BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Distribusi adalah kegiatan untuk memindahkan produk dari pihak supplier ke pihak konsumen dalam suatu *supply chain* (Chopra, 2001). Distribusi terjadi di antara tahapan dari *supply chain*. Aliran bahan baku yang diperlukan berpindah dari supplier menuju suatu perusahaan pembuat produk dan perusahaan tersebut akan memindahkan barang jadi yang dihasilkan ke tangan konsumen. Distribusi merupakan suatu kunci dari keuntungan yang akan diperoleh perusahaan karena distribusi secara langsung akan mempengaruhi biaya dari *supply chain* dan kebutuhan konsumen. Jaringan distribusi yang tepat dapat digunakan untuk mencapai berbagai macam tujuan dari *supply chain*, mulai dari biaya yang rendah sampai respon yang tinggi terhadap permintaan dari pelanggan (Chopra, 2001).

Manajemen distribusi dan transportasi pada umumnya melakukan sejumlah fungsi dasar yaitu melakukan segmentasi dan menentukan target *service level*, menentukan model transportasi yang akan digunakan, melakukan penjadwalan dan penentuan rute pengiriman, memberikan pelayanan nilai tambah, penyimpanan persediaan dan menangani pengembalian (Pujawan dan Mahendrawathi, 2010). Dalam sebuah proses distribusi, faktor yang diutamakan adalah kepuasan pelanggan. Sebuah proses distribusi yang baik mencakup ketepatan waktu pengiriman, efisiensi biaya, produk yang berkualitas, dan layanan yang baik.

Untuk menjamin ketepatan pengiriman produk baik waktu, kualitas maupun jumlah produk ke konsumen diperlukan perencanaan distribusi dan transportasi yang baik. Distribusi yang optimal tergantung dari kompleksitas pendistribusian produk, yang akan semakin meningkat tingkat kesulitannya karena adanya beberapa tujuan (*allocation*), kapasitas dan keterbatasan sumber daya (*source*) yang harus dipenuhi bersama-sama dengan tujuan untuk meminimalkan biaya distribusi (Ikfan dan Masudin, 2014).

Terdapat beberapa penelitian terdahulu yang membahas mengenai distribusi. Pada umumnya penelitian-penelitian tersebut diselesaikan dengan evaluasi rute

distribusi, kemudian mengelompokkan rute baru yang optimal sehingga menghasilkan jarak yang terpendek dan biaya minimum.

Penelitian yang dilakukan Mardiani *et al.* (2014) mengenai penentuan rute untuk pendistribusian BBM menggunakan algoritma *Nearest Neighbour* di PT. X . Dalam pendistribusian BBM, PT. X menggunakan 4 jenis kendaraan dengan kapasitas 16 kilo liter sebanyak 29 kendaraan, 24 kilo liter sebanyak 17 kendaraan, 32 kilo liter sebanyak 17 kendaraan dan 40 kilo liter sebanyak 3 kendaraan. Pendistribusian dilakukan ke 104 SPBU yang tersebar di masing-masing daerah di antaranya 55 SPBU untuk daerah Karawang, 18 SPBU untuk daerah Purwakarta dan 32 SPBU untuk daerah Subang. Mardiani *et al.* (2014) menggunakan metode *Nearest Neighbour* dengan tujuan dapat mereduksi waktu keterlambatan yang kerap terjadi. Metode *Nearest Neighbour* dapat mereduksi waktu keterlambatan karena menghubungkan lokasi distribusi berdasarkan jarak terpendek dari lokasi yang terpilih sebelumnya. Dari hasil peneletiannya Mardiani *et al.* (2014) menetapkan 98 tur untuk pengiriman BBM untuk semua daerah di antaranya 56 tur untuk daerah Bekasi, 17 tur untuk daerah Karawang, 7 tur untuk daerah Purwakarta dan 18 tur untuk daerah Subang. Pengurangan kendaraan untuk proses distribusi juga mengalami pengurangan, yang awalnya menggunakan 66 kendaraan menjadi 51 kendaraan. Penggunaan metode *Nearest Neighbour* selain berdampak pada penurunan jumlah penggunaan kendaraan, juga berdampak pada penurunan biaya total operasional proses distribusi sebesar 18,88%.

Slamet *et al.* (2014) mengenai pembuatan jalur pendistribusian sayuran di dataran tinggi untuk meminimalkan biaya total transportasi tanpa mengorbankan waktu penyelesaian tujuan untuk mengurnagi resiko penurunan kualitas sayuran selama perjalanan. Penelitian dilakukan di PT. Saung Mirwan yang memiliki pelanggan tetap yang tersebar di daerah Jakarta. PT. Saung Mirwan memiliki 5 armada dengan kapasitas 75 peti dan 2 armada dengan kapasitas 175 peti. Sayuran yang didistribusikan adalah selada, kembang kol, tomat, sawi dan seledri. Pembuatan jalur pendistribusian sayur di dataran tinggi menggunakan algoritma genetika. Mencari solusi optimal dengan melakukan simulasi ke lokasi distribusi sebagai titik tujuan. Algoritma genetika dikembangkan untuk menyelesaikan masalah dengan cara mencari himpunan solusi terbaik yang bertahan hidup dan melakukan rekombinasi solusi yang kurang baik untuk mendapatkan kromosom lain yang lebih baik pada generasi berikutnya.

Penerapan algoritma genetika menghasilkan pemabgian rute distribusi sayuran dari depot sentra produksi ke 6 klaster konsumen. Dari hasil penerapan algoritma genetika diperoleh penghematan waktu pendistribusian dari depot ke seluruh konsumen sebesar 32,22% dan penghematan penggunaan armada distribusi sebanyak 1 buah atau 14,28%.

2.2. Penelitian Sekarang

Penelitian yang dilakukan saat ini mengenai implementasi *heterogeneous fleet vehicle routing problem* di UD. Garuda. UD. Garuda merupakan agen pendistribusian multiplek untuk 20 toko di Daerah Istimewa Yogyakarta. Pendistribusian multiplek di UD. Garuda menggunakan 4 kendaraan angkut yang memiliki kapasitas yang berbeda antara lain, kendaraan angkut dengan kapasitas 350, 240, 180 dan 120 lembar per sekali angkut. Pendistribusian multiplek dilakukan 2 kali dalam 1 minggu, tepatnya pada hari Jumat dan Sabtu. Pendistribusian yang dilakukan pada hari Sabtu dilakukan apabila pendistribusian yang dilakukan pada hari Jumat belum dapat memenuhi seluruh permintaan pelanggan. Pesanan pelanggan akan dikumpulkan terlebih dahulu kemudian didistribusikan pada hari Jumat dan Sabtu. Masing-masing kendaraan hanya boleh melakukan pendistribusian 1 kali pada tiap hari pengiriman. Penelitian dilakukan dengan menggunakan metode *Route-First, Cluster-Second.* Proses pembuatan rute menggunakan algoritma *Travelling Salesman Problem* (TSP) dengan bantuan *software* LINGO 13, kemudian pengklasteran dilakukan dengan memotong-motong jalur yang telah terbentuk berdasarkan kapasitas kendaraan yang dapat memenuhi pesanan pelanggan. Pemotongan jalur distribusi memperhatikan permintaan pelanggan terhadap kapasitas kendaraan serta biaya operasional yang terdiri dari biaya tetap dan biaya variabel dari setiap kendaraan. Pemotongan jalur dilakukan dengan pembuatan program sederhana di *Microsoft Excel* 2007. Penelitian ini bertujuan untuk menemukan jalur yang optimal dan mendapatkan biaya yang optimal dalam pendistribusian multiplek.

Tabel 2.1. Perbedaan Penelitian Terdahulu dan Sekarang

Peneliti	Tujuan penelitian	Kasus Penelitian	Metode yang digunakan	Hasil penelitian
Mardiani *et al.* (2014)	Penentuan rute distribusi untuk pendistribusian BBM	*Heterogeneous Fleet Vehicle Routing Problem* (HVRP)	*Nearest Neighbour*	Keterlambatan waktu pengiriman dapat dihilangkan serta biaya total operasional kendaraan untuk pendistribusian BBM dapat dihemat sebanyak 18,88%
Slamet *et al.* (2014)	Pembuatan jalur distribusi sayur di dataran tinggi untuk meminimalkan biaya total transportasi	*Heterogeneous Fleet Vehicle Routing Problem (*HVRP)	Algoritma genetika dengan simulasi	Pembentukan enam cluster untuk pendistribusian ke pelanggan. Mengurangi waktu pendistribusian 32,22% dan mengurangi penugasan armada distribusi yang awalnya tujuh menjadi enam armada
Saputra (2015)	Penugasan kendaraan dengan jalur yang optimal dan dengan biaya terendah	*Heterogeneous Fleet Vehicle Routing Problem* (HVRP)	*Travelling Salesman Problem (TSP), Routing first cluster second*	Menghasilkan satu rute tetap dengan berdasarkan algoritma TSP, terbentuknya program yang dapat membantu penugasan kendaraan berdasarkan permintaan yang dapat dipenuhi oleh kapasitas maksimal kendaraan serta dengan biaya operasional kendaraan yang rendah

2.3. Dasar Teori

Dalam sub bab ini akan dipaparkan mengenai teori-teori pendukung penyelesaian penelitian yang dilakukan,

2.3.1. Saluran Distribusi

Saluran distribusi adalah saluran yang digunakan oleh produsen untuk menyalurkan barang dari produsen sampai ke konsumen. Saluran distribusi merupakan struktur yang mengambarkan alternatif saluran yang dipilih dan menggambarkan situasi pemasaran yang berbeda oleh berbagai macam perusahaan atau lembaga usaha (Swastha, 1984).

Saluran distribusi juga diartikan oleh Kotler (1997), sebagai serangkaian organisasi yang saling bergantung dan terlibat dalam proses menjadikan barang atau jasa untuk digunakan. Pemilihan saluran distribusi harus benar-benar dilakukan, karena kesalahan dalam menentukan saluran distribusi akan berdampak pada terhambatnya proses penyaluran barang atau jasa tersebut.

Dalam saluran distribusinya, produsen sering menggunakan perantara sebagai penyalurnya. Perantara (*middleman*) merupakan kegiatan usaha yang berdiri sendiri. Perantara berada di antara produsen dan konsumen akhir atau pemakai industri. Mereka memberikan pelayanan dalam hubungannya dengan pembelian dan penjualan barang dari produsen ke konsumen. Penghasilan yang diterima adalah hasil dari transaksi antar produsen dan konsumen tersebut (Swastha, 1984).

Menurut Swastha (1984) Dalam pendistribusian, perantara (*middleman*) digolongkan menjadi dua golongan yaitu:

a. Perantara Pedagang

   Perantara pedagang (*merchant middleman*) bertanggung jawab terhadap pemilikan semua barang yang dipasarkannya. Ada dua kelompok yang termasuk dalam perantara pedagang yaitu pedagang besar *(wholesaler)* dan pengecer (*retailer).*

b. Perantara Agen

   Perantara agen *(agent middleman)* tidak memiliki hak miliki atas semua barang yang mereka tangani. Perantara agen dapat digolongkan menjadi dua golongan yaitu:

i. Agen penunjang

Agen penunjang adalah agen yang ikut secara aktif dalam pemindahan barang dari Produsen ke konsumen. Contohnya : agen pengangkutan

ii. Agen Pelengkap

Agen pelengkap adalah agen yang tidak ikut secara aktif dalam pemindahan barang tetapi memberikan bantuan serta memperlancar pemindahan tersebut. Contoh: perusahaan asuransi dan bank.

2.3.2. Faktor-Faktor Yang Mempengaruhi Kegiatan Distribusi

Faktor-faktor yang mempengaruhi kegiatan distribusi menurut Nitisemito (1977), meliputi:

a. Faktor pasar

Saluran distribusi dipengaruhi oleh pola pembelian konsumen, yaitu jumlah konsumen, letak geografis konsumen, jumlah pesanan dan kebiasaan dalam pembelian.

b. Faktor Barang

Faktor barang yang menjadi pertimbangan adalah nilai unit, besar dan berat barang, mudah rusaknya barang, standar barang dan pengemasan.

c. Faktor Perusahaan

Faktor perusahaan yang menjadi pertimbangan adalah sumber dana, pengalaman dan kemampuan manajemen serta pengawasan dan pelayanan yang diberikan.

d. Faktor kebiasaan dalam pembelian

Faktor kebiasaan dalam pembelian yang menjadi pertimbangan adalah kegunaan perantara, sikap perantara terhadap kebijaksanaan produsen, volumen penjualan dan ongkos penyaluran barang.

2.3.3. Fungsi Dasar Manajemen Distribusi dan Transportasi

Menurut Pujawan dan Mahendrawathi (2010), manajemen distribusi dan transportasi pada umumnya melakukan sejumlah fungsi dasar yang terdiri dari:

a. Melakukan segmentasi dan menentukan target *service level*

   Segmentasi pelanggan perlu dilakukan karena kontribusi mereka pada *revenue* perusahaan bisa sangat bervariasi dan karakteristik setiap pelanggan bisa sangat berbeda antar satu pelanggan dengan pelanggan lainnya.

b. Menentukan mode transportasi yang akan digunakan

   Setiap mode transportasi memiliki karakteristik yang berbeda dan mempunyai keunggulan serta kelemahan yang berbeda juga. Manajemen transportasi harus bisa menentukan mode apa yang akan digunakan dalam mengirimkan atau mendistribusikan produk ke pelanggan.

c. Melakukan konsolidasi informasi dan pengiriman

   Konsolidasi informasi dapat dilakukan dengan pertukaran informasi permintaan dari berbagai regional *distribution center* oleh *central warehouse* untuk keperluan pembuatan jadwal pengiriman. Sedangkan konsolidasi pengiriman dilakukan dengan menyatukan permintaan beberapa toko atau ritel yang berbeda dalam sebuah truk.

d. Melakukan penjadwalan dan penentuan rute pengiriman

   Salah satu kegiatan operasional yang dilakukan oleh distributor adalah menentukan kapan sebuah truk harus berangkat dan rute mana yang harus dilalui untuk memenuhi permintaan dari sejumlah pelanggan.

e. Memberikan pelayanan nilai tambah

   Beberapa proses nilai tambah yang dilakukan oleh distributor seperti pengepakan, pelabelan harga, pemeberian *barcode* dan lain-lain.

f. Penyimpanan persediaan

   Jaringan distribusi selalu melibatkan proses penyimpanan produk baik di satu gudang pusat atau gudang regional maupun di toko mana produk tersebut akan dijual.

g. Menangani pengembalian

   Manajemen distribusi juga punya tanggung jawab untuk melaksanakan kegiatan pengembalian. Pengembalian dapat terjadi karena barang rusak atau tidak terjual sampai batas waktu penjualan habis.

2.3.4. Tipe Strategi Distribusi

Tipe-tipe strategi distribusi menurut *Levi et al.* (2003), yaitu:

a. *Cross Docking*

Pada strategi ini, produk didistribusikan secara terus-menerus dari supplier menuju *warehouse* kemudian dilanjutkan ke konsumen. *Cross docking* merupakan salah satu teknik logistik yang relatif masih baru, yang digunakan pada pusat distribusi dan industri transportasi. Sistem ini berfungsi untuk mengkonsilidasikan antara produk yang tiba di pusat distribusi untuk selanjutnya dikirim ke retailer dengan memperhatikan faktor waktu dan beban muatan *transporter.* Produk yang bagus untuk penerapan *cross docking* sama seperti *Just In Time Manufacturing*, dapat berjalan pada variasi yang rendah dan terdapat jumlah yang cukup untuk memenuhi permintaan konsumen.

b. *Direct Shipment*

Dalam strategi ini, produk dikirim langsung dari supplier ke ritel tanpa melalui pusat distribusi. Tidak terdapat penyimpanan persediaan pada *warehouse*, karena *warehouse* atau pusat distribusi tidak ada.

c. *Warehousing*

Strategi ini merupakan strategi klasik, di mana warehouse menyediakan stok dan melayani pelanggan sesuai dengan permintaan. *Warehousing* dapat diartikan sebagai bagian logistik yang mengatur masalah penyimpanan produk pada produksi, konsumsi, dan di antara produksi dan konsumsi. Aktivitas *warehousing* juga menyediakan informasi bagi pihak manajemen tentang status, kondisi dan disposisi item produk yang disimpan. Dalam pelaksanaannya *warehouse* adalah istilah yang lebih umum dibandingkan dengan *distribution center.*

2.3.5. Saluran Distribusi Barang Konsumsi

Dalam penyaluran barang konsumsi yang ditujukan untuk pasar konsumen, Swastha (1984), membaginya menjadi lima macam saluran, di antaranya sebagai berikut:

a. Produsen – Konsumen

   Saluran ini disebut saluran distribusi langsung, karena memungkinkan produsen langsung dapat menjual barangnya langsung kepada konsumen.

b. Produsen – Pengecer – Konsumen

   Saluran distribusi ini juga disebut saluran distribusi langsung. Pengecer melakukan pembelian barang kepada produsen. Adapula beberapa produsen yang mendirikan toko pengecer sehingga dapat secara langsung melayani konsumen.

c. Produsen – Pedagang Besar – Pengecer – Konsumen

   Saluran distribusi seperti ini banyak digunakan oleh produsen dan dinamakan sebagai saluran saluran distribusi tradisional. Dalam saluran distribusi ini produsen hanya melayani penjualan dalam jumlah yang besar kepada pedagang besar saja. Pembelian oleh pengecer dilayani oleh pedagang besar dan pembelian oleh konsumen dilayani oleh pengecer saja.

d. Produsen – Agen – Pengecer – Konsumen

   Dalam saluran distribusi ini, produsen memilih agen (agen penjualan atau agen pabrik) sebagai penyalurnya. Agen menjalankan perdagangan besar dalam saluran distribusi yang ada. Sasaran penjualannya ditujukan kepada pengecer besar.

e. Produsen – Agen – Pedagang Besar – Pengecer – Konsumen

   Dalam saluran distribusi ini, produsen sering menggunakan agen sebagai perantara untuk menyalurkan barangnya kepada pedagang besar yang kemudian menjualnya kepada toko-toko kecil. Agen yang terlibat adalah agen penjualan.

2.3.6. *Vehicle Routing Problem* (VRP)

Vehicle Routing Problem (VRP) adalah masalah penentuan rute-rute yang optimal dari satu depot menuju sejumlah pelanggan yang tersebar secara geografis dengan memperhatikan sejumlah batasan (Laporte,1992). Batasan yang muncul antar lain berupa setiap pelanggan dikunjungi hanya satu kali oleh satu kendaraan, setiap kendaraan berawal dan berakhir di depo, setiap kendaraan dapat melayani lebih dari satu rute atau banyak trip (*multiple trips*), waktu pengiriman tiap rute tidak melebihi watu tertentu (*time horison*).

Menurut (Toth dan Vigo, 2002), secara umum bentuk dasar *Vehicle Routing Problem* (VRP) berkaitan dengan masalah penentuan suatu himpunan rute kendaraan (*vehicle*) yang melayani satu himpunan konsumen yang diasosiasikan dengan *vertex* dan *demand* (permintaan) yang diketahui dan rute yang menghubungkan depot dengan konsumen dan satu konsumen dengan konsumen yang lain yang dinamakan dengan *arc*.

Ada beberapa karakterisitik dalam VRP yang perlu diperhatikan. Yang pertama adalah adalah komponen-komponen yang berkaitan dalam VRP (Toth dan Vigo , 2002) ,yaitu :

a. Pelanggan
b. Depo
c. Pengemudi
d. Rute Kendaraan

Variasi bentuk VRP muncul tergantung pada suatu kondisi atau karakteristik yang ada. Kondisi tersebut terdiri dari sejumlah faktor, kendala, dan fungsi tujuan.

Suprayogi (2003), memberikan beberapa contoh karakteristrik dari VRP dalam hal kendala yang ada dalam VRP tersebut berdasarkan batasan atau kendala yang ada antara lain:

a. VRP *Time Windows* (VRPTW)
   Setiap pelanggan memiliki rentang waktu dalam pelayanan, pelayanan harus dilakukan pada renatang waktu (*time windows*) masing-masing pelanggan.
b. VRP *Split Delivery* (VRPSD)
   Pelanggan dapat dilayani lebih dari satu kendaraan, hal ini biasanya terjadi karena terbatasnya kapasitas kendaraan dalam melayani pelanggan.
c. VRP *Pick Up and Delivery* (VRPPD)
   Kendaraan melakukan dua tugas sekaligus, yaitu melakukan pengambilan dan pengantaran produk pada pelanggan.
d. VRP *Multiple Depots* (VRPMD)
   VRP ini memiliki depo lebih dari satu.

e. VRP *Multiple Products* (VRPMP)
   Karakteristik VRP ini adalah permintaan pelanggan lebih dari satu produk.

f. VRP *Multiple Trips* (VRPMT)
   Karakteristik dari VRP ini adalah satu kendaraan dapat menempuh beberapa rute untuk memenuhi kebutuhan pelanggan.

g. VRP *Heterogeneous Fleet of Vehicles* (VRPHFV)
   Kendaraan yang digunakan bermacam-macam dengan karakteristik yang berbeda-beda.

h. *Periodic* VRP (PVRP)
   Dalam VRP standar, horison perencanaan hanya berlaku pada satu hari, pada variasi VRP ini pelayanan kepada pelanggan dapat dilakukan dalam beberapa waktu selama horison perencanaan.

i. *Stochastic* VRP (SVRP)
   Parameter angka (seperti jumlah pelanggan, permintaan masing-masing pelanggan, waktu layanan) bersifat acak atau tidak pasti, setiap pelanggan memiliki kemungkinan untuk tidak dikunjungi setiap hari.

j. *Dynamic* VRP (DVRP)
   VRP jenis ini bertujuan untuk mengantisipasi apabila terdapat pelanggan baru pada rute tertentu, pelanggan baru ini harus disisipkan pada rute tambahan saat pembuatan rute pengiriman utama.

2.3.7. VRP *with Heterogeneous Vehicle* (HVRP)

Dantzig dan Ramser (1959) merupakan peneliti yang pertama kali memperkenalkan model *Vehicle Routing Problem* (VRP). Terdapat dua variasi umum VRP yaitu *homogeneous dan heterogeneous*. Penelitian ini difokuskan pada model *Heterogeneous* VRP (HVRP).

Pada umumnya banyak perusahaan yang memiliki kegiatan pengiriman barang dan jasa dengan kendaraan angkut yang heterogen. Persoalan HVRP dapat digambarkan sebagai berikut: Kendaraan yang telah teridentifikasi dengan jumlah kapasitas angkut (*Q)* tertentu harus mengirim sejumlah kuantitas pesanan $q_1(i = 1,...,n)$ suatu barang pada *n* pelanggan dari sebuah single depo *(i=0).* Di mana diketahui bahwa d_{ij} adalah jarak antara customer *i* dan *j (i,j=0,...,n),* sehingga problemnya adalah bagaimana menemukan suatu tur untuk kendaraan

di mana jarak tempuh kendaraan adalah yang paling singkat, setiap customer dilayani oleh satu kendaraan, kuantitas barang yang dikirimkan tidak melebihi kapasitas kendaraan pengangkut *(Q)* (Azmi *et al.,* 2008).

HVRP berbeda dengan VRP klasik, di mana HVRP berhubungan dengan armada kendaraan yang heterogen sedangkan VRP klasik berhubungan dengan armada kendaraan yang bersifat homogen jenisnya. HVRP membentuk sekumpulan rute kendaraan yang di mulai dan diakhiri pada depo sehingga setiap pelanggan dikunjungi tepat satu kali dan *demand* total dari satu rute tidak melebihi kapasitas angkut kendaraan yang ditugaskan pada rute tersebut. Pendekatan penyelesaian diaplikasikan pada karakteristik sebagai berikut: Diberikan sekumpulan distributor dan sebuah depo. Setiap distributor memiliki demand d_i . Jarak antar distributor *i* dan *j* adalah d_{ij}. terdapat tipe truk K, tiap truk berkapasitas Q_k dan Biaya C_k. (Azmi *et al.,* 2008).

2.3.8. *Route –First, Cluster-Second Method*

Metode ini termasuk dalam *two-phase method.* Menurut Toth dan Vigo (2002), *Two-phase method* merupakan metode yang membagi proses pembuatan rute ke dalam dua kondisi natural: *route* dan *cluster* . Pembuatan *route* fokus pada urutan perjalanan dalam rute (urutan pelanggan mana yang terlebih dahulu dilayani). Pembuatan *cluster* fokus pada pengelompokkan pelanggan berdasarkan karakteristik atau kendala yang dimiliki.

Dalam penelitian ini *route-first, cluster-second* diaplikasikan dengan membuat rute yang menghubungkan kedua puluh pelanggan dalam satu rute. Rute berawal dari depo dan diakhiri di depo. Pembuatan rute menggunakan algoritma *Travelling Salesman Problem* (TSP) dengan bantuan *software integer programming* yaitu LINGO 13. Setelah rute terbentuk, kemudian rute tersebut dibagi kedalam beberapa rute. Dalam penelitian ini pembagian rute didasarkan pada kapasitas kendaraan. Banyaknya pelanggan yang dapat dilayani oleh kendaraan tanpa melebihi kapasitas yang dimiliki kendaraan yang ditugaskan maka sejumlah toko tersebut manjadi rute yang harus dilayani oleh kendaraan tersebut.

2.3.9. *Integer Programming*

Menurut Winston (2003), *Integer programming* (IP) adalah masalah *linear programming* (LP) di mana beberapa atau semua variabelnya harus berupa

bilangan bulat non negatif. *Linear Programming* (LP) sendiri adalah *tools* untuk menyelesaikan masalah optimasi. Pada formulasi IP terdapat fungsi tujuan dan kendala-kendala. Dalam penelitian ini, IP digunakan untuk pembuatan rute yang optimal berdasarkan algoritma *Travelling Salesman Problem* (TSP). IP yang digunakan untuk pembuatan rute pada penelitian ini adalah LINGO 13.

2.3.10. *Travelling Salesman Problem* (TSP)

Travelling salesman problem (TSP) merupakan masalah yang terjadi pada seorang *salesman* untuk mengunjungi sejumlah tempat yang telah ditentukan dan telah diketahui jarak antar satu tempat ke tempat lainnya (Taha, 2007).

Travelling salesman problem (TSP) merupakan suatu permasalahan untuk seorang *salesman* yang harus berangkat dari sebuah depo untuk mengunjungi *n* Node atau kota, di mana setiap node hanya boleh dikunjungi sebanyak 1 (satu) kali, kemudian kembali ke depo semula dengan mengambil rute yang optimal. Rute yang optimal adalah rute yang memberikan total biaya, waktu tempuh, dan jarak yang paling minimum. *Output* yang dihasilkan dari perutean adalah urutan rangkaian lokasi yang harus dikunjungi oleh *salesman* dalam 1 kali *tour*.

Menurut Johnson dan McGeoch (1997), *Traveling Salesman Problem* (TSP) adalah suatu permasalahan mencari sebuah rute tertutup untuk mengunjungi sejumlah kota, di mana setiap kota hanya dikunjungi sekali dan kembali ke kota awal setelah semua kota dikunjungi. Beberapa algoritma yang pernah dipakai untuk memecahkan masalah TSP menurut Johnson dan McGeoch (1997) adalah algoritma *genetic,* algoritma *simulated annealing,* algoritma *tabu search,* algoritma *lin Kernighan,* algoritma *neutral network,* dan algoritma *local search.*

Berikut model matematis dari Travelling Salesman Problem (TSP)

Fungsi tujuan TSP menurut LINDO (2011):

$$\text{Minimasi: } \Sigma_i \Sigma_j C_{ij}\ X_{ij} \tag{2.1}$$

Fungsi Kendala

- Setiap titik j harus dikunjungi satu kali

$$\sum_{x \neq j}^{n} X_{ij} = 1 \text{ (untuk j = 1 sampai n)} \tag{2.2}$$

- Setiap titik i harus ditinggalkan tepat satu kali

$$\sum_{j \neq i}^{n} X_{ij} = 1 \text{ (untuk i = 1 sampai n)} \tag{2.3}$$

- *Subtours* tidak diperbolehkan untuk setiap subset S, tidak termasuk depo

$$\sum_{ij} \in_s X_{ij} \leq |S| - 1 \quad \text{untuk setiap S} \tag{2.4}$$

$|S|$ merupakan ukuran dari S

- Kendala 3 dapat diganti dengan

$$U_j \geq U_i + 1 - (1 - X_{ij})n \; untuk \; j = 2, 3, 4, \ldots; j \neq 1 \tag{2.5}$$

Keterangan:

C_{ij} = menunjukkan biaya perjalanan dari outlet I ke outlet j

X_{ij} = variabel keputusan di mana:

X_{ij} = 1, salesman mengunjungi outlet j setelah 1

X_{ij} = 0, kondisi lain

N = jumlah outlet

S = menunjukkan *subtours*

U_j = outlet ke j pada urutan rute yang dibentuk

2.3.11. Penyelesaian Menggunakan LINGO 13

LINGO 13 merupakan suatu alat komprehensif yang dirancang untuk memecahkan masalah Linear, *Nonlinear (convex* dan *nonconvex/global), Quadratic, Quadratically Constrained, Second Order Cone, Stochastic,* dan model optimisasi integer dengan lebih cepat, lebih mudah, dan lebih efisien. Beberapa hal yang perlu diperhatikan pada model LINGO menurut LINDO (2011):

a. *Comment* pada model ini harus dimulai dengan tanda seru (!) dan akan muncul dalam teks berwarna hijau.

b. Fungsi operator yang telah ditetapkan dalam LINGO akan muncul dalam teks berwarna biru

c. Teks lainnya akan muncul dalam teks berwarna hitam

d. Setiap pernyataan dalam LINGO harus diakhiri dengan tanda (;)

e. Nama variabel harus dimulai dengan sebuah huruf (A-Z). Karakter lainnya dalam nama variabel bisa menggunakan huruf-huruf atau kata, atau karakter *underscore* (_). Nama variabel dapat mencapai panjang hingga 32 karakter.

Penyelesaian menggunakan LINGO 13, menggunakan fitur sebagai berikut:

a. Menggunakan Command SETS pada LINGO 13

Command SETS digunakan untuk mengelompokkan hal-hal dari *variabel* yang sama pada program LINGO 13. SETS digunakan sebelum model *constraint* pada program LINGO 13 dan diakhiri dengan *command* ENDSETS. Beberapa fungsi set juga tersedia untuk digunakan dalam program LINGO 13. Fungsi-fungsi tersebut menurut LINDO (2011):

i. @FOR

Digunakan untuk menentukan setiap member suatu set dalam suatu *constraint.*

ii. @SUM

Digunakan untuk menyatakan jumlah suatu pernyataan dari seluruh member set.

iii. @MIN

Digunakan untuk memperhitungkan nilai minimum dari sebuah pernyataan dari seluruh member set.

iv. @MAX

Digunakan untuk memperhitungkan nilai maksimum dari sebuah pernyataan dari seluruh member set.

b. LINGO DATA *Section*

LINGO 13 menyediakan suatu bagian tersendiri untuk menetapkan setiap nilai dari *variabel* yang berbeda yaitu dengan menggunakan *command* DATA. Penulisan *command* DATA dilakukan setelah *command* SETS pada setiap model LINGO 13. Bagian ini dimulai dengan label DATA dan diakhiri dengan ENDDATA. Pernyataan dalam DATA diikuti dengan kalimat *object_list = value_list*. *Object list* memuat tentang nama-nama dari setiap atribut suatu set di mana nilainya telah ditetapkan.

c. Operasional dan Fungsi LINGO 13

LINGO 13 menyediakan suatu fungsi dan operasional sebagai *problem solving* dalam suatu model. Terdapat tiga tipe operasional yang digunakan oleh LINGO 13 yaitu aritmatika, logika dan relasi operasional. Operasional aritmatika adalah sebagai berikut:

i. Eksponen (^)

ii. Perkalian (*)

iii. Pembagian (/)

iv. Penjumlahan (+)

v. Pengurangan (-)

Operasional logika digunakan dalam kumpulan fungsi untuk menentukan kondisi *TRUE*/*FALSE* menurut LINDO (2011):

i. #LT# : *TRUE* jika argumen disebelah kiri secara tepat kurang dari argumen sebelah kanan, lainnya *FALSE*

ii. #LE# : *TRUE* jika argumen disebelah kiri kurang dari atau sama dengan argumen disebelah kanan,lainnya *FALSE*

iii. #GT# : *TRUE* jika argumen disebelah kiri secara tepat lebih besar dari argumen disebelah kanan, lainnya *FALSE*

iv. #GE# : *TRUE* jika argumen disebelah kiri secara lebih besar atau sama dengan argumen disebelah kanan, lainnya *FALSE*

v. #EQ# : *TRUE* jika kedua argumen sama, lainnya *FALSE*

vi. #NE# : *TRUE* jika kedua argument tidak sama, lainnya *FALSE*

vii. #AND# : *TRUE* jika kedua argumen bernilai *TRUE*, lainnya *FALSE*

viii. #OR# : *FALSE* hanya jika kedua argumen bernilai *FALSE*, lainnya *TRUE*

ix. #NOT# : *TRUE* jika argument adalah *FALSE*, lainnya *FALSE*

Relasi operasional digunakan ketika mendefinisikan batasan-batasan untuk sebuah model, antara lain:

a. Ekspresi sama dengan (=)

b. Pernyataan di sisi kiri kurang dari atau sama dengan sisi kanan (<=)

c. Pernyataan di sisi kiri lebih besar dari atau sama dengan sisi kanan (>=)

2.3.12. Fungsi dan Fitur *Microsoft Excel* yang Dipakai dalam Pembuatan Program

Dalam *Microsoft Excel* terdapat berbagai macam fungsi dalam dan fitur yang dapat membantu perhitungan dan pengolahan data. Berikut ini adalah beberapa fungsi dan fitur yang digunakan untuk pembuatan program dalam penelitian ini:

a. Fungsi *IF*

Fungsi *IF* membandingkan nilai suatu *cell*, dengan suatu kriteria tertentu yang dijadikan acuan. Jika nilai *cell* tersebut sesuai atau sama dengan nilai dari kriteria yang diinginkan maka akan bernilai *TRUE* dan apabila tidak sama maka bernilai *FALSE*. Sintaks penulisan fungsi IF adalah sebagai berikut:

=IF(*logical_test*;*value_if_true*;*value_if_false*)

Logical_test merupakan pernyataan yang akan dievaluasi ke nilai *TRUE* atau *FALSE*. *Value_if_true* merupakan nilai tetapan yang dibuat apabila argumen *Logical_test* mengevaluasi ke *TRUE. Value_if_false* merupakan nilai tetapan yang dibuat apabila argumen *Logical_test* mengevaluasi ke *FALSE*.

b. Fungsi *AND*

Fungsi *AND* digunakan untuk menetapkan nilai *TRUE* apabila semua argumen mengevaluasi ke nilai *TRUE* dan menetapkan nilai *FALSE* apabila satu atau beberapa argumen mengevaluasi ke nilai *FALSE*. Sintaks penulisan fungsi *AND* adalah sebagai berikut:

=AND(*logical1;[logical2];...*)

Logical1 merupakan kondisi pertama yang akan dievaluasi ke *TRUE* atau *FALSE*. *Logical2* merupakan kondisi tambahan yang akan dievaluasi ke *TRUE* atau *FALSE*.

c. Fungsi *INDEX*

Fungsi *INDEX* digunakan untuk mencari suatu nilai milik *cell* di dalam tabel yang dipilih oleh indeks nomor baris dan kolom . Sintaks penulisan fungsi *INDEX* adalah sebagai berikut :

=INDEX(array;row_num;[column_num])

Array merupakan *range* tempat pencarian nilai yang diinginkan, *row_num* merupakan posisi baris dari *cell* yang ingin dicari nilainya relatif terhadap *cell*

yang terletak pada bagian paling kiri atas *range* sedangkan *column_num* merupakan posisi kolom dari *cell* yang ingin dicari nilainya relatif terhadap *cell* yang terletak pada bagian paling kiri atas *range*.

d. Fungsi *MATCH*

Fungsi *MATCH* digunakan untuk mencari posisi relatif dari suatu *cell* spesifik dalam suatu *range*. Sintaks penulisan fungsi *MATCH* adalah sebagai berikut:

=MATCH(lookup_value;lookup_array;[match_type])

Lookup_value merupakan nilai referensi sebagai pembanding, *lookup_array* adalah *range* tempat pencarian dilakukan, sedangkan *match_type* adalah tipe kecocokan. Bila *match type* diisikan angka 0 maka pencarian akan dilakukan pencarian untuk menemukan posisi relatif dari *cell* yang memiliki nilai sama persis seperti *lookup_value*. Jika diisikan angka -1 maka akan dilakukan pencarian untuk menemukan posisi relatif dari *cell* yang memiliki nilai lebih besar dari *lookup_value*. Sedangkan jika diisikan angka 1 maka akan mencari posisi relatif dari *cell* yang memiliki nilai lebih kecil dari *lookup_value*. Fungsi *MATCH* dapat disarangkan ke dalam Fungsi *INDEX* sehingga akan berfungsi seperti *VLOOKUP*.

e. Fungsi *MIN*

Fungsi *MIN* digunakan untuk mencari nilai minimal dari sekumpulan nilai. Sintaks penulisan fungsi *MIN* adalah :

=MIN(number1;[number2];...)

Number diperlukan sebagai nilai dari satu atau beberapa *cell* yang akan dicari nilai terkecilnya.

f. Fungsi *SUM*

Fungsi *SUM* digunakan untuk melakukan operasi penjumlahan terhadap sekumpulan nilai. Sintaks penulisan fungsi *SUM* adalah :

=SUM(number1;[number2];...)

Number diperlukan sebagai nilai dari satu atau beberapa *cell* yang akan dijumlahkan nilainya.

g. Fungsi *COUNTIF*

Fungsi *COUNTIF* adalah menghitung jumlah *cell* dalam suatu rentang, yang memenuhi kriteria yang telah ditentukan. Sintaks penulisan fungsi *COUNTIF* adalah*:*

=COUNTIF(range;criteria)

Range diperlukan untuk menetapkan rentang *cell* yang akan dihitung jumlahnya sesuai denga kriteria yang telah ditentukan*. Criteria* diperlukan sebagai tetapan nilai yang akan dihitung jumlah *cell*nya pada *range* yang terpilih.

h. Fungsi *SMALL*

Fungsi *SMALL* mencari urutan angka terkecil ke-sekian dari suatu *array.* Sintaks penulisan fungsi *SMALL* adalah:

=SMALL(array;k)

Array diperlukan untuk menentukan baris atau kolom yang akan dicari urutan nilai terkecilnya*. K* diperlukan untuk menentukan urutan nilai terkecil yang akan dicari.

i. Fungsi *ROW*

Fungsi *Row* untuk mengambil data baris dari suatu referensi. Sintaks penulisan fungsi *ROW* adalah:

=ROW([reference])

Reference dibutuhkan sebagai rentang *cell* yang hendak didapatkan nomor barisnya.

j. Fungsi *OR*

Fungsi *OR* untuk menetapkan suatu argumen bernilai *TRUE* apabila terdapat satu dari beberapa argumen bernilai *TRUE* dan menetapkan argumen *FALSE* apabila seluruh argumen bernilai *FALSE.* Sintaks penulisan fungsi *OR* adalah:

=OR(logical1;[logical2];...)

Logical diperlukan sebagai nilai yang akan ditetapkan nilainya apakah bernilai *TRUE* atau *FALSE.*

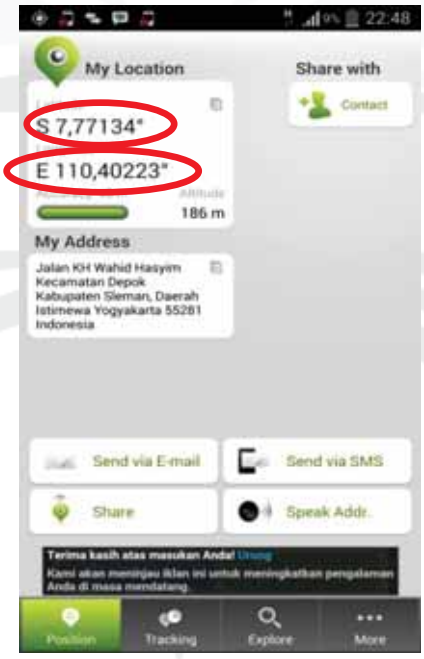
k. Fitur *Conditional Formatting*

Fitur *Conditional Formating* dapat memudahkan kita untuk memformat *cell* secara otomatis bila *cell* tersebut memenuhi kriteria tertantu. *Cell* yang diformat biasanya mempunyai nilai-nilai tertentu yang khas sehingga perlu dibedakan formatnya dengan *cell* lain supaya mudah dikenali.

2.3.13. Aplikasi *One Touch Location*

Aplikasi *One Touch Location* adalah aplikasi yang memberikan kemudahan dalam mencari titik koordinat keberadaan seseorang disuatu daerah. Aplikasi *One Touch Locaton* dapat di akses melalui *smart phone* dengan sistem operasi android. Aplikasi *One Touch Location dapat* dijalankan apabila *smart phone* terhubung dengan jaringan internet, selain itu *smart phone* harus dilengkapi dengan *Global Positioning System* (GPS).

Langkah awal dalam pencarian titik koordinat menggunakan aplikasi *One Touch Location* dengan terlebih dahulu berada disekitar lokasi yang hendak diketahui titik koordinatnya, setelah itu menjalankan aplikasi ini dari *smart phone* yang terhubung dengan jaringan internet.



Gambar 2.1. Tampilan Aplikasi *One Touch Location* Saat di Jalankan

Titik koordinat yang diberikan aplikasi *One Touch Location* terdiri dari titik latitude dan longitude yang pada contoh ini diberi lingkaran merah. Titik koordinat yang

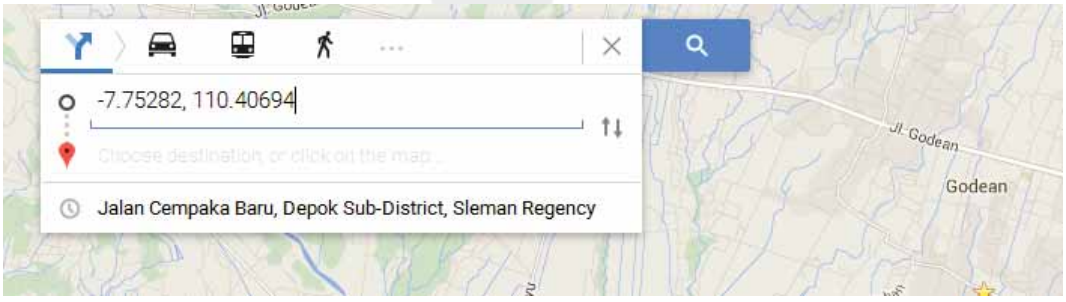
didapatkan dari aplikasi *One Touch Location* akan dimanfaatkan untuk pencarian lokasi dan pengukuran jarak tempuh di aplikasi *Google Maps.*

Titik koordinat yang didapatkan melalui aplikasi *One Touch Location* mengalami sedikit perubahan pada saat dimasukkan kedalam aplikasi *Google Maps.* Setiap titik koordinat di bawah garis khatulistiwa pada *Google Maps* diawali dengan tanda (-) dan titik koordinat lokasi yang berada di sebelah kanan *Greenwich* akan bertanda (+). Begitu pula titik koordinat lokasi yang berada di atas garis khatulistiwa diawali dengan tanda (+) dan titik koordinat yang berada disebelah kiri *Greenwich* akan diawali dengan tanda (-).

Penulisan titik koordinat yang terdiri dari titik longitude dan latitude di dalam aplikasi *Google Maps* juga harus diperhatikan. Penulisan titik latitude terlebih dahulu dituliskan kemudian disusul dengan penulisan titik longitude dengan tanda (,) sebagai pemisah. Dari gambar 2.1. titik longitude dan latitude dituliskan seperti berikut: -7.7134,110.40223.

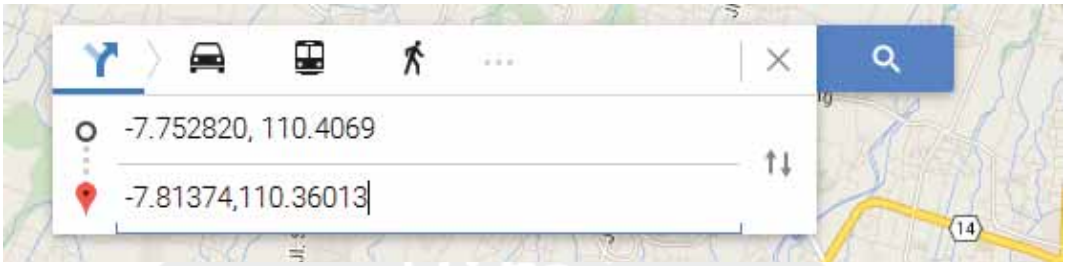
2.3.14. Aplikasi *Google Maps*

Google Maps merupakan aplikasi peta elektronik berbasis web yang disediakan oleh google secara gratis. *Google Maps* dapat diakses melalui website http;//maps.google.com. *Google Maps* dapat digunakan untuk mencari jarak dari suatu tempat ke tempat lain. Pencarian jarak dapat dilakukan dengan memasukkan nama tempat, titik koordinat atau alamat dari suatu tempat yang hendak diketahui keberadaan dan jaraknya dari titik keberadaan saat ini atau titik awal. Untuk mencari suatu lokasi yang hendak diketahui jaraknya dan keberadaan lokasinya, langkah awal yang dilakukan adalah memasukkan data lokasi suatu tempat pada *search bar*, pada contoh kali ini data lokasi yang hendak dicari jaraknya berupa titik koordinat yang terdiri dari titik longitude dan latitude yang diperoleh dari aplikasi *One Touch Location.*



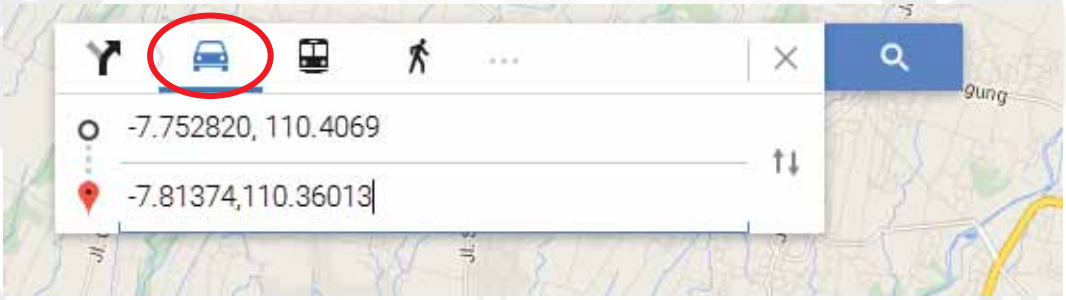
Gambar 2.2. Proses Memasukkan Data Lokasi Tujuan

Langkah selanjutnya adalah memasukkan titik koordinat lokasi awal. Lokasi awal ini berguna sebagai titik awal pengukuran jarak ke lokasi tujuan.



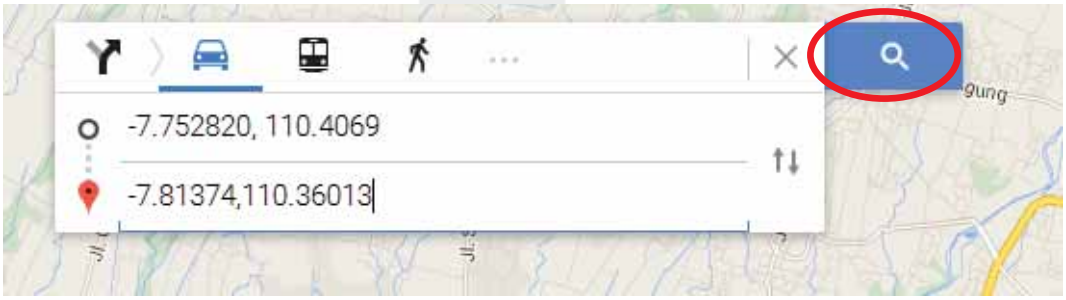
Gambar 2.3. Proses Memasukkan Data Lokasi Awal

Setelah memasukkan data lokasi awal, pilih menu pencarian rute yang dapat dilewati oleh kendaraan roda empat, yang pada gambar 2.3. diberi tanda lingkaran merah. Pencarian rute yang dapat dilewati kendaraan roda empat dilakukan agar seluruh kendaraan yang dimiliki UD. Garuda dapat menjangkau pelanggan yang ada.



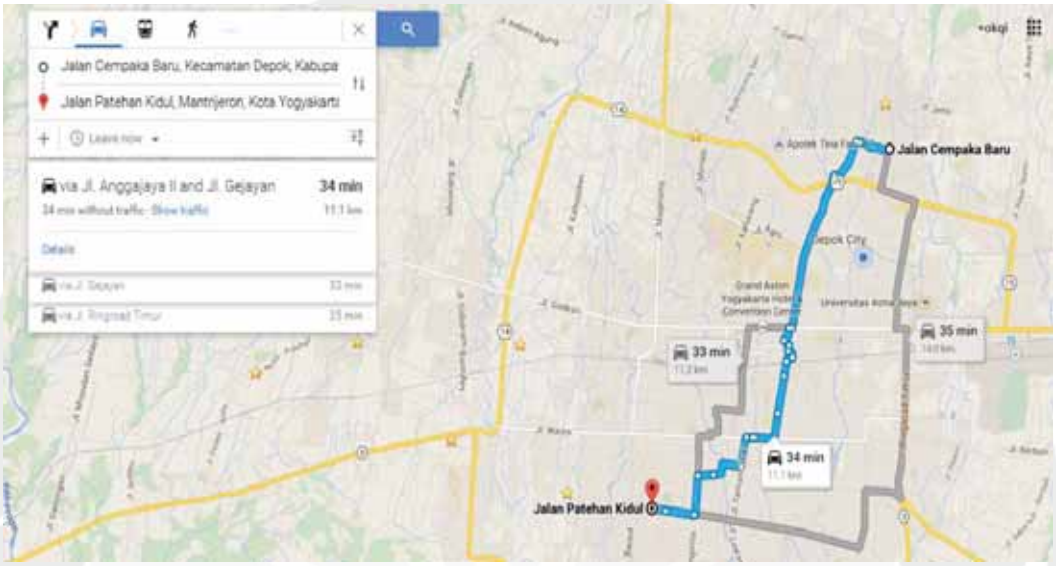
Gambar 2.4. Pemilihan menu untuk pencarian rute yang dapat dilewati kendaraan roda empat

Setelah memilih menu pencarian rute, langkah selanjutnya adalah klik tombol *search* yang pada contoh ini diberi tanda lingkaran merah untuk mengetahui rute alternatif yang dapat di tempuh dari lokasi awal dan lokasi tujuan.



Gambar 2.5. Tombol *search* pada aplikasi *Google Maps*

Tunggu beberapa saat untuk mendapatkan rute alternatif dari aplikasi *Google Maps*. Jika terdapat rute alternatif lebih dari satu, maka dipilih rute dengan jarak tempuh terpendek yang dapat dilalui oleh kendaraan angkut dengan kapasitas paling besar yaitu 350 lembar per sekali angkut sampai kendaraan angkut dengan kapasitas paling kecil yaitu 120 lembar per sekali angkut.



Gambar 2.6. Alternatif Rute Hasil Aplikasi *Google Maps*