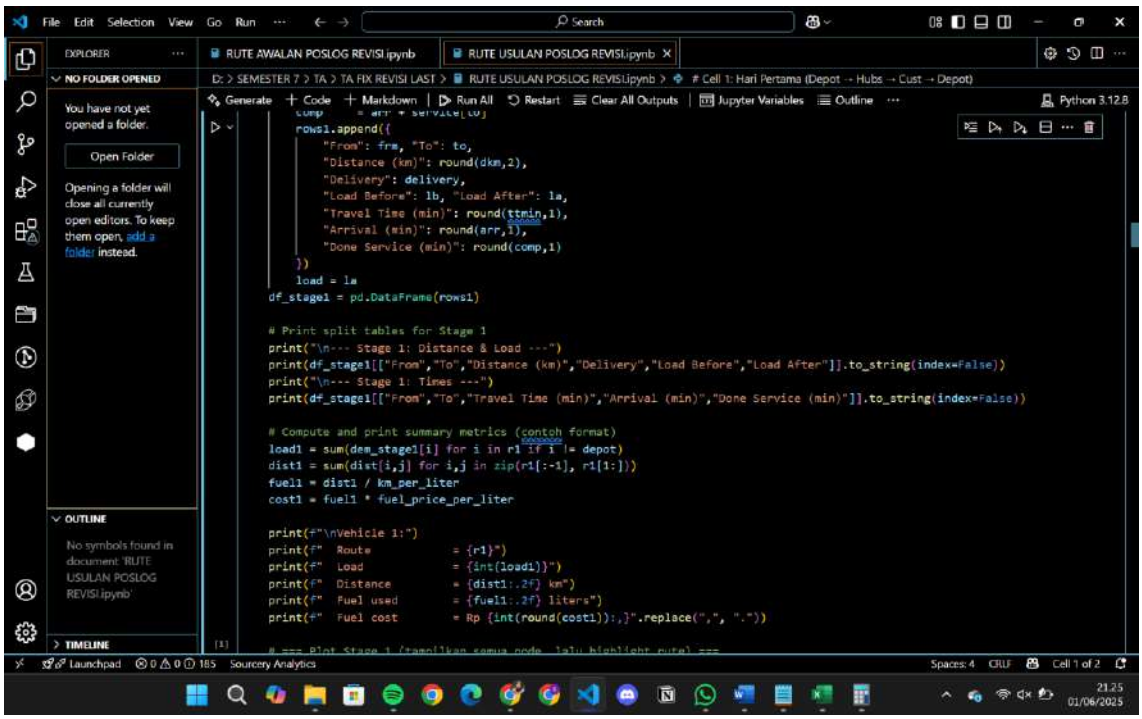
```
            route.append(nxt)
            times.append(t[nxt])
            cur = nxt

        back = times[-1] + service[cur] + travel_time[(cur, start)]
        if back > depot_latest:
            raise ValueError(f"Return at {back:.1f} > {depot_latest}")
        route.append(start)
        times.append(back)
        return route, times, ampl

    # === 4) Solve Stage 1 & build table ===
    nodes1, c1, tw1 = [depot] + stage1_cust + hubs, stage1_cust + hubs, stage1_cust
    r1, t1, ampl1   = solve_vrp(nodes1, c1, tw1, dem_stage1, start=depot)

    # Retrieve objective value
    obj1 = ampl1.getObjective("Obj").value()

    # Detailed Stage 1 table
    rows1 = []
    load = sum(dem_stage1[i] for i in r1 if i != depot)
    for i in range(len(r1)-1):
        frm, to = r1[i], r1[i+1]
        delivery = dem_stage1[to]
        dkm      = dist[frm, to]
        ttmin    = travel_time[(frm, to)]
        arr      = t1[i+1]
        lb, la   = load, load - delivery
        comp     = arr + service[to]
        rows1.append({
            "From": frm, "To": to,
            "Distance (km)": round(dkm,2),
```

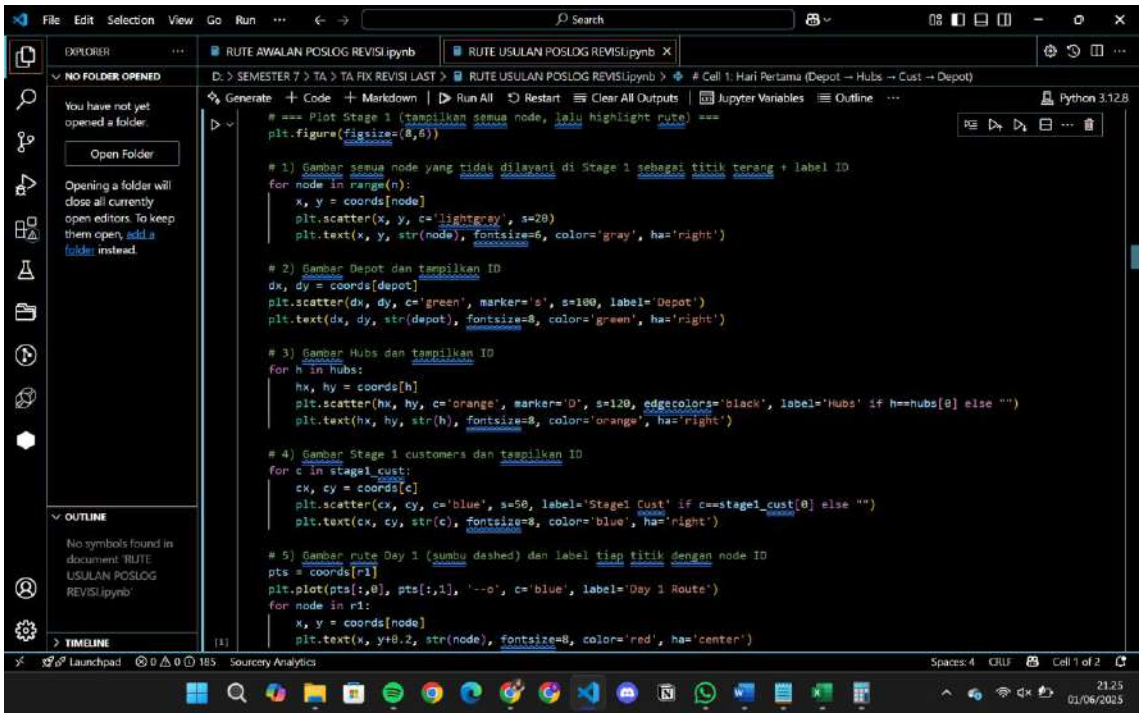


```
        rows1.append({
            "From": frm, "To": to,
            "Distance (km)": round(dkm,2),
            "Delivery": delivery,
            "Load Before": lb, "Load After": la,
            "Travel Time (min)": round(ttmin,1),
            "Arrival (min)": round(arr,1),
            "Done Service (min)": round(comp,1)
        })
        load = la
    df_stage1 = pd.DataFrame(rows1)

    # Print split tables for Stage 1
    print("\n--- Stage 1: Distance & Load ---")
    print(df_stage1[["From","To","Distance (km)","Delivery","Load Before","Load After"]].to_string(index=False))
    print("\n--- Stage 1: Times ---")
    print(df_stage1[["From","To","Travel Time (min)","Arrival (min)","Done Service (min)"]].to_string(index=False))

    # Compute and print summary metrics (contoh format)
    load1 = sum(dem_stage1[i] for i in r1 if i != depot)
    dist1 = sum(dist[i,j] for i,j in zip(r1[:-1], r1[1:]))
    fuel1 = dist1 / km_per_liter
    cost1 = fuel1 * fuel_price_per_liter

    print(f"\nVehicle 1:")
    print(f"  Route           = {r1}")
    print(f"  Load            = {int(load1)}")
    print(f"  Distance        = {dist1:.2f} km")
    print(f"  Fuel used       = {fuel1:.2f} liters")
    print(f"  Fuel cost       = Rp {int(round(cost1)):,}".replace(",", "."))
```



```
    # === Plot Stage 1 (tampilkan semua node, lalu highlight rute) ===
    plt.figure(figsize=(8,6))

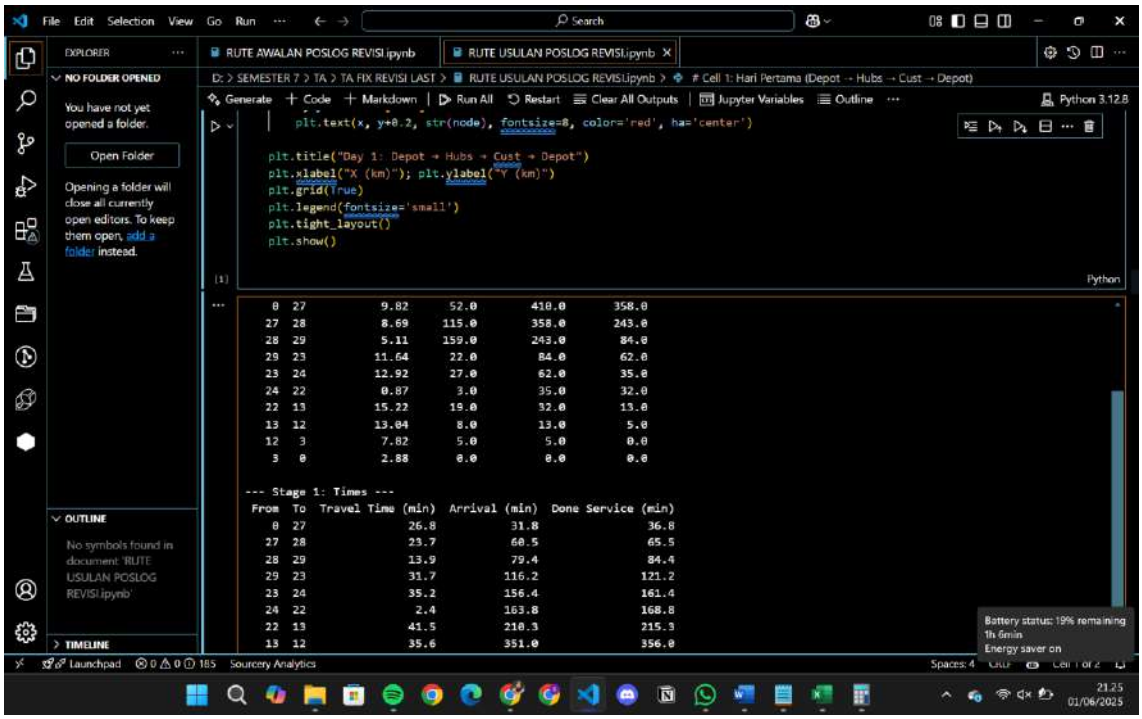
    # 1) Gambar semua node yang tidak dilayani di Stage 1 sebagai titik terang + label ID
    for node in range(n):
        x, y = coords[node]
        plt.scatter(x, y, c='lightgray', s=20)
        plt.text(x, y, str(node), fontsize=6, color='gray', ha='right')

    # 2) Gambar Depot dan tampilkan ID
    dx, dy = coords[depot]
    plt.scatter(dx, dy, c='green', marker='s', s=100, label='Depot')
    plt.text(dx, dy, str(depot), fontsize=8, color='green', ha='right')

    # 3) Gambar Hubs dan tampilkan ID
    for h in hubs:
        hx, hy = coords[h]
        plt.scatter(hx, hy, c='orange', marker='D', s=120, edgecolors='black', label='Hubs' if h==hubs[0] else "")
        plt.text(hx, hy, str(h), fontsize=8, color='orange', ha='right')

    # 4) Gambar Stage 1 customers dan tampilkan ID
    for c in stage1_cust:
        cx, cy = coords[c]
        plt.scatter(cx, cy, c='blue', s=50, label='Stage1 Cust' if c==stage1_cust[0] else "")
        plt.text(cx, cy, str(c), fontsize=8, color='blue', ha='right')

    # 5) Gambar rute Day 1 (sumbu dashed) dan label tiap titik dengan node ID
    pts = coords[r1]
    plt.plot(pts[:,0], pts[:,1], '--o', c='blue', label='Day 1 Route')
    for node in r1:
        x, y = coords[node]
        plt.text(x, y+0.2, str(node), fontsize=8, color='red', ha='center')
```



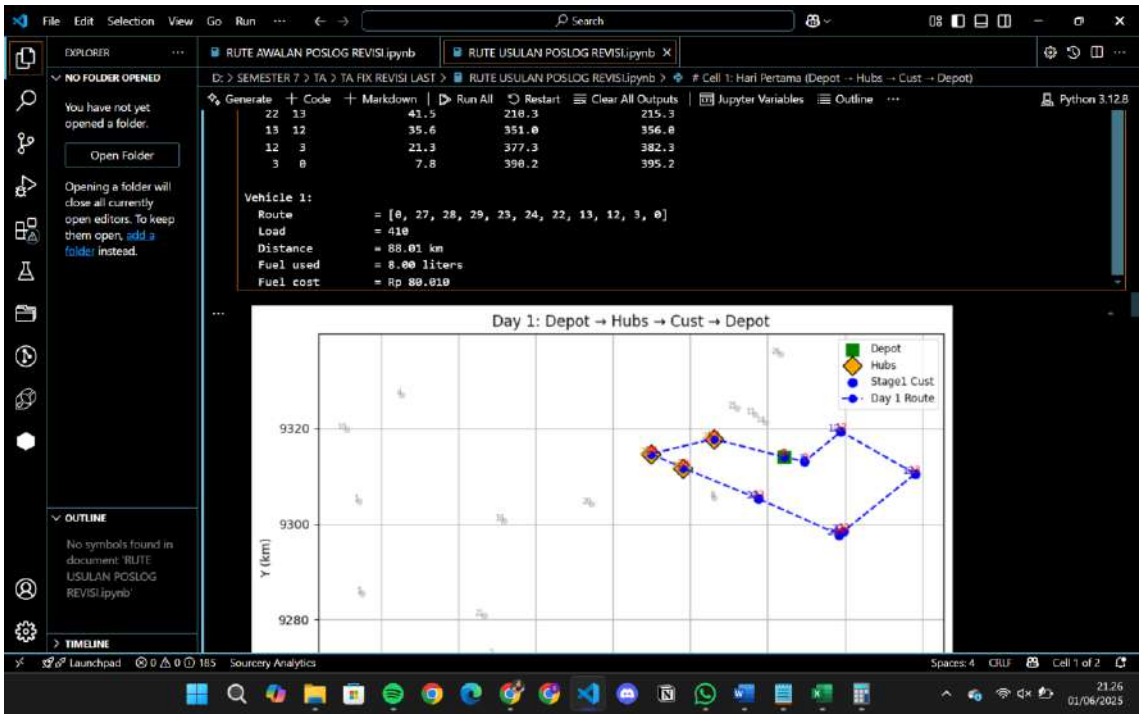
```
    plt.text(x, y+0.2, str(node), fontsize=8, color='red', ha='center')

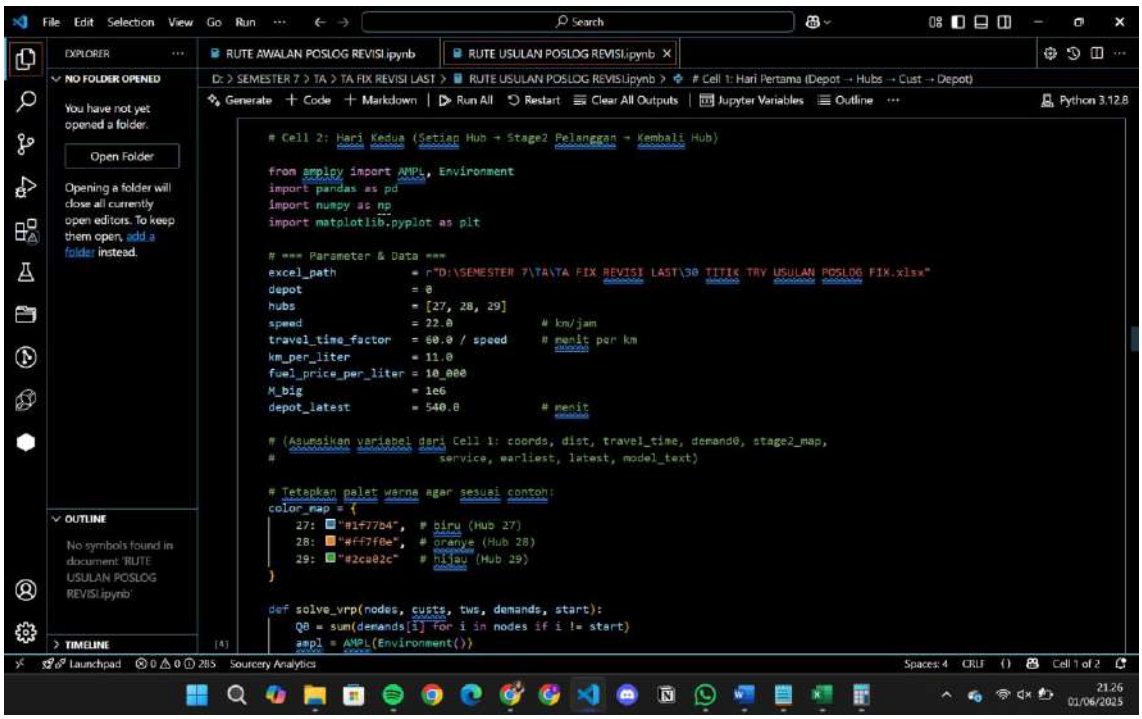
plt.title("Day 1: Depot → Hubs → Cust → Depot")
plt.xlabel("X (km)"); plt.ylabel("Y (km)")
plt.grid(True)
plt.legend(fontsize='small')
plt.tight_layout()
plt.show()
```

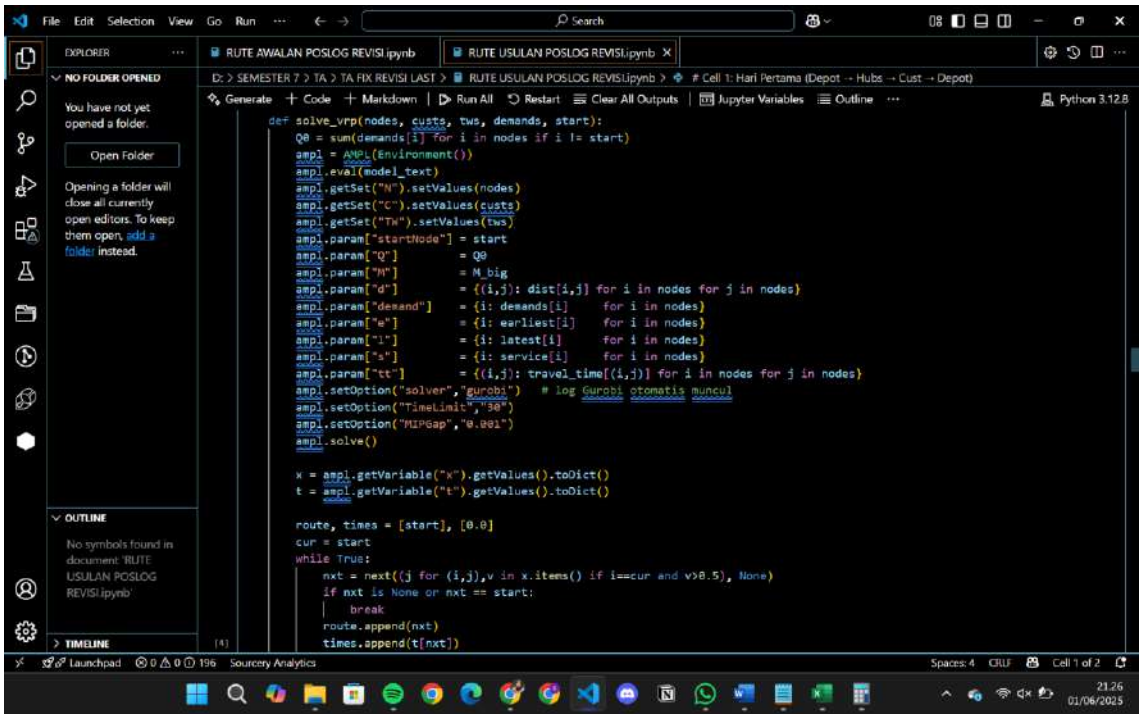
```
  0  27          9.82      52.0       410.0       358.0
 27  28          8.69     115.0       358.0       243.0
 28  29          5.11     159.0       243.0        84.0
 29  23         11.64      22.0        84.0        62.0
 23  24         12.92      27.0        62.0        35.0
 24  22          0.87       3.0        35.0        32.0
 22  13         15.22      19.0        32.0        13.0
 13  12         13.04       8.0        13.0         5.0
 12   3          7.82       5.0         5.0         0.0
  3   0          2.88       0.0         0.0         0.0

--- Stage 1: Times ---
 From  To  Travel Time (min)  Arrival (min)  Done Service (min)
    0  27              26.8           31.8                36.8
   27  28              23.7           60.5                65.5
   28  29              13.9           79.4                84.4
   29  23              31.7          116.2               121.2
   23  24              35.2          156.4               161.4
   24  22               2.4          163.8               168.8
   22  13              41.5          210.3               215.3
   13  12              35.6          351.0               356.0
```

Battery status: 19% remaining
1h 6min
Energy saver on


File Edit Selection View Go Run
Search
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Opening a folder will close all currently open editors. To keep them open, add a folder instead.
OUTLINE
No symbols found in document 'RUTE USULAN POSLOG REVISI.ipynb'
TIMELINE
RUTE AWALAN POSLOG REVISI.ipynb
RUTE USULAN POSLOG REVISI.ipynb
D: > SEMESTER 7 > TA > TA FIX REVISI LAST > RUTE USULAN POSLOG REVISI.ipynb > # Cell 1: Hari Pertama (Depot → Hubs → Cust → Depot)
Generate Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline
Python 3.12.8
22 13 41.5 210.3 215.3
13 12 35.6 351.0 356.8
12 3 21.3 377.3 382.3
3 0 7.8 390.2 395.2
Vehicle 1:
Route = [0, 27, 28, 29, 23, 24, 22, 13, 12, 3, 0]
Load = 410
Distance = 88.01 km
Fuel used = 8.00 liters
Fuel cost = Rp 80.010
Day 1: Depot → Hubs → Cust → Depot
Depot
Hubs
Stage1 Cust
Day 1 Route
9320
9300
9280
Y (km)
Launchpad
Sourcery Analytics
Spaces: 4 CRLF Cell 1 of 2
21.26
01/06/2025

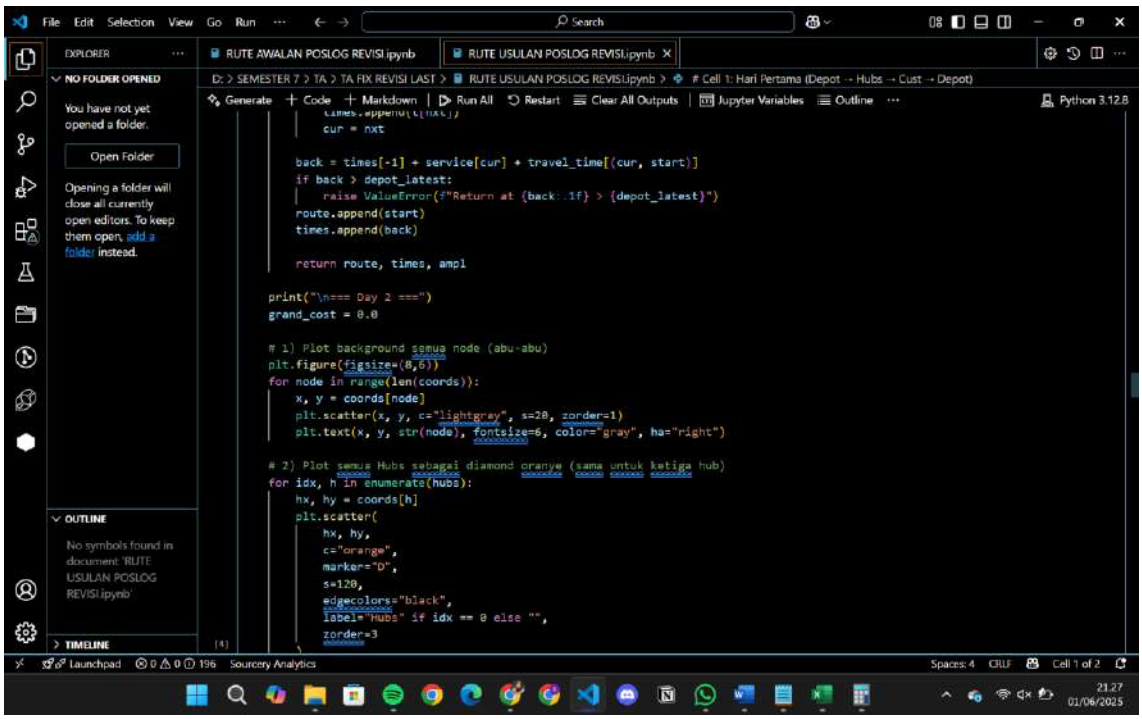

File Edit Selection View Go Run
Search
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Opening a folder will close all currently open editors. To keep them open, add a folder instead.
OUTLINE
No symbols found in document 'RUTE USULAN POSLOG REVISI.ipynb'
TIMELINE
RUTE AWALAN POSLOG REVISI.ipynb
RUTE USULAN POSLOG REVISI.ipynb
D: > SEMESTER 7 > TA > TA FIX REVISI LAST > RUTE USULAN POSLOG REVISI.ipynb > # Cell 1: Hari Pertama (Depot → Hubs → Cust → Depot)
Generate Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline
Python 3.12.8
# Cell 2: Hari Kedua (Setiap Hub → Stage2 Pelanggan → Kembali Hub)
from amplpy import AMPL, Environment
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
# === Parameter & Data ===
excel_path = r"D:\SEMESTER 7\TA\TA FIX REVISI LAST\30 TITIK TRY USULAN POSLOG FIX.xlsx"
depot = 0
hubs = [27, 28, 29]
speed = 22.0 # km/jam
travel_time_factor = 60.0 / speed # menit per km
km_per_liter = 11.0
fuel_price_per_liter = 10_000
M_big = 1e6
depot_latest = 540.0 # menit
# (Asumsikan variabel dari Cell 1: coords, dist, travel_time, demand0, stage2_map,
# service, earliest, latest, model_text)
# Tetapkan palet warna agar sesuai contoh:
color_map = {
27: "#1f77b4", # biru (Hub 27)
28: "#ff7f0e", # oranye (Hub 28)
29: "#2ca02c" # hijau (Hub 29)
}
def solve_vrp(nodes, custs, tws, demands, start):
Q0 = sum(demands[i] for i in nodes if i != start)
ampl = AMPL(Environment())
Launchpad
Sourcery Analytics
Spaces: 4 CRLF Cell 1 of 2
21.26
01/06/2025



```
    def solve_vrp(nodes, custs, tws, demands, start):
        Q0 = sum(demands[i] for i in nodes if i != start)
        ampl = AMPL(Environment())
        ampl.eval(model_text)
        ampl.getSet("N").setValues(nodes)
        ampl.getSet("C").setValues(custs)
        ampl.getSet("TW").setValues(tws)
        ampl.param["startNode"] = start
        ampl.param["Q"]         = Q0
        ampl.param["M"]         = M_big
        ampl.param["d"]         = {(i,j): dist[i,j] for i in nodes for j in nodes}
        ampl.param["demand"]    = {i: demands[i]     for i in nodes}
        ampl.param["e"]         = {i: earliest[i]    for i in nodes}
        ampl.param["l"]         = {i: latest[i]      for i in nodes}
        ampl.param["s"]         = {i: service[i]     for i in nodes}
        ampl.param["tt"]        = {(i,j): travel_time[(i,j)] for i in nodes for j in nodes}
        ampl.setOption("solver","gurobi")   # log Gurobi otomatis muncul
        ampl.setOption("TimeLimit","30")
        ampl.setOption("MIPGap","0.001")
        ampl.solve()

        x = ampl.getVariable("x").getValues().toDict()
        t = ampl.getVariable("t").getValues().toDict()

        route, times = [start], [0.0]
        cur = start
        while True:
            nxt = next((j for (i,j),v in x.items() if i==cur and v>0.5), None)
            if nxt is None or nxt == start:
                break
            route.append(nxt)
            times.append(t[nxt])
```



```
            times.append(t[nxt])
            cur = nxt

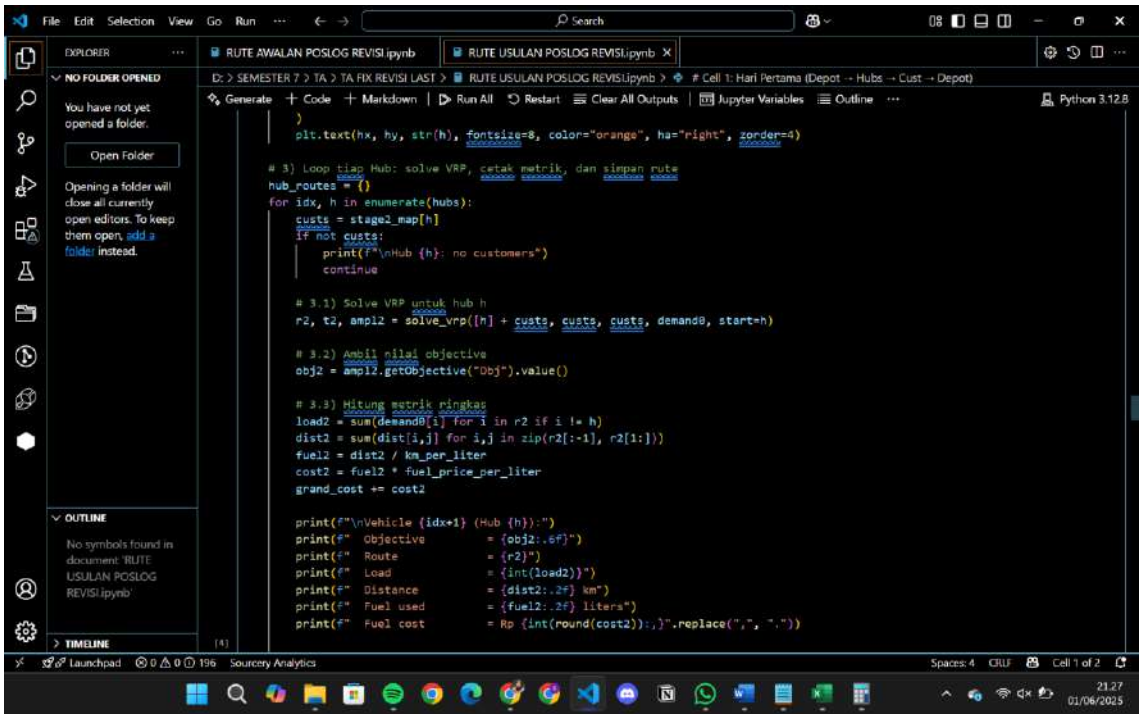
        back = times[-1] + service[cur] + travel_time[(cur, start)]
        if back > depot_latest:
            raise ValueError(f"Return at {back:.1f} > {depot_latest}")
        route.append(start)
        times.append(back)

        return route, times, ampl

    print("\n=== Day 2 ===")
    grand_cost = 0.0

    # 1) Plot background semua node (abu-abu)
    plt.figure(figsize=(8,6))
    for node in range(len(coords)):
        x, y = coords[node]
        plt.scatter(x, y, c="lightgrey", s=28, zorder=1)
        plt.text(x, y, str(node), fontsize=6, color="gray", ha="right")

    # 2) Plot semua Hubs sebagai diamond oranye (sama untuk ketiga hub)
    for idx, h in enumerate(hubs):
        hx, hy = coords[h]
        plt.scatter(
            hx, hy,
            c="orange",
            marker="D",
            s=120,
            edgecolors="black",
            label="Hubs" if idx == 0 else "",
            zorder=3
```



```
        )
        plt.text(hx, hy, str(h), fontsize=8, color="orange", ha="right", zorder=4)

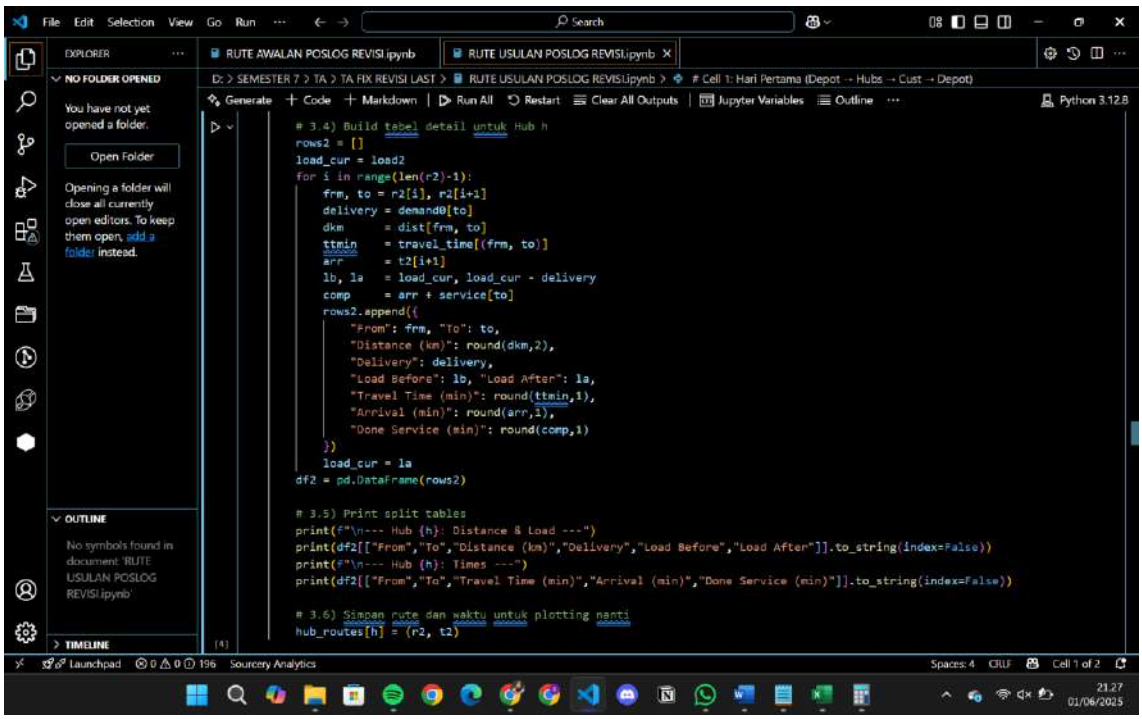
    # 3) Loop tiap Hub: solve VRP, cetak metrik, dan simpan rute
    hub_routes = {}
    for idx, h in enumerate(hubs):
        custs = stage2_map[h]
        if not custs:
            print(f"\nHub {h}: no customers")
            continue

        # 3.1) Solve VRP untuk hub h
        r2, t2, ampl2 = solve_vrp([h] + custs, custs, custs, demand0, start=h)

        # 3.2) Ambil nilai objective
        obj2 = ampl2.getObjective("Dbj").value()

        # 3.3) Hitung metrik ringkas
        load2 = sum(demand0[i] for i in r2 if i != h)
        dist2 = sum(dist[i,j] for i,j in zip(r2[:-1], r2[1:]))
        fuel2 = dist2 / km_per_liter
        cost2 = fuel2 * fuel_price_per_liter
        grand_cost += cost2

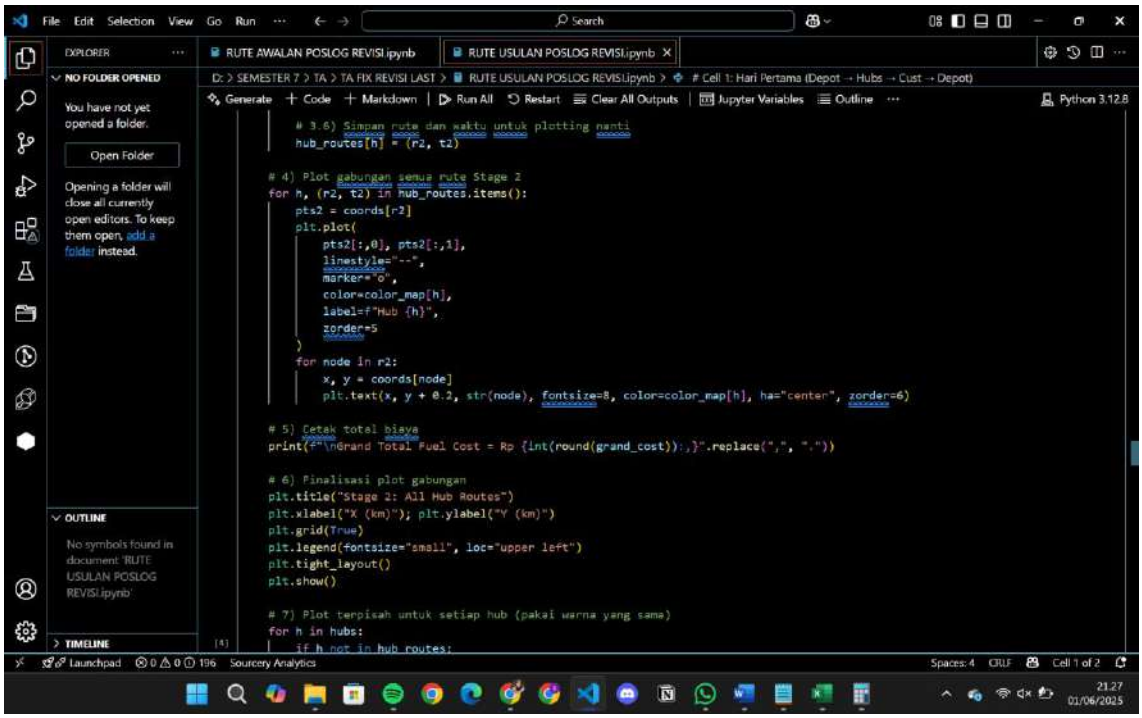
        print(f"\nVehicle {idx+1} (Hub {h}):")
        print(f"  Objective      = {obj2:.6f}")
        print(f"  Route          = {r2}")
        print(f"  Load           = {int(load2)}")
        print(f"  Distance       = {dist2:.2f} km")
        print(f"  Fuel used      = {fuel2:.2f} liters")
        print(f"  Fuel cost      = Rp {int(round(cost2)):,}".replace(",", "."))
```



```
    # 3.4) Build tabel detail untuk Hub h
    rows2 = []
    load_cur = load2
    for i in range(len(r2)-1):
        frm, to = r2[i], r2[i+1]
        delivery = demand0[to]
        dkm      = dist[frm, to]
        ttmin    = travel_time[(frm, to)]
        arr      = t2[i+1]
        lb, la   = load_cur, load_cur - delivery
        comp     = arr + service[to]
        rows2.append({
            "From": frm, "To": to,
            "Distance (km)": round(dkm,2),
            "Delivery": delivery,
            "Load Before": lb, "Load After": la,
            "Travel Time (min)": round(ttmin,1),
            "Arrival (min)": round(arr,1),
            "Done Service (min)": round(comp,1)
        })
        load_cur = la
    df2 = pd.DataFrame(rows2)

    # 3.5) Print split tables
    print(f"\n--- Hub {h}: Distance & Load ---")
    print(df2[["From","To","Distance (km)","Delivery","Load Before","Load After"]].to_string(index=False))
    print(f"\n--- Hub {h}: Times ---")
    print(df2[["From","To","Travel Time (min)","Arrival (min)","Done Service (min)"]].to_string(index=False))

    # 3.6) Simpan rute dan waktu untuk plotting nanti
    hub_routes[h] = (r2, t2)
```



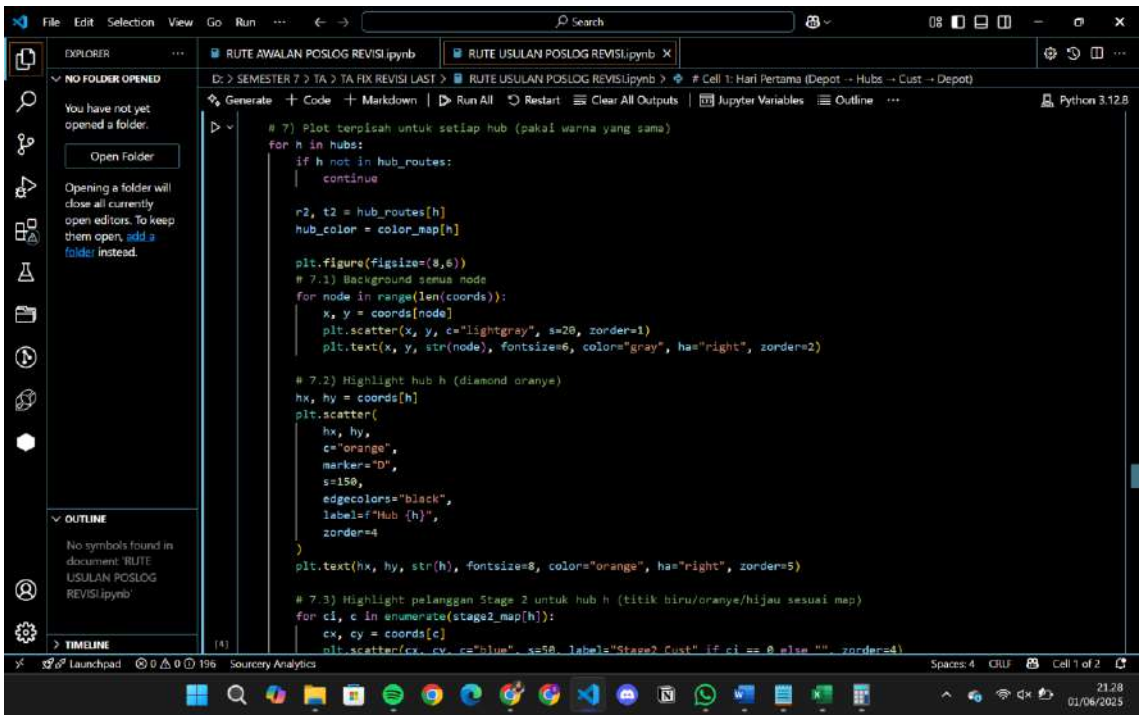
```
        # 3.6) Simpan rute dan waktu untuk plotting nanti
        hub_routes[h] = (r2, t2)

    # 4) Plot gabungan semua rute Stage 2
    for h, (r2, t2) in hub_routes.items():
        pts2 = coords[r2]
        plt.plot(
            pts2[:,0], pts2[:,1],
            linestyle="--",
            marker="o",
            color=color_map[h],
            label=f"Hub {h}",
            zorder=5
        )
        for node in r2:
            x, y = coords[node]
            plt.text(x, y + 0.2, str(node), fontsize=8, color=color_map[h], ha="center", zorder=6)

    # 5) Cetak total biaya
    print(f"\nGrand Total Fuel Cost = Rp {int(round(grand_cost)):,}".replace(",", "."))

    # 6) Finalisasi plot gabungan
    plt.title("Stage 2: All Hub Routes")
    plt.xlabel("X (km)"); plt.ylabel("Y (km)")
    plt.grid(True)
    plt.legend(fontsize="small", loc="upper left")
    plt.tight_layout()
    plt.show()

    # 7) Plot terpisah untuk setiap hub (pakai warna yang sama)
    for h in hubs:
        if h not in hub_routes:
```



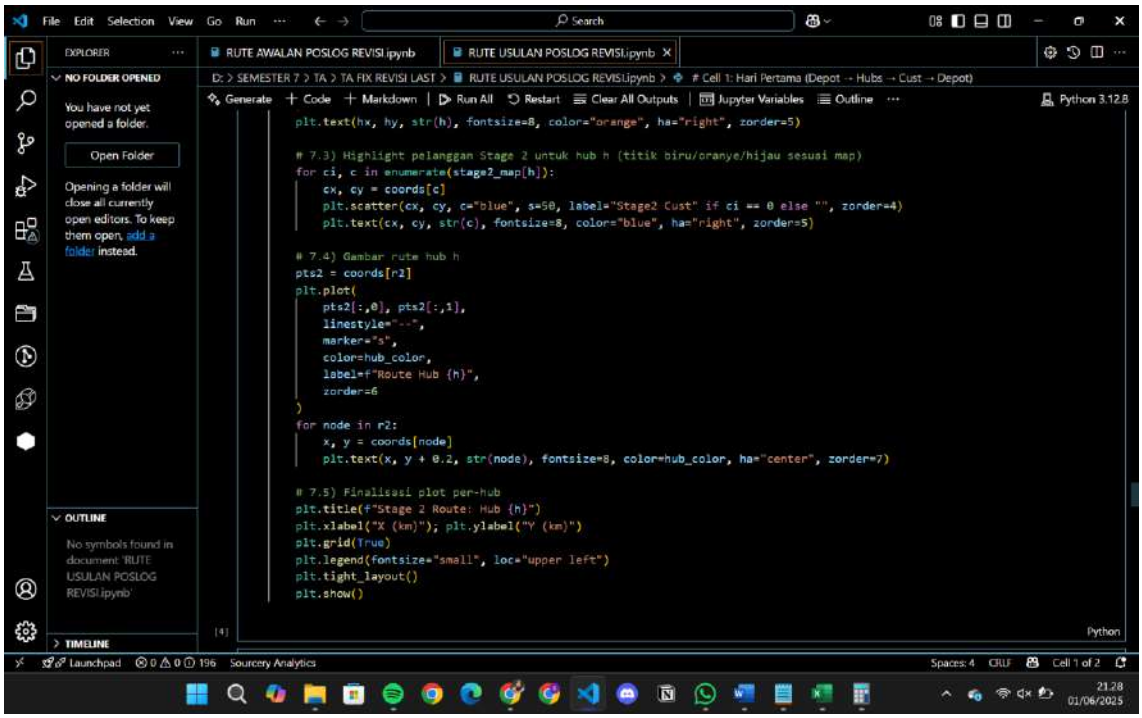
```
    # 7) Plot terpisah untuk setiap hub (pakai warna yang sama)
    for h in hubs:
        if h not in hub_routes:
            continue

        r2, t2 = hub_routes[h]
        hub_color = color_map[h]

        plt.figure(figsize=(8,6))
        # 7.1) Background semua node
        for node in range(len(coords)):
            x, y = coords[node]
            plt.scatter(x, y, c="lightgray", s=20, zorder=1)
            plt.text(x, y, str(node), fontsize=6, color="gray", ha="right", zorder=2)

        # 7.2) Highlight hub h (diamond oranye)
        hx, hy = coords[h]
        plt.scatter(
            hx, hy,
            c="orange",
            marker="D",
            s=150,
            edgecolors="black",
            label=f"Hub {h}",
            zorder=4
        )
        plt.text(hx, hy, str(h), fontsize=8, color="orange", ha="right", zorder=5)

        # 7.3) Highlight pelanggan Stage 2 untuk hub h (titik biru/oranye/hijau sesuai map)
        for ci, c in enumerate(stage2_map[h]):
            cx, cy = coords[c]
            plt.scatter(cx, cy, c="blue", s=50, label="Stage2 Cust" if ci == 0 else "", zorder=4)
```

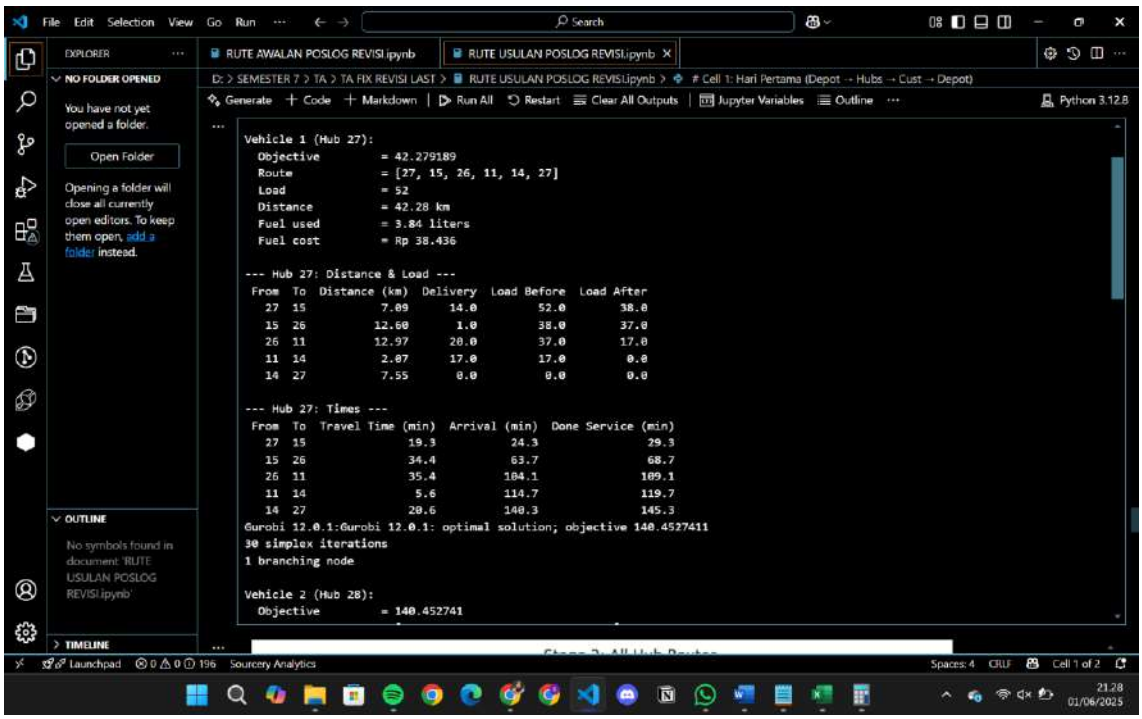


```
    plt.text(hx, hy, str(h), fontsize=8, color="orange", ha="right", zorder=5)

    # 7.3) Highlight pelanggan Stage 2 untuk hub h (titik biru/oranye/hijau sesusi map)
    for ci, c in enumerate(stage2_map[h]):
        cx, cy = coords[c]
        plt.scatter(cx, cy, c="blue", s=50, label="Stage2 Cust" if ci == 0 else "", zorder=4)
        plt.text(cx, cy, str(c), fontsize=8, color="blue", ha="right", zorder=5)

    # 7.4) Gambar rute hub h
    pts2 = coords[r2]
    plt.plot(
        pts2[:,0], pts2[:,1],
        linestyle="--",
        marker="s",
        color=hub_color,
        label=f"Route Hub {h}",
        zorder=6
    )
    for node in r2:
        x, y = coords[node]
        plt.text(x, y + 0.2, str(node), fontsize=8, color=hub_color, ha="center", zorder=7)

    # 7.5) Finalisasi plot per-hub
    plt.title(f"Stage 2 Route: Hub {h}")
    plt.xlabel("X (km)"); plt.ylabel("Y (km)")
    plt.grid(True)
    plt.legend(fontsize="small", loc="upper left")
    plt.tight_layout()
    plt.show()
```



```
Vehicle 1 (Hub 27):
  Objective         = 42.279189
  Route             = [27, 15, 26, 11, 14, 27]
  Load              = 52
  Distance          = 42.28 km
  Fuel used         = 3.84 liters
  Fuel cost         = Rp 38.436

--- Hub 27: Distance & Load ---
 From  To  Distance (km)  Delivery  Load Before  Load After
   27  15          7.09      14.0         52.0        38.0
   15  26         12.60       1.0         38.0        37.0
   26  11         12.97      20.0         37.0        17.0
   11  14          2.07      17.0         17.0         0.0
   14  27          7.55       0.0          0.0         0.0

--- Hub 27: Times ---
 From  To  Travel Time (min)  Arrival (min)  Done Service (min)
   27  15              19.3           24.3                29.3
   15  26              34.4           63.7                68.7
   26  11              35.4          104.1               109.1
   11  14               5.6          114.7               119.7
   14  27              20.6          140.3               145.3
Gurobi 12.0.1:Gurobi 12.0.1: optimal solution; objective 140.4527411
30 simplex iterations
1 branching node

Vehicle 2 (Hub 28):
  Objective         = 140.452741
```