

Pemecahan Traveling Salesman Problem Menggunakan Teknik Branch and Bound dan Cheapest Insertion Heuristic (Studi Kasus : PT. Paris Jaya Mandiri – Ambon)

Daniel B. Paillin^{*1)}, Johan M Tupan²⁾

^{1,2)}Jurusan Teknik Industri, Fakultas Teknik, Universitas Pattimura, Kampus Poka, Ambon, 97233, Indonesia

Email: dani_ti_fatek@yahoo.co.id, johan.tupan@fatek.unpatti.ac.id

ABSTRAK

Penelitian ini membahas perbandingan teknik Branch and Bound dengan Cheapest Insertion dalam pemecahan Traveling Salesmen Problem (TSP) berdasarkan penghematan jarak dan waktu tempuh kendaraan. Salah satu kasus nyata dari permasalahan ini adalah penentuan rute kendaraan untuk pengiriman produk nestle pada PT. Paris Jaya Mandiri di kota Ambon. Hasil penelitian menunjukkan bahwa teknik Branch and Bound memberikan hasil terbaik dibandingkan dengan Cheapest Insertion, dengan persentase penghematan jarak sebesar 17.96% dan penghematan waktu sebesar 18.92% dari rute regular perusahaan.

Kata kunci: branch and bound, cheapest insertion, rute kendaraan, traveling salesman problem

1. Pendahuluan

Pendistribusian produk perusahaan ke konsumen sering menimbulkan permasalahan yang perlu diperhatikan terkhususnya perusahaan yang bergerak di bidang distributor sehingga perusahaan dapat melakukan penghematan dari segi jarak dan waktu tempuh yang lebih minimum, selain itu pendistribusian yang baik akan meningkatkan kepercayaan konsumen terhadap perusahaan tersebut. PT. Paris Jaya Mandiri merupakan salah satu perusahaan distributor produk Nestle di Ambon. Untuk daerah Ambon distribusi produk ini meliputi kecamatan Sirimau dan kecamatan Nusaniwe. Saat ini jumlah outlet yang akan dikunjungi oleh perusahaan tersebar pada 39 titik lokasi berbeda. Kondisi ini menyebabkan banyak pilihan rute yang dapat digunakan, oleh karena itu perusahaan harus berhati-hati dalam menentukan pilihan rute yang dituju sehingga dapat dicapai jarak tempuh yang optimal. Selain itu, kondisi perusahaan saat ini belum memiliki rute pasti pengiriman produk. Hal ini menyebabkan rute pengiriman tidak terstruktur, dan menyebabkan terjadinya pertambahan jarak dan waktu tempuh yang berakibat terhadap adanya keterlambatan pengiriman produk. Untuk itu, perlu dilakukan penentuan rute optimal sehingga perusahaan dapat melakukan aktivitas distribusi secara efektif dan efisien.

Permasalahan diatas lebih dikenal dengan nama *Travelling Salesman Problem* (TSP). TSP adalah perjalanan seorang *salesman* menuju ke semua tempat yang akan dituju dan akhirnya kembali ke tempat awal dengan menggunakan jalur terpendek, akan tetapi semua tempat hanya boleh dilalui satu kali. Terdapat beberapa algoritma yang digunakan untuk menemukan jalur terpendek dari TSP tersebut, seperti algoritma branch and bound digunakan untuk menyelesaikan TSP untuk pengangkutan barang kantor pos di Palembang (Putra dkk, 2015). Konsep utama dari metode Branch and Bound adalah membagi masalah aslinya yang berukuran besar menjadi sub masalah yang lebih kecil kemudian menjadi anak gugus yang lebih kecil lagi sampai semua submasalah dapat diselesaikan (Hillier andLieberman, 2001). Gea Aristi (2014) menggunakan algoritma *Greedy*, Algoritma *Cheapest Insertion* dan *dynamic programming* untuk mencari rute terpendek dari TSP degan cara membandingkan performansi solusi dari ketiga metode tersebut. Banyak metode dan algoritma yang dapat digunakan untuk menyelesaikan masalah TSP. Masing-masing algoritma memiliki kelebihan dan kekurangannya. Suatu algoritma memiliki hasil yang berbeda-beda, karena belum tentu suatu algoritma yang

memiliki optimasi yang tinggi untuk suatu kasus memiliki optimasi yang tinggi pula untuk kasus yang lain.

Berdasarkan masalah tersebut, penulis tertarik untuk membahas penentuan rute distribusi produk yang optimal yang harus dilalui, sehingga penyaluran produk dapat menjadi lebih efisien.

2. Metode

Traveling Salesman Problem

Hoffman dan Wolfe (1985), mendefinisikan *TSP* sebagai berikut: *TSP is one which has commanded much attention of mathematicians and computer scientists specifically because it is so easy to describe and so difficult to solve. The problem can simply be stated as: if a Traveling salesman wishes to visit exactly once each of a list of m cities (where the cost of Traveling from city i to city j is C_{ij}) and then return to the home city, what is the least costly route the Traveling salesman can take?* Berdasarkan definisi di atas, Traveling salesman problem merupakan suatu masalah yang mudah dideskripsikan namun sulit untuk diselesaikan, yaitu masalah bagaimana menentukan jarak terpendek dalam perjalanan melwati titik-titik tertentu di mana satu titik harus dilalui satu kali dan hanya boleh dilalui satu kali saja dan perjalanan harus berakhir dengan kembali ke titik pertama. Dalam hal ini titik-titik tersebut merupakan kota-kota dalam suatu wilayah tertentu

Dalam kasus penentuan rute formulasi model TSP dapat ditulis sebagai berikut: Jarak perjalanan dari titik i ke titik j direpresentasikan oleh C_{ij} . Selanjutnya didefinisikan variabel keputusan X_{ij} yang merepresentasikan ada tidaknya perjalanan dari titik i ke j dalam suatu rute sebagai berikut :

$$X_{ij} = \begin{cases} 1, & \text{Jika terdapat perjalanan kendaraan dari } i \text{ ke } j \\ 0, & \text{Jika tidak ada perjalanan kendaraan dari } i \text{ ke } j \end{cases} \quad (1)$$

Jika Z merupakan fungsi tujuan TSP, maka fungsi tujuan Z dirumuskan dengan meminimumkan

$$Z = \sum_{i=1}^n \sum_{j=1}^n C_{ij} X_{ij} \quad (2)$$

dengan batasan kendala,

$$\sum_{i=1}^n X_{ij} = 1 \quad (i = 1, 2, 3, \dots, N)$$

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

$$X_{ij} = 0 \text{ atau } 1$$

Batasan yang pertama dan kedua memastikan bahwa rute yang terpilih mendatangi setiap kota 1 kali dan meninggalkan kota tersebut 1 kali.

Algoritma Branch And Bound

Branch and bound merupakan metode algoritma yang umum digunakan untuk menentukan penyelesaian optimal dari masalah optimisasi, khususnya pada diskrit dan optimisasi kombinatorial. Pada intinya algoritma ini menggunakan pendekatan enumerasi dengan cara mematikan search space yang tidak mengarah pada penyelesaian. Pada tahun 1960, algoritma branch and bound diperkenalkan oleh A.H. Land dan A.G. Doig.

Sesuai dengan namanya *branch and bound* memiliki dua alat yaitu *branching* dan *bounding*. *Branching* dilakukan dengan cara meng-cover daerah penyelesaian yang layak dengan beberapa sub daerah layak yang lebih kecil. *Bounding* dilakukan dengan cara

menentukan nilai batas atas dan batas bawah untuk penyelesaian optimal di dalam sub daerah yang layak.

Proses *branching* menggunakan skema *Breadth First Search (BFS)*. Simpul atau titik yang dibangkitkan terlebih dahulu merupakan simpul yang bertetangga dengan simpul akar. Namun proses pemilihan simpul yang akan diperluas (simpul *expand*) tidak seperti pada *BFS* murni. Tidak seperti *BFS* murni yang memilih simpul *expand* berdasarkan urutan pembangkitan, pada algoritma *branch and bound* pemilihan simpul *expand* didasarkan pada nilai fungsi objektif.

Masalah optimisasi biasanya memiliki fungsi untuk menghasilkan nilai batas yang unik. Sedangkan pada algoritma *branch and bound* pemilihan formula fungsi menggunakan metode *heuristic*, berdasarkan pada pengalaman dan *instinc* pembuat program dan tidak ada bukti matematisnya. Hasil optimisasi yang diperoleh melalui algoritma ini bergantung pada keakuratan pemilihan fungsi batas tersebut. Tidak ada cara baku untuk menentukan fungsi batas karena masalah yang sama bisa saja memiliki rumus perhitungan nilai batas yang berbeda.

Langkah-langkah algoritma *branch and bound* untuk menyelesaikan *TSP* adalah sebagai berikut.

Langkah 1:

Tetapkan penyelesaian awal masalah. Penyelesaian yang ditetapkan merupakan rute perjalanan lengkap. Tentukan batas tertinggi pada nilai minimum fungsi objektif dengan mencari berbagai kemungkinan rute perjalanan. Batas atas ini dinotasikan dengan f_u .

Langkah 2:

Buat cabang awal dengan mengatur $x_j = 1$ untuk masing-masing kota $j = 2, 3, \dots, n$. Untuk $i = j$, nilai $C_{ij} = M$ untuk menyatakan rute yang tidak mungkin. Hitung batas terendah yang dinotasikan dengan f_L pada nilai minimum fungsi objektif di setiap titik. Dari data awal, hapus baris pertama dan kolom ke j , serta ganti $C_{ij} = M$. Tentukan penyelesaian masalah dan tambahkan harga f ke C_{ij} untuk memperoleh f_L sehingga $f_L = C_{ij} + f$. Jika $f_L \leq f_u$ aktifkan simpul, dan jika sebaliknya hapus simpul.

Langkah 3:

Jika tidak ada simpul aktif pada langkah ini maka penyelesaian terbaik saat ini adalah optimal. Jika tidak, pilih simpul dengan nilai f_L terkecil dan buat cabang baru dengan mengatur $x_{jk} = 1$ untuk setiap kota yang belum dikunjungi sebelumnya.

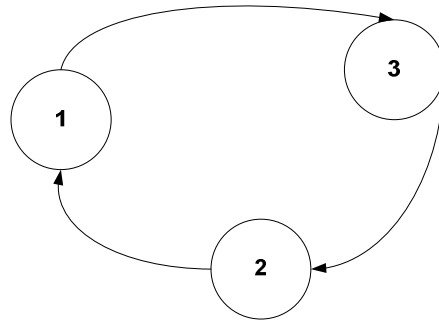
Langkah 4:

Buat batasan f pada setiap simpul dengan menghapus baris j dan kolom k dari data pada simpul aktif di atasnya. Tambahkan nilai f ke C_{jk} dengan seluruh nilai sebelumnya.

Algoritma Cheapest Insertion Heuristics

Algoritma *Cheapest Insertion Heuristics* adalah algoritma yang membangun suatu *tour* dari sikel-sikel kecil dengan bobot minimal dan secara berturut-turut ditambah dengan titik baru sampai semua titik berhasil dilalui. Berikut ini adalah tata cara urutan algoritma CIH berdasarkan *paper* Winston, Wayne L. dan Goldberg, Jeffrey B pada tahun 2004 yang berjudul *Operations Research Application And Algorithms 4th Edition*

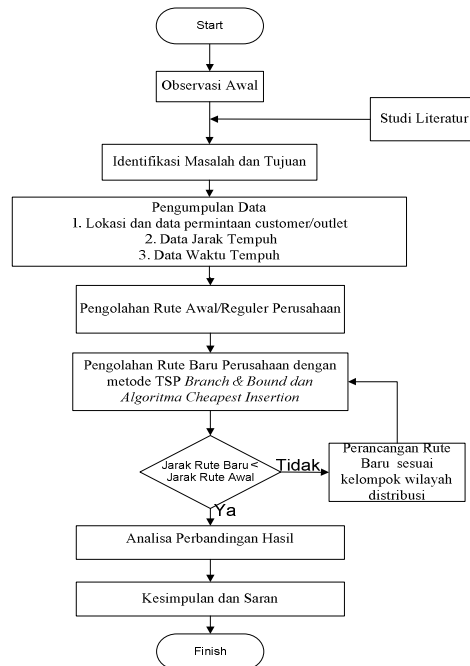
1. Penelusuran dimulai dari sebuah kota pertama yang dihubungkan dengan sebuah kota terakhir.
2. Dibuat sebuah hubungan *subtour* antara 2 kota tersebut. Yang dimaksud *subtour* adalah perjalanan dari kota pertama dan berakhir di kota pertama. Seperti $(1,3) \rightarrow (3,2) \rightarrow (2,1)$ pada gambar 1.



Gambar 1. Subtour

3. Ganti salah satu arah hubungan (*arc*) dari dua kota dengan kombinasi dua *arc*, yaitu *arc* (*i,j*) dengan *arc* (*i,k*) dan *arc* (*k,j*), dengan *k* diambil dari kota yang belum masuk *subtour* dan dengan tambahan jarak terkecil. Jarak diperoleh dari: $C_{ik} + C_{kj} - C_{ij}$
 - C_{ik} adalah jarak dari kota *i* ke kota *k*
 - C_{kj} adalah jarak dari kota *k* ke kota *j*
 - C_{ij} adalah jarak dari kota *i* ke kota *j*
4. Ulangi langkah 3 sampai seluruh kota masuk dalam *subtour*.

Berikut ini adalah tahapan penelitian yang dilakukan dapat dilihat pada gambar 2. berikut ini



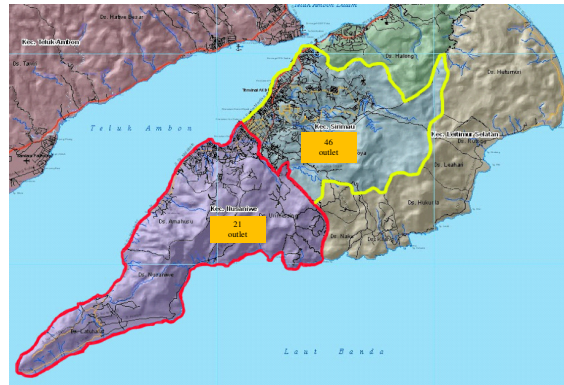
Gambar 2. Flowchart Penelitian

3. Hasil dan Pembahasan

Deskripsi Sistem Distribusi Produk Nestle di Daerah Ambon

Kegiatan distribusi produk Nestle yang dilakukan oleh PT. Paris Jaya Mandiri mencakup 2 wilayah yang ada di Kepulauan Ambon yaitu wilayah kecamatan Sirimau dan kecamatan Nusaniwe. Cakupan wilayah kecamatan Sirimau dimulai dari Galala – Air Mata Cina dan

cakupan wilayah kecamatan Nusaniwe dimulai dari Waihaong – Seri. Berikut merupakan gambaran wilayah distribusi produk Nestle di pulau Ambon.



Gambar 3. Peta Umum Sebaran Outlet di Pulau Ambon

Data Lokasi dan Permintaan Outlet

Berikut merupakan daftar lokasi dari kode customer (KC) dan demand (D) dari outlet yang akan dikunjungi. KC (PJM) PT. Paris Jaya Mandiri

Tabel 1. Lokasi dan Jumlah Permintaan Outlet PT. Paris Jaya Mandiri Ambon

KC	Nama Outlet	Alamat	D(krt)/minggu	KC	Nama Outlet	Alamat	D(krt)/minggu
C1	Cahaya Fatan	Pertokoan Batu Merah	16	C35	Anda	Jl. Ahmad Yani	13
C2	Al – Bagir	Pertokoan Batu Merah	18	C36	Surya Indah Toko	Batu Gajah	9
C3	Mas Sri	Pertokoan Batu Merah	30	C37	Prison	Batu Meja	22
C4	Nuryanti	Pertokoan Batu Merah	14	C38	Biasa Toko	Jl. Setia Budi	8
C5	Blihar	Pertokoan Batu Merah	31	C39	Berkat	Trikora	9
C6	Manna Toko	Galala	26	C40	Maluku Jaya Abadi Toko	Diponegoro (Kl. Ahusen)	15
C7	Kios Julais	Galala	9	C41	Anugerah Toko	Dekat Hotel Pasifik	5
C8	Herley	Tantui	18	C42	The Indah Toko	A. Y. Patty	50
C9	Fandi	Mardika	10	C43	Ci Nona	Yos Soedarso	50
C10	Melissa 2	Mardika	36	C44	Sovia Toko	Pala	15
C11	Ko Han	Mardika	50	C45	Andi Kios	Air Mata Cina	8
C12	Adin	Mardika	18	C46	Lily Toko	A.Y.Patty	22
C13	Akbar	Mardika	12	C47	Kios Anes	Depan Pasar Tagalaya	13
C14	Om Mon	Belakang Soya	42	C48	Cempaka Toko	Dr. Kayadoe	10
C15	Liana	Tanah Tinggi	15	C49	Safari Toko	Dr. Kayadoe	17
C16	Meter Toko	D. I. Panjaitan	7	C50	Empi Toko	Dr. Kayadoe	20
C17	Toko Mitra Jaya	Jl. Pantai Mardika	16	C51	Darren	Dr. Kayadoe	39
C18	...	Jl. Pantai Mardika	10	C52	Kios Ema	Dr. Kayadoe	13
C19	Fata Rahmat	Jl. Pantai Mardika	13	C53	Damai Saparua Toko	Dr. Kayadoe	14
C20	Toko Ivan	Jl. Pantai Mardika	25	C54	Spout Toko	Dr. Kayadoe	21
C21	...	Jl. Pantai Mardika	19	C55	Mikael Toko	Gudang Arang	11
C22	Teiki Toko	Karpan	15	C56	Amnyong Toko	Kusu – Kusu	11
C23	Harmoni	Wara	6	C57	Sinar Mulia Toko	Benteng	19
C24	Kios 2 Putri	Kebun Cengkeh	4	C58	Benteng Indah Toko	Benteng	11
C25	Balvis Kios	BTN Kanawa	11	C59	Jaya Abadi Toko	Bentas	18
C26	Haikal	Kebun Cengkeh	11	C60	Natsepa Kios	Amahusu	6
C27	Mas No	Skip	12	C61	Linda Toko	OSM	7
C28	Mustika Toko	Jl. Rijali No. 40	10	C62	Beringin Baru	Jl. Nn Saar Sopacua	12
C29	Aping Toko	Batu Meja	8	C63	Kace Toko	Wainitu	8
C30	Sakura Toko	Bere – Bere	14	C64	Baru Toko	Talake	7
C31	Jhon Coa	Bere – Bere	12	C65	Aliang Toko	Talake	26
C32	Rejeki Toko	Bere – Bere	12	C66	Olive	Waihaong	14
C33	Alan	Kayu Putih	18	C67	Umar Toko	Jl. Sultan Babulah	20
C34	Jerry Cell	Jl. Ahmad Yani	8				

Data Jarak Tempuh

Data jarak tempuh adalah data jarak dari depot PT. Paris Jaya Mandiri menuju sejumlah outlet yang dituju dan juga jarak antaroutlet. Perhitungan jarak diperoleh dengan rumus *Euclidean* $d_{(ij)} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ dengan titik x dan y merupakan titik latitude dan longitude outlet yang diperoleh melalui *google maps*.

Tabel 2. Matriks Jarak Antaroutlet Wilayah Sirimau 1

		Data Jarak Tempuh (KM)																								
Dari/ke	PJM	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C17	C18	C19	C20	C21	C41	C42	C43	C44	C45	C46	
PJM																										
C1	2.80																									
C2	2.80	0																								
C3	2.83	0.03	0.03																							
C4	2.90	0.09	0.09	0.09																						
C5	2.81	0.03	0.03	0.04	0.07																					
C6	1.50	2.98	2.98	2.99	3.07	3																				
C7	1.40	3.06	3.06	3.07	3.15	3.08	0.14																			
C8	1.15	2.36	2.36	2.37	2.45	2.39	0.62	0.72																		
C9	2.60	0.13	0.13	0.15	0.12	0.12	3.02	3.10	2.40																	
C10	3.10	0.27	0.27	0.27	0.18	0.24	3.21	3.30	2.59	0.19																
C11	3.10	0.27	0.27	0.27	0.18	0.24	3.21	3.30	2.59	0.19	0															
C12	3.12	0.18	0.18	0.17	0.09	0.15	3.15	3.24	2.54	0.17	0.12	0.12														
C13	3.12	0.18	0.18	0.17	0.09	0.15	3.15	3.24	2.54	0.17	0.12	0.12	0													
C17	4.27	0.49	0.49	0.48	0.40	0.46	3.46	3.55	2.85	0.45	0.27	0.27	0.31	0.31												
C18	3.50	0.49	0.49	0.46	0.40	0.46	3.46	3.54	2.85	0.48	0.31	0.31	0.32	0.32	0.10											
C19	3.30	0.50	0.50	0.49	0.41	0.47	3.48	3.55	2.86	0.48	0.31	0.31	0.32	0.32	0.08	0.03										
C20	3.41	0.51	0.51	0.49	0.42	0.48	3.48	3.56	2.86	0.51	0.34	0.34	0.34	0.34	0.14	0.04	0.06									
C21	3.34	0.36	0.36	0.34	0.28	0.34	3.32	3.40	2.71	0.38	0.25	0.25	0.21	0.21	0.21	0.15	0.17	0.16								
C41	3.30	0.52	0.52	0.54	0.46	0.50	3.32	3.42	2.70	0.40	0.30	0.30	0.42	0.42	0.40	0.50	0.48	0.54	0.52							
C42	4.78	1.23	1.23	1.22	1.14	1.21	4.21	4.29	3.59	1.19	1.00	1.00	1.05	1.05	0.74	0.76	0.74	0.75	0.91	0.96						
C43	4.60	1.15	1.15	1.13	1.06	1.12	4.12	4.20	3.51	1.12	0.93	0.93	0.97	0.97	0.67	0.66	0.65	0.64	0.80	0.96	0.20					
C44	4.56	0.93	0.93	0.91	0.83	0.90	3.90	3.98	3.29	0.90	0.70	0.70	0.75	0.75	0.44	0.44	0.43	0.43	0.59	0.74	0.32	0.23				
C45	4.85	1.81	1.81	1.81	1.72	1.79	4.74	4.84	4.12	1.74	1.55	1.55	1.64	1.64	1.34	1.38	1.36	1.38	1.52	1.42	0.66	0.84	0.97			
C46	4.70	1.22	1.22	1.21	1.13	1.20	4.20	4.28	3.58	1.18	0.98	0.98	1.04	1.04	0.73	0.75	0.73	0.75	0.90	0.94	0.03	0.21	0.32	0.65		

Tabel 3. Matriks Jarak Antaroutlet Wilayah Sirimau 2

		Data Jarak Tempuh (KM)																							
Dari/ke	PJM	C14	C15	C16	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38	C39	C40		
PJM																									
C14	2.70																								
C15	3.40	0.59																							
C16	4.00	0.45	0.51																						
C22	4.50	0.19	0.77	0.57																					
C23	3.60	3.35	3.75	3.80	3.24																				
C24	1.60	1.76	2.24	2.20	1.63	1.65																			
C25	3	2.79	3.22	3.23	2.67	0.60	1.05																		
C26	2.30	2.44	2.87	2.88	2.32	0.93	0.72	0.35																	
C27	3.70	0.69	0.15	0.66	0.87	3.73	2.26	3.21	2.87																
C28	3.50	0.69	0.12	0.62	0.88	3.78	2.3	3.26	2.92	0.07															
C29	4.10	0.81	0.23	0.65	1	3.96	2.47	3.43	3.09	0.23	0.18														
C30	3.89	0.9	0.32	0.79	1.09	3.96	2.5	3.44	3.1	0.24	0.22	0.14													
C31	3.98	1	0.44	0.94	1.19	3.93	2.51	3.43	3.09	0.32	0.33	0.31	0.17												
C32	4.02	1	0.44	0.94	1.19	3.93	2.51	3.43	3.09	0.32	0.33	0.31	0.17	0.00											
C33	4.40	0.99	0.41	0.86	1.18	4.02	2.57	3.51	3.17	0.32	0.30	0.21	0.08	0.14	0.14										
C34	3.71	0.99	0.40	0.82	1.18	4.07	2.61	3.56	3.22	0.35	0.32	0.18	0.12	0.23	0.25	0.09									
C35	3.71	0.99	0.40	0.82	1.18	4.07	2.61	3.56	3.22	0.35	0.32	0.18	0.12	0.23	0.23	0.09	0								
C36	4.10	1.42	0.84	1.19	1.6	4.47	3.04	3.97	3.63	0.78	0.75	0.60	0.54	0.54	0.47	0.44	0.44								
C37	4.20	1.06	0.48	0.88	1.25	4.14	2.68	3.63	3.29	0.43	0.39	0.25	0.19	0.26	0.26	0.13	0.08	0.08	0.36						
C38	4.53	1.15	0.69	0.8	1.32	4.44	2.9	3.9	3.55	0.74	0.68	0.51	0.59	0.72	0.72	0.58	0.50	0.50	0.54	0.48					
C39	4.67	1.29	0.79	0.94	1.46	4.54	3.02	4.01	3.66	0.82	0.76	0.59	0.63	0.75	0.75	0.61	0.53	0.53	0.45	0.49	0.15				
C40	4.40	1.35	0.60	1.07	1.53	4.50	3.03	3.99	3.64	0.78	0.73	0.56	0.55	0.61	0.61	0.49	0.43	0.43	0.20	0.36	0.35	0.25			

Tabel 4. Matriks Jarak Antaroutlet Wilayah Nusaniwe

		Data Jarak Tempuh (KM)																					
Dari/Ke	PJM	C47	C48	C49	C50	C51	C52	C53	C54	C55	C56	C57	C58	C59	C60	C61	C62	C63	C64	C65	C66	C67	
PJM																							
C47	5.20																						
C48	6	0.73																					
C49	5.60	0.28	0.46																				
C50	5.80	0.50	0.26	0.23																			
C51	5.82	0.52	0.25	0.24	0.02																		
C52	5.82	0.52	0.25	0.24	0.02	0																	
C53	6.33	1.02	0.32	0.75	0.58	0.57	0.57																
C54	6.60	1.23	0.55	0.97	0.81	0.80	0.80	0.23															
C55	7.03	1.62	1.16	1.44	1.37	1.36	1.36	0.89	0.73														
C56	6	0.71	0.36	0.51	0.47	0.46	0.46	0.42	0.56	0.93													
C57	6.90	1.55	1.01	1.34	1.24	1.23	1.23	0.71	0.52	0.23	0.84												
C58	6.91	1.70	1.15	1.49	1.38	1.37	1.37	0.85	0.64	0.23	0.99	0.15											
C59	7.70	1.95	1.30	1.71	1.56	1.54	1.54	0.98	0.75	0.66	1.25	0.51	0.43										
C60	10	4.05	3.32	3.78	3.57	3.55	3.55	3.04	2.86	2.92	3.43	2.79	2.70	2.28									
C61	6.40	1.15	0.66	0.95	0.86	0.85	0.85	0.44	0.37	0.50	0.44	0.40	0.55	0.85	3.10								
C62	6.50	1.24	0.73	1.04	0.95	0.94	0.94	0.48	0.36	0.43	0.54	0.31	0.45	0.76	3.02	0.09							
C63	5.80	0.51	0.31	0.28	0.28	0.28	0.28	0.53	0.72	1.16	0.23	1.06	1.21	1.45	3.57	0.67	0.76						
C64	5.70	0.38	0.74	0.43	0.62	0.63	0.63	0.93	1.10	1.33	0.53	1.31	1.45	1.76	3.96	0.92	1.01	0.43					
C65	5.73	0.40	0.76	0.45	0.64	0.65	0.65	0.94	1.10	1.33	0.54	1.31	1.44	1.76	3.97	0.92	1.01	0.45	0.02				
C66	5.30	0.77	1.25	0.93	1.14	1.14	1.14	1.40	1.53	1.57	0.99	1.62	1.74	2.11	4.37	1.27	1.36	0.94	0.52	0.50			
C67	5	0.65	1.25	0.87	1.09	1.10	1.10	1.46	1.61	1.75	1.05	1.77	1.90	2.25	4.47	1.40	1.49	0.95	0.52	0.51	0.25		

Waktu Tempuh

Waktu tempuh merupakan waktu yang dibutuhkan oleh kendaraan dalam proses pendistribusian barang. Kecepatan kendaraan yang digunakan adalah 40km/jam. Rumus perhitungan waktu tempuh sebagai berikut:

$$\text{Waktu tempuh} = \frac{d_{(ij)}}{v} \times 60 \text{ menit}$$

Keterangan: d_{ij} : Jarak tempuh (km)
 v : Kecepatan kendaraan (km/jam)

Untuk melakukan kegiatan pengangkutan dan bongkar muat barang tersebut dibutuhkan waktu 0.15 menit tiap kartonnya sehingga perhitungan waktu total dapat diperoleh dengan menjumlahkan waktu tempuh dengan waktu bongkar muat tiap karton pada masing – masing outlet (Waktu tempuh total)

Algoritma dalam TSP

Pemecahan kasus ini digunakan salah satu algoritma dalam TSP yaitu *branch and bound*. Algoritma ini menggunakan pohon pencarian (*search tree*), setiap simpul di pohon merupakan representasi dari sejumlah kemungkinan solusi dari TSP, serta algoritam *cheapest insertion heuristic*. Untuk penyelesaian dengan ketentuan jumlah $n > 10$ maka algoritma tidak dapat dikerjakan secara manual sehingga dilakukan dengan bantuan *software WINQSB (Windows Quantitative System for Business)*.

Rute Reguler/ Rute Awal PT. Paris Jaya Mandiri

Rute reguler merupakan rute yang biasanya dilalui petugas distribusi.

Tabel 5. Hasil Data Reguler

Rute	Destinasi Outlet	Jumlah		
		Jarak (Km)	Waktu	Total Waktu
1	PJM - C1 - C2 - C3 - C4 - C5 - C6 - C7 - C8 - PJM	8	12.02'	39.02'
2	PJM - C9 - C10 - C11 - C12 - C13 - C16 - C17 - C18 - C19 - C20 - C21 - PJM	7.28	10.91'	46.16'
3	PJM - C22 - C23 - C24 - C25 - C26 - PJM	12.19	18.28'	28.33'
4	PJM - C14 - C27 - C28 - C15 - C29 - C30 - C31 - C32 - C33 - C34 - C35 - C36 - C37 - C38 - C39 - C40 - C41 - PJM	10.55	20.31'	50.59'
5	PJM - C42 - C46 - C43 - C44 - PJM	9.81	16.6'	33.31'
6	PJM - C45 - C56 - C47 - C48 - C49 - C50 - C51 - C52 - C53 - C54 - C55 - C57 - C58 - C59 - C60 - C61 - C62 - C63 - C64 - C65 - C66 - C67 - PJM	22.93	34.39'	87.1'
Total		70.76	107.32'	284.51'

Penentuan Rute Menggunakan Algoritma *Branch and Bound*

Berdasarkan 3 kelompok wilayah distribusi dilakukan penyusunan rute baru berdasarkan letak outlet yang berdekatan. Kelompok wilayah 1 sebanyak 24 outlet dapat diperoleh 2 rute distribusi, kelompok wilayah sebanyak 22 outlet diperoleh 2 rute distribusi dan kelompok wilayah 3 sebanyak 21 outlet dapat diperoleh 1 rute distribusi. Dari hasil pengolahan data diperoleh 5 rute optimal dari 3 wilayah distribusi (WD) berdasarkan *Branch and Bound* sebagai berikut:

Tabel 6. Hasil Pengolahan Data *Branch and Bound*

W D	Rute	Destinasi Outlet	Jumlah		
			Jarak (Km)	Waktu	Total Waktu
1	1	PJM - C7 - C6 - C8 - C2 - C1 - C5 - C4 - C13 - C12 - C21 - C3 - PJM	8.09	12.13'	46.03'
	2	PJM - C9 - C10 - C11 - C17 - C19 - C18 - C20 - C44 - C43 - C42 - C46 - C45 - C41 - PJM	9.47	14.2'	59.2'
2	3	PJM - C14 - C15 - C16 - C22 - C24 - C25 - C23 - C26 - PJM	10.88	16.3'	35.35'
	4	PJM - C28 - C29 - C35 - C34 - C37 - C38 - C39 - C40 - C36 - C33 - C32 - C31 - C30 - C27 - PJM	9.74	14.58'	43.08'
3	5	PJM - C47 - C49 - C50 - C52 - C51 - C48 - C53 - C54 - C60 - C59 - C58 - C55 - C57 - C62 - C61 - C56 - C63 - C65 - C64 - C66 - C67 - PJM	19.87	29.8'	83.05'
Total			58.05	87.01'	266.71'

Penentuan Rute Menggunakan Algoritma *Cheapest Insertion Heuristic*

Pada hasil olah data dengan metode ini ditemukan kesamaan hasil berupa total jarak dan urutan destinasi outlet dengan pengolahan data menggunakan metode *Branch and Bound*. Hasil olah data yang serupa tersebut terjadi pada rute 1, 3, 4, dan 5.

Selain itu dapat dilihat juga bahwa sesuai dengan lokasi hasil output rute ini juga mengalami kesamaan dengan lokasi pada rute 2 *Branch and Bound*, tetapi pembeda pada hasil ini dengan metode *Branch and Bound* adalah pada urutan outlet yang dituju, dimana outlet tersebut berada pada suatu alamat yang sama tetapi terletak pada titik lokasi yang berbeda.

Berikut merupakan tabel ringkasan pengolahan data rute terpendek dengan metode *Cheapest Insertion Heuristic*.

Tabel 7. Hasil Pengolahan Data *Cheapest Insertion Heuristic*

Rute	Destinasi Outlet	Jumlah		
		Jarak (Km)	Waktu	Total Waktu
1	PJM – C7 – C6 – C8 – C2 – C1 – C5 – C4 – C13 – C12 – C21 – C3 – PJM	8.09	12.13'	46.03'
2	PJM - C9 - C11 - C10 - C20 - C18 - C19 - C17 - C44 - C43 - C42 - C46 - C45 - C41 – PJM	9.55	14.33'	62.78'
3	PJM – C14 – C15 – C16 – C22 – C24 – C25 – C23 – C26 – PJM	10.88	16.3'	35.35'
4	PJM – C28 – C29 – C35 – C34 – C37 – C38 – C39 – C40 – C36 – C33 – C32 – C31 – C30 – C27 – PJM	9.74	14.58'	43.08'
5	PJM – C47 – C49 – C50 – C52 – C51 – C48 – C53 – C54 – C60 – C59 - C58 – C55 - C57 – C62 – C61 – C56 – C63 – C65 – C64 – C66 - C67 – PJM	19.87	29.8'	83.05'
Total		58.13	87.14'	270.29'

Analisa Perbandingan dengan Rute Reguler

Hasil perbandingan pada tabel diatas menunjukkan perubahan dari segi jarak akan mempengaruhi waktu tempuh and total waktu setiap outlet. Antara rute regular dan rute hasil *branch and bound* dapat diperoleh selisih jarak tempuh sebesar 12.71 km, selisih waktu tempuh sebesar 20.31 menit dan selisih total waktu adalah sebesar 22 menit. Sedangkan rute regular dengan rute hasil *cheapest insertion* dapat diperoleh selisih jarak tempuh sebesar 12.63 km, selisih waktu tempuh sebesar 20.18 menit dan selisih total waktu adalah sebesar 14.22 menit. Table 8 berikut menunjukkan persentase penghematan hasil olahan dengan rute regular perusahaan

Tabel.8 Persentase Penghematan Hasil Olahan Data Dengan Rute Reguler Perusahaan

Metode	Penghematan Jarak	Penghematan Waktu	Penghematan Total Waktu
<i>Branch and Bound</i>	17.96%	18.92%	6.27%
<i>Cheapest Insertion Heuristic</i>	17.85%	18.80%	4.89%

Dari hasil pengolahan data diatas menunjukkan algoritma *branch and bound* memberikan hasil rute yang paling optimal dibandingkan dengan algoritma *cheapest insertion heuristic*. Hasil persentase penghematan tersebut dapat memberikan keuntungan positif bagi perusahaan dimana terjadinya pengurangan jarak tempuh sebesar 17.96% yang bukan hanya mempengaruhi waktu tetapi juga berpengaruh pada penghematan biaya transportasi.

4. Simpulan

Dari hasil penelitian ini dapat ditarik kesimpulan sebagai berikut :

Rute optimal yang dapat ditempuh oleh perusahaan terdiri dari 5 rute yang telah diolah dengan *Branch and Bound* yaitu; rute 1 PJM – C7 – C6 – C8 – C2 – C1 – C5 – C4 – C13 – C12 – C21 – C3 – PJM (8.09 Km), rute 2 PJM – C9 – C10 – C11 – C17- C19 – C18 – C20 – C44 – C43 – C42 – C46 – C45 – C41 – PJM (9.47 Km), rute 3 PJM – C14 – C15 – C16 – C22 – C24 – C25 – C23 – C26 – PJM (10.88 Km), rute 4 PJM – C28 – C29 – C35 – C34 – C37 – C38 – C39 – C40 – C36 – C33 – C32 – C31 – C30 – C27 – PJM (9.74 Km), rute 5 PJM – C47 – C49 – C50 – C52 – C51 – C48 – C53 – C54 – C60 – C59 - C58 – C55 - C57 – C62 – C61 – C56 – C63 – C65 – C64 – C66 - C67 – PJM (19.87).

Total jarak tempuh minimum adalah 58.05 Km. Hasil persentase penghematan *Branch and Bound* menunjukkan penghematan jarak sebesar 17.96%, waktu tempuh sebesar 18.92% dan total waktu tempuh sebesar 6.27%.

Daftar Pustaka

- Bangun BJ Putra, Sisca Octarina, Bran Valbert Purba. (2015). Penyelesaian Travelling Salesman Problem (TSP) dengan Metode Branch and Bound, *Prosiding Semirata 2015 Bidang MIPA BKS-PTN Barat*, Universitas Tanjung Pura Pontianak, pp. 399-408.
- Eka, 2012. Penentuan Rute Distribusi Produk Minuman Ringan PT. Coca-Cola Distribution Indonesia DC Pontianak Menggunakan Metode Travelling Salesman Problem. Pontianak: FT-UNTAN.
- Gea Aristi (2014), Perbandingan algoritma *greedy*, algoritma *cheapest insertion heuristics* dan *dynamic programming* dalam penyelesaian *travelling salesman problem*. *Jurnal Paradigma*, Vol XVI, No. 2, pp. 52-58.
- Hillier, F & Lieberman, G.J (2001). *Introduction to Operation Research*. Seventh Edition. Mc Graw Hill . New York , USA.
- Hoffman, A.J. and Wolfe, P. (1985). "History" in *The Traveling Salesman Problem*, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, eds., John Wiley, pp. 1-16.
- Paillin, D. B dan Soseboko I (2017) Penentuan Rute Optimal Distribusi Produk Nestle dengan Metode TSP. *Jurnal ARIKA*, Vol. 11, No. 1, pp.31-40.
- Richard Wiener. (2003). *Branch and Bound Implementations for the Traveling Salesperson Problem - Part 1*, in *Journal of Object Technology*, Vol. 2, No. 2, March-April 2003, pp. 65-86.