

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Permasalahan VRP merupakan permasalahan penting dibidang distribusi, logistik, dan transportasi. Banyak *literature* yang membahas tentang permasalahan ini. Penelitian terdahulu menampilkan beberapa penelitian yang terkait dengan *Vehicle Routing Problem with Time Windows* (VRPTW), Algoritma *Sweep*, dan *Mixed Integer Linear Programming* (MILP). Hal ini berguna sebagai bahan rujukan dan pembandingan antara penelitian yang telah dilakukan sebelumnya dengan penelitian yang akan dilakukan oleh peneliti.

Kallehauge et al. (2001) dalam *technical reportnya* yang berjudul *Lagrangian Duality Applied on Vehicle Routing Problem with Time Windows Experimental result* merumuskan model matematis dengan fungsi tujuan meminimalkan biaya rute perjalanan. Model matematis yang dikembangkan memiliki batasan antara lain: setiap *customer* harus dikunjungi tepat satu kali, permintaan tidak boleh melebihi kapasitas kendaraan, rute berawal dan berakhir di depot dimana setelah mengunjungi satu *customer* kendaraan akan pergi meninggalkan *customer* tersebut, waktu penjadwalan fisibel, memenuhi batasan *time windows*, dan variabel keputusan yang merupakan bilangan biner. Kallehauge et al. (2001) membentuk *Langrangian Relaxation* dimana kemudian diselesaikan menggunakan algoritma *cutting plane* yang dikombinasikan dengan algoritma *Dantzig-Wolfe* untuk menyelesaikan permasalahan perbandingan yang sebelumnya dikemukakan Solomon dan Homberger. Hasil menunjukkan bahwa, penelitian ini

mampu menyelesaikan 14 permasalahan Solomon yang belum terpecahkan serta menyelesaikan permasalahan Homberger dengan 1000 *customer*.

Dondo & Cerda (2007) dalam jurnalnya yang berjudul *A cluster-based optimization approach for multi-depot heterogeneous fleet vehicle routing problem with time windows* mengemukakan pendekatan algoritma/heuristic baru yang terdiri dari tiga fase atau pendekatan *a three-phase hierarchical hybrid* untuk permasalahan VRPTW dengan multi depot dan kendaraan pengangkut yang berbeda. Fase pertama terdiri dari pengelompokan *customer* atau *node* ke dalam kelompok rute yang fisibel dengan biaya efektif. Fase kedua terdiri dari pengaplikasian MILP dengan *Branch and Bound* untuk menemukan set optimal dalam penugasan dan urutan kendaraan dari rute secara efisien. Fase ketiga terdiri dari pemisahan kelompok *node* menjadi *node* yang asli dengan memecahkan formulasi umum VRPTW yang telah dimodelkan ke dalam MILP. Hasil numerik menunjukkan bahwa metode yang dikembangkan Dondo & Cerda (2007) yakni pengelompokan berdasarkan metode optimasi, terbukti cukup berhasil menyelesaikan contoh permasalahan VRPTW dengan multi depot dan kendaraan pengangkut yang dikemukakan Solomon.

Azi et al. (2007) dalam penelitiannya yang berjudul *An exact algorithm for a single-vehicle routing problem with time windows and multiple routes* mendeskripsikan sebuah algoritma untuk menyelesaikan VRPTW dimana satu kendaraan mampu melakukan beberapa rute. Algoritma matematis yang dikembangkan oleh Azi et al. (2007) memiliki tujuan meminimalkan total jarak yang ditempuh untuk melayani semua *customer* dengan memenuhi batasan

kapasitas, *time windows*, dan *deadline* pengiriman. Metode yang digunakan untuk menyelesaikan model ini adalah algoritma dasar jalur terpendek yang dibagi menjadi dua fase. Fase pertama digunakan untuk membangun rute yang fisibel yang akan digunakan sebagai *building block* untuk fase kedua. Sementara fase kedua digunakan menggabungkan hasil rute pada fase pertama dalam bentuk hari kerja kendaraan. Penelitian ini memberikan hasil bahwa algoritma ini sangat sensitif dengan batasan *deadline*. Ketika batasan *deadline* tidak cukup sempit, maka banyaknya rute fisibel akan membludak dan terlalu besar untuk diselesaikan algoritma.

Suthikarnnarunai (2008) melakukan penelitian untuk menentukan rute bus kampus bagi staf University of the Thai Chamber of Commerce (UTCC) yang efisien dan mampu melayani semua *customer*. Penelitian ini diawali dengan melakukan *cluster* berdasarkan Algoritma *Sweep* dilanjutkan penentuan rute menggunakan model *Integer Programming* (IP) pada *cluster* yang terbentuk berdasarkan Algoritma *Sweep*. *2-Opt exchanges* kemudian diterapkan untuk memperbaiki *cluster* yang terbentuk dan meningkatkan solusi. Hasil penelitian heuristik ini kemudian dibandingkan dengan metode eksak. Pada jam kerja pagi, hasil menunjukkan bahwa metode heuristik yang terdiri dari Algoritma *Sweep*, IP, dan *2-opt exchanges* cukup baik dan membutuhkan waktu komputasi yang lebih sedikit jika dibandingkan dengan metode eksak. Sementara pada jam kerja sore, metode eksak tidak dapat diterapkan karena permasalahan terlalu besar.

Priyandari et al. (2011) melakukan penelitian pada PT. Pupuk Sriwidjaja (Pusri) di Karanganyar untuk menentukan rute pengiriman pupuk dari distributor

ke sejumlah pengecer. Solusi rute pengiriman dalam penelitian ini menggunakan lintasan jalan pada peta digital karena dianggap menggambarkan jarak tempuh yang sebenarnya dilalui kendaraan. Model rute pengiriman memiliki tujuan meminimasi total biaya transportasi menggunakan MILP. Hasil dari penelitian ini menunjukkan bahwa model usulan mampu menghemat biaya 2.28% dari sistem awal. Model ini memiliki keterbatasan karena belum memasukkan gudang Pusri sebagai fasilitas perantara (*intermediate*) antara garasi distributor dengan pengecer, sehingga perlu dilakukan koreksi manual. Hasil koreksi manual menunjukkan bahwa sistem usulan mampu menghemat biaya 4.17%.

Suwansuksamran & Ongkunaruk (2013) melakukan penelitian VRPTW pada perusahaan bumbu bubuk yang ada di Thailand menggunakan metode *Mixed Integer Programming* (MIP). Tujuan dari penelitian ini yakni untuk mengurangi biaya transportasi dan waktu penjadwalan proses distribusi yang dilakukan oleh pihak ketiga atau *third party logistics* (3PLs) yang memiliki 690 *customer*. Peneliti mengelompokkan lokasi *customer* menjadi 6 grup untuk mempermudah perhitungan, kemudian dilakukan pemodelan matematis menggunakan MIP untuk diselesaikan dengan *software* optimasi IBM ILOG CPLEX versi 12.4. Hasil dari penelitian ini kemudian dibandingkan dengan penelitian yang sebelumnya dilakukan oleh Ongraj & Ongkunaruk (2013), yakni penggunaan IP untuk mengurangi total biaya pada permasalahan *bin packing with time windows* dengan menyerahkan masalah penentuan rute pada supir. Hasil penelitian menunjukkan bahwa alokasi terhadap *customer* sama dengan penelitian sebelumnya, total pengiriman turun 23%, *overtime* disebabkan penjadwalan manual turun 50% atau

2 jam per hari sehingga supir tidak perlu membuang waktu dengan menentukan rute tersendiri.

Cahyaningsih (2015) melakukan penelitian dengan tujuan meminimalkan jarak tempuh kendaraan pada rute distribusi surat kabar Kedaulatan Rakyat di wilayah kabupaten Sleman, DIY menggunakan Algoritma *Sweep* dengan tipe permasalahan *Capacitated Vehicle Routing Problem (CVRP)*. Pada penelitian ini *customer* dikelompokkan terlebih dahulu berdasarkan sudut polar, kemudian ditentukan masing-masing rute setiap *cluster* dengan metode *Nearest Neighbour*. Hasil penelitian menunjukkan bahwa rute usulan 32 km lebih sedikit dari rute awalan dan waktu tempuh kendaraan berkurang 23 menit. Presentase penghematan jarak tempuh kendaraan sebesar 18.29%.

Savitri (2017) melakukan penelitian pada CV. Jogja Transport untuk menentukan rute kendaraan yang optimal dengan jarak tempuh terpendek. Penelitian dilakukan menggunakan metode heuristik *cluster first route second*, dimana pengclustering dilakukan dengan modifikasi Algoritma *Sweep* kemudian dilanjutkan penyelesaian TSP untuk menentukan rute masing-masing *cluster* dengan metode MILP.

Rangkuman penelitian terdahulu dengan penelitian saat ini dapat dilihat pada tabel sebagai berikut:

Tabel 2.1. Posisi Penelitian Terdahulu

No	Peneliti	Judul	Metode	Hasil
1	Kallehauge et al. (2001)	<i>Lagrangian Duality Applied on Vehicle Routing Problem with Time Windows Experimental result</i>	VRPTW, <i>Lagrangian Duality</i> , Algoritma Dantzig-Wolfe	Penelitian ini mampu menyelesaikan empat belas permasalahan Solomon yang belum terpecahkan serta

				menyelesaikan permasalahan Homberger dengan 1000 <i>customer</i> .
2	Dondo & Cerda (2007)	<i>A cluster-based optimization approach for multi-depot heterogeneous fleet vehicle routing problem with time windows</i>	<i>Multi-depot routing problem with time windows and heterogeneous vehicles, A three-phase hierarchical hybrid, MILP, Branch and Bound</i>	Hasil numerik menunjukkan bahwa pengelompokan <i>node</i> berdasarkan metode optimasi, terbukti cukup berhasil menyelesaikan contoh permasalahan VRPTW dengan multi depot dan kendaraan pengangkut yang dikemukakan Solomon.
3	Azi et al. (2007)	<i>An exact algorithm for a single-vehicle routing problem with time windows and multiple routes</i>	VRPTW, Algoritma Jalur Terpendek	Penelitian ini memberikan hasil bahwa algoritma jalur terpendek sangat sensitif dengan batasan <i>deadline</i> . Ketika batasan <i>deadline</i> tidak cukup sempit, maka banyaknya rute <i>feasible</i> akan membludak dan terlalu besar untuk diselesaikan algoritma.
4	Suthikarnnarunai (2008)	<i>A Sweep Algorithm for the Mix Fleet Vehicle Routing Problem</i>	Algoritma <i>Sweep, Integer Programming,</i>	Metode heuristic yang terdiri dari <i>Algoritma Sweep, IP,</i>

			<i>2-Opt Exchange</i>	dan <i>2-opt exchange</i> memberikan hasil yang cukup baik dibanding metode eksak dengan waktu komputasi yang jauh lebih sedikit.
5	Priyandari et al. (2011)	Penentuan Rute Pengiriman Pupuk Urea Bersubsidi di Karanganyar	VRPTW, MILP	Sistem usulan menggunakan model menghemat biaya sebesar 2.28%, dan koreksi manual terhadap solusi model, didapat penghematan sebesar 4.17%.
6	Suwansuksamran & Ongkunaruk (2013)	<i>A Mixed Integer Programming For A Vehicle Routing Problem With Time Windows: A Case Study Of A Thai Seasoning Company</i>	VRPTW, MIP	Rute yang dihasilkan mampu menurunkan total pengiriman sebesar 23% dan menurunkan <i>overtime</i> sebesar 50% yang disebabkan penjadwalan manual.
7	Cahyaningsih (2015)	Penyelesaian <i>Capacitated Vehicle Routing Problem</i> (CVRP) Menggunakan Algoritma <i>Sweep</i> Untuk Optimasi Rute Distribusi Surat Kabar Kedaulatan Rakyat	Algoritma <i>Sweep</i> , <i>Nearest Neighbour</i>	Rute usulan 32 km lebih sedikit dari rute awalan dan waktu tempuh kendaraan berkurang 23 menit. Presentase penghematan jarak tempuh kendaraan sebesar 18.29%.
8	Savitri (2017)	Pemodelan <i>Vehicle Routing Problem With Time Windows</i> untuk Mengoptimasi Rute	VRPTW, Modifikasi Algoritma <i>Sweep</i> , MILP	Rute pengiriman produk Sari Roti di wilayah Bantul, DIY

		Distribusi Produk Sari Roti dengan Metode Algoritma <i>Sweep</i> dan <i>Mixed Integer Linear Programming</i> (Studi Kasus Pada Cv. Jogja Transport)		dengan jarak tempuh minimum.
--	--	---	--	------------------------------

2.2 Transportasi

Salim dalam bukunya yang berjudul Manajemen Transportasi mengemukakan definisi transportasi secara umum, yakni “rangkaian kegiatan memindahkan/mengangkut barang dari produsen sampai kepada konsumen dengan menggunakan salah satu moda transportasi, yang dapat meliputi moda transportasi darat, laut/sungai maupun udara” (Salim, 1993, hlm. 27)

Metode transportasi berkembang menjadi metode penyusunan angkutan untuk mengirimkan barang atau jasa dari satu atau beberapa sumber yang kemudian didistribusikan ke beberapa tempat atau lokasi yang membutuhkan barang atau jasa sesuai dengan kapasitas permintaannya. Metode transportasi memperhitungkan biaya perjalanan yang dikeluarkan oleh perusahaan selama proses pendistribusian dimana tujuannya adalah meminimalisir biaya tersebut (Kakiay, 2008)

Transportasi dalam bidang pengangkutan barang memegang peran penting dan mendapat perhatian khusus dalam manajemen logistik dan *supply chain* perdagangan di masa ini dimana kegiatan ini mendukung aktivitas ekonomi maupun sosial. Oleh sebab itu, banyak perusahaan menggunakan cara yang rasional dan *tools* yang efektif untuk mengurangi biaya transportasi sehingga dapat menekan pengeluaran perusahaan (Yousefikhoshbakht, 2015).

2.3 *Vehicle Routing Problem*

2.3.1 *Vehicle Routing Problem*

Vehicle Routing Problem (VRP) dapat dideskripsikan sebagai permasalahan perancangan rute pengiriman yang optimal atau kumpulan rute dari satu atau beberapa depot menuju beberapa kota atau *customer* yang tersebar secara geografis dengan dibatasi beberapa kendala. VRP terdiri dari merancang rute dengan biaya minimal sehingga setiap kota atau *customer* hanya dikunjungi tepat satu kali oleh satu kendaraan, dimana semua rute kendaraan dimulai dan berakhir di depot, serta terdapat beberapa kendala yang harus dipenuhi (Laporte, 1992).

Menurut Adewuni & Adeleke (2016), VRP klasik didefinisikan sebagai menentukan rute optimal dari beberapa kendaraan yang terletak di depot, mengirimkan suatu produk ke beberapa *customer* yang berbeda lokasi. VRP merujuk pada permasalahan truk pengiriman yang umumnya dijumpai pada organisasi dengan sistem operasi yang kompleks. Perhatian utama dalam permasalahan ini adalah menemukan rute yang optimal untuk lokasi yang berbeda khususnya dengan melihat kenyataan bahwa biaya bahan bakar dan supir akan meningkat seiring dengan peningkatan waktu untuk pengiriman.

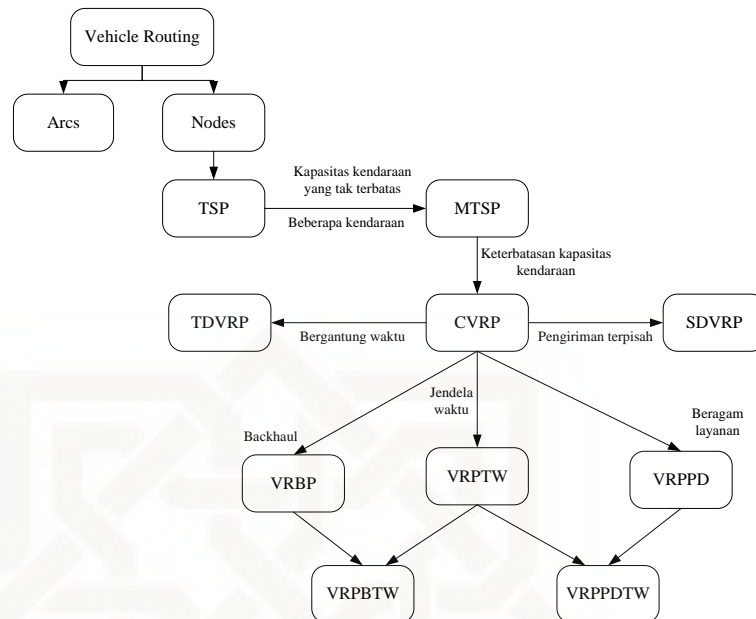
VRP bertujuan untuk melakukan pengiriman kepada beberapa *customer* yang diketahui permintaannya dengan biaya rute yang minimal dimana rute dimulai dan diakhiri di depot. VRP klasik secara umum digambarkan dengan grafik $G = (V, E)$, dimana $V = \{v_0, v_1, \dots, v_n\}$ merupakan Vertex yang menyatakan kumpulan lokasi, E menyatakan n kota.

Selain itu terdapat pula notasi C yang menyatakan matriks *non negative* atau jarak c_{ij} antara *customer* v_i dan v_j , R_i menyatakan rute untuk kendaraan i , m menyatakan banyaknya kendaraan (semua identik dan berkendara dengan kecepatan konstan), satu rute ditugaskan untuk setiap kendaraan (Faied et al., 2010).

VRP tergolong dalam persoalan NP-Hard dan digunakan manajemen *supply chain* dalam pengiriman fisik dari barang maupun jasa. Walaupun tergolong NP-Hard, VRP banyak dipelajari karena penerapannya yang luas dan peran pentingnya dalam menentukan strategi yang efisien untuk mengurangi biaya operasional dalam jaringan distribusi (Kumar & Panneerselvam, 2012).

2.3.2 Macam-Macam *Vehicle Routing Problem*

Seringkali permasalahan VRP yang ditemui lebih rumit dengan beberapa batasan yang kemudian memunculkan beberapa variasi. Berikut adalah gambaran beberapa variasi VRP (Sandhya & Kumar, 2013):



Gambar 2.1 Variasi VRP

Sumber: Sandhya & Kumar, 2013, hlm. 669

Vehicle Routing and Scheduling diklasifikasikan menjadi permasalahan *Arc* (busur) dan *Node* (titik). VRP termasuk ke dalam permasalahan *Node* dimana permintaan pelayanan terkait dengan *Node* (Sandhya & Kumar, 2013). Berikut adalah beberapa variasi VRP:

- a. *TDVRP (Time Dependent Vehicle Routing Problem)*. Merupakan VRP dimana waktu perjalanan atau biaya perjalanan antara dua lokasi bergantung pada waktu dalam sehari (Toth & Vigo, 2014).
- b. *CVRP (Capacitated Vehicle Routing Problem)*. Merupakan VRP dengan tambahan batasan berupa setiap kendaraan pengangkut harus memiliki kapasitas yang seragam (Faied et al., 2010).

- c. SDVRP (*Split Delivery Vehicle Routing Problem*). Merupakan VRP dimana *customer* yang sama dapat dilayani kendaraan berbeda jika hal tersebut mampu mengurangi biaya (Faied et al., 2010).
- d. VRPB (*Vehicle Routing Problem with Backhaul*). Merupakan perluasan CVRP dimana *customer* dibagi menjadi dua bagian yakni *linehaul customer* (masing-masing *customer* menerima barang yang dikirimkan) dan *backhaul customer* (barang harus diambil dari *customer*) (Toth & Vigo, 2014).
- e. VRPTW (*Vehicle Routing Problem with Time Windows*). Merupakan VRP dengan tambahan batasan berupa *time windows* yang menghubungkan antar *customer* dimana pada interval waktu inilah *customer* dapat dilayani (Faied et al., 2010).
- f. VRPPD (*Vehicle Routing Problem with Pick-up and Delivery*). Merupakan perluasan CVRP dimana barang diambil dari suatu lokasi (jemput) untuk kemudian diantar ke lokasi yang lain (antar) oleh kendaraan yang sama (Toth & Vigo, 2014).
- g. VRPBTW (*Vehicle Routing Problem with Backhaul and Time Windows*). Merupakan VRP dengan *linehaul* dan *backhaul customer* dimana *customer* harus dilayani dalam interval waktu tertentu (Toth & Vigo, 2014).
- h. VRPPDTW (*Vehicle Routing Problem with Pick-up and Delivery and Time Windows*). Merupakan VRP dimana barang diambil dari suatu *customer*/lokasi (jemput) untuk kemudian diantar ke *customer*/lokasi

yang lain (antar) oleh kendaraan yang sama dengan tambahan batasan bahwa tiap *customer*/lokasi memiliki interval waktu pelayanan masing-masing (Toth & Vigo, 2014)

Selain beberapa variasi di atas, menurut Toth & Vigo (2014) terdapat pula variasi lain dari VRP yakni *Periodic VRP* (PVRP) atau perluasan VRP dimana *customer* dikunjungi beberapa kali selama horizon perencanaan, *Heterogeneous or Mixed Fleet VRP* atau VRP dengan kendala bahwa kendaraan pengangkut memiliki kapasitas dan biaya yang berbeda, *Stochastic VRP* atau VRP dimana terdapat *random value* yang bisa berasal dari permintaan, *customer*, atau waktu perjalanan dan *Dynamic VRP* atau VRP dimana kendaraan melayani permintaan yang sudah diketahui sebelumnya dan permintaan yang baru diketahui secara *real time*.

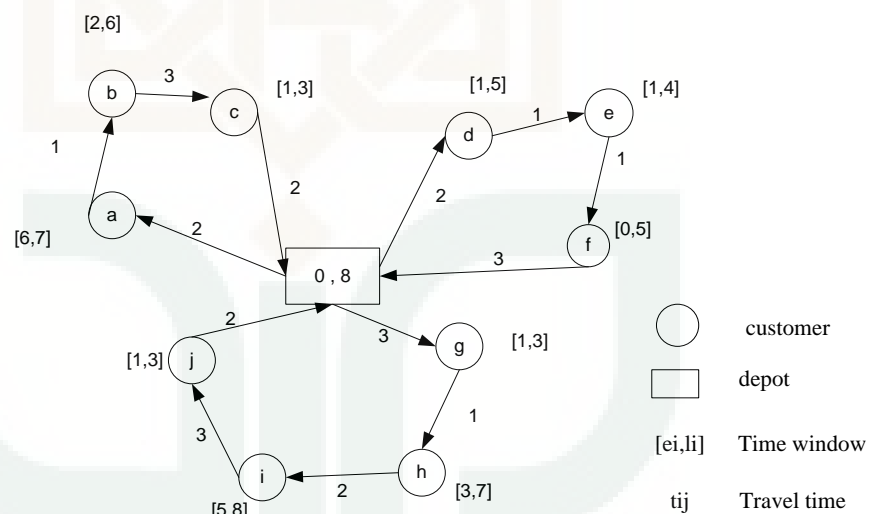
2.3.3 Vehicle Routing Problem with Time Windows

Menurut Toth & Vigo (2014, hlm. 119) “*Vehicle Routing Problem with Time Windows* (VRPTW) adalah perluasan dari *Capacitated Vehicle Routing Problem* (CVRP) dimana pelayanan untuk setiap *customer* harus dimulai dalam interval waktu yang berhubungan dan disebut *time windows* atau jendela waktu.” Dalam kasus *hard time windows*, kendaraan datang terlalu cepat dan harus menunggu hingga *customer* siap dilayani dimana pada umumnya tidak diperlukan biaya menunggu. Sementara dalam kasus *soft time windows*, setiap *time windows* dapat dilanggar dengan menanggung biaya pinalti.

VRPTW memiliki tujuan yakni meminimalkan banyaknya keseluruhan kendaraan yang digunakan untuk melayani *customer* dan meminimalkan biaya perjalanan seluruh kendaraan dengan tetap memenuhi batasan-batasan. Batasan-batasan tersebut antara lain (Sandhya & Kumar, 2013):

- Setiap *customer* hanya dilayani tepat satu kali
- Batasan *time windows* harus dipenuhi
- Total permintaan dari setiap rute tidak boleh melampaui batasan kapasitas kendaraan
- Setiap kendaraan harus mulai dan berakhir di depot.

Sandhya & Kumar (2013) menggambarkan ilustrasi dari permasalahan VRPTW sederhana sebagai berikut:



Gambar 2.2. Vehicle Routing Problem with Time Window

Sumber: Sandhya & Kumar, 2013, hlm.668

2.4 Model Matematis VRPTW

Model matematis VRPTW yang dikembangkan seringkali memiliki fungsi tujuan yang beragam. Azi et al. (2007) memformulasikan VRPTW dengan tujuan untuk mengurangi jarak keseluruhan untuk melayani *customer* dengan tetap

memenuhi batasan kapasitas, *time window*, dan *deadline*. Kallehauge et al. (2001) mengembangkan model matematis VRPTW dalam bentuk secara umum dengan fungsi tujuan meminimalkan *total cost* dan batasan bahwa setiap *customer* dilayani satu kali, setiap rute berawal dan berakhir di depot, batasan *time windows* serta batasan kapasitas.

Model matematis Kallehauge et al. (2001) memiliki notasi diantaranya, sebagai berikut:

V	= kumpulan kendaraan dengan kapasitas yang sama
C	= kumpulan <i>customer</i>
G	= grafik berarah yang terdiri dari $ C + 2$ <i>vertices</i>
N	= kumpulan titik yang terdiri dari <i>customer</i> dan depot
A	= kumpulan <i>arc</i> /jalur
0	= depot sebagai awal rute
$n + 1$	= depot sebagai akhir rute
q	= kapasitas kendaraan
d_i	= permintaan <i>customer</i>
c_{ij}	= biaya
t_{ij}	= waktu perjalanan ditambah waktu pelayanan
a_i, b_i	= <i>time windows</i>

Model VRPTW memaksa kendaraan harus tiba pada *customer* sebelum b_i , tetapi dapat datang sebelum a_i dengan konsekuensi kendaraan harus menunggu sampai *customer* siap dilayani yakni setelah a_i . Depot memiliki *time window* yang diasumsikan identik yang disebut *scheduling horizon* atau horizon penjadwalan dan

dinotasikan dengan $[a_0, b_0]$. Kendaraan tidak boleh meninggalkan depot sebelum a_0 dan harus kembali sebelum atau tepat ketika b_{n+1} .

Model Kallehauge et al. (2001) terdiri dari dua variabel keputusan yakni x dan s . Untuk setiap (i, j) , dimana $i \neq j, i \neq n + 1, j \neq 0$, dan setiap kendaraan k , x_{ij} didefinisikan:

$$x_{ijk} = \begin{cases} 0, & \text{jika kendaraan } k \text{ tidak melakukan perjalanan dari titik } i \text{ ke titik } j \\ 1, & \text{jika kendaraan } k \text{ melakukan perjalanan dari titik } i \text{ ke titik } j \end{cases}$$

Variabel keputusan s_{ik} menunjukkan waktu dimulai pelayanan pada *customer* i oleh kendaraan k . Jika kendaraan k tidak melayani *customer* i , maka s_{ik} tidak berarti apapun. Adapun model matematisnya dituliskan sebagai berikut:

$$Z_{VRPTW} = \text{minimize} \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk} \quad (1)$$

Batasan:

$$\sum_{k \in V} \sum_{i \in N} x_{ijk} = 1 \quad \forall_i \in C \quad (2)$$

$$\sum_{i \in C} d_i \sum_{j \in N} x_{ijk} \leq q \quad \forall_k \in V \quad (3)$$

$$\sum_{j \in N} x_{0jk} = 1 \quad \forall_k \in V \quad (4)$$

$$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{hjk} = 0 \quad \forall_h \in C, \forall_k \in V \quad (5)$$

$$\sum_{i \in N} x_{i,n+1,k} = 1 \quad \forall_k \in V \quad (6)$$

$$s_{ik} + t_{ij} - K(1 - x_{ijk}) \leq s_{jk} \quad \forall_{i,j} \in N, \forall_k \in V \quad (7)$$

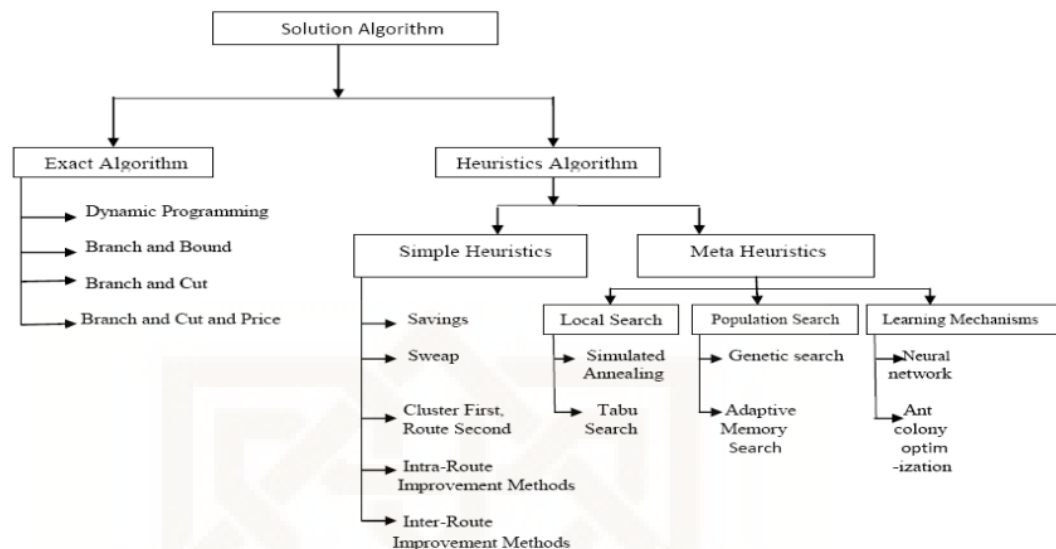
$$a_i \leq s_{ik} \leq b_i \quad \forall_i \in N, \forall_k \in V \quad (8)$$

$$x_{ijk} \in \{0, 1\} \quad \forall_{i,j} \in N, \forall_k \in V \quad (9)$$

Fungsi tujuan yang digambarkan dengan persamaan (1) menyatakan bahwa tujuan dari model Kallehauge et al. (2001) yakni untuk meminimalkan biaya perjalanan. Batasan yang dirumuskan dengan persamaan (2) menyatakan bahwa setiap *customer* dikunjungi tepat satu kali. Batasan (3) menunjukkan batasan bahwa kendaraan tidak boleh mengangkut melebihi kapasitas yang diperbolehkan. Batasan (4) menunjukkan bahwa setiap kendaraan bermula dari depot, batasan (5) menunjukkan bahwa setelah mengunjungi satu *customer* maka kendaraan akan pergi meninggalkan *customer* tersebut untuk menuju *customer* selanjutnya, dan batasan (6) menyatakan bahwa setiap kendaraan akan berakhir di depot. Batasan (7) digunakan untuk menyatakan bahwa kendaraan k tidak diperbolehkan sampai di *customer* j sebelum $s_{ik} + t_{ij}$ atau sebelum waktu dimulai pelayanan dan waktu perjalanan dari i ke j , dimana K merupakan bilangan riil yang bernilai besar. Batasan yang dituliskan oleh batasan (8) memastikan bahwa batasan *time windows* masing-masing *customer* terpenuhi dan batasan (9) menyatakan bahwa variable keputusan x_{ijk} bernilai biner. Sebagai catatan bahwa, kendaraan yang tidak dipergunakan akan memiliki rute kosong $0, n + 1$.

2.5 Metode Penyelesaian *Vehicle Routing Problem*

Permasalahan VRP dapat diselesaikan dengan menggunakan beberapa metode diantaranya dengan algoritma eksak, heuristik, dan metaheuristik. Saat ini, algoritma eksak memiliki keterbatasan hanya dapat merespon 50 hingga 100 titik *customer* berdasarkan variasi VRP dan waktu yang diperlukan untuk penyelesaian (Kumar et al., 2012). Sandhya & Kumar (2013) menggambarkan algoritma solusi permasalahan VRP dalam bagan sebagai berikut:



Gambar 2.3 Variasi Algoritma Penyelesaian VRP

Sumber: Sandhya & Kumar, 2013, hlm.671

Selain metode yang disebutkan diatas, beberapa peneliti juga mengembangkan beberapa metode penyelesaian seperti *Lagrangean Duality* yang dikemukakan Kallehauge et al. (2001), model IP yang diselesaikan dengan Algoritma Jalur Terpendek yang dikemukakan Azi et al. (2006), model *mixed integer programming* yang diselesaikan dengan GIDEON System yang dikemukakan Tangiah (1995), serta model MILP yang dikemukakan Toth & Vigo (2014) serta Dondo & Cendra (2006).

2.6 Cluster First Route Second

Menurut Cordeau et al. (2007) salah satu metode penyelesaian VRP yakni dengan metode heuristik dua fase. Metode tersebut berdasarkan penguraian proses penentuan solusi VRP ke dalam dua permasalahan terpisah yakni:

1. *Clustering* atau pengelompokkan: menentukan pembagian *customer* ke dalam kelompok-kelompok masing-masing sesuai rute, dan
2. Menentukan rute: menentukan urutan *customer* yang dikunjungi untuk masing-masing rute.

Pada metode *cluster first route second*, *customer* dikelompokkan kedalam *cluster-cluster* terlebih dahulu untuk kemudian ditentukan urutan *customer* yang dikunjungi. Terdapat beberapa teknik berbeda dalam melakukan pengelompokkan, sementara penentuan rute dilakukan dengan TSP. Pengelompokkan tersebut diantara dapat menggunakan Algoritma *Sweep*, Algoritma Fisher dan Jaikumar, dan Algoritma Petal.

2.6.1 Algoritma *Sweep*

Menurut Nurcahyo et al. (2002), Algoritma *Sweep* terdiri dari dua tahap yakni *clustering* pada tahap pertama, dan pembangkitan rute pada tahap keduanya. Algoritma *Sweep* pertama kali dikenalkan Gillet & Miller pada 1974 dimana *clusteringnya* dimulai dengan menempatkan depot sebagai titik pusat koordinat dan dikelilingi *nodes* yang tersebar secara acak sesuai letak geografis. Langkah selanjutnya yakni “menyapu” mulai dari depot kemudian dilanjutkan dengan *nodes* terdekat atau *nodes* yang memiliki sudut polar terkecil, secara terus menerus hingga kapasitas kendaraan terpenuhi. Kemudian setelah *cluster* terbentuk, dilanjutkan dengan pembangkitan rute. Algoritma *Sweep* dapat diimplementasikan menggunakan dua metode yang berbeda yakni *forward sweep* dan *backward sweep*. Pada *forward sweep*

sapuan dilakukan searah jarum jam, sementara pada *backward sweep* sapuan dilakukan berlawanan arah jarum jam.

Suthikarnnarunai (2008) menjelaskan bahwa Algoritma *Sweep* merupakan suatu metode untuk *clustering customer* ke dalam kelompok-kelompok sehingga *customer* dalam kelompok yang sama terdiri dari *customer* yang dekat secara geografis dan dapat dilayani dengan kendaraan yang sama. Tahapan *clustering* menggunakan Algoritma *Sweep* sebagai berikut:

1. Tempatkan depot sebagai titik tengah dari bidang dua dimensi.
2. Hitung koordinat polar masing-masing *customer* terhadap depot.
3. Mulai “menyapu” seluruh *customer* menurut kenaikan sudut polar.
4. Memasukkan masing-masing *customer* yang dicakup “sapuan” kedalam *cluster* saat ini.
5. Berhenti “menyapu” ketika tambahan *customer* selanjutnya berakibat melanggar batasan maksimal kapasitas kendaraan.
6. Buat *cluster* baru dengan melanjutkan “sapuan” dari titik yang ditinggalkan terakhir kali.
7. Ulangi langkah 4-6 hingga seluruh *customer* masuk ke dalam *cluster*.

2.6.2 Travelling Salesman Problems

Diaby (2007) menuliskan definisi TSP sebagai permasalahan menentukan urutan kota yang dikunjungi oleh kendaraan agar biaya yang dibutuhkan minimum, dimana kendaraan harus mengunjungi beberapa kota, berawal dan berakhir pada kota yang sama, dan setiap kota harus dikunjungi tepat satu kali.

Saiyed (2012) menyatakan bahwa TSP pertama kali dipelajari pada tahun awal 1930an oleh seorang matematikawan dan ekonom Karl Menger di Vienna dan Harvard. Sementara secara historis, ilmu matematika yang berhubungan dengan TSP dikembangkan pada 1800an oleh Sir Willian Rowan Hamilton dan Thomas Penyngton Kirkman yang kemudian berkembang hingga saat ini. Permasalahan TSP dapat diselesaikan dengan menggunakan metode eksak, solusi pendekatan, maupun teknik optimasi.

Menurut Klansek (2011), TSP merupakan permasalahan optimasi kombinatorial untuk menentukan rute yang optimal dimana setiap lokasi hanya dapat dikunjungi tepat satu kali. TSP dapat diformulasikan dalam bentuk MILP dimana parameternya dapat berupa diskrit maupun kontinu. Model formulasi TSP secara umum dapat dituliskan sebagai berikut:

$$\min z = \sum_{i=1}^n \sum_{j=1}^n d_{i,j} y_{i,j} \quad (1)$$

Batasan:

$$\sum_{j=1}^n y_{i,j} = 1 \quad i = 1, 2, 3, \dots, n \quad (2)$$

$$\sum_{i=1}^n y_{i,j} = 1 \quad j = 1, 2, 3, \dots, n \quad (3)$$

$$t_i - t_j \leq n(1 - y_{i,j}) - 1 \quad i, j = 1, 2, 3, \dots, n \quad (4)$$

$$y_{i,j} \in \{0, 1\} \quad i, j = 1, 2, 3, \dots, n$$

$$t_i \in \{1, 2, 3, \dots, n\} \quad i = 1, 2, 3, \dots, n$$

Dimana:

$d_{i,j}$ = merupakan batasan yakni jarak lintasan,

i = merupakan index yakni titik pertama pada lintasan,

j = merupakan index yakni titik kedua pada lintasan,

n = merupakan batasan yakni jumlah *node*

t_i = merupakan variabel yakni posisi dalam rute dimana *node* tersebut dikunjungi

$y_{i,j}$ = merupakan variabel keputusan yakni keputusan dipilih atau tidaknya lintasan tersebut dalam sebuah rute

z = merupakan fungsi tujuan yakni meminimalkan total jarak tempuh rute kendaraan

Setiap *node* i merepresentasikan lokasi yang harus dikunjungi, dimana lintasan i, j menyatakan terjadinya hubungan antara dua lokasi. $d_{i,j}$ dapat merepresentasikan biaya perjalanan, waktu perjalanan atau jarak antara dua lokasi. Oleh karena itu, fungsi tujuan z dapat berupa meminimalkan biaya perjalanan, waktu perjalanan, atau jarak perjalanan.

Fungsi tujuan yang dituliskan dalam persamaan (1) menyatakan bahwa fungsi matematis bertujuan untuk meminimalkan jarak atau waktu atau biaya perjalanan. Batasan (2) dan batasan (3) memastikan bahwa setiap lokasi dikunjungi dan ditinggalkan tepat satu kali. Batasan

(2) dan (3) dalam persamaan tidak cukup memastikan solusi optimal yang terbentuk tidak menyertakan sub rute oleh karena itu digunakan batasan (4) untuk menghalangi adanya sub rute.

TSP dapat sangat berguna diterapkan dalam bidang industri diantaranya dalam memilih urutan barang yang diambil pada gudang, menentukan urutan pekerjaan, dan *vehicle routing*.

2.6.3 Mixed Integer Linear Programming

Penentuan jalur terpendek dalam distribusi tergolong dalam permasalahan riset operasi, dimana penyelesaian permasalahan tersebut selalu didahului dengan pembuatan model matematika dari persoalan yang ada. Program linier merupakan salah satu model yang paling banyak aplikasinya, meskipun secara umum penyelesaian bukanlah bilangan bulat (Siang, 2011).

Permasalahan program linier atau *Linier Programming Problems* dalam beberapa kasus nyata, variabelnya tidak hanya berupa bilangan riil akan tetapi berupa bilangan integer atau yang lebih membatasi lagi berupa berupa bilangan biner yang hanya bernilai 0 atau 1 (Castillo et al., 2002). *Mixed Integer Linier Programming* (MILP) merupakan *integer linier programming* dimana beberapa variabelnya dapat berupa integer dan yang lain dapat bernilai *continue*. MILP dapat digunakan untuk memodelkan berbagai permasalahan dengan cakupan yang luas. Saat ini, MILP sangat diperlukan sebagai *tools* dalam bidang bisnis dan teknik (Vielma, 2015). Kelebihan MILP ini terletak pada hasilnya yang optimal dan adanya *software*

komersil yang dapat digunakan untuk menyelesaikan permasalahan MILP (Richards et al., 2002).

Menurut Smith (2007) membuat model MILP dapat dilakukan dalam tiga langkah, yakni:

1. Mendefinisikan variabel keputusan yang akan dioptimasi dalam sistem
2. Menyatakan batasan dalam model yang dibuat
3. Menyatakan fungsi tujuan yang hendak dicapai

BAB III

METODOLOGI PENELITIAN

3.1 Objek Penelitian

Objek penelitian tugas akhir ini dilakukan pada bagian distribusi CV. Jogja Transport selaku distributor produk Sari Roti di Daerah Istimewa Yogyakarta. CV Jogja Transport beralamat di Komplek Pertokoan Kembar Swalayan, Jalan Tri Tunggal No.1, Bangunharjo, Sewon, Bantul.

3.2 Jenis Data

Jenis data yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Data Primer

Data primer merupakan data yang diperoleh dari hasil pengamatan langsung di lapangan dan wawancara dengan pihak terkait. Data primer yang diperoleh menyajikan informasi yang berhubungan dengan data yang akan diolah. Selain itu, data primer berhubungan dengan permasalahan di lapangan serta dapat diidentifikasi gejalanya secara langsung. Data primer yang digunakan dalam penelitian ini meliputi:

- a. Waktu administrasi dan waktu bongkar muat di depot.
- b. Data lokasi depot dan *customer*.
- c. Data jarak tempuh antara depot dengan *customer* dan jarak tempuh antar *customer*.
- d. Data waktu tempuh antara depot dengan *customer* dan jarak tempuh antar *customer*.
- e. Data koordinat masing-masing *customer*.

2. Data Sekunder

Data sekunder merupakan data yang diperoleh dari referensi yang berasal dari perusahaan, perpustakaan, jurnal ilmiah, atau *literature* lain sesuai dengan permasalahan yang dibahas. Data sekunder yang digunakan dalam penelitian ini meliputi:

- a. Data alamat masing-masing *customer*.
- b. Data permintaan setiap *customer* untuk hari Rabu selama bulan September 2016.
- c. Data kapasitas alat angkut.
- d. Data waktu bongkar muat dan pelayanan pada masing-masing *customer*.
- e. Data rute distribusi perusahaan saat ini.

3.3 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data dalam penelitian ini adalah:

1. Metode Observasi

Metode observasi merupakan metode yang digunakan dengan melakukan pengamatan secara langsung di area perusahaan untuk mengetahui proses distribusi perusahaan saat ini.

2. Metode Wawancara

Metode wawancara dilakukan dengan pihak terkait untuk melengkapi gambaran proses distribusi perusahaan dan permasalahan distribusi yang dialami perusahaan. Wawancara dilakukan dengan pihak terkait yakni

Bapak Saikhu Rohman selaku pimpinan perusahaan dan Bapak Mujib selaku wakil pimpinan perusahaan.

3. Studi *literature*.

Studi *literature* dilakukan dengan cara penelusuran *literature-literature* terkait baik berupa buku, jurnal, maupun sumber-sumber lain yang berfungsi menambah pengetahuan dan informasi mengenai metode yang digunakan.

3.4 Metode Pengolahan Data

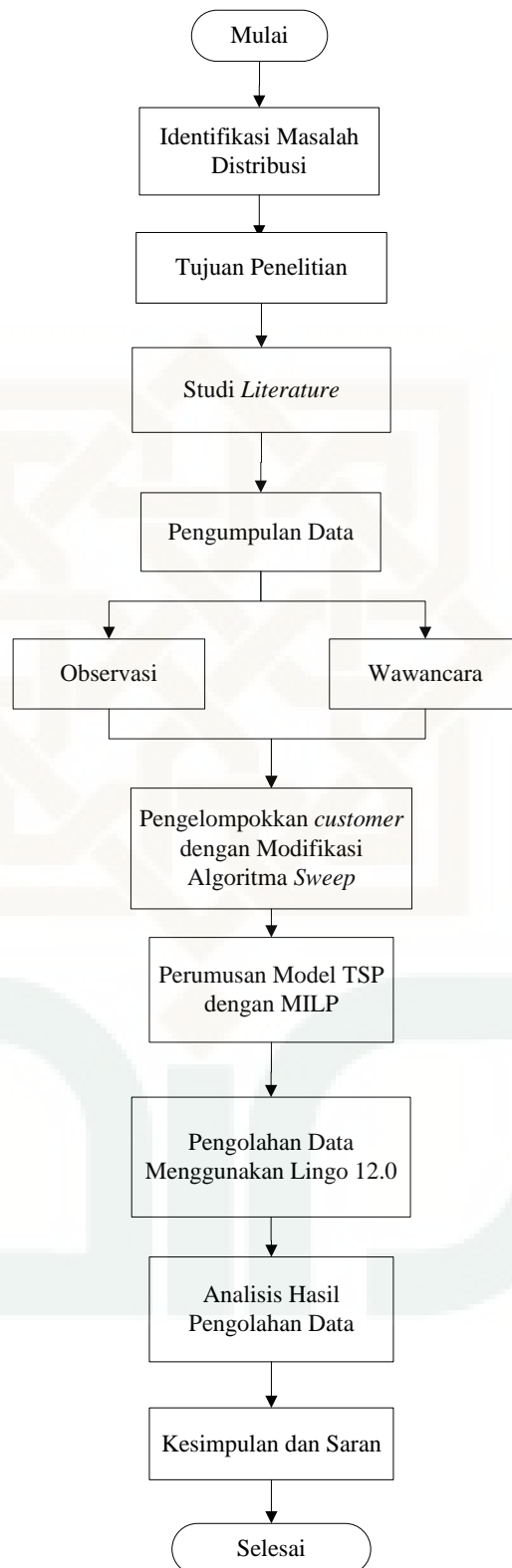
Metode yang digunakan untuk pengolahan data dalam penelitian ini yakni:

1. Modifikasi Algoritma *Sweep*. Algoritma *Sweep* merupakan salah satu metode heuristik yang digunakan sebagai fase pertama dalam penyelesaian kasus VRP dengan cara *cluster first route second*. Metode ini digunakan untuk mengelompokkan *customer* ke dalam *cluster-cluster* guna menyederhanakan permasalahan. Sementara modifikasi Algoritma *Sweep* merupakan metode Algoritma *Sweep* dengan tambahan batasan bahwa setiap *cluster* maksimal terdiri dari 9 *nodes* termasuk depot. Hal ini dikarenakan keterbatasan kemampuan *software* Lingo12.0 berbasis edukasi untuk mengolah data dalam jumlah besar.
2. *Travelling Salesman Problems* (TSP). TSP merupakan permasalahan menentukan jarak minimum bagi sebuah kendaraan yang mengunjungi beberapa *node*, dimana setiap *node* dikunjungi tepat satu kali. TSP pada penelitian ini merupakan kelanjutan modifikasi Algoritma *Sweep* atau

fase kedua dari metode heuristik *cluster first-route second*. TSP digunakan untuk menentukan rute pada setiap *cluster* yang terbentuk. Permasalahan TSP tersebut dapat diselesaikan dengan metode eksak *Mixed Integer Linier Programming* (MILP).

3. MILP merupakan percabangan dari *Integer Linier Programming* dimana formulasi model MILP memungkinkan variabelnya tidak hanya berupa *integer* dan pecahan melainkan dapat pula berupa biner. Variabel biner diperlukan sebagai pengambilan keputusan apakah rute pengiriman dilakukan oleh kendaraan. Variabel integer dalam penelitian ini berupa variabel data permintaan dan data *time windows*. Sementara variabel pecahan berupa data jarak dan waktu tempuh kendaraan.
4. Lingo 12.0. Lingo 12.0 merupakan salah satu *software* optimasi berbasis komputer yang mampu melakukan kalkulasi dalam proses penyelesaian model matematika. Lingo biasa digunakan dalam penyelesaian permasalahan optimisasi dibidang bisnis, industri, dan pemerintahan. Pada penelitian ini Lingo yang digunakan yakni Lingo 12.0 versi edukasi dimana *software* ini memiliki keterbatasan jumlah variabel maupun batasan meskipun bersifat *full version*. Lingo pada penelitian ini digunakan untuk mengolah model MILP.

Untuk mengetahui alir penelitian dari awal sampai akhir dapat digambarkan sebagai berikut:



Gambar 3.1 Diagram Alir Penelitian

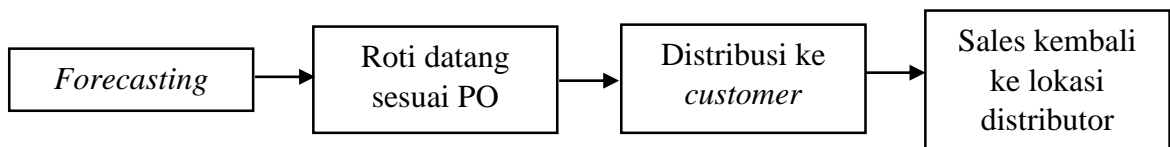
BAB IV

ANALISIS DAN PEMBAHASAN

4.1 Proses Distribusi Perusahaan

CV. Jogja Transport merupakan salah satu perusahaan yang bergerak dibidang distribusi produk khususnya produk Sari Roti. CV Jogja Transport mulai bergerak dibidang pendistribusian produk Sari Roti terhitung sejak awal tahun 2016. Sebelum bergerak dalam bidang distribusi, CV Jogja Transport telah terlebih dahulu menjadi salah satu perusahaan yang bergerak dibidang jasa sewa mobil dan penginapan di wilayah Yogyakarta. Hingga pada tahun 2016, CV Jogja Transport memulai kontribusi dibidang pendistribusian produk Sari Roti sebagai salah satu mata rantai *supply chain* PT. Nippon Indosari Corporindo. Saat ini CV Jogja Transport telah memiliki kurang lebih 140 titik *customer/outlet* yang tersebar di wilayah Daerah Istimewa Yogyakarta.

Proses pendistribusian yang dilakukan CV Jogja Transport dimulai dari tahap *forecasting* yang dilakukan untuk mengetahui jumlah produk yang dibutuhkan masing-masing titik *customer*, hingga pendistribusian kepada setiap titik *customer*. Berikut adalah gambaran proses pendistribusian yang dilakukan oleh CV Jogja Transport:



Gambar 4.1 Bagan Proses Distribusi

a. *Forecasting*.

Tahap pertama dalam proses pendistribusian CV Jogja Transport yakni tahap *forecasting*. Pada tahap ini perusahaan melakukan *forecasting* atau peramalan untuk mengetahui kebutuhan setiap jenis roti untuk setiap *customer*. Hasil *forecasting* kemudian dikirimkan ke pabrik Sari Roti yang berlokasi di Semarang maksimal pukul 23.00 WIB tiga hari sebelum roti dibutuhkan atau tiga hari sebelum roti sampai ke distributor. Pemesanan dilakukan melalui surat elektronik.

b. Roti datang sesuai PO.

Roti yang sudah dipesan melalui surat elektronik akan tiba sesuai *purchase order* yang dikirimkan perusahaan berdasarkan hasil *forecasting*. Sales kemudian membagi roti sesuai dengan kebutuhan masing-masing *customer*.

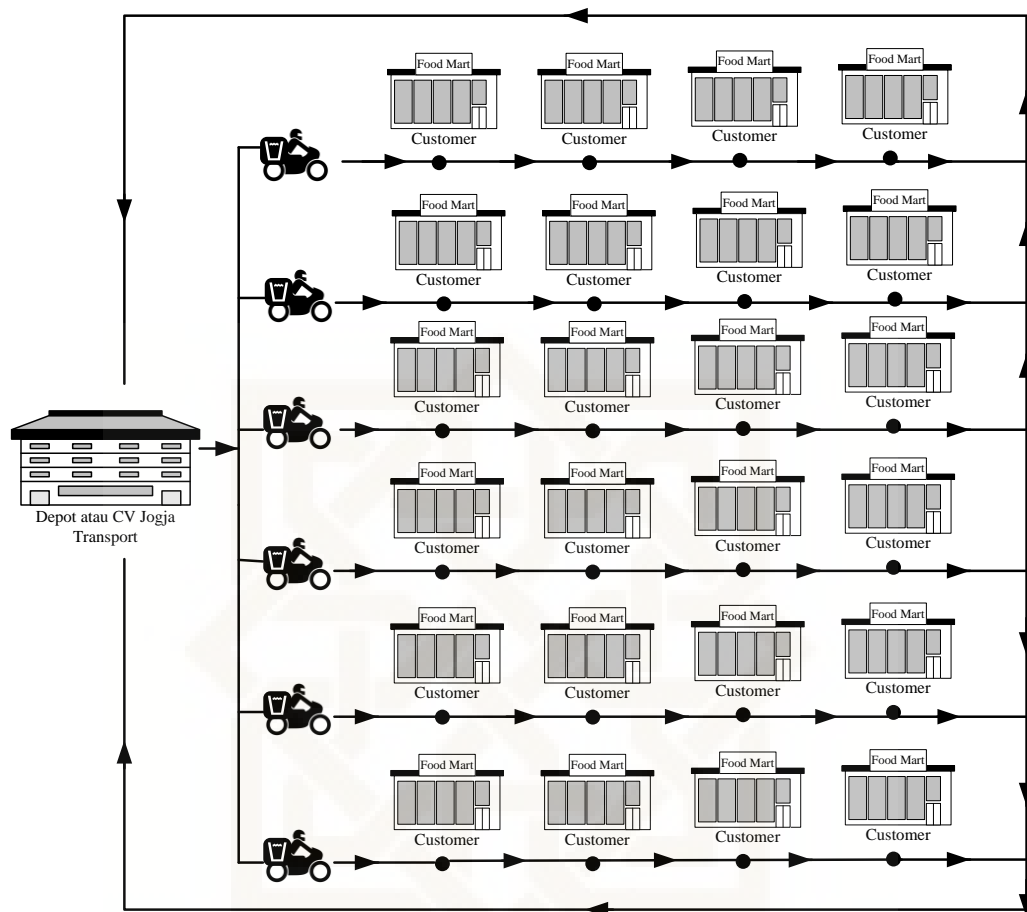
c. Distribusi ke *customer*.

Proses selanjutnya yakni pendistribusian roti oleh pihak CV Jogja Transport. Proses distribusi kepada *customer* dilakukan oleh masing-masing sales. Setiap sales memiliki zona pengiriman yang berbeda dan bertanggung jawab atas roti yang dikirimkan dan roti yang kembali karena masa kadaluarsa. Sales melakukan penarikan roti satu hari sebelum masa kadaluarsa roti tersebut untuk menjamin kualitas roti yang ada pada *customer* tetap terjaga sekaligus mendistribusikan roti baru kepada *customer*. Setiap *customer* mengalami pengiriman dan pengembalian roti dua kali dalam satu minggu dengan pasangan hari Senin dengan Kamis, Selasa dengan Jumat, dan Rabu dengan Sabtu. Jadi untuk *customer* yang menerima roti pada hari Senin, akan dilakukan penarikan roti yang mendekati kadaluarsa pada hari

Rabu sekaligus mendapatkan roti baru. Roti yang tersisa dari hari Rabu kemudian diambil kembali pada hari Senin. Proses ini berulang untuk setiap *customer* dengan masing-masing pasangan hari.

Saat ini, CV Jogja Transport memiliki enam orang sales dengan masing-masing wilayah pengiriman yang berbeda. Pembagian wilayah pengiriman didasarkan pada zona yang dibagi perusahaan. Sementara untuk menentukan *customer* mana yang terlebih dahulu dikunjungi oleh sales, perusahaan menyerahkan keputusan pada sales. Masing-masing sales dibekali satu buah kendaraan bermotor dengan tipe sepeda motor bebek yang dibagian belakang diberi wadah untuk meletakkan roti. Setiap wadah memiliki kapasitas 550 buah roti. Sales juga berhak mendapat uang sejumlah masing-masing lima belas ribu rupiah per hari untuk biaya pembelian bahan bakar.

Selain melakukan pengambilan roti, sales juga bertugas untuk mengambil uang dari hasil penjualan sebelumnya. Hal ini tidak berlaku bagi semua *customer*, terdapat pula beberapa *customer* yang proses penarikan uangnya dilakukan setiap seminggu sekali sesuai dengan aturan *customer*. Pada saat ini, CV Jogja Transport sedang berupaya untuk memisahkan tugas ini dari sales, sehingga sales hanya bertugas mengantar dan mengambil roti sementara penagihan dilakukan oleh bagian lain. Berikut adalah gambaran skema proses pengiriman produk Sari Roti:



Gambar 4.2 Skema Distribusi Produk Sari Roti

d. Sales kembali ke lokasi distributor.

Setelah menyelesaikan tugas melakukan pengiriman dan penarikan roti, sales akan kembali ke lokasi distributor untuk melakukan pertanggung jawaban. Setiap sales bertanggung jawab dalam bentuk pengisian *proforma invoice* dimana dalam *proforma invoice* akan tertera berapa roti yang dikirimkan pada periode sebelumnya, berapa roti yang kembali, dan perhitungan dalam bentuk rupiah. Sales akan memberikan pertanggung jawaban berupa fisik yakni uang dan roti yang kembali untuk kemudian direkap oleh sales admin. Angka jumlah penjualan inilah

yang digunakan CV Jogja Transport sebagai dasar untuk melakukan peramalan periode berikutnya.

4.2 Pengumpulan Data

a. Data Alamat Lokasi Depot dan *Customer*

Tabel 4.1 Data Alamat Lokasi Depot dan *Customer*

No	Nama Toko	Alamat
0	CV Jogja Transport (depot)	Jl Tri Tunggal No. 1, Bangunharjo, Sewon, Bantul
1	Shafira	Jl Imogiri Barat
2	Govinda	Jl Imogiri Barat
3	DM 6	Jl Parangtritis
4	Almira	Jl Ali Maksum
5	Fadli	Jl Ali Maksum
6	JM Mini Market	Jl Parangtritis
7	3 M	Jl Imogiri Barat
8	DM 4	Jl Imogiri Barat
9	Kembar 2 Swalayan	Jl Tri Tunggal
10	Toko Rani	Jl Bantul
11	JA Mart	Jl Bantul
12	WS Bantul	Jl Bantul
13	WS Niten	Jl Bantul
14	Mega 4	Jl Srandakan
15	Prima Srandakan	Jl Srandakan
16	Madina	Jl Bantul
17	Familia Swalayan	Jl Bantul
18	Atmaja Toserba	Jl Srandakan
19	Toko Janat	Jl Srandakan
20	Lestari	Jl Bantul
21	Rizky UMY	Kampus UMY
22	Hans Mini Market	Kasongan
23	WS Madukismo	Madukismo
24	WS Ridang	Jl Ridang
25	Toko Bu Sudi	Jl Kasongan
26	WS Pasar Mini	Jl Rindang
27	Toko Martiyem	Jl Kasongan
28	Rizky Celluler	Jl Bantul
29	Cahaya Swalayan	Jl Gunung Sempu
30	Agromart	Jl Parangtritis

31	Konde Mart 3	Jl Ali Maksum
32	Damai Indah S	Jl Bantul
33	Numart	Jl Gunung Sempu
34	Ridho Jaya	Jl Ali Maksum
35	KUD Tani Makmur	Jl Madukismo
36	Kembar 1	Jl Tritunggal
37	Toko Amin	Jl Ali Maksum
38	Kuncoro	Jl Ali Maksum
39	MM An-Nur	Ponpes An Nur Bantul
40	Amanda 4	Jl Pleret
41	Toko Mugiharjo	Jl Ali Maksum
42	Ada Swalayan	Jl Ngrukeman Gatak, Kasihan
43	Devia Putri	JL Rajawali, Selatan UMY

Sumber: Data Perusahaan CV Jogja Transport

b. Data Jarak Antar Lokasi

Data jarak antar lokasi menunjukkan jarak yang ditempuh kendaraan dari satu titik/*node* depot atau *customer* ke titik/*node* depot atau *customer* lain. Data jarak antar titik yang diperoleh dengan bantuan *google maps*. Data jarak antar depot dengan customer dan antar customer dilampirkan pada Lampiran A.

c. Data Waktu Tempuh Antar Lokasi

Data waktu tempuh antar lokasi menunjukkan waktu yang dibutuhkan kendaraan untuk pergi dari satu titik/*node* depot atau *customer* ke titik/*node* depot atau *customer* lain. Data waktu tempuh diperoleh dari rumus hubungan antara jarak, kecepatan, dan waktu tempuh. Berdasarkan Purnomo (2010), rumus waktu tempuh dalam satuan menit dituliskan sebagai berikut:

$$\text{waktu tempuh} = \frac{\text{jarak (km)}}{\text{kecepatan rata - rata}} \times 60$$

Keterangan:

Kecepatan rata-rata kendaraan pengangkut 40 *km/jam* dan 1 *jam* = 60 *menit*

Contoh perhitungan waktu tempuh kendaraan dari depot menuju titik 1 dapat dituliskan sebagai berikut:

$$\text{waktu tempuh} = \frac{1.8}{40} \times 60 = 2.7 \text{ menit}$$

Matriks data waktu tempuh antar titik/*node* depot atau *customer* ke titik/*node* depot atau *customer* lain berdasarkan perhitungan dilampirkan pada Lampiran B.

d. *Data Time Windows*

Data *time windows* depot dan *customer* menyatakan jam buka dan jam tutup pada depot dan *customer*. Dalam hal ini, depot dan *customer* hanya dapat dilayani oleh kendaraan dalam kurun waktu diantara jam buka dan jam tutup tersebut. Data tersebut diperoleh dengan bantuan *google maps* dan wawancara dengan pihak terkait yang mengetahui jam buka dan tutup toko bersangkutan. Berikut data *time windows* masing-masing titik:

Tabel 4.2 Data Time Windows

Node	Nama Outlet	Buka (a)		Tutup (b)	
		Jam	Menit	Jam	Menit
0	CV Jogja Transport (depot)	8.00	480	14.00	840
1	Shafira	8.00	480	21.00	1260
2	Govinda	7.00	420	21.00	1260
3	DM 6	9.00	540	22.00	1320
4	Almira	7.30	438	20.00	1200
5	Fadli	0.00	0	24.00	1440
6	JM Mini Market	8.00	480	22.00	1320
7	3 M	9.00	540	22.00	1320
8	DM 4	9.00	540	22.00	1320
9	Kembar 2 Swalayan	7.00	420	22.00	1320
10	Toko Rani	8.00	480	21.00	1260
11	JA Mart	7.00	420	21.00	1260
12	WS Bantul	8.00	480	20.30	1218
13	WS Niten	8.00	480	20.30	1218
14	Mega 4	7.00	420	21.00	1260
15	Prima Srandakan	8.30	498	20.30	1218

16	Madina	9.00	540	21.00	1260
17	Familia Swalayan	8.00	480	21.00	1260
18	Atmaja Toserba	8.00	480	21.00	1260
19	Toko Janat	7.00	420	21.00	1260
20	Lestari	8.00	480	20.30	1218
21	Rizky UMY	8.00	480	22.00	1320
22	Hans Mini Market	9.00	540	21.00	1260
23	WS Madukismo	8.00	480	20.30	1218
24	WS Ridang	8.00	480	20.30	1218
25	Toko Bu Sudi	8.00	480	22.00	1320
26	WS Pasar Mini	8.00	480	20.30	1218
27	Toko Martiyem	8.00	480	21.00	1260
28	Rizky Celluler	9.00	540	22.00	1320
29	Cahaya Swalayan	8.00	480	21.30	1278
30	Agromart	5.00	300	21.00	1260
31	Konde Mart 3	6.30	378	22.00	1320
32	Damai Indah S	8.00	480	21.00	1260
33	Numart	8.00	480	21.30	1278
34	Ridho Jaya	7.00	420	19.00	1140
35	KUD Tani Makmur	8.00	480	16.00	960
36	Kembar 1	6.00	360	22.00	1320
37	Toko Amin	8.00	480	21.00	1260
38	Kuncoro	8.00	480	21.00	1260
39	MM An-Nur	9.00	540	21.00	1260
40	Amanda 4	9.00	540	22.00	1320
41	Toko Mugiharjo	7.00	420	21.00	1260
42	Ada Swalayan	8.00	480	22.00	1320
43	Devia Putri	8.00	480	22.00	1320

e. Data Permintaan Setiap *Customer*

Data permintaan setiap *customer* menyatakan kebutuhan *customer* yang harus dipenuhi oleh perusahaan. Pada penelitian ini, data permintaan setiap *customer* berkaitan dengan kapasitas angkut kendaraan. Kendaraan harus memenuhi permintaan setiap customer tetapi tidak diperbolehkan membawa permintaan melebihi kapasitas kendaraan. Selain itu, data permintaan perusahaan untuk hari Rabu selama satu bulan tersebut digunakan untuk mengevaluasi model yang telah

dirumuskan. Data permintaan setiap *customer* tersebut diperoleh dari dokumen perusahaan. Berikut data permintaan masing-masing *customer*:

Tabel 4.3 Data Permintaan Customer

Node	Nama Outlet	Bulan September			
		Minggu ke-1	Minggu ke-2	Minggu ke-3	Minggu ke-4
1	Shafira	11	11	11	11
2	Gofinda	19	19	19	19
3	DM 6	51	47	50	48
4	Almira	84	82	84	95
5	Fadly	30	29	29	47
6	JM Mart	13	13	13	13
7	3 M	63	62	64	64
8	DM 4	50	50	50	50
9	Kembar 2	63	63	65	65
10	Rani Swalayan	10	10	19	12
11	JA Mart	11	11	15	13
12	WS Bantul	15	15	15	20
13	WS Niten	37	37	37	39
14	Mega 4	14	14	14	19
15	Prima Srandakan	84	84	91	76
16	Madina	22	22	25	29
17	Familia Swalayan	21	21	21	21
18	Atmaja Toserba	54	54	54	54
19	Toko Janat	56	57	71	70
20	Lestari	44	44	56	57
21	Rizky UMY	256	171	154	117
22	Hans	11	11	11	16
23	WS Madukismo	55	55	55	65
24	WS Rindang	13	13	13	27
25	Bu Sudi	14	14	14	13
26	WS Pasar Mini	13	13	13	13
27	Toko Martiyem	19	19	19	19
28	Rizky Seluler	6	6	6	6
29	Cahaya Swalayan	8	8	10	8
30	Agromart	7	7	7	12
31	Konde Mart 3	30	30	30	30
32	Damai Indah	44	44	44	39
33	NU Mart	7	7	7	7
34	Ridho Jaya	10	10	7	10

35	KUD Tani Makmur	12	12	12	19
36	Kembar 1	10	10	28	15
37	Toko Amin	14	14	14	31
38	Kuncoro	8	8	8	13
39	MM An Nur	41	41	50	46
40	Amanda 4	128	128	130	132
42	Mugiharjo	92	99	140	210
43	Toko Ada	28	28	18	18
44	Devia Putri	212	136	154	117

Sumber: Data Perusahaan CV Jogja Transport yang diolah

f. Rute Awal

Rute awal menyatakan rute yang saat ini ditempuh sales atau kendaraan dalam mengirimkan roti. Berikut rute awal atau rute yang saat ini digunakan oleh CV Jogja Transport:

Tabel 4.4 Rute Awal

No Kendaraan	Sales	Rute	Jarak Tempuh (km)	Waktu Tempuh (m)	Ongkos Bahan Bakar (Rp)
1	Sogi	0 → 4 → 5 → 41 → 7 → 8 → 0	14.7	22.05	15000
2	Wahyu	0 → 9 → 31 → 32 → 28 → 34 → 27 → 22 → 29 → 6 → 3 → 1 → 2 → 0	41.11	60.465	15000
3	Dian	0 → 23 → 25 → 35 → 33 → 26 → 24 → 42 → 30 → 0 → 43 → 21 → 0	37.916	61.899	15000
4	Pandu	0 → 19 → 18 → 15 → 14 → 10 → 20 → 0	46.95	69.075	15000
5	Fandy	0 → 40 → 17 → 12 → 11 → 16 → 13 → 0	28.33	43.2	15000
6	Feri	0 → 36 → 37 → 38 → 39 → 0	39.2	28.83	15000
Total Jarak Tempuh			208.206 km		
Total Waktu Tempuh			285.519 menit		
Total Ongkos Bahan Bakar			Rp 90000		

Sumber: Data Perusahaan CV Jogja Transport yang diolah

g. Ongkos Bahan Bakar per km

Ongkos bahan bakar per km diperoleh dari perhitungan total ongkos yang dikeluarkan perusahaan dalam satu kali hari Rabu, dibagi dengan jarak yang

ditempuh perusahaan saat ini. Hal ini dikarenakan perusahaan tidak mematok biaya bahan bakar per km, melainkan dengan sistem pukul rata masing-masing sales lima belas ribu rupiah. Ongkos bahan bakar per km dirumuskan sebagai berikut:

$$\text{Ongkos bahan bakar per km} = \frac{\text{jumlah sales} \times \text{Rp } 15000.00}{\text{jarak yang ditempuh}}$$

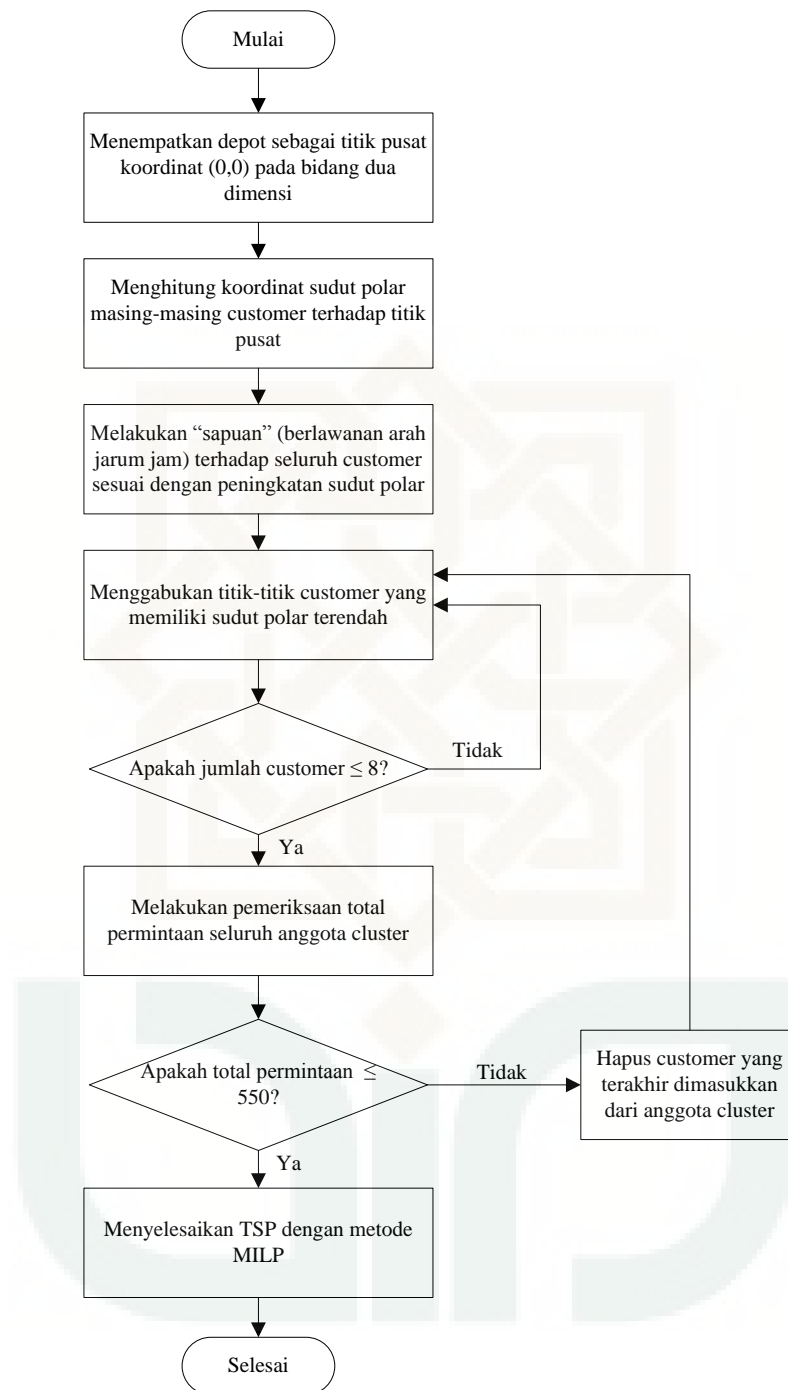
$$\text{Ongkos bahan bakar per km} = \frac{6 \times \text{Rp } 15000.00}{208.206}$$

$$\text{Ongkos bahan bakar per km} = \text{Rp } 432.26$$

4.3 Pengolahan Data

Langkah pengolahan data pada penelitian ini merujuk pada penelitian yang dilakukan Suthikarnnarunai (2008), dimana penelitian tersebut dimulai dari mengelompokkan *customer* ke dalam *cluster-cluster* menggunakan Algoritma *Sweep*, kemudian dilanjutkan dengan pembangkitan rute masing-masing *cluster* menggunakan *Integer Programming* (IP), dan diakhiri dengan perbaikan rute menggunakan *2 opt-exchanges*. Perbedaan penelitian ini dengan penelitian tersebut terletak pada batasan *cluster* Algoritma *Sweep* yang digunakan, metode pembangkitan rute yang digunakan, dan penelitian ini tidak dilanjutkan hingga tahap perbaikan rute.

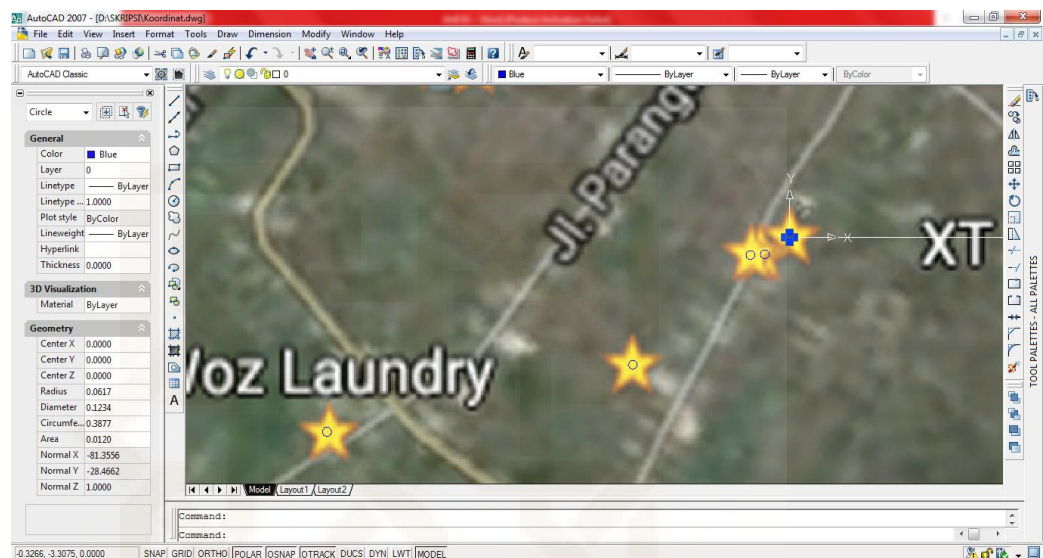
Langkah pertama pada pengolahan data penelitian ini dimulai dari pengelompokkan *customer* menggunakan Modifikasi Algoritma *Sweep* sebagai fase pertama. Kemudian dilanjutkan dengan penyelesaian TSP menggunakan metode MILP sebagai fase kedua. Berikut adalah diagram alir pengelompokkan *customer* menggunakan Modifikasi Algoritma *Sweep*:



Gambar 4.3 Diagram Alir Modifikasi Algoritma Sweep

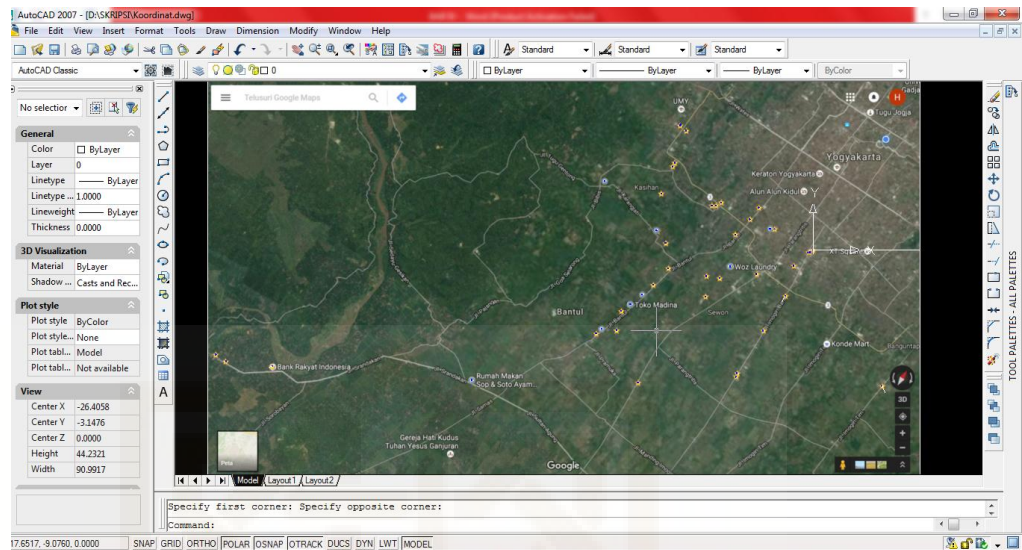
4.3.1 Pengelompokkan dengan Modifikasi Algoritma *Sweep*

a. Menempatkan depot sebagai titik pusat koordinat dua dimensi



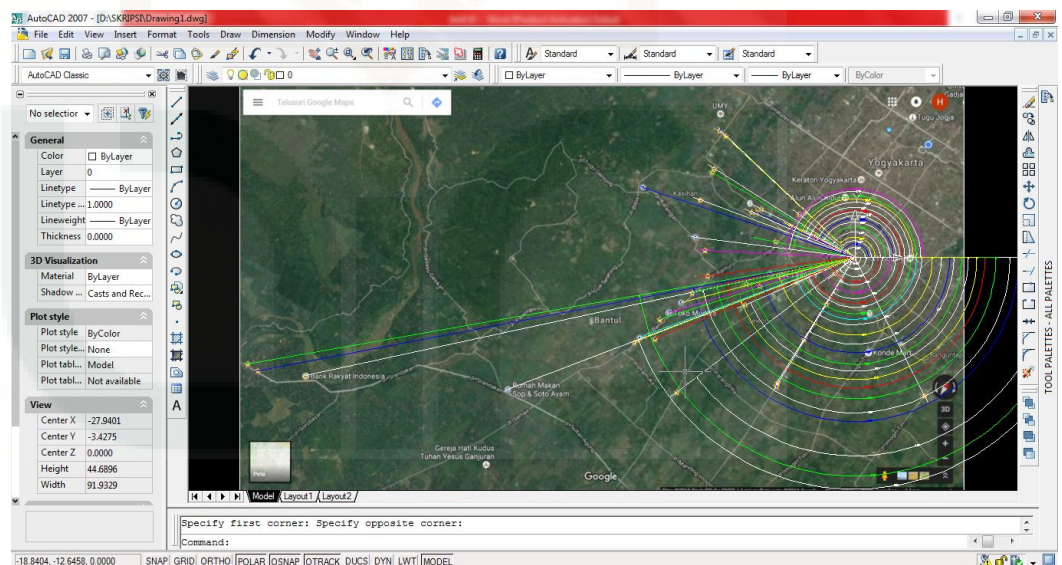
Gambar 4.4 Lokasi depot pada titik pusat (0.0)

CV Jogja Transport sebagai depot diletakkan pada koordinat (0.0) bidang dua dimensi. Setiap *customer* ditentukan titik koordinatnya terhadap depot. Lokasi depot dan *customer* diperoleh dari *google maps*, yang kemudian diubah ke dalam bentuk gambar untuk dilakukan penentuan titik koordinat dua dimensi. Penentuan titik koordinat depot dan masing-masing *customer* menggunakan bantuan *software* AUTOCAD.



Gambar 4.5 Lokasi koordinat depot dan *customer* pada bidang dua dimensi

- b. Menghitung koordinat sudut polar masing-masing titik *customer* terhadap titik pusat



Gambar 4.6 Sudut polar depot dan *customer* pada bidang dua dimensi

Setiap *customer* dihitung sudut polarnya terhadap depot sebagai titik pusat. Perhitungan sudut dilakukan berlawanan arah jarum jam

atau *counter clockwise* dengan bantuan *tools software* AUTOCAD untuk mencari *dimension angular*. Perhitungan sudut ini dilakukan dengan membuat garis lurus dari titik pusat depot hingga ke titik koordinat *customer* dan garis lurus terhadap sumbu x. Sudut yang terbentuk antara kedua garis tersebut merupakan sudut polar yang digunakan untuk pengelompokkan. Berikut data koordinat masing-masing *customer* dengan sudut polarnya:

Tabel 4.5 Koordinat dan Sudut Polar

Node	Nama Outlet	x	y	Sudut Polar (°)
0	CV Jogja Transport (depot)	0	0	0
1	Shafira	-3.1323	-5.1974	239
2	Govinda	-8.6784	-14.2932	239
3	DM 6	-19.7419	-14.9282	217
4	Almira	-9.3599	-1.9111	192
5	Fadli	-10.5355	-3.1587	197
6	JM Mini Market	-6.3929	-2.6872	203
7	3 M	-5.6242	-9.0769	238
8	DM 4	-8.5753	-13.8813	238
9	Kembar 2 Swalayan	-0.3433	-0.2383	215
10	Toko Rani	-18.0315	-3.9187	192
11	JA Mart	-23.8578	-8.8573	200
12	WS Bantul	-24.466	-9.3607	201
13	WS Niten	-16.3223	-2.0514	187
14	Mega 4	-60.9916	-13.1093	192
15	Prima Srandakan	-67.3656	-11.7845	190
16	Madina	-20.5812	-6.0855	196
17	Familia Swalayan	-21.6267	-6.8417	198
18	Atmaja Toserba	-66.227	-12.556	191
19	Toko Janat	-38.5099	-14.6624	201
20	Lestari	-19.285	-4.9872	194
21	Rizky UMY	-14.7461	14.1811	136
22	Hans Mini Market	-17.6616	2.2385	173
23	WS Madukismo	-10.3688	5.3705	153
24	WS Ridang	-15.5894	9.8735	148

25	Toko Bu Sudi	-10.9417	5.0105	155
26	WS Pasar Mini	-15.83	9.4288	149
27	Toko Martiyem	-16.6213	0.8108	177
28	Rizky Celluler	-8.7666	1.7566	169
29	Cahaya Swalayan	-23.481	7.8063	162
30	Agromart	-2.1684	-1.7636	219
31	Konde Mart 3	-4.3356	2.3767	151
32	Damai Indah S	-6.1181	4.8033	142
33	Numart	-17.1172	6.3762	160
34	Ridho Jaya	-4.8436	2.4228	153
35	KUD Tani Makmur	-1.4224	5.0076	156
36	Kembar 1	-0.5415	-0.248	205
37	Toko Amin	-7.1259	3.4783	154
38	Kuncoro	-12.025	-2.9468	194
39	MM An-Nur	-22.1605	-9.0193	202
40	Amanda 4	7.7623	-15.3357	297
41	Toko Mugiharjo	-12.1429	-5.2437	203
42	Ada Swalayan	-14.3057	13.4872	137
43	Devia Putri	-15.0094	14.0697	137

c. Melakukan “sapuan” terhadap seluruh *customer* sesuai dengan peningkatan sudut polar.

Melakukan “sapuan” terhadap seluruh *customer* dimulai dari sudut polar terkecil hingga sudut polar terbesar terhadap depot. Sapuan dilakukan berlawanan arah jarum jam (*counter clockwise*). Sapuan dengan arah berlawanan jarum jam tersebut disebut dengan *backward sweep*. Berikut data urutan sudut polar berdasarkan “sapuan” dari yang terkecil hingga terbesar:

Tabel 4.6 Urutan Sudut Polar

No	Nama Outlet	Urutan Sudut Polar (°)
1	CV Jogja Transport (depot)	0
2	Rizky UMY	136
3	Devia Putri	137
4	Ada Swalayan	137
5	Damai Indah S	142

6	WS Ridang	148
7	WS Pasar Mini	149
8	Konde Mart 3	151
9	Ridho Jaya	153
10	WS Madukismo	153
11	Toko Amin	154
12	Toko Bu Sudi	155
13	KUD Tani Makmur	156
14	Numart	160
15	Cahaya Swalayan	162
16	Rizky Celluler	169
17	Hans Mini Market	173
18	Toko Martiyem	177
19	WS Niten	187
20	Prima Srandakan	190
21	Atmaja Toserba	191
22	Almira	192
23	Toko Rani	192
24	Mega 4	192
25	Lestari	194
26	Kuncoro	194
27	Madina	196
28	Fadli	197
29	Familia Swalayan	198
30	JA Mart	200
31	WS Bantul	201
32	Toko Janat	201
33	MM An-Nur	202
34	Toko Mugiharjo	203
35	JM Mini Market	203
36	Kembar 1	205
37	Kembar 2 Swalayan	215
38	DM 6	217
39	Agromart	219
40	3 M	238
41	DM 4	238
42	Shafira	239
43	Govinda	239
44	Amanda 4	297

d. Memasukkan setiap *customer* ke dalam *cluster* sesuai urutan sapuan

Tahap ini dilakukan untuk mengelompokkan setiap *customer* berdasarkan urutan sudut polar. Pada Algoritma *Sweep* asli, pengelompokkan berdasarkan “sapuan” ini dilakukan dari satu *customer* ke *customer* lain (sesuai urutan sudut polar) hingga kapasitas kendaraan terpenuhi. Ketika kapasitas kendaraan sudah tidak mencukupi, *customer* akan dimasukkan ke dalam *cluster* selanjutnya. Begitu seterusnya hingga seluruh *customer* tersapu.

Pada penelitian ini dilakukan beberapa aturan tambahan bahwa pengelompokkan *customer* tidak hanya dilakukan berdasarkan urutan sudut polar dan kapasitas kendaraan, melainkan dengan tambahan bahwa setiap *cluster* terdiri tidak lebih dari 8 *customer*. Penambahan aturan ini dilakukan agar penyelesaian rute masing-masing *cluster* dapat dilakukan dengan *software* Lingo 12.0. Berikut hasil *cluster* masing-masing *customer*:

Tabel 4.7 Cluster berdasarkan Modifikasi Algoritma Sweep

Urutan Sudut Polar (°)	Nama Outlet	Demand				Cluster
0	CV Jogja Transport (depot)	0	0	0	0	
136	Rizky UMY	256	171	154	117	1
137	Devia Putri	212	136	154	117	
137	Ada Swalayan	28	28	18	18	
142	Damai Indah S	44	44	44	39	
148	WS Ridang	13	13	13	27	2
149	WS Pasar Mini	13	13	13	13	
151	Konde Mart 3	30	30	30	30	
153	Ridho Jaya	10	10	7	10	
153	WS Madukismo	55	55	55	65	
154	Toko Amin	14	14	14	31	
155	Toko Bu Sudi	14	14	14	13	

156	KUD Tani Makmur	12	12	12	19	3
160	Numart	7	7	7	7	
162	Cahaya Swalayan	8	8	10	8	
169	Rizky Celluler	6	6	6	6	
173	Hans Mini Market	11	11	11	16	
177	Toko Martiyem	19	19	19	19	
187	WS Niten	37	37	37	39	
190	Prima Srandakan	84	84	91	76	
191	Atmaja Toserba	54	54	54	54	
192	Almira	84	82	84	95	4
192	Toko Rani	10	10	19	12	
192	Mega 4	14	14	14	19	
194	Lestari	44	44	56	57	
194	Kuncoro	8	8	8	13	
196	Madina	22	22	25	29	
197	Fadli	30	29	29	47	
198	Familia Swalayan	21	21	21	21	5
200	JA Mart	11	11	15	13	
201	WS Bantul	15	15	15	20	
201	Toko Janat	56	57	71	70	
202	MM An-Nur	41	41	50	46	
203	Toko Mugiharjo	92	99	140	210	
203	JM Mini Market	13	13	13	13	
205	Kembar 1	10	10	28	15	
215	Kembar 2 Swalayan	63	63	65	65	
217	DM 6	51	47	50	48	6
219	Agromart	7	7	7	12	
238	3 M	63	62	64	64	
238	DM 4	50	50	50	50	
239	Shafira	11	11	11	11	
239	Govinda	19	19	19	19	
297	Amanda 4	128	128	130	132	

4.3.2 Penentuan Rute Masing-Masing *Cluster* dengan MILP

4.3.2.1 Formulasi Model Matematika

Model matematika dalam penelitian ini mengadopsi model yang dikembangkan oleh Kallehauge et al (2001). Model yang

dikembangkan Kallehauge et al. (2001) dianggap sesuai dengan karakteristik sistem yang diteliti. Perbedaan model ini dengan model Kallehauge et al. (2001) terletak pada fungsi tujuan, fungsi tujuan pada model ini yakni untuk meminimalkan jarak yang ditempuh kendaraan pengangkut sementara pada model Kallehauge et al. (2001) fungsi tujuan yang ingin dicapai yakni meminimalkan biaya distribusi. Selain itu, model yang digunakan pada penelitian ini bersifat TSP atau hanya menggunakan satu kendaraan, dengan mengabaikan batasan kapasitas kendaraan. Hal ini dikarenakan batasan kapasitas telah terpenuhi atau diselesaikan pada tahap *clustering customer*.

Asumsi yang digunakan dalam model ini adalah sebagai berikut:

1. Kondisi kendaraan baik (tidak rusak) dan perjalanan lancar (tidak macet).
2. *Volume* semua jenis roti dianggap sama.
3. Waktu *setup* di depot 2 jam atau 120 menit.
4. Waktu pelayanan di tiap *customer* 1.5 jam atau 90 menit.

Notasi himpunan yang digunakan dalam model matematika penelitian ini adalah sebagai berikut:

C = himpunan *customer*

N = himpunan *vertices* atau node terdiri dari *customer* dan depot.

Indeks yang digunakan dalam model matematika meliputi sebagai berikut:

i = menyatakan tiap-tiap *customer*, dimana $i = \{1,2,3, \dots, n\}$
 $0, n + 1$ = menyatakan depot

Parameter yang digunakan dalam model matematika meliputi sebagai berikut:

d_{ij} = jarak yang ditempuh dari depot/*customer* i ke depot/*customer* j

m_i = waktu dimulai pelayanan pada *customer* i

s_i = waktu pelayanan di *customer* i

t_{ij} = waktu perjalanan dari depot/*customer* i ke depot/*customer* j

R = bilangan riil yang sangat besar

a_i = waktu awal dimulai pelayanan di *customer* i

b_i = waktu berakhirnya pelayanan di *customer* i

a. Variabel Keputusan

Variabel keputusan dalam penelitian ini adalah:

x_{ij} = variabel biner (0,1), bernilai 1 jika terjadi perjalanan dari depot/*customer* i ke depot/*customer* j dan bernilai 0 jika tidak.

m_i = waktu dimulai pelayanan pada depot/*customer* i .

b. Model Matematis

Model matematis yang digunakan dalam menentukan rute kendaraan pada penelitian ini dituliskan sebagai berikut:

$$Z_{VRPTW} = \min \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ij} \quad (1)$$

Batasan:

$$\sum_{i \in N} x_{ij} = 1 \quad \forall_i \in C \quad (2)$$

$$\sum_{j \in N} x_{0j} = 1 \quad \forall_j \in C \quad (3)$$

$$\sum_{i \in N} x_{ih} - \sum_{j \in N} x_{hj} = 0 \quad \forall_h \in C \quad (4)$$

$$\sum_{i \in N} x_{i(n+1)} = 1 \quad \forall_i \in C \quad (5)$$

$$m_i + s_i + t_{ij} - R(1 - x_{ij}) \leq m_j \quad \forall_{i,j} \in N \quad (6)$$

$$a_i \leq s_i \leq b_i \quad \forall_i \in N \quad (7)$$

$$x_{ij} \in \{0,1\} \quad \forall_{i,j} \in N \quad (8)$$

Fungsi tujuan yang hendak dicapai dituliskan dalam bentuk persamaan (1) yakni untuk meminimalkan jarak keseluruhan yang ditempuh oleh kendaraan. Batasan permasalahan yang ditunjukkan dengan batasan (2) menyatakan bahwa setiap *customer* dikunjungi tepat satu kali. Batasan (3), (4), dan (5) menyatakan jalur yang dilalui kendaraan yakni kendaraan berangkat dari depot, kemudian mengunjungi salah satu *customer*, dimana setelah mengunjungi satu *customer*, kendaraan akan pergi meninggalkan *customer* tersebut untuk menuju *customer* selanjutnya, hingga kemudian kendaraan kembali ke depot. Batasan (6) digunakan untuk menyatakan bahwa kendaraan tidak

diperbolehkan sampai di *customer j* sebelum $m_i + s_i + t_{ij}$ atau sebelum waktu dimulai pelayanan ditambah waktu pelayanan pada *customer i* dan ditambah waktu perjalanan dari *i* ke *j*, dimana R merupakan bilangan riil yang bernilai besar. Batasan (7) digunakan untuk memastikan bahwa batasan *time window* terpenuhi. Batasan (8) menyatakan bahwa variabel keputusan x_{ij} merupakan variabel keputusan biner yang bernilai 0 atau 1.

4.3.2.2 Input Lingo 12.0

Input Lingo 12.0 menyatakan model matematis yang diubah ke dalam bahasa Lingo 12.0. Lingo 12.0 yang digunakan merupakan Lingo versi edukasi, dimana *software* ini memiliki keterbatasan hanya mampu menjalankan 4000 kendala, 8000 variabel, 800 variabel integer, 800 variabel non linier, dan 20 variabel global. Penulisan bahasa LINGO 12.0 untuk *cluster* 1 adalah sebagai berikut:

```

model:
!TSP CLUSTER 1 terdiri dari 5 titik (1 sebagai depot)
setiap customer memiliki time window (a,b), waktu pelayanan
ditiap customer (s2-s5) = 90, sementara pelayanan di depot s1
= 120
customer dan kendaraan terhubung dalam -> perjalanan (x),
waktu perjalanan (dur), jarak (d), waktu dimulai pelayanan
(m);

!parameter input:
A(I)           = WAKTU BUKA CUSTOMER I
B(I)           = WAKTU TUTUP CUSTOMER I
S(I)           = WAKTU PELAYANAN DI CUSTOMER I
D(I,J)         = JARAK I KE J
DUR(I,J)       = WAKTU PERJALANAN DARI I KE J

!variabel yang dicari:
X(I,J)         = 1 JIKA TERJADI PERJALANAN DARI TITIK I KE TITIK J,
0 SEBALIKNYA
M(I)           = WAKTU DIMULAI PELAYANAN DI TITIK I OLEH K
;

```

```

sets:
  NODE/1..5 /: S, A, B, M;
  PERJALANAN(NODE, NODE): X, D, DUR;

endsets

data:
  A = @OLE('E:\REAL SKRIPSI\OLAH DATA\OLAH_DATA.xlsx','a_1');
  B = @OLE('E:\REAL SKRIPSI\OLAH DATA\OLAH_DATA.xlsx','b_1');
  D = @OLE('E:\REAL SKRIPSI\OLAH DATA\OLAH_DATA.xlsx','d_1');
  DUR = @OLE('E:\REAL SKRIPSI\OLAH
DATA\OLAH_DATA.xlsx','t_1');
  S = 120 90 90 90 90;
  R = 10000000;

enddata

!fungsi tujuan: minimasi jarak;
MIN = @SUM(NODE(I): @SUM(NODE(J) | I#NE#J :
      D(I,J) * X(I,J)));

!Batasan:
!setiap titik dikunjungi sekali;
@FOR(NODE(J) | J #GT# 1 : @SUM(NODE(I) | I #NE# J : X(I,J)) =
1);

!berawal di depot;
@FOR(NODE(I) | I #EQ# 1 : @SUM (NODE(J) | J #GT# 1 : X(I,J))=
1);

!jalur;
@FOR(NODE(H) : @SUM(NODE(I) | I #NE# H : X(I,H)) -
@SUM(NODE(J) | J #NE# H : X(H,J)) = 0);

!berakhir di depot;
@FOR(NODE(J) | J #EQ# 1 : @SUM (NODE(I) | I #GT# 1 : X(I,J)) =
1);

!fisibilitas;
@FOR(NODE(I) | I #NE# 1 : @FOR(NODE(J) :
      M(J) >= M(I) + S (I) + DUR (I,J) - R * (1-X(I,J)) ));

!time windows;
@FOR(NODE(I) | I #NE# 1 : A(I) <= M(I));
@FOR(NODE(I) | I #ne# 1 : B(I) >= M(I) + S(I));

!biner;
@FOR(PERJALANAN(I,J) : @BIN(X(I,J)));

end

```

Sementara *input* Lingo 12.0 untuk *cluster* 2, 3, 4, 5 dan 6 dilampirkan dalam lampiran C.

4.3.2.3 Output Lingo 12.0

Output Lingo 12.0 menunjukkan hasil yang didapat setelah *input* dikerjakan menggunakan perintah *solve*. *Output* Lingo 12.0 berupa *solution report* yang merupakan hasil pengolahan input. Berikut merupakan *solution report cluster 1*:

Sementara untuk *solution report cluster 2, 3, 4, 5, dan 6* dilampirkan dalam lampiran D.

```

Global optimal solution found.
Objective value:                22.31600
Objective bound:                22.31600
Infeasibilities:                0.000000
Extended solver steps:          1
Total solver iterations:        252

Model Class:                    MILP

Total variables:                30
Nonlinear variables:            0
Integer variables:              25

Total constraints:              40
Nonlinear constraints:          0

Total nonzeros:                 144
Nonlinear nonzeros:            0

```

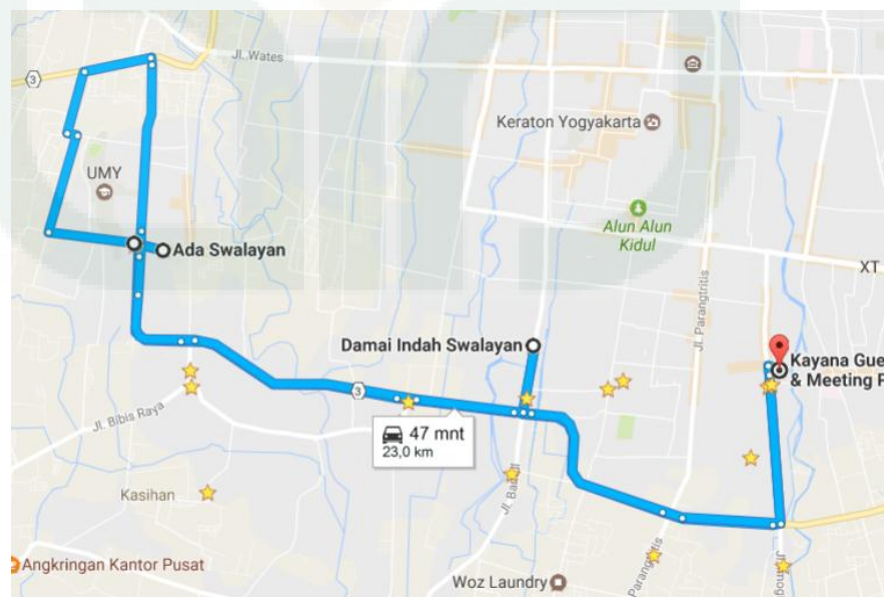
Variable	Value	Reduced Cost
R	0.1000000E+08	0.000000
S (1)	120.0000	0.000000
S (2)	90.00000	0.000000
S (3)	90.00000	0.000000
S (4)	90.00000	0.000000
S (5)	90.00000	0.000000
A (1)	480.0000	0.000000
A (2)	480.0000	0.000000
A (3)	480.0000	0.000000
A (4)	480.0000	0.000000
A (5)	480.0000	0.000000
B (1)	840.0000	0.000000
B (2)	1320.000	0.000000
B (3)	1320.000	0.000000
B (4)	1320.000	0.000000
B (5)	1260.000	0.000000
M (1)	1335.900	0.000000
M (2)	667.3740	0.000000
M (3)	577.3500	0.000000
M (4)	763.3740	0.000000
M (5)	480.0000	0.000000
X (1, 1)	0.000000	0.000000
X (1, 2)	0.000000	8.500000
X (1, 3)	0.000000	8.500000
X (1, 4)	0.000000	8.200000
X (1, 5)	1.000000	4.800000
X (2, 1)	0.000000	10.60000
X (2, 2)	0.000000	0.000000
X (2, 3)	0.000000	0.1600000E-01
X (2, 4)	1.000000	4.000000
X (2, 5)	0.000000	7.000000
X (3, 1)	0.000000	10.60000
X (3, 2)	1.000000	0.1600000E-01
X (3, 3)	0.000000	0.000000

X(3, 4)	0.000000	4.000000
X(3, 5)	0.000000	7.000000
X(4, 1)	1.000000	8.600000
X(4, 2)	0.000000	2.600000
X(4, 3)	0.000000	2.600000
X(4, 4)	0.000000	0.000000
X(4, 5)	0.000000	5.000000
X(5, 1)	0.000000	4.800000
X(5, 2)	0.000000	4.900000
X(5, 3)	1.000000	4.900000
X(5, 4)	0.000000	4.600000
X(5, 5)	0.000000	0.000000
D(1, 1)	0.000000	0.000000
D(1, 2)	8.500000	0.000000
D(1, 3)	8.500000	0.000000
D(1, 4)	8.200000	0.000000
D(1, 5)	4.800000	0.000000
D(2, 1)	10.600000	0.000000
D(2, 2)	0.000000	0.000000
D(2, 3)	0.16000000E-01	0.000000
D(2, 4)	4.000000	0.000000
D(2, 5)	7.000000	0.000000
D(3, 1)	10.600000	0.000000
D(3, 2)	0.16000000E-01	0.000000
D(3, 3)	0.000000	0.000000
D(3, 4)	4.000000	0.000000
D(3, 5)	7.000000	0.000000
D(4, 1)	8.600000	0.000000
D(4, 2)	2.600000	0.000000
D(4, 3)	2.600000	0.000000
D(4, 4)	0.000000	0.000000
D(4, 5)	5.000000	0.000000
D(5, 1)	4.800000	0.000000
D(5, 2)	4.900000	0.000000
D(5, 3)	4.900000	0.000000
D(5, 4)	4.600000	0.000000
D(5, 5)	0.000000	0.000000
DUR(1, 1)	0.000000	0.000000
DUR(1, 2)	12.750000	0.000000
DUR(1, 3)	12.750000	0.000000
DUR(1, 4)	12.300000	0.000000
DUR(1, 5)	7.200000	0.000000
DUR(2, 1)	15.900000	0.000000
DUR(2, 2)	0.000000	0.000000
DUR(2, 3)	0.24000000E-01	0.000000
DUR(2, 4)	6.000000	0.000000
DUR(2, 5)	10.500000	0.000000
DUR(3, 1)	15.900000	0.000000
DUR(3, 2)	0.24000000E-01	0.000000
DUR(3, 3)	0.000000	0.000000
DUR(3, 4)	6.000000	0.000000
DUR(3, 5)	10.500000	0.000000
DUR(4, 1)	12.900000	0.000000
DUR(4, 2)	3.900000	0.000000
DUR(4, 3)	3.900000	0.000000
DUR(4, 4)	0.000000	0.000000
DUR(4, 5)	7.500000	0.000000
DUR(5, 1)	7.200000	0.000000
DUR(5, 2)	7.350000	0.000000
DUR(5, 3)	7.350000	0.000000
DUR(5, 4)	6.900000	0.000000
DUR(5, 5)	0.000000	0.000000

Row	Slack or Surplus	Dual Price
1	22.31600	-1.000000
2	0.000000	0.000000
3	0.000000	0.000000
4	0.000000	0.000000
5	0.000000	0.000000
6	0.000000	0.000000
7	0.000000	0.000000
8	0.000000	0.000000
9	0.000000	0.000000
10	0.000000	0.000000
11	0.000000	0.000000
12	0.000000	0.000000
13	0.1000056E+08	0.000000
14	9999910.	0.000000
15	9999820.	0.000000
16	0.000000	0.000000
17	9999712.	0.000000

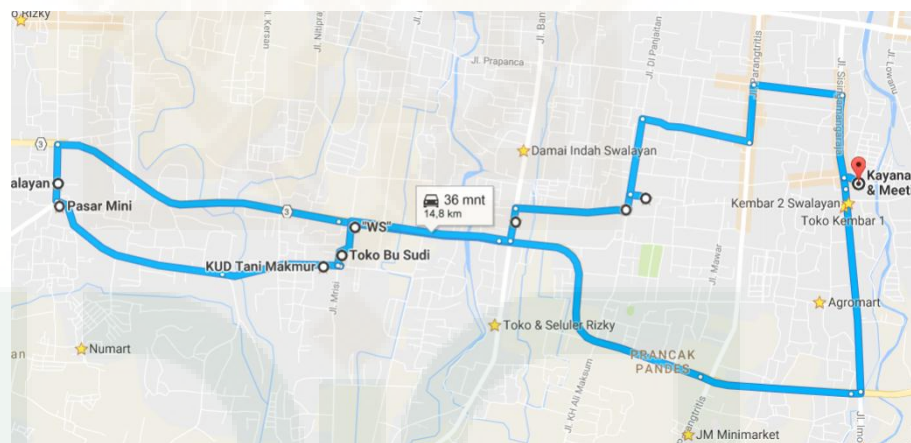
18	0.1000065E+08	0.000000
19	0.000000	0.000000
20	9999910.	0.000000
21	0.1000009E+08	0.000000
22	9999802.	0.000000
23	469.6260	0.000000
24	9999810.	0.000000
25	9999720.	0.000000
26	9999910.	0.000000
27	9999619.	0.000000
28	0.1000076E+08	0.000000
29	0.1000009E+08	0.000000
30	0.000000	0.000000
31	0.1000019E+08	0.000000
32	9999910.	0.000000
33	187.3740	0.000000
34	97.35000	0.000000
35	283.3740	0.000000
36	0.000000	0.000000
37	562.6260	0.000000
38	652.6500	0.000000
39	466.6260	0.000000
40	690.0000	0.000000

Berdasarkan *solution report cluster 1*, diketahui bahwa rute yang terbentuk memiliki total jarak tempuh sejauh 22.316 km dengan urutan toko yang dikunjungi yakni 1 → 5 → 3 → 2 → 4 → 1. Node 1 menyatakan CV Jogja transport atau depot, node 5 menyatakan Damai Indah S, node 3 menyatakan Devia Putri, node 2 menyatakan Rizky UMY, dan node 4 menyatakan Ada Swalayan. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 1:



Gambar 4.7 Rute Cluster 1

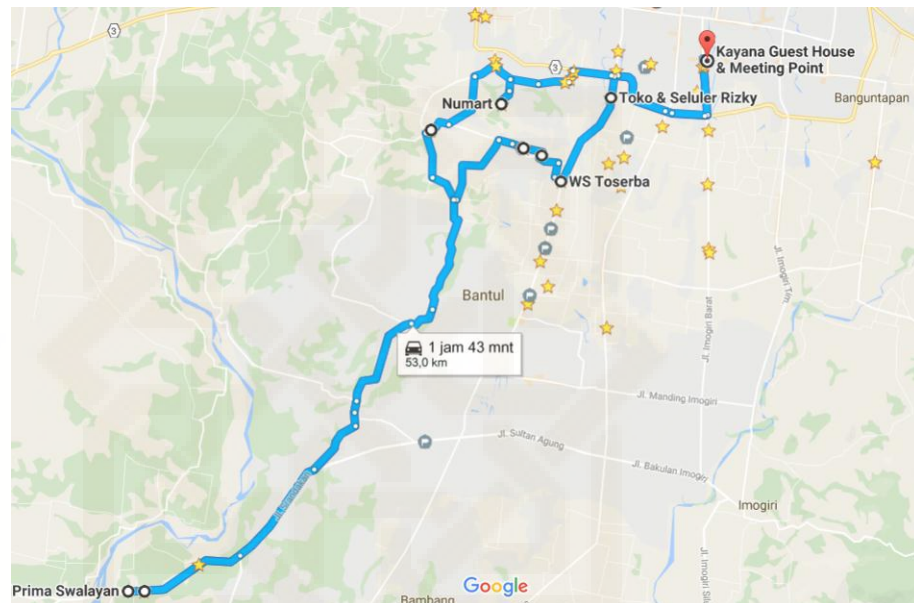
Berdasarkan *solution report cluster 2*, diketahui rute yang terbentuk memiliki total jarak tempuh sejauh 14.75 km dengan urutan toko yang dikunjungi yakni $1 \rightarrow 6 \rightarrow 2 \rightarrow 3 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 5 \rightarrow 4 \rightarrow 1$. Node 1 menyatakan CV Jogja Transport atau depot, node 6 menyatakan WS Madukismo, node 2 menyatakan WS Ridang, node 3 menyatakan WS Pasar Mini, node 9 menyatakan KUD Tani Makmur, node 8 menyatakan Toko Bu Sudi, node 7 menyatakan Toko Amin, node 5 menyatakan Ridho Jaya, dan node 4 menyatakan Konde Mart 3. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 2:



Gambar 4.8 Rute Cluster 2

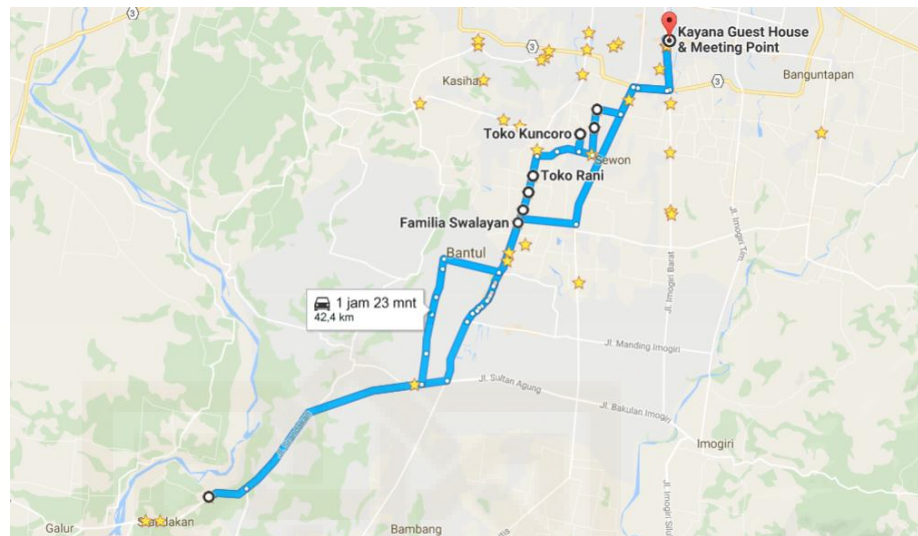
Berdasarkan *solution report cluster 3*, diketahui rute yang terbentuk memiliki total jarak tempuh sejauh 53.05 km dengan urutan toko yang dikunjungi yakni $1 \rightarrow 2 \rightarrow 3 \rightarrow 9 \rightarrow 8 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 4 \rightarrow 1$. Node 1 menyatakan CV Jogja Transport atau depot, node 2 menyatakan Numart, node 3 menyatakan Cahaya Swalayan, node 9 menyatakan Atmaja Toserba, node 8 menyatakan Prima Srandakan,

node 5 menyatakan Hans Mini Market, node 6 menyatakan Toko Martiyem, node 7 menyatakan WS Niten, dan node 4 menyatakan Rizky Celluler. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 3:



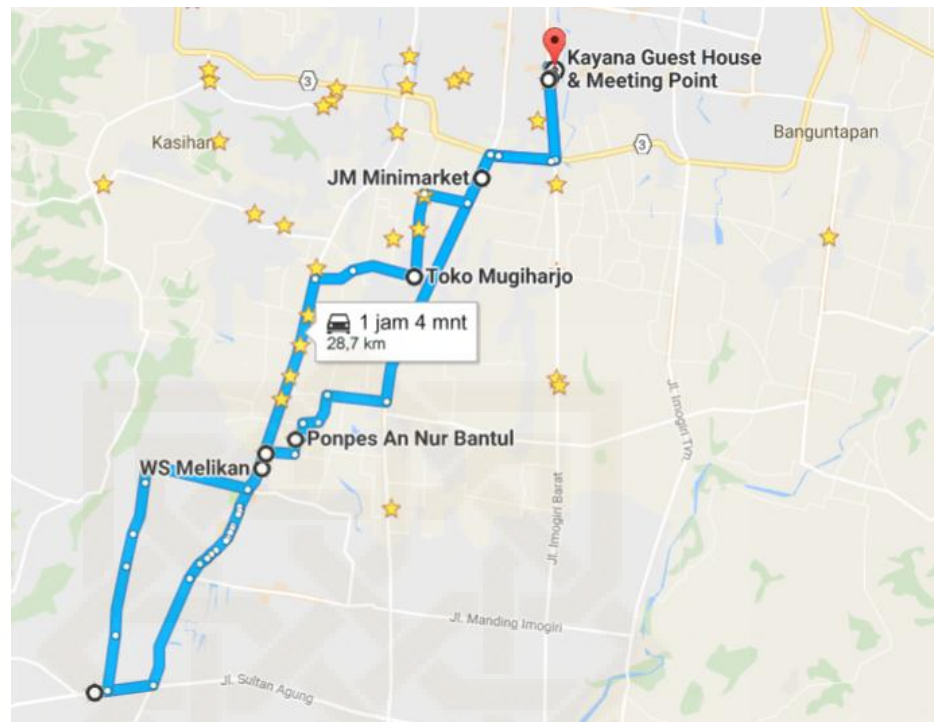
Gambar 4.9 Rute Cluster 3

Berdasarkan *solution report cluster 4*, diketahui rute yang terbentuk memiliki total jarak tempuh sejauh 41.45 km dengan urutan toko yang dikunjungi yakni $1 \rightarrow 4 \rightarrow 9 \rightarrow 7 \rightarrow 5 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 2 \rightarrow 1$. Node 1 menyatakan CV Jogja Transport atau depot, node 4 menyatakan Mega 4, node 9 menyatakan Familia Swalayan, node 7 menyatakan Madina, node 5 menyatakan Lestari, node 3 menyatakan Toko Rani, node 6 menyatakan Kuncoro, node 8 menyatakan Fadli, dan node 2 menyatakan Almira. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 4:



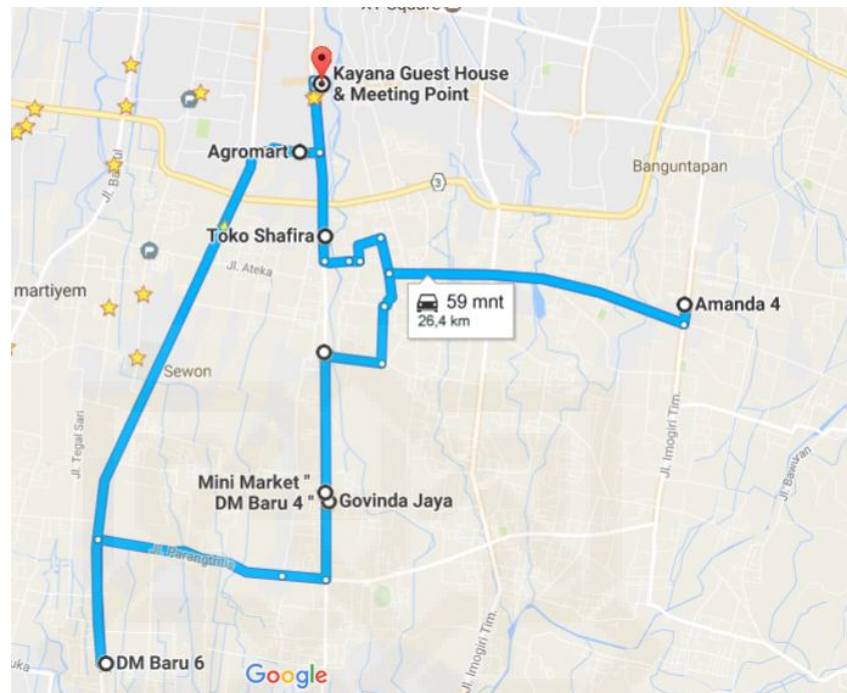
Gambar 4.10 Rute *Cluster 4*

Berdasarkan *solution report cluster 5*, diketahui rute yang terbentuk memiliki total jarak tempuh sejauh 27.54 km dengan urutan toko yang dikunjungi yakni $1 \rightarrow 8 \rightarrow 7 \rightarrow 5 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 9 \rightarrow 1$. Node 1 menyatakan CV Jogja Transport atau depot, node 8 menyatakan Kembar 1, node 7 menyatakan JM Mini Market, node 5 menyatakan MM An-Nur, node 2 menyatakan JA Mart, node 3 menyatakan WS Bantul, node 4 menyatakan Toko Janat, node 6 menyatakan Toko Mugiharjo, dan node 9 menyatakan Kembar 2 Swalayan. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 5:



Gambar 4.11 Rute *Cluster 5*

Berdasarkan *solution report cluster 6*, diketahui rute yang terbentuk memiliki total jarak tempuh sejauh 26.3 km dengan urutan toko yang dikunjungi yakni $1 \rightarrow 6 \rightarrow 8 \rightarrow 4 \rightarrow 5 \rightarrow 7 \rightarrow 2 \rightarrow 3 \rightarrow 1$. Node 1 menyatakan CV Jogja Transport atau depot, node 6 menyatakan Shafira, node 8 menyatakan Amanda 4, node 4 menyatakan 3M, node 5 menyatakan DM 4, node 7 menyatakan Govinda, node 2 menyatakan DM 6, dan node 3 menyatakan Agromart. Berikut adalah gambaran rute untuk *cluster* atau kendaraan 6:



Gambar 4.12 Rute *Cluster 6*

4.3.3 Ongkos Bahan Bakar

a. Kendaraan 1

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 22.316 = \text{Rp } 9646.41 \end{aligned}$$

b. Kendaraan 2

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 14.75 = \text{Rp } 6375.90 \end{aligned}$$

c. Kendaraan 3

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 53.05 = \text{Rp } 22931.62 \end{aligned}$$

d. Kendaraan 4

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 41.45 = \text{Rp } 17917.35 \end{aligned}$$

e. Kendaraan 5

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 27.54 = \text{Rp } 11904.56 \end{aligned}$$

f. Kendaraan 6

$$\begin{aligned} \text{Ongkos bahan bakar} &= \text{Ongkos bahan bakar per km} \times \text{jarak tempuh kendaraan} \\ &= \text{Rp } 432.26 \times 26.3 = \text{Rp } 11368.55 \end{aligned}$$

4.3.4 Rute Usulan

Rute usulan yang terbentuk dari hasil pengolahan data yang dilakukan, ditampilkan pada tabel berikut:

Tabel 4.8 Rute Usulan

No Kendaraan	Sales	Rute	Jarak Tempuh (km)	Waktu Tempuh (menit)	Ongkos Bahan Bakar (Rp)
1	Sogi	0 → 32 → 43 → 21 → 42 → 0	22.316	33.474	9646.41
2	Wahyu	0 → 23 → 24 → 26 → 35 → 25 → 37 → 34 → 31 → 0	14.75	22.125	6375.90
3	Dian	0 → 33 → 29 → 18 → 15 → 22 → 27 → 13 → 28 → 0	53.05	79.575	22931.62
4	Pandu	0 → 14 → 17 → 16 → 20 → 10 → 38 → 5 → 4 → 0	41.45	62.175	17917.35
5	Fandy	0 → 36 → 6 → 39 → 11 → 12 → 19 → 41 → 9 → 0	27.54	41.31	11904.56
6	Feri	0 → 1 → 40 → 7 → 8 → 2 → 3 → 30 → 0	26.3	39.45	11368.55
Total Jarak Tempuh			185.406 km		
Total Waktu Tempuh			278.109 menit		
Total Ongkos Bahan Bakar			Rp 80144.38		

4.4. Analisis dan Pembahasan

Pengolahan data yang dilakukan untuk menentukan rute optimal bagi pendistribusian produk Sari Roti oleh CV Jogja Transport dilakukan dengan metode heuristik *cluster first route second*. Fase pertama pada metode ini dimulai dengan

cluster first dimana pada penelitian ini pengelompokan dilakukan menggunakan Modifikasi Algoritma *Sweep*. Pengelompokan menggunakan Algoritma *Sweep* ditujukan untuk memecah permasalahan menjadi kelompok-kelompok kecil. Pemecahan permasalahan tersebut dikarenakan keterbatasan kemampuan *software* pengolah data untuk mencari rute optimal. *Software* yang digunakan dalam penentuan rute merupakan *software* berbasis edukasi sehingga memiliki keterbatasan dari jumlah batasan maupun variabel.

Rute awalan yang saat ini digunakan CV Jogja Transport merupakan rute yang berdasarkan zona wilayah. Zona wilayah ini merupakan hasil pengelompokan perusahaan yang membagi peta wilayah Bantul, DIY menjadi sejumlah sales atau kendaraan yang dimiliki perusahaan yakni 6 zona. Pembagian zona wilayah seperti ini berdampak pada adanya sales atau kendaraan yang harus kembali ke depot ditengah proses distribusi akibat keterbatasan kapasitas angkut kendaraan. Kendaraan yang kembali ke depot akan kembali melakukan pengiriman ke *customer* yang sebelumnya belum terpenuhi. Hal ini terjadi pada dua dari enam kendaraan. Untuk menyelesaikan permasalahan ini, maka pada penelitian ini, pengelompokan dilakukan menggunakan metode modifikasi Algoritma *Sweep*. Algoritma *Sweep* merupakan salah satu metode *cluster first-routes second*, dimana proses pencarian rute dimulai dari mengelompokkan *customer* sesuai dengan “sapuan” sudut polar dari yang terkecil hingga terbesar baru kemudian ditentukan rute masing-masing kelompoknya sesuai dengan kapasitas *software* dan kapasitas kendaraan. Pengelompokan dengan cara ini memungkinkan *customer* dengan letak geografis berdekatan dapat masuk ke dalam satu kelompok.

Fase kedua pada penelitian ini yakni menentukan rute kendaraan untuk masing-masing *cluster*. Penentuan rute pada tahap kedua metode heuristik ini menggunakan TSP, dimana permasalahan TSP bertujuan menentukan rute dengan jarak minimum untuk sebuah kendaraan. Permasalahan TSP ini kemudian diselesaikan menggunakan metode eksak MILP karena pada karakteristik sistem yang diteliti terdapat variabel berupa variabel integer, pecahan, maupun biner. Variabel integer diantaranya terdapat pada data permintaan dan data *time windows*, sementara variabel pecahan terdapat pada data jarak tempuh kendaraan dan waktu tempuh kendaraan. Variabel biner pada penelitian ini digunakan sebagai variabel keputusan ada tidaknya perjalanan dari titik i ke titik j . Model TSP yang berbentuk MILP ini kemudian diselesaikan menggunakan *software* Lingo 12.0 versi edukasi.

Perusahaan membagi rute kendaraan 1 dengan urutan depot → Almira → Fadli → Toko Mugiharjo → 3M → DM 4 → depot, dengan jarak tempuh 14.7 km dan waktu tempuh 22.05 menit. Sementara itu, kendaraan 2 memiliki urutan depot → Kembar 2 Swalayan → Konde Mart 3 → Damai Indah S → Rizky Celluler → Ridho Jaya → Toko Martiyem → Hans Mini Market → Cahaya Swalayan → JM Mini Market → DM 6 → Shafira → Govinda → depot, dengan jarak tempuh 41.11 km dan waktu tempuh 60.465 menit. Kendaraan 3 rute awalan meliputi depot → WS Madukismo → Toko Bu Sudi → KUD Tani Makmur → Numart → WS Pasar Mini → WS Ridang → Ada Swalayan → Agromart → depot → Devia Putri → Rizky UMY → depot, dengan total jarak tempuh 37.916 km dan waktu tempuh 61.899 menit.

Kendaraan 4 pada rute awalan memiliki urutan depot → Toko Janat → Atmaja Toserba → Prima Srandakan → Mega 4 → Toko Rani → Lestari → depot, dengan jarak tempuh 46.95 km, dan waktu tempuh 69.075. Sementara itu, kendaraan 5 memiliki urutan depot → Amanda 4 → Familia Swalayan → WS Bantul → JA Mart → Madina → WS Niten → depot, dengan jarak tempuh 28.33 km dan waktu tempuh 43.2 menit. Kendaraan 6 memiliki urutan depot → Kembar 1 → Toko Amin → Kuncoro → depot, dengan jarak tempuh 39.2 km dan waktu tempuh 28.83 menit.

Jarak tempuh keseluruhan rute awalan yang digunakan perusahaan saat ini yakni sejauh 208.206 km dengan waktu tempuh selama 285.519 menit. Dengan total jarak sedemikian, perusahaan memberikan ongkos bahan bakar sebesar Rp 15,000.00 untuk masing-masing kendaraan. Pembagian ongkos oleh perusahaan yang tidak berdasarkan jarak tempuh masing-masing kendaraan, mampu menimbulkan rasa iri antara sales satu dengan yang lain. Oleh karena itu, pada penelitian ini dilakukan perhitungan ongkos bahan bakar sesuai jarak yang ditempuh oleh masing-masing kendaraan. Hal ini dilakukan untuk mengatasi kemungkinan permasalahan ketidakpuasan sales terhadap ongkos yang sama rata sementara jarak yang ditempuh tidak sama rata.

Berdasarkan pengolahan data yang dilakukan menggunakan metode heuristik *cluster first route second, customer* terbagi menjadi enam kelompok. Pembagian ini sesuai dengan kemampuan *software* pengolah data pencarian rute sekaligus sesuai dengan jumlah kendaraan perusahaan.

Kendaraan 1 pada rute usulan berdasarkan pengolahan data yang dilakukan memiliki urutan depot → Damai Indah Sawalayan → Devia Putri → Rizky UMY → Ada Swalayan → depot, dengan total jarak tempuh sejauh 22.316 km, waktu tempuh selama 33.474 menit, dan ongkos bahan bakar sebesar Rp 9,646.41. Sementara itu, kendaraan 2 memiliki urutan depot → WS Madukismo → WS Ridang → WS Pasar Mini → KUD Tani Makmur → Toko Bu Sudi → Toko Amin → Ridho Jaya → Konde Mart 3 → depot, dengan jarak tempuh 14.75 km, waktu tempuh 22.125 menit, dan ongkos bahan bakar Rp 6,375.90. Kendaraan 3 memiliki urutan yakni depot → Numart → Cahaya Swalayan → Atmaja Toserba → Prima Srandakan → Hans Mini Market → Toko Martiyem → WS Niten → Rizky Celluler → depot, dengan total jarak tempuh 53.05 km, waktu tempuh 79.575 menit, dan ongkos bahan bakar Rp 22,931.62.

Kendaraan 4 pada rute usulan memiliki urutan depot → Mega 4 → Familia Swalayan → Madina → Lestari → Toko Rani → Kuncoro → Fadli → Almira → depot, dengan jarak tempuh sejauh 41.45 km, waktu tempuh selama 62.175 menit, dan ongkos bahan bakar sebesar Rp 17,917.35. Sementara kendaraan 5 memiliki urutan depot → Kembar 1 → JM Mini Market → MM An-Nur → JA Mart → WS Bantul → Toko Janat → Toko Mugiharjo → Kembar 2 Swalayan → depot, dengan jarak tempuh sejauh 27.54 km, waktu tempuh selama 41.31 menit, dan ongkos bahan bakar Rp 11,904.56. Kendaraan 6 memiliki urutan yakni depot → Shafira → Amanda 4 → 3M → DM 4 → Govinda → DM 6 → Agromart → depot, dengan jarak tempuh sejauh 26.3 km, waktu tempuh selama 39.45 menit, dan ongkos bahan bakar Rp 11,368.55.

Secara keseluruhan, perbedaan antara rute awalan dan rute usulan cukup signifikan. Hal ini terlihat dari perbedaan *customer* yang harus dilayani masing-masing kendaraan, total jarak tempuh, total waktu tempuh, dan ongkos bahan bakar. Jarak tempuh rute usulan mampu mengurangi jarak tempuh hingga 22.8 km, mengurangi waktu tempuh 7.41 menit, dan mengurangi ongkos bahan bakar sebesar Rp 9,855.62. Perbandingan rute awalan dan rute usulan dapat dilihat pada tabel berikut:

Tabel 4.9 Perbandingan Rute Awalan dan Rute Usulan

No	Perbandingan	Rute Awalan	Rute Usulan	% Penghematan
1	Kendaraan 1	0 → 4 → 5 → 41 → 7 → 8 → 0	0 → 32 → 43 → 21 → 42 → 0	NA
2	Kendaraan 2	0 → 9 → 31 → 32 → 28 → 34 → 27 → 22 → 29 → 6 → 3 → 1 → 2 → 0	0 → 23 → 24 → 26 → 35 → 25 → 37 → 34 → 31 → 0	
3	Kendaraan 3	0 → 23 → 25 → 35 → 33 → 26 → 24 → 42 → 30 → 0 → 43 → 21 → 0	0 → 33 → 29 → 18 → 15 → 22 → 27 → 13 → 28 → 0	
4	Kendaraan 4	0 → 19 → 18 → 15 → 14 → 10 → 20 → 0	0 → 14 → 17 → 16 → 20 → 10 → 38 → 5 → 4 → 0	
5	Kendaraan 5	0 → 40 → 17 → 12 → 11 → 16 → 13 → 0	0 → 36 → 6 → 39 → 11 → 12 → 19 → 41 → 9 → 0	
6	Kendaraan 6	0 → 36 → 37 → 38 → 39 → 0	0 → 1 → 40 → 7 → 8 → 2 → 3 → 30 → 0	
7	Total Jarak Tempuh	208.206 km	185.406 km	10.95%
8	Total Waktu Tempuh	285.519 menit	278.109 menit	2.60%
9	Total Ongkos Bahan Bakar	Rp 90000	Rp 80144.38	10.95%

Keterangan: NA = *not available*

Persentase penghematan total jarak tempuh yang didapat jika perusahaan menerapkan rute usulan adalah sebesar 10.95%, ditambah penghematan total waktu tempuh 2.60% dan penghematan ongkos bahan bakar sebesar 10.95%. Dengan jarak tempuh yang lebih pendek, diharapkan perusahaan mampu menekan

pengeluaran terkait bahan bakar, dan *maintenance* yang mungkin diperlukan jika jarak tempuh kendaraan terlalu jauh dan waktu tempuh kendaraan terlalu lama.

Penyelesaian TSP pada masing-masing *cluster* menggunakan metode eksak MILP memang menjamin bahwa rute usulan yang terbentuk pada setiap kelompok memiliki hasil yang optimal dalam meminimalkan jarak tempuh kendaraan. Akan tetapi, peng*clusteran* yang pada awalnya dilakukan untuk memecah permasalahan menjadi potongan-potongan kecil agar dapat diselesaikan *software* dan memenuhi batasa kapasitas, menyebabkan hasil rute tersebut belum dapat dikatakan optimal secara keseluruhan. Pemecahan *customer* dalam kelompok-kelompok tersebut, berdampak pada kemungkinan adanya kombinasi rute lain yang mungkin lebih optimal.