

Implementasi Algoritma Cheapest Insertion Heuristic (CIH) dalam Penyelesaian Travelling Salesman Problem (TSP)

Rio Guntur Utomo¹, Dian Sa'adillah Maylawati², Cecep Nurul Alam³

¹University of Southampton, United Kingdom

²Sekolah Tinggi Teknologi Garut, Indonesia

³Asia E University, Malaysia

¹rgu1n15@soton.ac.uk, ²dsaadillah@sttgarut.ac.id, ³c70101170003@aeu.edu.my

Abstract- *Traveling salesman problem (TSP) is the problem of a salesman to visit the city of each city connected to each other and there is the weight of travel between the cities so as to form a complete weighted graph. Departing from a certain initial city, a salesman had to visit (n-1) another city exactly once and return on the initial city of departure. The purpose of TSP is to find the route of all cities with minimum total weight. Many algorithms have been found to solve the TSP, one of which is the Cheapest Insertion Heuristic (CIH) algorithm in the process of inserting weighted steps obtained from the equation $c(i, k, j) = d(i, k) + d(k, j) - d(i, j)$. This algorithm provides different travel routes depending on the order of insertion of cities on the subtour in question. In this final project, the writer took the problem of distribution route of mineral water of al-ma'soem 240 ml cup type, with vehicle capacity to meet 1200 carton and have different customer / agent demand that is the distance of depot and agent far from each other, distribution costs.*

Keywords- *Traveling Salesman Problem (TSP), Cheapest Insertion Heuristic (CIH), demand.*

Abstrak- *Travelling salesman problem (TSP) adalah permasalahan seorang salesman untuk mengunjungi kota n dari masing-masing kota terhubung satu sama lain dan terdapat bobot perjalanan antar kota tersebut sehingga membentuk suatu graf lengkap berbobot. Berangkat dari suatu kota awal tertentu, seorang salesman harus mengunjungi (n-1) kota lainnya tepat satu kali dan kembali pada kota awal keberangkatan. Tujuan TSP adalah mencari rute perjalanan semua kota dengan total bobot minimum. Banyak algoritma yang telah ditemukan untuk menyelesaikan TSP, salah satunya algoritma Cheapest Insertion Heuristic (CIH) dalam pengerjaannya melakukan langkah bobot penyisipan yang diperoleh dari persamaan $c(i,k,j) = d(i,k) + d(k,j) - d(i,j)$. Algoritma ini memberikan rute perjalanan yang berbeda tergantung dari urutan penyisipan kota-kota pada subtour yang bersangkutan. Pada pembahasan tugas akhir ini penulis mengambil permasalahan rute distribusi Air mineral al-ma'soem jenis cup 240 ml, dengan mempunyai kapasitas kendaraan memenuhi 1200 karton dan memiliki demand pelanggan/agen yang berbeda yaitu jarak depot dan agen yang saling berjauhan, hal ini menyebabkan tingginya biaya distribusi.*

Kata Kunci : *Traveling Salesman Problem (TSP), Cheapest Insertion Heuristic (CIH), demand.*

I. PENDAHULUAN

Perencanaan transportasi erat kaitannya dengan penentuan rute kendaraan yang efisien untuk menghemat ongkos operasional yang dikeluarkan oleh perusahaan. Rute kendaraan yang efektif dan efisien akan mempermudah dan memperlancar proses distribusi[1], [2].

Air mineral PT. Al-Ma'soem Muawanah adalah sebagai obyek penelitian untuk menentukan jalur distribusi dikarenakan jarak antara depot dengan agen yang saling berjauhan, Dan hal tersebut yang menyebabkan tingginya biaya distribusi pada saat melakukan pengiriman. Kesulitan yang dialami PT. Al-

Ma'soem Muawanah termasuk kedalam permasalahan VRP (*Vehicle Routing Problem*) VRP merupakan permasalahan optimasi penentuan rute dengan keterbatasan kapasitas kendaraan. Ada satu buah depot dan beberapa agen untuk dikunjungi dengan demand yang dapat berbeda-beda, dan dibatasi dengan kapasitas kendaraan sebesar 1200 karton sehingga kendaraan pengangkut diharapkan dapat memenuhi permintaan dari tiap-tiap agen[3][4][5].

Permasalahan ini bisa disebut dengan *Travelling Salesman Problem (TSP)* yakni pencarian rute optimum dalam suatu perjalanan yang mencari jarak terpendek untuk mencapai tempat tujuan agen yang akan dikunjungi semuanya dan setiap tempat agen hanya sekali

dikunjungi[6]. Penelitian ini digunakan Algoritma *Cheapest Insertion Heuristic* untuk menentukan rute kendaraan yang harus dilalui agar diperoleh jarak tempuh yang minimal atau terpendek[4][7]. Algoritma CIH adalah Algoritma *Insertion* yang pada setiap penambahan kota baru yang akan disisipkan ke dalam subtour mempunyai bobot penyisipan paling minimal. Bobot penyisipan diperoleh dari persamaan $c(i,k,j) = d(i,k) + d(k,j) - d(i,j)$. Algoritma ini memberikan rute perjalanan yang berbeda tergantung dari urutan penyisipan kota- kota pada subtour yang bersangkutan dan algoritma *cheapest insertion heuristic* ini baik digunakan untuk kasus TSP dengan jumlah kota yang besar. Tujuan yang ingin dicapai dalam penelitian di PT. Al-Ma'soem Muawanah adalah, dapat menerapkan Algoritma *Cheapest Insertion Heuristic* (CIH) dalam proses pencarian optimasi rute untuk pendistribusian air mineral PT. Al-Ma'soem Muawanah. Berikut ini merupakan pembatasan masalah yang digunakan dalam penelitian:

1. Data jarak, data ini didapatkan dari hasil pemetaan melalui koordinat (Latitude dan Longitude) pada google maps yang diambil dari alamat seluruh agen wilayah pengiriman Eksternal air mineral PT. Al-Ma'soem Muawanah.
2. Titik depot (titik nol) berada pada koordinat - 6.935242, 107.743846 Jalan Cikalang No. 168, Cimekar - Cileunyi, Bandung.

II. METODE PENELITIAN

A. Identifikasi Masalah

PT. Al-Ma'soem Muawanah sampai saat ini masih mengalami kesulitan dalam menentukan jalur distribusi air mineral keseluruhan agen, hal ini dikarenakan jarak antara depot dengan agen yang berjauhan, permasalahan tersebut yang menyebabkan tingginya biaya distribusi pada saat melakukan pengiriman air mineral ke seluruh agen. Sampai saat ini PT. Al-Ma'soem Muawanah belum memiliki metode khusus dalam menentukan jalur distribusi, salah satu cara untuk meminimasi biaya distribusi adalah dengan memperpendek jarak tempuh kendaraan ke tiap-tiap lokasi pengiriman, dengan jalur terpendek tersebut maka biaya distribusi air mineral dapat lebih murah. Untuk menyelesaikan kesulitan yang dihadapi PT. Al-Ma'soem Muawanah dalam menentukan jalur distribusi, diperlukan suatu metode yang dapat menyelesaikan masalah pendistribusian dengan cara menentukan rute distribusi terpendek, sehingga dapat menghasilkan rute distribusi yang lebih cepat, dan dapat mengurangi biaya distribusi [8].

B. Studi Literatur

Permasalahan yang dialami PT. Al-Ma'soem Muawanah dalam menentukan rute distribusi termasuk kedalam permasalahan VRP (Vehicle Routing Problem)[9]. Vehicle Routing Problem (VRP) adalah suatu permasalahan pengiriman/distribusi yang

melibatkan sekumpulan rute kendaraan-kendaraan yang berpusat pada satu depot atau lebih untuk melayani pelanggan yang tersebar diberbagai wilayah pengiriman dengan permintaannya masing-masing. Pada permasalahan ini, ada satu buah depot dan sejumlah 34 agen untuk dikunjungi dengan demand yang dapat berbeda-beda, dan dibatasi dengan kapasitas kendaraan sebesar 1200 Karton. Persoalan distribusi ini dapat diselesaikan dengan menggunakan metode heuristik yang didalamnya terdapat beberapa metode, metode yang dapat digunakan adalah algoritma *Cheapest Insertion Heuristic* (CIH)[10], Berikut adalah langkah-langkah yang harus dilakukan untuk penggunaan algoritma *Cheapest Insertion Heuristic* (CIH):

Langkah 1: Penulusuran dimulai dari sebuah lokasi pertama yang dihubungkan dengan sebuah lokasi terakhir. Langkah 2 : Bangun subtour antara 2 lokasi tersebut. Yang dimaksud adalah perjalanan dari lokasi pertama dan berakhir di lokasi pertama.

Langkah 3: Ganti salah satu arah hubunngan (arc) dari 2 lokasi dengan kombinasi 2 arc yaitu arc (I,j) dengan arc (I,k), dan arc (k,j), dengan k diambil dari lokasi yang belum masuk subtour dan dengan tambahan jarak terkecil. Langkah 4: Lakukan langkah 3 sampai seluruh lokasi masuk ke dalam subtour.

Sebagai contoh diberikan 5 kota dengan jarak antar kota seperti tertera dalam Tabel 1.

Tabel 1. Jarak Antar Kota [4]

Kota Asal	Kota Tujuan	Jarak
1	2	132
1	3	217
1	4	164
1	5	58
2	3	290
2	4	201
2	5	79
3	4	113
3	5	303
4	5	196

Untuk mencari jarak terpendek melalui ke 5 kota tersebut sebagaimana terdapat dalam Tabel 3.1, ambil langkah-langkah sebagai berikut:

1. Ambil perjalanan dari kota 1 ke 5,
2. Buat subtour $\rightarrow(1,5) \rightarrow (5,1)$,
3. Buat tabel yang menyimpan kota yang bisa disisipkan dalam subtour beserta tambahan jaraknya, seperti ditampilkan dalam Tabel 2.

Tabel 2. ArcPenambah Subtour ke 1 [4]

Arc Yang Akan Diganti	Arc Yang Ditambahkan Ke Subtour	Tambahkan Jarak
(1,5)	(1,2) - (2,5)	$C_{12} + C_{25} - C_{15} = 153$
(1,5)	(1,3) - (3,5)	$C_{13} + C_{35} - C_{15} = 462$
(1,5)	(1,4) - (4,5)	$C_{14} + C_{45} - C_{15} = 302$

Arc Yang Akan Diganti	Arc Yang Ditambahkan Ke Subtour	Tambahkan Jarak
(5,1)	(5,2) – (2,1)	$C_{52} + C_{21} - C_{51} = 153$
(5,1)	(5,3) – (3,1)	$C_{53} + C_{31} - C_{51} = 462$
(5,1)	(5,4) – (4,1)	$C_{54} + C_{41} - C_{51} = 302$

Dari Tabel 2 diperoleh tambahan jarak terkecil apabila:

Arc (1,5) diganti dengan arc (1,2) dan arc (2,5) atau arc (5,1) diganti dengan arc (5,2) dan arc (2,1) dari kemungkinan tersebut, bisa dipilih salah satu. Misal dipilih kemungkinan pertama maka subtour yang baru menjadi: $\rightarrow(1,2) \rightarrow(2,5) \rightarrow(5,1)$

- Selanjutnya dibuat tabel yang menyimpan kota yang bisa disisipkan dalam subtour beserta tambahan jaraknya, seperti ditampilkan dalam Tabel 3.

Tabel 3. Arc Penambah Subtour ke 2 [4]

Arc Yang Akan Diganti	Arc Yang Ditambahkan Ke Subtour	Tambahkan Jarak
(1,2)	(1,3) - (3,2)	$C_{13} + C_{32} - C_{12} = 375$
(1,2)	(1,4) – (4,2)	$C_{14} + C_{42} - C_{12} = 233$
(2,5)	(2,3) – (3,5)	$C_{23} + C_{35} - C_{25} = 514$
(2,5)	(2,4) – (4,5)	$C_{24} + C_{45} - C_{25} = 318$
(5,1)	(5,3) – (3,1)	$C_{53} + C_{31} - C_{51} = 462$
(5,1)	(5,4) – (4,1)	$C_{54} + C_{41} - C_{51} = 302$

Dari Tabel 3.3 diperoleh tambahan jarak terkecil adalah 233 dengan menggantikan arc(1,2) dengan arc(1,4) dan arc(4,2), sehingga subtour baru yang dihasilkan adalah: $\rightarrow(1,4) \rightarrow(4,2) \rightarrow(2,5) \rightarrow(5,1)$.

- Karena masih ada kota yang belum masuk, perlu dibuat tabel yang menyimpan kota yang bisa disisipkan dalam subtour beserta tambahan jaraknya, seperti ditampilkan dalam Tabel 4.

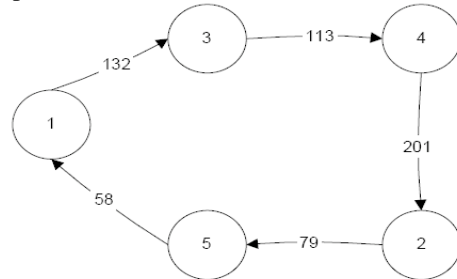
Tabel 4. Arc Penambah Subtour ke 3 [4]

Arc Yang Akan Diganti	Arc Yang Ditambahkan Ke Subtour	Tambahkan Jarak
(1,4)	(1,3) - (3,4)	$C_{13} + C_{34} - C_{14} = 166$
(4,2)	(4,3) – (3,2)	$C_{43} + C_{32} - C_{42} = 202$
(2,5)	(2,3) – (3,5)	$C_{23} + C_{35} - C_{25} = 514$
(5,1)	(5,3) – (3,1)	$C_{53} + C_{31} - C_{51} = 462$

Dari Tabel 4 diperoleh tambahan jarak terkecil adalah 166 dengan menggantikan arc (1,4) dengan arc

(1,3) dan arc (3,4), sehingga subtour baru yang dihasilkan adalah: $\rightarrow(1,3) \rightarrow(3,4) \rightarrow(4,2) \rightarrow(2,5) \rightarrow(5,1)$.

Dari langkah-langkah tersebut diatas dapat diperoleh lintasan terpendek untuk mengunjungi 5 kota adalah $\rightarrow(1,3) \rightarrow(3,4) \rightarrow(4,2) \rightarrow(2,5) \rightarrow(5,1)$ seperti terlihat pada Gambar 1.



Gambar 1. Lintasan Terpendek antar 5 Kota

Dengan lintasan tersebut diperoleh jarak tempuhnya adalah:

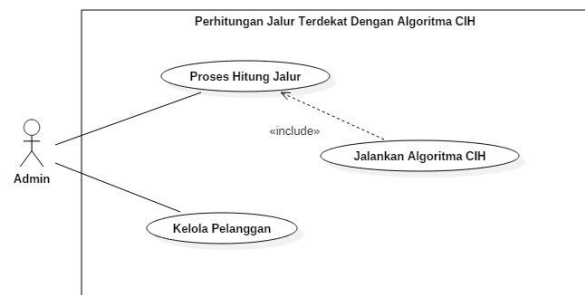
$$c_{13} + c_{34} + c_{42} + c_{25} + c_{51} = 132 + 113 + 201 + 79 + 58 = 668$$

Dengan menggunakan kasus 5 kota dengan jarak seperti tertera pada tabel 3.4 diperoleh lintasan $\rightarrow(1,3) \rightarrow(3,4) \rightarrow(4,2) \rightarrow(2,5) \rightarrow(5,1)$. Lintasan tersebut sama dengan yang tertera pada hasil hitungan manual pada Gambar 1. Jarak lintasan adalah 668.

C. Perancangan Sistem

1. Use Case Diagram

Pada bagian use case diagram ini menjelaskan apa yang akan dilakukan oleh aplikasi yang akan dibangun dan siapa saja yang akan berinteraksi dengan aplikasi. Use case diagram menjadi aktifitas pengguna. Bisa dilihat pada Gambar 2.

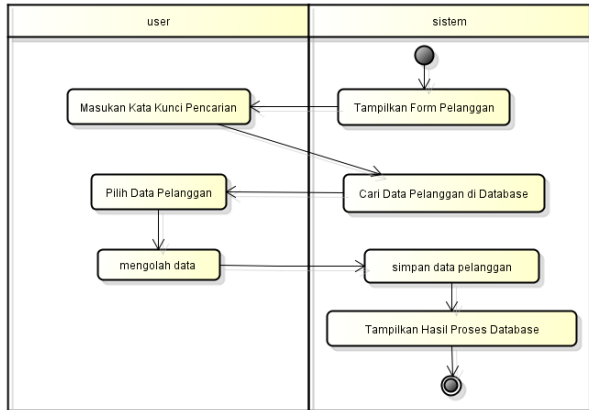


Gambar 2. Use Case Diagram

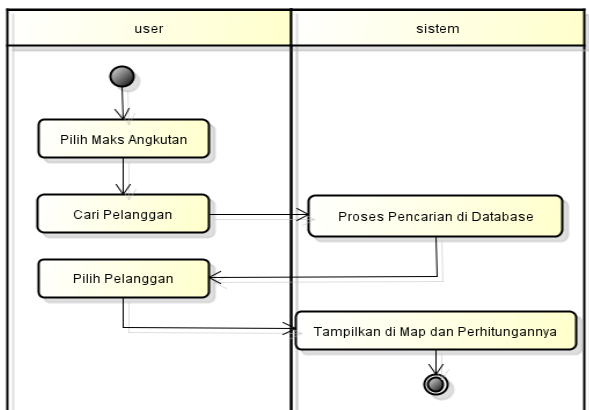
Pada Gambar 2 Use Case Diagram terdapat satu hak akses atau actor yaitu admin, actor admin ini menjadi aktifitas pengguna system melakukan menu-menu yang ada dalam aplikasi seperti menu proses hitung rute, dan kelola pelanggan/agen.

2. Activity Diagram

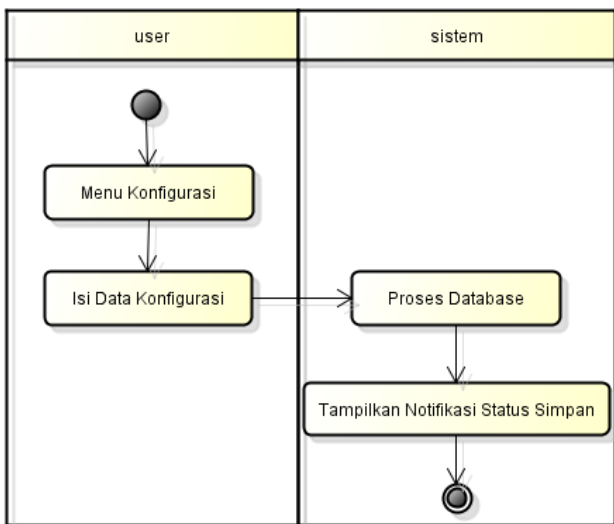
Activity diagram pada aplikasi pencarian rute terpendek distribusi Air mineral PT. Al- Ma'soem Muawanah dirancang seperti Gambar 3 dan 4 dibawah ini :



Gambar 3. Activity Diagram kelola Pelanggan



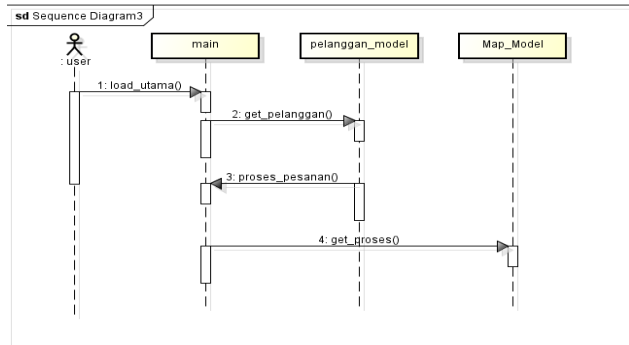
Gambar 4. Activity Diagram Proses Perjalanan/hitung rute



Gambar 5. Activity Diagram Konfigurasi

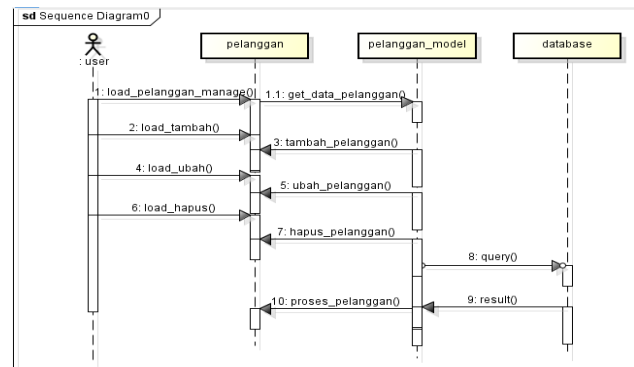
3. Sequence Diagram

Sequence diagram ini menjelaskan secara detail proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case pada Gambar 6, dan 7 beberapa sequence diagram antara lain :



Gambar 5. Sequence Diagram proses hitung jalur

Pada Gambar 5 Squence diagram proses hitung jalur terlihat bahwa user akan memilih menu utama, memilih pelanggan, kemudian system akan menampilkan form search untuk pilihan pelanggan, asal dan tujuan. Kemudian Pengguna dapat memilih proses hitung menggunakan algoritma CIH dan tampilan perhitungan atau hasil perhitungan maka system akan menampilkan rute terpendeknya di peta.

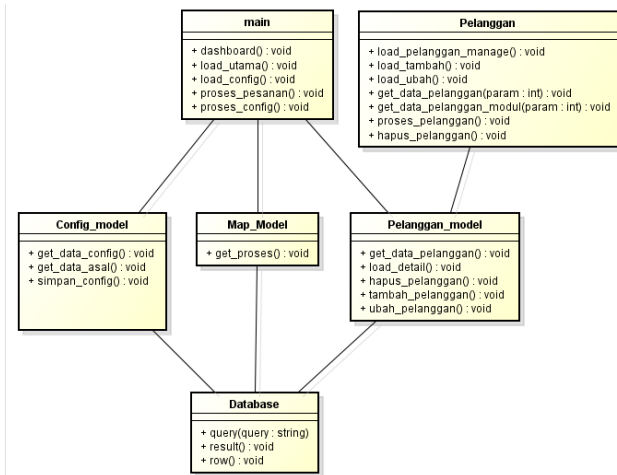


Gambar 6. Sequence Diagram kelola Pelanggan

Pada Gambar 6 Sequence diagram Kelola Pelanggan terlihat bahwa user memilih menu kelola/mengatur data pelanggan, proses dikelola dalam source code pelanggan_model yang berfungsi seperti menambahkan, menghapus, dan mengupdate data pelanggan baru ataupun lama, kemudian akan muncul notifikasi simpan dengan otomatis masuk ke dalam database.

4. Class Diagram

Class diagram memberikan gambaran pemodelan elemen-elemen class serta fungsi dan relasinya dengan class lain dalam sebuah sistem. Terdapat enam class pada aplikasi ini yaitu class main, config_model, map_model, pelanggan_model, Database, Pelanggan. Relasi antar class pada perancangan perangkat lunak ini ditunjukkan pada Gambar 7 berikut ini:



Gambar 7. Diagram class sistem

III. HASIL DAN PEMBAHASAN

A. Data Jarak Pelanggan/Agen.

Data jarak didapatkan melalui penentuan titik koordinat pada google maps menggunakan point to point, artinya dengan menggunakan point to point padagoogle maps maka jarak antara satu tempat dengan tempat lainnya akan diketahui jaraknya, dengan ketentuan sudah menentukan terlebih dahulu point-point mana yang ingin diketahui jaraknya.

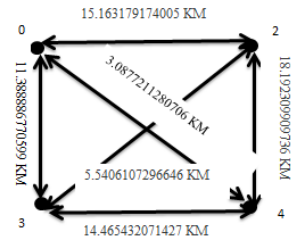
B. Data Kapasitas Kendaraan.

Data jarak didapatkan melalui penentuan titik koordinat pada google maps menggunakan point to point, artinya dengan menggunakan point to point padagoogle maps maka jarak antara satu tempat dengan tempat lainnya akan diketahui jaraknya, dengan ketentuan sudah menentukan terlebih dahulu point-point mana yang ingin diketahui jaraknya.

C. Pembentukan rute menggunakan *Cheapest Insertion Heuristic* (CIH).

Titik depot (titik nol) berada pada koordinat - 6.935242, 107.743846 Jalan Cikalang No. 168, Cimekar Cileunyi, Bandung. Hal-hal yang perlu dilakukan dalam pencarian/pembentukan rute awal yaitu beberapa node (titik) yang mewakili pelanggan/agen, Depot air mineral PT. Al- Ma’soem Muawanah. Data edge (jarak antar node) diperoleh dari kondisi riil. Secara lengkap, solusi awal diperoleh dengan langkah-langkah berikut :

Langkah 1 : Buat inialisasi graf yaitu sebagai contoh untuk wilayah bandung dari depot PT. Al-Masoem (0), BEC (2), Transmart Carefour Kiara Condong (3), Jatinangor Town Square (4). Kemudian memasukan angkutan maksimum sebesar 1200. Dapat dilihat pada gambar 8.



Gambar 8. Graf wilayah bandung

Langkah 2: system melakukan perjalanan ke-1 ke dalam bentuk table dengan kapasitas kendaraan 1200 karton, sebagai contoh dapat dilihat pada tabel 5.

Tabel 5. Jarak Antar kota

Tempat Asal	Tempat Tujuan	Angkutan maksimum	Jarak
PT. Al-Masoem	BEC (2)	1200 karton	15.163179174005 KM
PT. Al-Masoem	Transmart Carefour Kiara Condong (3)	1200 karton	11.388886770599 KM
PT. Al-Masoem	Jatinangor Town Square (4)	1200 karton	3.0877211280706 KM

Langkah 3: Lakukan pengambilan jarak minimum dari PT. Al-Ma’soem Muawanah/Depot (0) ke Jatinangor Town Square (4) dengan jarak 3.0877211280706 KM, dapat dilihat pada tabel . Kemudian buat Subtour = (0,4)→(4,0). Subtour dibuat bolak balik karena berangkat dan kembali harus ke titik yang sama. Subtour ditujukan untuk menandakan bahwa suatu perjalanan akan berangkat dan kembali ke titik yang sama.

1. Penyisipan

Adapun langkah-langkah yang dilakukan dalam penyisipan ini adalah sebagai berikut:

Langkah 1: Buat tabel yang menyimpan node (titik) yang bisa disisipkan dalam subtour beserta tambahan jaraknya. Node yang ditambahkan adalah node yang belum pernah dilewati, seperti terlihat pada Tabel 6.

Tabel 6. Arc Penambah Subtour ke 1

Arc yang akan diganti	Arc yang akan ditambahkan ke subtour	Tambahan Jarak	Pehitungan Jarak (km)
(0,4)	(0,2) - (2,4)	$C_{02} + C_{24} - C_{04}$	$15.163179174005 + 18.19230909736 - 3.0877211280706 = 30.267767955671$ KM
(0,4)	(0,3) - (3,4)	$C_{03} + C_{34} - C_{04}$	$11.388886770599 + 14.465432071427 - 3.0877211280706 = 22.766597713955$ KM
(4,0)	(4,2) - (2,0)	$C_{42} + C_{20} - C_{40}$	$18.19230909736 + 15.163179174005 - 3.0877211280706 = 30.267767955671$ KM
(4,0)	(4,3) - (3,0)	$C_{43} + C_{30} - C_{40}$	$14.465432071427 + 11.388886770599 - 3.0877211280706 = 22.766597713955$ KM

Langkah 2: Selanjutnya ganti rute minimum awal dengan rute minimum (sementara yaitu Arc (0,4) diganti dengan state (0,3) dan (3,4) atau Arc (4,0) diganti dengan state (4,3) dan (3,0). Dan apabila 2 rute yang memiliki total jarak yang sama, maka yang akan diganti adalah nilai minimum yang kedua.

Langkah 3: Lakukan penyisipan node yang belum terlewati seperti Tabel 7 .

Tabel 7. Arc Penambah Subtour ke 2

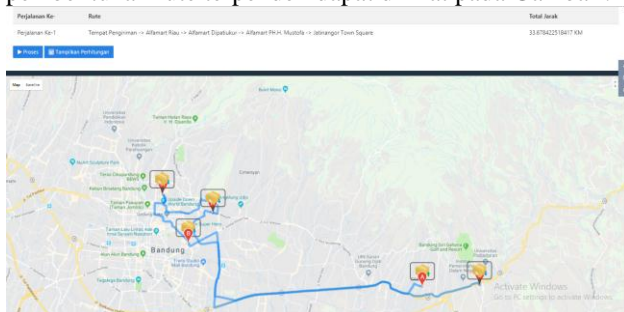
Arc yang akan diganti	Arc yang akan ditambahkan ke subtour	Tambah an Jarak	Pehitungan Jarak (km)
(0,3)	(0,2) - (2,3)	$C_{02} + C_{23} - C_{03}$	15.163179174005+5.5406107296646 -11.388886770599 = 9.3149031330715 KM
(3,4)	(3,2) - (2,4)	$C_{32} + C_{24} - C_{34}$	5.5406107296646+18.192309909736 -14.465432071427 = 9.2674885679739 KM
(4,3)	(4,2) - (2,3)	$C_{42} + C_{23} - C_{43}$	18.192309909736+5.5406107296646 -14.465432071427 = 9.2674885679739 KM
(3,0)	(3,2) - (2,0)	$C_{32} + C_{20} - C_{30}$	5.5406107296646+15.163179174005 -11.388886770599 = 9.2674885679739 KM

Langkah 4: Dari tabel 4, diperoleh penambahan jarak terkecil yaitu, maka ganti rute minimum yang lama dengan rute minimum (sementara) baru lagi yaitu Arc (3,4) diganti dengan state(3,2) dan (2,4). Hasil akhir perhitungan jarak terpendek seperti pada Tabel 8.

Tabel 8. Rekapitulasi Rute dengan Algoritma *Cheapest Insertion Heuristic* (CIH)

Perjalanan ke-	Rute	Total Jarak (km)
1	Tempat Pengiriman -> Transmart Carefour Kiara Condong -> BEC -> Jatinangor Town Square	35.12180741 KM

Hasil dari perhitungan diatas dengan aplikasi pembentukan rute terpendek dapat dilihat pada Gambar .



Gambar 9. Tampilan Maps Rute Terpendek

IV. PENUTUP

A. Kesimpulan

1. Aplikasi pendistribusian air mineral Al-Ma'soem Muawanah dirancang dan dibangun dengan melalui beberapa tahapan dari permasalahan jumlah kapasitas kendaraan yang tersedia, dengan menginput jumlah kapasitas maksimum kendaraan dan memilih data pelanggan sebagai objek pencarian rute terpendek yang nantinya akan dibutuhkan untuk melakukan sebuah perhitungan dengan menggunakan algoritma CIH dan data jarak diambil dari koordinat titik awal menggunakan *latitude* dan *longitude*.
2. Pencarian rute terpendek menggunakan Algoritma *Cheapest Insertion Heuristic* (CIH) adalah dengan cara menyisipkan kota yang belum terlewati dengan tambahan jarak minimum secara bergantian dengan menggunakan proses pengulangan hingga semua pelanggan/agen yang masuk menjadi sebuah subtour atau rute dengan perhitungan jarak untuk mendapatkan total jarak yang terkecil.

B. Saran

Beberapa saran dan masukan berikut diharapkan dapat memberikan perbaikan dalam penelitian, selanjutnya yaitu dengan membandingkan algoritma apa yang cocok untuk penyelesaian travelling salesman problem.

V. REFERENSI

- [1] M. Irfan, "IMPLEMENTASI COMPUTER BASED INSTRUCTION MODEL INSTRUCTIONAL GAMES PADA PEMBELAJARAN INTERAKTIF," vol. VIII, no. 2, pp. 162–176, 2014.
- [2] D. Suryani, M. Irfan, W. Uriawan, and W. Budiawan, "Implementasi Algoritma Divide and Conquer Pada Aplikasi Belajar Ilmu Tajwid," *J. Online Inform.*, vol. 1, no. 1, pp. 13–19, 2016.
- [3] A. Zarman, M. Irfan, and W. Uriawan, "Implementasi Algoritma Ant Colony Optimization Pada Aplikasi Pencarian Lokasi Tempat Ibadah Terdekat Di Kota Bandung," *JOIN (Jurnal Online Inform.)*, vol. 1, no. 1, pp. 6–12, 2016.
- [4] R. Naufal, "Implementasi Firefly Algorithm - Tabu Search Untuk Penyelesaian Traveling Salesman Problem," *JOIN (Jurnal Online Inform.)*, vol. 2, no. 1, pp. 42–48, 2017.
- [5] X. Li, D. Zhou, Z. Yang, J. Huang, K. Zhang, and Q. Pan, "UAV route evaluation algorithm based on CSA-AHP and TOPSIS," in *2017 IEEE International Conference on Information and Automation (ICIA)*, 2017, pp. 915–914.
- [6] C. N. Alam, K. Manaf, A. R. Atmadja, and D. K. Aurum, "Implementation of Haversine Formula for Counting Event Visitor in The Radius Based on Android Application," in *Proceedings of 2016*

- 4th International Conference on Cyber and IT Service Management, CITSM 2016*, 2016.
- [7] Jumadi, "Penentuan Rute Terpendek Menuju Kampus Menggunakan Algoritma Dynamic Programming," *ISTEK*, vol. VIII, no. 1, pp. 214–225, 2014.
- [8] A. Wahana, D. S. Maylawati, M. Irfan, and H. Effendy, "Supply chain management using fp-growth algorithm for medicine distribution," in *Journal of Physics: Conference Series*, 2018, vol. 978, no. 1.
- [9] B. L. Golden, S. Raghavan, and E. A. Wasil, "The Vehicle Routing Problem: Latest Advances and New Challenges: latest advances and new challenges," *Oper. Res. - Comput. Sci. Interface*, no. 43, p. 584, 2008.
- [10] K. Saleh and B. Prihandono, "PENENTUAN RUTE TERPENDEK DENGAN MENGGUNAKAN ALGORITMA (STUDI KASUS : PT . Wicaksana Overseas International Tbk . Cabang Pontianak) a b d," vol. 4, no. 3, pp. 295–304, 2015.