

IMPLEMENTASI ALGORITMA *FLOYD WARSHALL* DAN *NEAREST NEIGHBOUR* DALAM PENGOPTIMALAN RUTE *CAPACITATED VEHICLE ROUTING PROBLEM WITH TIME WINDOWS* (CVRPTW)

ARTIKEL JURNAL SKRIPSI

Diajukan kepada Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan
guna Memperoleh Gelar Sarjana Sains



Oleh

Intrada Reviladi

NIM 11305144037

**PROGRAM STUDI MATEMATIKA
JURUSAN PENDIDIKAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI YOGYAKARTA**

2016

PERSETUJUAN

Jurnal yang berjudul:

IMPLEMENTASI ALGORITMA *FLOYD WARSHALL* DAN *NEAREST NEIGHBOUR* DALAM PENGOPTIMALAN RUTE *CAPACITATED VEHICLE ROUTING PROBLEM WITH TIME WINDOWS (CVRPTW)*

Yang disusun oleh,

Nama : Intrada Reviladi

NIM : 11305144037

Prodi : Matematika

Telah disetujui oleh dosen pembimbing dan direview oleh dosen penguji untuk memenuhi sebagian persyaratan guna memperoleh Gelar Sarjana Sains.

Yogyakarta, Juli 2016

Direview

Dosen Penguji



Nur Insani, M.Sc

NIP. 198104062005012005

Disetujui

Dosen Pembimbing



Bambang S.H.M., M.Kom

NIP. 196802101998021001

IMPLEMENTASI ALGORITMA *FLOYD WARSHALL* DAN *NEAREST NEIGHBOUR* DALAM PENGOPTIMALAN RUTE *CAPACITATED VEHICLE ROUTING PROBLEM WITH TIME WINDOWS* (CVRPTW)

IMPLEMENTATION OF *FLOYD WARSHALL* ALGORITHM AND *NEAREST NEIGHBOUR* ALGORITHM IN ROUTE OPTIMIZATION *CAPACITATED VEHICLE ROUTING PROBLEM WITH TIME WINDOWS* (CVRPTW)

Oleh: Intrada Reviladi ¹⁾, Bambang S.H.M., M.Kom ²⁾
 Program Studi Matematika, Jurusan Pendidikan Matematika
 Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Yogyakarta
intradarms@gmail.com ¹⁾, bambang@uny.ac.id ²⁾

Abstrak

Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) merupakan masalah penentuan rute tercepat kendaraan untuk memenuhi permintaan konsumen yang terdiri dari pelayanan antar dengan kendala kapasitas kendaraan, *time windows*, dan kecepatan pada tiap jalur berdasarkan waktu per jam. Dalam menyelesaikan masalah CVRPTW akan digunakan dua algoritma, yakni algoritma *Floyd Warshall* dan *Nearest Neighbour*. Pada penelitian ini, dijelaskan mengenai penggunaan algoritma *Floyd Warshall* dan *Nearest Neighbour* dalam penyelesaian masalah CVRPTW yang diimplementasikan pada data simulasi secara manual dan menggunakan perangkat lunak MatLab. Selanjutnya akan dibandingkan efektifitas kedua algoritma tersebut yang diukur berdasarkan waktu penyelesaian dan hasil pembentukan rute. Berdasarkan hasil penelitian, diperoleh bahwa algoritma *Floyd Warshall* dapat membentuk rute dengan total waktu tempuh 939 menit, yang lebih efektif dibandingkan dengan algoritma *Nearest Neighbor* dengan total waktu tempuh 1.006 menit. Namun dalam proses penerapannya, algoritma *Nearest Neighbour* jauh lebih cepat dan praktis dibandingkan dengan algoritma *Floyd Warshall*.

Kata kunci: *Capacitated Vehicle Routing Problem With Time Windows* (CVRPTW), *Floyd Warshall*, *Nearest Neighbour*

Abstract

Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) is a method to find the fastest route for vehicle in order to fulfil the demands from consumers which consists of delivery service with vehicle capacity problem, time windows problem, and velocity at every track based on time (per hour). To solve the CVRPTW problem is solved using 2 algorithm, *Floyd Warshall* and *Nearest Neighbour*. This research explains the use of *Floyd Warshall* algorithm and *Nearest Neighbour* algorithm concerning CVRPTW problem solution which is implemented at the data simulation manually and using MatLab software. Then, the effectiveness of both algorithms is compared based on computational time process and the final routes. Based on the research, *Floyd Warshall* algorithm can form a route with more effective of total amount of traveling time 939 minutes compared to the nearest *Nearest Neighbour* algorithm with total amount of traveling time 1.006 minutes. However, in the implementation process, *Nearest Neighbour* algorithm is faster and more practical compared to *Floyd Warshall* algorithm.

Keywords: *capacitated vehicle routing problem with time windows* (CVRPTW), *floyd warshall*, *nearest neighbour*

PENDAHULUAN

Distribusi adalah kegiatan yang selalu menjadi bagian dalam menjalankan sebuah usaha. Distribusi merupakan suatu proses pengiriman barang dari suatu depot ke konsumen. Dalam distribusi, salah satu hal yang harus diperhatikan adalah kepuasan konsumen. Salah satu faktor kepuasan konsumen adalah barang sampai ke konsumen dengan tepat waktu.

Beberapa kendala yang harus dihadapi dalam proses distribusi, seperti:

1. jumlah permintaan barang yang berbeda-beda pada setiap konsumen,
2. kapasitas kendaraan,
3. batas waktu pengiriman,
4. kecepatan rata-rata yang dapat ditempuh pada jalur dan waktu tertentu,
5. dan lokasi konsumen yang berbeda pula.

Oleh karena itu diperlukan suatu cara agar proses distribusi dapat berjalan dengan lancar dan tepat waktu. Salah satu cara yang dapat dilakukan dalam proses distribusi adalah dengan mengoptimalkan rute kendaraan agar waktu yang digunakan untuk melayani konsumen lebih efisien dan barang dapat sampai ke konsumen tepat waktu.

Permasalahan optimisasi rute kendaraan dikenal dengan *vehicle routing problem* (VRP). VRP yang bertujuan membentuk rute optimal dalam melayani konsumen dengan kendala kapasitas dan *time windows* disebut *capacitated vehicle routing problem with time windows* (CVRPTW). Beberapa metode yang dapat digunakan untuk menyelesaikan kasus ini adalah metode eksak dan metode heuristik. Metode eksak merupakan algoritma yang menghitung setiap solusi sampai didapat solusi yang optimal namun waktu penyelesaiannya relatif lama, sedangkan metode heuristik merupakan algoritma yang menggunakan performa komputasi sederhana dalam penyelesaian masalah sehingga dapat memberikan solusi yang mendekati optimal yang relatif cepat.

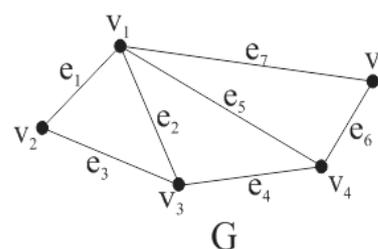
Pada makalah ini, penyelesaian permasalahan CVRPTW, menggunakan dua metode yaitu metode eksak dan metode

heuristik, salah satu metode eksak yang akan digunakan adalah algoritma *floyd warshall* dan metode heuristik yang digunakan adalah algoritma *nearest neighbor*, serta akan dibandingkan efektifitas algoritma *floyd warshall* dan *nearest neighbour* dalam menentukan rute tercepat.

Graf

Definisi graf menurut Edgar G. Goodaire dan Michael M. Parmanter (2002) adalah kumpulan simpul (vertex atau nodes) yang dihubungkan satu dengan lainnya melalui busur (edges). Secara matematis, suatu graf G didefinisikan sebagai pasangan himpunan $G(V,E)$ dengan $V(G)$ adalah himpunan tidak kosong dari simpul (vertex atau nodes), $V(G) = \{v_1, v_2, v_3, \dots, v_n\}$, dan $E(G)$ adalah himpunan busur (edges atau arcs) $E(G) = \{e_1, e_2, e_3, \dots, e_n\}$, yang menghubungkan sepasang simpul pada graf tersebut.

Sebuah graf direpresentasikan dalam bentuk gambar (Gambar 1). Simpul pada graf digambarkan dengan lingkaran kecil, sedangkan busur yang menghubungkan dua simpul digambarkan dengan kurva sederhana atau segmen garis dengan titik akhir di kedua simpul tersebut.



Gambar 1. Graf G

Keterhubungan

Pada bagian ini akan dijelaskan keterhubungan graf yang digunakan dalam makalah ini. Jika seluruh rusuk dan simpul pada sebuah trayek berbeda, maka trayek tersebut disebut lintasan. (Robin J. Wilson & John J. Watkin, 1990 : 35). Suatu graf G dikatakan terhubung jika untuk setiap pasang dua simpulnya terhubung. Simpul u dan v disebut terhubung jika terdapat lintasan dari u ke v . Jika terdapat lintasan yang titik awal dan titik

akhirnya berhimpit, maka trayek tersebut disebut *cycle* (sikel).

Representasi Graf dalam Matriks Ketetangaan

Keterhubungan antar simpul pada graf akan disajikan dalam sebuah matriks agar dapat lebih mudah pada penyelesaiannya. Matriks tersebut dinamakan matriks ketetangaan. Jong Jek Siang (2006 : 273) mendefinisikan matriks ketetangaan sebagai berikut. Misalkan G adalah graf tak berarah dengan simpul-simpul $v_1, v_2, v_3, \dots, v_n$ dengan n berhingga. Matriks ketetangaan pada graf G_4 dinyatakan dalam Gambar 2.

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Gambar 2. Matriks Ketetangaan

Pada matriks ketetangaan dapat dilihat simpul yang saling berhubungan maupun tidak. Jika $a_{ij} = 0$ maka simpul v_i tidak terhubung dengan simpul v_j , sedangkan jika $a_{ij} = 1$ maka simpul v_i terhubung dengan simpul v_j .

Vehicle Routing Problem

Menurut Rahmi dan Murti (2013): *Vehicle Routing Problem* (VRP) merupakan permasalahan dalam sistem distribusi yang bertujuan untuk membuat suatu rute yang optimal, dengan sekelompok kendaraan yang sudah diketahui kapasitasnya, agar dapat memenuhi permintaan konsumen dengan lokasi dan jumlah permintaan yang telah diketahui.

Suatu rute dikatakan optimal jika rute dapat memenuhi kendala atau batasan yang ada. Berikut ini adalah beberapa kendala atau batasan yang harus dipenuhi dalam VRP yaitu:

1. Rute kendaraan dimulai dari depot dan berakhir di depot,
2. Masing-masing konsumen harus dikunjungi sekali dengan satu kendaraan,
3. Kendaraan yang digunakan adalah homogen dengan kapasitas tertentu, sehingga permintaan konsumen pada setiap rute yang dilalui tidak boleh melebihi kapasitas kendaraan.

4. Jika kapasitas kendaraan sudah mencapai batas, maka konsumen berikutnya akan dilayani oleh shift berikutnya.

Tujuan umum VRP menurut Toth dan Vigo (2002) adalah

1. Meminimalkan jarak dan biaya tetap yang berhubungan dengan penggunaan kendaraan,
2. Meminimalkan banyaknya kendaraan yang dibutuhkan untuk melayani permintaan seluruh konsumen,
3. Menyeimbangkan rute-rute dalam hal waktu perjalanan dan muatan kendaraan, dan
4. Meminimalkan pinalti sebagai akibat dari pelayanan yang kurang memuaskan terhadap konsumen, seperti keterlambatan pengiriman dan lain sebagainya.

Capacitated Vehicle Routing Problem with Time Windows (CVRPTW)

Capacitated vehicle routing problem with time windows (CVRPTW) adalah salah satu jenis VRP yang merupakan kombinasi dari bentuk umum *capacitated vehicle routing problem* (CVRP) dan *vehicle routing problem with time windows* (VRPTW). CVRPTW bertujuan untuk membentuk rute optimal untuk memenuhi permintaan konsumen yang dilakukan secara *delivery* dengan kendala kapasitas dan *time windows*.

Kendala pertama pada CVRPTW adalah kendala kapasitas. Kendala kapasitas yang dimaksud adalah bahwa setiap kendaraan memiliki kapasitas tertentu dan jika kapasitas kendaraan sudah penuh, maka kendaraan tersebut tidak dapat melayani konsumen selanjutnya. Kendala berikutnya adalah kendala *time windows* pada masing-masing konsumen dan *time windows* pada depot. *Time windows* pada masing-masing konsumen $[a_i, b_i]$ adalah interval waktu yang ditentukan oleh masing-masing konsumen bagi setiap kendaraan untuk dapat melakukan pelayanan pada konsumen tersebut. Kendaraan dapat memulai pelayanan di antara waktu awal konsumen (a_i) dan waktu akhir konsumen (b_i). Namun kendaraan juga harus menunggu sampai waktu awal konsumen dapat dilayani apabila kendaraan tersebut datang

sebelum waktu awal konsumen, sedangkan *time windows* pada depot $[a_0, b_0]$ didefinisikan sebagai interval waktu yang menunjukkan waktu awal keberangkatan kendaraan dari depot dan waktu kembalinya kendaraan ke depot, itu artinya kendaraan tidak boleh meninggalkan depot sebelum waktu awal depot (a_0) dan harus kembali ke depot sebelum waktu akhir depot (b_0).

Model CVRPTW

Masalah CVRPTW dapat direpresentasikan sebagai suatu graf berarah $G = (V, A)$ dengan $V = \{v_0, v_1, v_2, \dots, v_n\}$ adalah himpunan simpul (verteks), v_0 menyatakan depot dengan v_{n+1} merupakan depot semu dari v_0 yaitu tempat kendaraan memulai dan mengakhiri rute perjalanan. Sedangkan $A = \{(v_i, v_j) \mid v_i, v_j \in V, i \neq j\}$ adalah himpunan rusuk atau garis berarah (arc) yang menghubungkan dua simpul yaitu ruas jalan penghubung antar konsumen ataupun antara depot dengan konsumen.

Setiap simpul $\{v_i \in V, i \neq 0\}$ memiliki permintaan (*demand*) sebesar q_i dengan q_i adalah *integer positif*. Himpunan $K = \{k_1, k_2, \dots, k_n\}$ merupakan kumpulan kendaraan yang homogen dengan kapasitas maksimal yang identik yaitu Q , sehingga panjang setiap rute dibatasi oleh kapasitas kendaraan. Setiap verteks (v_i, v_j) memiliki waktu tempuh Wt_{ij} yaitu waktu tempuh dari simpul i ke j . Waktu tempuh perjalanan ini diasumsikan asimetrik yaitu $Wt_{ij} \neq Wt_{ji}$ dan $Wt_{ii} = 0$.

Dari permasalahan CVRPTW tersebut kemudian diformulasikan ke dalam bentuk model matematika dengan tujuan meminimumkan total waktu tempuh kendaraan untuk melayani semua konsumen, jika z adalah fungsi tujuan maka

$$\min z = \sum_{i=1}^n \sum_{j=1}^n (Wt_{ij} \sum_{k=1}^m X_{ijk}) \quad \dots (2)$$

dengan variabel keputusan sebagai berikut.

1. Variabel $X_{ijk}, \forall i, j \in N, \forall k \in K, i \neq j$.

Variabel X_{ijk} mempresentasikan ada atau tidaknya perjalanan dari konsumen ke- i ke konsumen ke- j oleh kendaraan ke- k .

$$X_{ijk} = \begin{cases} 1, & \text{jika ada perjalanan dari konsumen } i \text{ ke konsumen } j \text{ oleh kendaraan } k. \\ 0, & \text{jika tidak ada perjalanan dari konsumen } i \text{ ke konsumen } j \text{ oleh kendaraan } k. \end{cases}$$

2. Variabel T_{ik}, T_{0k} , dan $s_{ik} \forall i \in N, \forall k \in K$. Variabel T_{ik} menyatakan waktu dimulainya pelayanan pada konsumen ke- i oleh kendaraan ke- k , T_{0k} menyatakan waktu saat kendaraan ke- k meninggalkan depot dan kembali ke depot, dan s_{ik} menyatakan lamanya pelayanan di konsumen ke- i .
3. Variabel Y_{ik} dan $q_j, \forall i, j \in N, \forall k \in K$. Variabel Y_{ik} menyatakan kapasitas total dalam kendaraan ke- k setelah melayani konsumen ke- i , sedangkan q_j menyatakan banyaknya permintaan konsumen ke- j .

Adapun kendala dari permasalahan CVRPTW adalah sebagai berikut.

1. Setiap konsumen hanya dikunjungi tepat satu kali oleh kendaraan yang sama.

$$\sum_{k=1}^m \sum_{j=1}^n \sum_{i=1}^n X_{ijk} = 1, \forall i, j \in N, \forall k \in K \quad \dots (3)$$

2. Total jumlah permintaan konsumen dalam satu rute tidak melebihi kapasitas kendaraan yang melayani rute tersebut.

Jika ada lintasan dari i ke j dengan kendaraan k , maka

$$Y_{ik} + q_j = Y_{jk}, \forall i, k \in N, \forall k \in K \quad \dots (4)$$

$$Y_{jk} \leq Q, \forall j \in N, \forall k \in K \quad \dots (5)$$

3. Jika ada perjalanan dari konsumen ke- i ke konsumen ke- j , maka waktu memulai pelayanan di konsumen ke- j lebih dari atau sama dengan waktu kendaraan ke- k untuk memulai pelayanan di konsumen ke- i ditambah waktu pelayanan konsumen ke- i dan ditambah waktu tempuh perjalanan dari konsumen ke- i ke konsumen ke- j .

Jika ada lintasan dari i ke j dengan kendaraan k , maka

$$T_{ik} + s_{ik} + Wt_{ij} \leq T_{jk}, \forall i, j \in N, \forall k \in K \quad \dots (6)$$

4. Waktu kendaraan untuk memulai pelayanan di konsumen ke- i harus berada pada selang waktu $[a_i, b_i]$.

$$a_i \leq T_{ik} \leq b_i, \forall i \in N, \forall k \in K \quad \dots (7)$$

5. Setiap rute perjalanan berawal dari depot dan berakhir di depot.

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^m X_{ijk} = 1, \forall i, j \in N, \forall k \in K \quad \dots (8)$$

6. Kekontinuan rute, artinya kendaraan yang mengunjungi setiap konsumen, setelah selesai melayani akan meninggalkan konsumen tersebut.

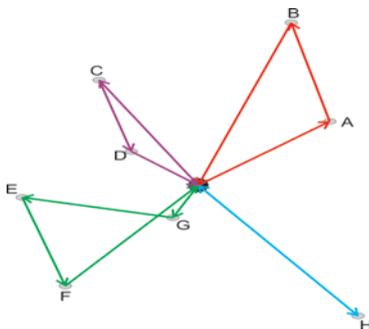
$$\sum_{i=1}^n X_{ijk} - \sum_{i=1}^n X_{ijk} = 0, \forall i, j \in N, \forall k \in K \quad \dots (9)$$

7. Variabel keputusan X_{ijk} merupakan integer biner

$$X_{ijk} \in \{0,1\}, \forall i, j \in N, \forall k \in K \quad \dots (10)$$

Representasi Solusi

Solusi layak dari formulasi CVRPTW yang dihasilkan adalah himpunan rute kendaraan yang memiliki total waktu tempuh minimal dengan memenuhi semua kendala yang ada. Himpunan rute tersebut dapat dituliskan sebagai berikut, $rute = \{rute1, rute2, \dots, rute n\}$. Solusi CVRPTW dapat digambarkan dalam bentuk graf yang setiap rute perjalanannya merupakan lintasan tertutup dengan depot sebagai simpul awal dan simpul akhir, sedangkan simpul lainnya adalah konsumen. Ilustrasi mengenai solusi layak dari CVRPTW seperti Gambar 3.



Gambar 3. Ilustrasi Solusi Layak CVRPTW

Pada Gambar 3 terdapat 8 konsumen yaitu A, B, C, D, E, F, G, H. Dengan penggunaan algoritma eksak ataupun heuristik didapat sebuah solusi yang terdiri dari empat rute perjalanan kendaraan yang diawali dan diakhiri di depot serta memenuhi kendala yang ada pada

CVRPTW. Rute pertama terdiri dari konsumen A dan B, rute kedua terdiri dari konsumen C dan D, rute ketiga terdiri dari konsumen E, F dan G, dan rute yang terakhir hanya terdiri dari konsumen H saja.

Dengan demikian, representasi solusi dari masalah CVRPTW tersebut adalah $rute = \{0 - A - B - 0, 0 - C - D - 0, 0 - E - F - G - 0, 0 - H - 0\}$ dengan total waktu tempuh kendaraan = $Wt_{0A,t} + Wt_{AB,t} + Wt_{B0,t} + Wt_{0C,t} + Wt_{CD,t} + Wt_{D0,t} + Wt_{0E,t} + Wt_{EF,t} + Wt_{FG,t} + Wt_{G0,t} + Wt_{0H,t} + Wt_{H0,t}$.

HASIL DAN PEMBAHASAN

Secara garis besar pada bab ini akan dijelaskan mengenai ilustrasi permasalahan CVRPTW.

1. Jenis dan Sumber Data

Jenis data yang digunakan dalam permasalahan ini adalah data simulasi yang dibuat berdasarkan karakteristik dari sebuah data nyata. Data yang akan dibuat dalam data simulasi ini antara lain letak depot dan konsumen, permintaan konsumen, dan *time windows*. Dalam hal ini akan ditambah dengan data matriks ketetanggaan dan alokasi kecepatan rata-rata pada tiap jalur berdasarkan waktu per jam. Penambahan kendala tersebut bertujuan untuk mengembangkan pemodelan matematika agar lebih kompleks dan dapat lebih real untuk diaplikasikan ke dalam permasalahan yang sebenarnya.

2. Teknik Analisis Data

Pengambilan sampel tidak dilakukan dilapangan namun membuat data simulasi antara pukul 07.00 - 12.00 yang dibagi menjadi lima bagian waktu yaitu masing-masing satu jam. Setelah data selesai dibuat, proses selanjutnya yaitu analisis data. Analisis data dalam penelitian ini dapat dibagi menjadi tiga tahap yaitu:

- Tahap pertama yaitu input data simulasi ke dalam program sesuai dengan perintah program.
- Tahap kedua yaitu data yang telah diinput akan diolah oleh program.

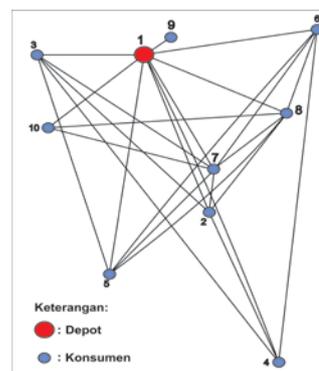
Tahap ketiga yaitu dihasilkan output dari hasil pengolahan data. Setelah data selesai diolah, hasil output diinterpretasikan secara kualitatif dengan tujuan menarik kesimpulan pada permasalahan ini.

3. Ilustrasi Permasalahan

Sebuah usaha yang bergerak di bidang jasa pengiriman barang menerapkan sistem layanan antar barang (*delivery service*) terhadap konsumennya. Distributor dari perusahaan akan mengantar barang ke konsumen. Wilayah operasi perusahaan adalah dalam lingkup satu propinsi.

Untuk satu hari kerja, perusahaan memiliki *transportation request* yang hendak dilayani. Setiap *transportation request* terdiri dari informasi mengenai konsumen, lokasi pengiriman barang, permintaan konsumen, dan *time windows* konsumen. Terkadang dalam melakukan distribusi barang, perusahaan menentukan rute berdasarkan jarak tempuh saja, namun tidak memperkirakan tingkat kemacetan. Karena banyaknya pelanggan yang harus dilayani dengan batasan waktu dan jumlah permintaan/muatan yang berbeda-beda, ada kalanya pengiriman barang tidak tepat waktu dan ada kalanya kapasitas kendaraan tidak mencukupi.

Berdasarkan uraian di atas, penentuan rute yang saat ini digunakan oleh perusahaan masih kurang efektif karena dengan adanya beberapa kendala tersebut, mengakibatkan kurang maksimalnya pihak perusahaan dalam melakukan proses pengiriman barang ke konsumen. Oleh karena itu, diperlukan suatu metode dalam penentuan rute yang efektif sehingga menghasilkan total waktu tempuh perjalanan yang optimal dengan mempertimbangkan kendala yang ada.



Gambar 8. Ilustrasi Permasalahan CVRPTW

4. Pengumpulan Data

Dalam menyelesaikan permasalahan CVRPTW, dibutuhkan beberapa data yang digunakan untuk mendapatkan solusi rute distribusi yang optimal. Adapun data-data tersebut mengenai lokasi konsumen, jumlah permintaan konsumen, kapasitas kendaraan, *time windows*, *service time*, matriks jarak antar konsumen, alokasi rata-rata kecepatan pada waktu dan jalur tertentu.

Tabel 1. Data Lokasi, Demand, Time Windows, dan Service Time

No	Lokasi		Demand	Time Windows		s_i
	Koor-x	Koor-y		a_i	b_i	
1	30	90	0	0	315	10
2	47	47	70	0	240	10
3	2	90	60	0	300	10
4	65	6	45	45	240	10
5	21	30	55	0	300	10
6	75	97	70	0	300	10
7	48	59	70	0	270	10
8	67	74	70	60	240	10
9	37	95	65	45	270	10
10	5	70	60	30	240	10

Keterangan :

a_i = batas awal time windows pada i

b_i = batas akhir time windows pada i

s_i = waktu pelayanan pada i

Tabel 2. Data Keterhubungan antar Lokasi

Dari Ke	1	2	3	4	5	6	7	8	9	10
1	0	1	1	1	1	1	1	1	1	1
2	1	0	1	1	0	0	1	1	0	0
3	1	1	0	1	1	0	1	0	0	0
4	1	1	1	0	0	1	0	0	0	0
5	1	0	1	0	0	1	1	1	0	0
6	1	0	0	1	1	0	1	1	0	0
7	1	1	1	0	1	1	0	1	0	1
8	1	1	0	0	1	1	1	0	0	1
9	1	0	0	0	0	0	0	0	0	0
10	1	0	0	0	0	0	1	1	0	0

Tabel 3. Data Matriks Jarak

	1	2	3	4	5	6	7	8	9	10
1	0	46.2385	28	91	60.6712	45.5412	35.8469	40.3113	8.6023	32.0156
2	46.2385	0	62.2415	44.7772	inf	inf	12.0416	33.6006	inf	inf
3	28	62.2415	0	105	62.9365	inf	55.4707	inf	inf	inf
4	91	44.7772	105	0	inf	91.5478	inf	inf	inf	inf
5	60.6712	inf	62.9365	inf	0	86.0523	39.6232	63.6553	inf	inf
6	45.5412	inf	inf	91.5478	86.0523	0	46.6154	24.3516	inf	inf
7	35.8469	12.0416	55.4707	inf	39.6232	46.6154	0	24.2074	inf	44.3847
8	40.3113	33.6006	inf	inf	63.6553	24.3516	24.2074	0	inf	62.1289
9	8.6023	inf	0	inf						
10	32.0156	inf	inf	inf	inf	inf	44.3847	62.1289	inf	0

Data alokasi kecepatan berdasarkan waktu adalah kecepatan rata-rata maksimal yang dapat ditempuh kendaraan pada pukul 07.00 – 12.00 dan di jalur tertentu. Untuk alokasi waktunya diambil kecepatan rata-rata maksimal setiap satu jam selama 5 jam.

Tabel 4. Data Kecepatan Rata-rata pada pukul 07.00-08.00

	1	2	3	4	5	6	7	8	9	10
1	0	35	35	45	40	50	30	40	40	35
2	38	0	38	35	0	0	30	36	0	0
3	34	36	0	50	40	0	39	0	0	0
4	30	30	40	0	0	37	0	0	0	0
5	50	0	42	0	0	45	35	34	0	0
6	50	0	0	34	36	0	37	37	0	0
7	30	40	35	0	30	31	0	30	0	33
8	45	35	0	0	30	35	30	0	0	36
9	50	0	0	0	0	0	0	0	0	0
10	40	0	0	0	0	0	40	38	0	0

Tabel 5. Data Kecepatan Rata-rata pada pukul 08.00-09.00

	1	2	3	4	5	6	7	8	9	10
1	0	50	40	45	45	50	35	45	45	45
2	42	0	43	40	0	0	40	39	0	0
3	38	45	0	55	48	0	45	0	0	0
4	30	35	45	0	0	39	0	0	0	0
5	40	0	43	0	0	55	38	40	0	0
6	45	0	0	38	36	0	40	45	0	0
7	35	40	40	0	35	37	0	35	0	40
8	47	45	0	0	40	35	35	0	0	40
9	53	0	0	0	0	0	0	0	0	0
10	45	0	0	0	0	0	45	40	0	0

Tabel 6. Data Kecepatan Rata-rata pada pukul 09.00-10.00

	1	2	3	4	5	6	7	8	9	10
1	0	60	55	55	45	60	40	50	60	40
2	50	0	45	48	0	0	47	47	0	0
3	45	50	0	55	50	0	50	0	0	0
4	40	42	45	0	0	50	0	0	0	0
5	45	0	50	0	0	52	43	40	0	0
6	50	0	0	40	42	0	45	45	0	0
7	45	40	40	0	40	43	0	39	0	48
8	50	50	0	0	50	50	45	0	0	45
9	50	0	0	0	0	0	0	0	0	0
10	55	0	0	0	0	0	50	45	0	0

Tabel 7. Data Kecepatan Rata-rata pada pukul 10.00-11.00

	1	2	3	4	5	6	7	8	9	10
1	0	65	70	60	55	65	55	60	70	55
2	80	0	60	65	0	0	55	60	0	0
3	65	70	0	65	70	0	70	0	0	0
4	65	60	55	0	0	60	0	0	0	0
5	50	0	60	0	0	58	59	60	0	0
6	55	0	0	60	62	0	70	68	0	0
7	50	60	60	0	65	58	0	60	0	55
8	60	65	0	0	70	65	59	0	0	60
9	80	0	0	0	0	0	0	0	0	0
10	60	0	0	0	0	0	75	77	0	0

Tabel 8. Data Kecepatan Rata-rata pada pukul 11.00-12.00

	1	2	3	4	5	6	7	8	9	10
1	0	60	60	55	40	55	45	50	55	50
2	65	0	50	60	0	0	45	50	0	0
3	58	55	0	55	55	0	57	0	0	0
4	63	60	50	0	0	62	0	0	0	0
5	50	0	60	0	0	55	54	50	0	0
6	50	0	0	50	56	0	60	62	0	0
7	40	60	56	0	65	55	0	55	0	53
8	50	60	0	0	65	62	55	0	0	58
9	75	0	0	0	0	0	0	0	0	0
10	50	0	0	0	0	0	70	65	0	0

5. Pengolahan Data

Penyelesaian permasalahan CVRPTW untuk mendapatkan solusi rute distribusi pada data simulasi dilakukan dengan mengolah data yang telah diperoleh dengan menggunakan Algoritma *Floyd Warshall* dan *Nearest Neighbour*.

Algoritma *Floyd Warshall*

Dalam hal ini akan dijelaskan penggunaan algoritma *Floyd Warshall* dalam membentuk rute kendaraan pada penyelesaian *capacitated vehicle routing problem with time windows* (CVRPTW). Berikut langkah-langkahnya.

a. Langkah 1

$$W_t = W_t^{(0)}$$

b. Langkah 2

Untuk $h = 1$ hingga n Untuk $i = 1$ hingga n Untuk $j = 1$ hingga n Jika $Wt_{ij,t} > Wt_{ih,t} + Wt_{hj,s}$, $s = t + Wt_{ih,t}$ maka tukar $Wt_{ij,t}$ dengan $Wt_{ih,t} + Wt_{hj,s}$, $s = t + Wt_{ih,t}$

c. Langkah 3

 $Wt^* = Wt$

d. Langkah 4

Ulangi langkah 2 sampai didapatkan hasil yang minimum.

e. Langkah 5

Jika rute masih memungkinkan untuk ditambah muatan, maka sisipkan konsumen lain dalam rute tersebut dan ulangi langkah 2.

Berikut langkah-langkah penyelesaian masalah CVRPTW dengan algoritma *floyd warshall*. Berikut langkah-langkahnya:

1. Pembentukan rute pertama

a. Pada pembentukan rute pertama, akandicari waktu tempuh terkecil dari depot ke masing-masing konsumen.

1) $1 - 2$ dengan waktu tempuh total 90 menit dan total *demand* 70 buah. Karena total *demand* 70 buah < kapasitas kendaraan, maka akan disisipkan konsumen lain.

2) $1 - x - 2$, dengan $\forall x \in N - \{1,2\}$.

Pilih $Wt_{1-2,0} = Wt_{1-3,0} + Wt_{3-2,58} = 154$.

Jika $Wt_{1-2,0} > Wt_{1-4,0} + Wt_{4-2,132} = 206$, maka $Wt_{1-2,0} = 154$.

Jika $Wt_{1-2,0} > Wt_{1-7,0} + Wt_{7-2,82} = 111$, maka tukar $Wt_{1-2,0} = 111$.

Jika $Wt_{1-2,0} > Wt_{1-8,0} + Wt_{8-2,71} = 126$, maka $Wt_{1-2,0} = 111$.

Untuk hasil perhitungannya dapat dilihat di Tabel 9.a.

Tabel 9.a. Total Waktu Tempuh Pembentukan Rute Pertama dengan *Floyd Warshall* pada iterasi 1

Rute	Demand	Total Waktu Tempuh
1 3 2	130	154
1 4 2	115	206
1 7 2	140	111
1 8 2	140	126

Diperoleh total waktu tempuh terkecil, yaitu $1 - 7 - 2$ dengan waktu tempuh total 111 menit dan total *demand* 140 buah. Karena total *demand* 140 buah < kapasitas kendaraan, maka akan disisipkan konsumen lain.

3) $1 - y - x - 2$, dengan $\forall x \in N - \{1,2\}$ dan $\forall y \in N - \{1,x,2\}$

Dilakukan perhitungan seperti 2) dan didapatkan hasil seperti pada Tabel 9.b.

Tabel 9.b. Total Waktu Tempuh Pembentukan Rute Pertama dengan *Floyd Warshall* pada iterasi 2

Rute	Demand	Total Waktu Tempuh
1 5 3 2	185	285
1 7 3 2	200	261
1 3 4 2	175	258
1 6 4 2	185	284
1 3 7 2	200	172
1 5 7 2	195	200
1 10 7 2	200	164
1 5 8 2	195	258
1 10 8 2	200	220

Diperoleh total waktu tempuh terkecil dari depot menuju konsumen-1, yaitu $1 - 10 - 7 - 2$ dengan waktu tempuh total 164 menit dan total *demand* 200 buah. Karena total *demand* 200 buah = kapasitas kendaraan, maka rute dengan total waktu tempuh terkecil menuju konsumen-1 adalah depot – konsumen-9 – konsumen-6 – konsumen-1.

4) Menggunakan cara yang sama untuk mencari total waktu tempuh terkecil dari konsumen lain.

- b. Kemudian akan dicari rute dengan waktu tempuh terkecil dari depot sampai kembali ke depot.

Tabel 9.c. Total Waktu Tempuh Pembentukan Rute Pertama dengan *Floyd Warshall* saat Kembali ke Depot

Rute	Demand	Total Waktu Tempuh
1 10 7 2 1	200	220
1 10 2 3 1	190	241
1 7 2 4 1	185	294
1 2 7 5 1	195	262
1 10 8 6 1	200	264
1 3 2 7 1	200	224
1 10 7 8 1	200	232
1 9 1	65	66
1 2 7 10 1	200	225

Dari Tabel 9.c diperoleh rute 1-10-7-2-1 dan 1-9-1. Kemudian jika ada konsumen yang belum dilayani, maka buat rute baru lagi.

Untuk rute selanjutnya digunakan cara yang sama seperti pembentukan rute pertama.

Berdasarkan perhitungan yang telah dilakukan menggunakan algoritma *Floyd Warshall*, permasalahan CVRPTW dalam data simulasi tersebut menghasilkan sebuah solusi yang terdiri dari 5 rute untuk melayani 9 konsumen. Adapun rekapitulasi hasil penyelesaian masalah menggunakan algoritma *Floyd Warshall*.

Tabel 10. Rekapitulasi Hasil penyelesaian CVRPTW dengan FW

Kendaraan ke-	Rute Perjalanan	Demand (buah)	Waktu Tempuh (menit)
1	1-9-1	65	66
2	1-10-7-2-1	200	220
3	1-6-8-5-1	195	281
4	1-3-1	60	103
5	1-4-1	45	269
Total		565	939

Pada Tabel 10, pembentukan rute menggunakan algoritma *Floyd Warshall* menghasilkan 5 rute untuk melayani 9 pelanggan dengan waktu penyelesaian selama 939 menit.

Algoritma *Nearest Neighbour*

Algoritma *Nearest Neighbour* merupakan algoritma yang memiliki prinsip dasar membentuk rute dengan memilih konsumen yang terdekat dari lokasi awal. Berikut langkah-langkah dari algoritma tersebut.

- a. Langkah 1
Cari konsumen ke- j yang memiliki waktu tempuh terpendek dari titik awal i .
- b. Langkah 2
Hitung total waktu tempuh kendaraan ($Wt = t + \text{waktu pelayanan } i + Wt_{ij,t}$). Untuk $Wt \geq a_j$ maka $Wt = a_j$. Jika $Wt \leq b_j$ maka lanjut ke Langkah 3. Jika $Wt > b_j$, maka lanjut ke Langkah 5.
- c. Langkah 3
Hitung permintaan/muatan kendaraan ($\text{demand} = \text{demand} + q_i$). Jika $\text{demand} \leq Q$, maka lanjut ke Langkah 4. Jika $\text{demand} > Q$, maka lanjut ke Langkah 5.
- d. Langkah 4
Set konsumen ke- j sebagai titik awal, kemudian ulangi ke Langkah 2.
- e. Langkah 5
Batalkan pemilihan konsumen, kemudian pilih konsumen yang belum dilayani dan yang terdekat dengan titik awal berdasarkan keterurutan dan kembali ke Langkah 2. Jika semua konsumen tidak ada yang layak, lanjutkan ke Langkah 6.
- f. Langkah 6
Kembali ke depot.
- g. Langkah 7
Jika pada saat kembali ke depot $Wt > b_{\text{depot}}$, maka batalkan konsumen terakhir dan kembali ke depot.
Berikut langkah-langkah penyelesaian masalah CVRPTW dengan algoritma *nearest neighbour*.
 1. Pembentukan rute pertama
Perjalanan diawali dari depot(1) dengan kendaraan-1 dan pada pukul 07.00 ($t=0$). Kemudian konsumen akan dilayani berturut-turut sesuai dengan batasan kendala dan kembali ke depot sebelum batas waktu akhir depot. Adapun langkah-langkah pembentukan rute pertama.
 - a. Membuat matriks waktu tempuh saat $t=0$. Kecepatan rata-rata kendaraan bersifat dinamis berdasarkan waktu dan kecepatan rata-rata maksimal

kendaraan yaitu 50 km/jam. Jika kecepatan rata-rata maksimal kendaraan < kecepatan rata-rata pada kondisi jalan dan waktu tertentu, maka kecepatan rata-rata yang digunakan adalah kecepatan rata-rata maksimal kendaraan. Jika kecepatan rata-rata maksimal kendaraan > kecepatan rata-rata pada kondisi jalan dan waktu tertentu, maka kecepatan rata-rata yang digunakan adalah kecepatan rata-rata pada kondisi jalan dan waktu tertentu.

Tabel 11.a. Matriks waktu tempuh kendaraan ke-1 saat t=0

	1	2	3	4	5	6	7	8	9	10
1	0	79.266	48	121.3333	91.0069	54.6494	71.6938	60.4669	12.9035	54.8839
2	73.0082	0	98.276	76.761	inf	inf	24.0832	56.001	inf	inf
3	49.4118	103.7358	0	126	94.4047	inf	85.3396	inf	inf	inf
4	182	89.5545	157.5	0	inf	148.4559	inf	inf	inf	inf
5	72.8055	inf	89.9093	inf	0	114.7364	67.9255	112.3329	inf	inf
6	54.6494	inf	inf	161.5549	143.4205	0	75.5926	39.4891	inf	inf
7	71.6938	18.0624	95.0927	inf	79.2465	90.2234	0	48.4149	inf	80.6994
8	53.7484	57.601	inf	inf	127.3106	41.7456	48.4149	0	inf	103.5482
9	10.3228	inf	inf	inf	inf	inf	inf	inf	0	inf
10	48.0234	inf	inf	inf	inf	inf	66.5770	98.0983	inf	0

- b. Memilih konsumen yang akan dilayani berdasarkan matriks waktu tempuh terkecil dari depot yaitu konsumen-8 dengan waktu tempuh 12.9035 menit dan *demand* 65 buah. Karena waktu tempuh dari depot ke konsumen-8 (Wt_{19}) adalah 12.9035 menit < a_9 , maka kendaraan sampai di tempat konsumen lebih awal dari *time windows*. Jadi konsumen mulai dilayani pada pukul 07.45 (a_9) dengan lama pelayanan (s_9) 10 menit dan total *demand* < kapasitas kendaraan. Jadi, rute sementara yang terbentuk adalah 1 – 9 dengan waktu tempuh total = $a_9 + s_9 = 45 + 10 = 55$ menit dan total *demand* = 65 buah.
- c. Karena kendaraan selesai melayani konsumen-8 pada pukul 07.55, maka buat matriks waktu tempuh saat t=55.

Tabel 11.b. Matriks waktu tempuh kendaraan ke-1 saat t=55

	1	2	3	4	5	6	7	8	9	10
1	0	61.6513	43.8261	121.3333	84.0063	54.6494	64.5244	55.8156	11.9109	46.1025
2	68.2208	0	90.3505	70.0861	inf	inf	19.7044	53.0536	inf	inf
3	45.8182	88.9164	0	126	83.2983	inf	77.401	inf	inf	inf
4	182	80.599	145.3846	0	inf	143.2922	inf	inf	inf	inf
5	84.0063	inf	88.5044	inf	0	103.2628	64.2539	100.5084	inf	inf
6	58.553	inf	inf	149.8055	143.4205	0	71.7161	34.5141	inf	inf
7	64.5244	18.0624	86.8237	inf	71.3218	79.9122	0	43.5734	inf	70.7013
8	52.2017	48.3849	inf	inf	104.1633	41.7456	43.5734	0	inf	96.4069
9	10.3228	inf	inf	inf	inf	inf	inf	inf	0	inf
10	44.3293	inf	inf	inf	inf	inf	61.4557	94.7729	inf	0

- d. Karena tidak ada akses dari konsumen-8 ke konsumen lain, maka kendaraan harus kembali ke depot. Waktu tempuh dari konsumen-8 ke depot adalah 10.3228 ≈ 11 menit. Waktu tempuh total = $t + Wt_{91} = 55 + 11 = 66$ menit < b_1 , maka memenuhi kendala *time windows*. Jadi kendaraan sampai di depot pada pukul 08.06. Jadi, rute yang terbentuk adalah 1 – 9 – 1 dengan waktu tempuh total 66 menit dan total *demand* = 65 buah.

Jika ada konsumen yang belum dilayani, maka buat rute baru lagi. Untuk rute selanjutnya digunakan cara yang sama seperti pembentukan rute pertama.

Berdasarkan perhitungan yang telah dilakukan menggunakan algoritma *Nearest Neighbour*, permasalahan CVRPTW dalam data simulasi tersebut menghasilkan sebuah solusi yang terdiri dari 5 rute untuk melayani 9 konsumen. Adapun rekapitulasi hasil penyelesaian masalah menggunakan algoritma *Nearest Neighbour*.

Tabel 12. Rekapitulasi Hasil penyelesaian CVRPTW dengan NN

Kendaraan ke-	Rute Perjalanan	Demand (buah)	Waktu Tempuh (menit)
1	1 – 9 – 1	65	66
2	1 – 3 – 7 – 2 – 1	200	229
3	1 – 6 – 8 – 10 – 1	200	248
4	1 – 5 – 1	55	194
5	1 – 4 – 1	45	269
Total		565	1006

Pada Tabel 12, pembentukan rute menggunakan algoritma *Nearest Neighbour* menghasilkan 5 rute untuk melayani 9 pelanggan dengan waktu penyelesaian selama 1.006 menit.

KESIMPULAN

Berdasarkan hasil dari pembahasan, terlihat bahwa algoritma *Floyd Warshall* dapat membentuk rute dengan total waktu tempuh yang lebih baik (optimal) dibandingkan dengan algoritma *Nearest Neighbour*. Namun dalam proses penyelesaiannya, algoritma *Nearest Neighbour* jauh lebih cepat dan praktis dibandingkan dengan algoritma *Floyd Warshall*.

Saran

Saran yang dapat diberikan untuk pengembangan tugas akhir ini adalah mengimplementasikan algoritma *Floyd Warshall* dan algoritma *Nearest Neighbour* pada kasus nyata yang serupa pada permasalahan skripsi ini. Penelitian ini juga dapat dikembangkan dengan menggunakan software selain Matlab, seperti Java, agar dapat diaplikasikan menggunakan perangkat bergerak. Dapat juga menggunakan parameter yang berbeda untuk melihat pengaruh terhadap hasil yang diperoleh, misalnya lebar jalan, kondisi jalan, dan kultur pada setiap lalu lintas. Selain itu, penggunaan algoritma lain seperti algoritma genetika, *tabu search*, *ant colony optimization*.

DAFTAR PUSTAKA

- Hidayat. (1986). *Teori Efektivitas dalam Kinerja Karyawan*. Yogyakarta : Gajah Mada University Press.
- Hindriyanto D.P. (2014). *Cara Mudah Belajar Metode Optimasi Metaheuristik Menggunakan Matlab*. Yogyakarta: Penerbit Gava Media.
- Johnsonbaugh, Richard. (2001). *Discrete Mathematics*. Fifth Editions. New Jersey: Prentice-Hall, Inc.
- Parment, M. Michael, Edgar G. Goodaire. (2002). *Discrete Mathematis with Graph Theory*. United States America: Prentice-Hall, Inc.
- Pius A. Partanto, & M. Dahlan Bahri. (1994). *Kamus Ilmiah Populer*. Surabaya: Arkoba.

- Rahmi Y., & Murti A., (2013). *Penerapan Metode Saving Matrix Dalam Penjadwalan Dan Penentuan Rute Distribusi Premium Di SPBU Kota Malang*. Jurnal Rekayasa Mesin. vol.04, no.01, hlm. 17-26.
- Siang, Jong Jek. (2006). *Matematika Diskret dan Aplikasinya pada Ilmu Komputer*. Yogyakarta: Andi Offset.
- Siang, Jong Jek. (2011). *Riset Operasi dalam Pendekatan Algoritmis*. Yogyakarta: Andi Offset.
- Sugeng Mardiyono.(1996). *Matematika Diskret*. Yogyakarta: FMIPA IKIP Yogyakarta.
- Talbi, E.-G. (2009). *Metaheuristik: from design to implementation*. Hoboken, New Jersey: John Wiley & Son, Inc.
- Toth, P. & Vigo, D. (2002). *Vehicle Routing Problem*. SIAM. Philadelphia.
- Wilson Robin J. & Watkins John J. (1990). *Graph An Introductory Approach. The Open University & Colorado Collage*.