

USULAN RUTE DISTRIBUSI MENGGUNAKAN ALGORITMA SWEEP DAN LOCAL SEARCH (STUDI KASUS DI PERUSAHAAN X)

Muhamad Ruben^{1*}, Arif Imran²

^{1,2}Program Studi Teknik Industri, Fakultas Teknologi Industri, Institut Teknologi Nasional
Jl. PHH Mustofa No 23, Bandung, 40124
*E-mail: muhamadruben87@gmail.com

Abstract

A company has to deliver its product to thirty-five points/customers. In delivering process, there are limitations such as the number of vehicles, vehicle capacity, and driver's working hours. The company wants to find a better/shorter route in distributing its product to customer in order to minimize the transportation cost. The problem can considered as the Capacitated Vehicle Routing Problem. In this paper, the Sweep algorithm are first used to find the initial solution. The local search procedures are used to the solution obtained by the Sweep algorithm. Four local searches are applied, 1-insertion intra-route, 1-insertion inter-route, 1-1 intra-route and 1-1 inter-route. The solutions produced by each local search are then compared, and the best one is selected. The best solution is given by the 1-Insertion intra-route procedure. It is 27,923 km shorter (15% better) than the current routes.

Keywords : *Vehicle Routing Problem, Sweep Algorithm, local search*

1. Pendahuluan

Jaringan distribusi dan transportasi memungkinkan produk pindah dari satu lokasi ke lokasi lainnya, dimana kemampuan mengirimkan produk ke konsumen tepat waktu dengan jumlah yang sesuai akan menentukan apakah produk tersebut akan kompetitif di pasar. Oleh karena itu, kemampuan untuk mengelola jaringan distribusi satu keunggulan kompetitif yang sangat penting. (Pujawan dan Mahendrawathi, 2017)

Industri Makanan Sizi merupakan sebuah usaha dengan produk yang dimilikinya adalah nasi tim ayam. Proses pendistribusian yang dilakukan oleh perusahaan hanya mencakup daerah-daerah yang ada di Kota Bandung dan Kabupaten Bandung. Pendistribusiannya dilakukan ke titik-titik *outlet* yang telah bekerja sama dengan perusahaan untuk menjual produk ke konsumen. Permintaan yang ada untuk setiap titik berbeda-beda dan kendaraan yang dimiliki memiliki kapasitas angkut yang terbatas sehingga diperlukannya penentuan rute distribusi yang optimal.

Proses pengiriman yang dilakukan selalu dimulai dari tempat produksi dan berakhir di tempat produksi. Pola pendistribusian yang berdasarkan pengalaman pengemudi dan pengelompokkan berdasarkan wilayah belum pasti memberikan rute terpendek, sehingga menyebabkan jarak tempuh kendaraan dapat menjadi lebih jauh dan waktu yang lebih lama.

Perusahaan memiliki banyak *outlet* yang setiap harinya harus didatangi untuk pengiriman produk. Masing-masing *outlet* tersebut memiliki permintaan yang berbeda-beda. Proses pengiriman dilakukan menggunakan kendaraan roda dua berupa motor berjumlah 4 motor yang setiap motor memiliki kapasitas 100 mangkok. Dalam proses pendistribusiannya perusahaan harus melakukan pendistribusian ke 35 titik dengan jumlah permintaan 685 mangkok untuk setiap harinya dan memiliki keterbatasan mulai dari jumlah kendaraan, kapasitas kendaraan, dan jam kerja pengemudi. Permasalahan perusahaan dapat dikategorikan dalam *Capacitated Vehicle Routing Problem (CVRP)*.

2. Landasan Teori

Menurut Toth dan Vigo (2002), terdapat banyak varian VRP, diantaranya, VRP *with Backhauls (VRPB)*, VRP *Split Delivery (VRPSD)*, VRP *Time Windows (VRPTW)*, VRP *Multiple Depot (VRPMD)*, VRP *Pick Up and Delivery (VRPPD)*, *Capacitated VRP (CVRP)*, *Stochastic VRP (SVRP)*, *Periodic VRP (PVRP)*, *Heterogeneous Fleet Vehicle Routing Problem (HFVRP)*, VRP *Multiple Trips (VRPMT)* dan *Multi-depot VRP (MDVRP)*. Review tentang VRP dan varian-varianya diberikan oleh Brackens et al (2016).

CVRP dapat diselesaikan dengan menggunakan

dua jenis pendekatan, yaitu metode eksak dan pendekatan heuristik. Pendekatan eksak memberikan solusi optimal akan tetapi metode eksak mempunyai permasalahan waktu perhitungan yang lama untuk mendapatkan solusi jika ukuran persoalan besar. Untuk itu dikembangkan metode heuristik/metaheuristik.

Banyak metode heuristik/metaheuristik yang telah dikembangkan untuk menyelesaikan masalah CVRP, sebagai contoh Algoritma Tabu Search (Barbazoglu & Ozgur, 1999), Variable Neighbour Search (Imran & Okdinawati 2012, Sze et al 2016), Algoritma Threshold Accepting (Tarantilis et al, 2002), algorithm Bee (Gomez et al 2013), Algoritma Ant Colony (Yu et al, 2009) dan lain lain. Review tentang CVRP dapat dilihat di Braekers et al 2016.

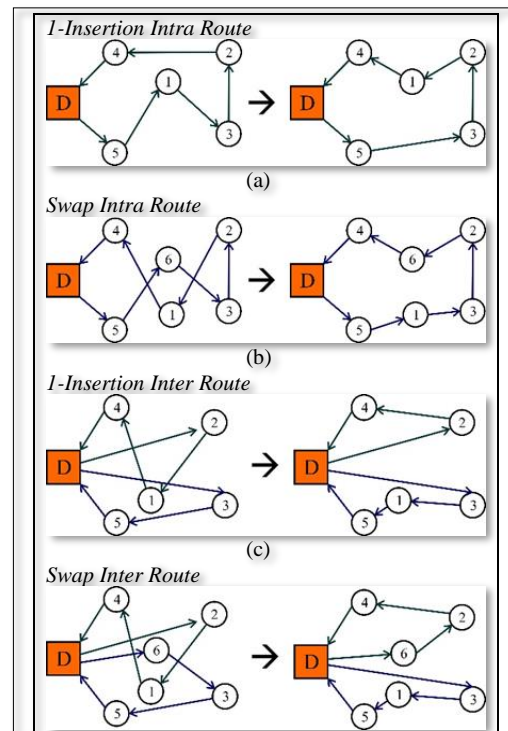
Algoritma *Sweep* merupakan suatu algoritma menggunakan metode dua fase dengan fase pertama berupa *clustering* pelanggan berdasarkan wilayah dan kendaraan yang tersedia, dan fase dua berupa membangun rute-rute untuk tiap *cluster*. Proses menyelesaikan model VRP dengan menggunakan algoritma *Sweep* diperlukan dua tahapan atau fase yaitu fase pengelompokan (*clustering*) dan fase pembentukan rute (Saraswati, R. et al., 2017):

1. Tahap pengelompokan (*clustering*) langkah-langkah yang dilakukan pada tahap pengelompokan adalah:
 - a. Menentukan posisi tiap konsumen dalam koordinat kartesius dan menetapkan lokasi depot sebagai pusat koordinat.
 - b. Menentukan seluruh koordinat polar tiap agen dengan depot awal.
 - c. Membentuk pengelompokan (*clustering*) dimulai dari konsumen yang memiliki sudut polar terkecil hingga terbesar dengan mempertimbangkan kapasitas kendaraan.
 - d. Memastikan semua konsumen yang terlibat telah dikelompokkan dalam *cluster* ini.
 - e. Pengelompokan dihentikan apabila terdapat satu *cluster* akan melebihi kapasitas maksimal kendaraan.
 - f. Jika hal tersebut terjadi maka dilakukan pembuatan *cluster* baru seperti langkah sebelumnya.
2. Tahap pembentukan rute distribusi, Tahap pembentukan rute distribusi, dari tiap *cluster* akan diselesaikan dengan algoritma *Nearest Neighbour* sehingga didapatkan diperoleh rute perjalanan dari tiap *cluster*. Langkah-langkahnya adalah sebagai berikut:
 - a. Langkah inisiasi dapat dilihat sebagai berikut:
 1. Menentukan satu titik sebagai titik awal perjalanan yaitu dari depot perusahaan.
 2. Menentukan himpunan titik (C) yang akan dikunjungi oleh kendaraan.
 3. Menentukan urutan rute distribusi sementara.
 - b. Memilih titik selanjutnya yang dikunjungi

kendaraan. Jika adalah titik di urutan terakhir dari rute R maka titik berikutnya yang memiliki jarak paling minimum dengan , dimana adalah anggota dari C. Apabila banyak pilihan optimal berarti terdapat lebih dari satu titik dengan jarak yang sama dari titik terakhir dalam rute R dan jarak tersebut merupakan jarak yang paling minimum maka pilih secara acak.

- c. Menambahkan titik terpilih untuk urutan rute berikutnya. Menambahkan titik pada urutan akhir rute sementara dan mengeluarkan titik yang terpilih dari daftar titik yang belum dikunjungi.
- d. Apabila semua titik telah dilewati selanjutnya dilakukan penutupan rute dengan menambahkan titik inisiasi atau titik awal perjalanan di akhir rute.

Algoritma *local search* merupakan algoritma yang digunakan untuk mendapatkan rute yang lebih pendek. Beberapa algoritma *Local search* yang digunakan dalam penelitian ini adalah *1-Insertion Intra Route*, *Swap Intra Route*, *1-Insertion Inter Route* dan *Swap Inter Route*. Algoritma-algoritma tersebut telah diaplikasikan sebelumnya untuk menyelesaikan berbagai jenis VRP. Sebagai contoh aplikasi algoritma-algoritma tersebut dapat dilihat di Imran et al 2016, Salhi et al (2014) dan Imran (2013). Algoritma *local search* yang digunakan dapat dilihat pada Gambar 1.



Gambar 1. Visualisasi Algoritma *Local Search*

Pada algoritma *1-Insertion Intra Route*, perbaikan rute dilakukan dengan cara melakukan memindahkan 1 konsumen ke tempat yang lain dalam

satu rute. Pada Gambar 1(a), pada awalnya urutan dalam rute adalah D-5-1-3-2-4-D. Dengan memindahkan konsumen 1 diantara konsumen 2 dan konsumen 4 didapatkan total jarak yang lebih pendek. Rutenya menjadi D-5-3-2-1-4-D. Algoritma *Swap Intra-Route* adalah algoritma perbaikan rute dilakukan dengan cara melakukan pertukaran posisi dua titik dalam rute yang sama. Sebagai contoh, pada Gambar 1(b) rute awal adalah D-5-6-3-2-1-4-D. Dengan menggunakan algoritma *Swap Intra-Route*, posisi konsumen 1 dan konsumen 6 ditukar (*swap*). Sehingga didapat rute baru D-5-1-3-2-6-4-D yang memiliki total jarak yang lebih pendek. Algoritma *1-Insertion Inter Route* adalah algoritma perbaikan rute dilakukan dengan cara melakukan penyisipan titik antara rute yang berbeda (Gambar 1(c)). Dalam algoritma *Swap Inter-Route* perbaikan rute dilakukan dengan cara melakukan pertukaran posisi dua titik antara rute yang berbeda (Gambar 1(d)).

3. Metodologi Penelitian

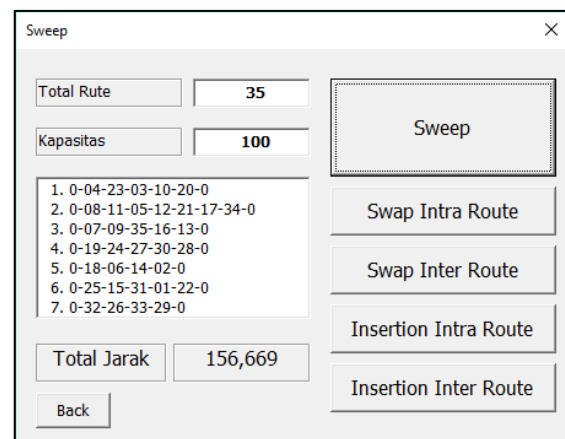
Pengumpulan data pada penelitian ini adalah data pelanggan mencakup alamat dan data permintaan, jumlah kendaraan, kapasitas kendaraan, koordinat polar, dan jarak tempuh. Proses pengumpulan data yang dilakukan pada penelitian ini dilakukan adalah wawancara dengan perusahaan dan menggunakan bantuan aplikasi dari *Google Maps* untuk mencari jarak antar outlet dan aplikasi *Geogebra* untuk mengetahui nilai titik koordinat polar.

Pada penelitian ini ada beberapa tahapan dalam proses penyelesaian masalahnya yang pertama penentuan koordinat depot dan pelanggan yang dikunjungi, Setelah proses pembentukan rute menggunakan algoritma *Sweep* dan algoritma *Nearest Neighbour*. Hasil yang diperoleh kemudian digunakan sebagai input untuk empat prosedur local search. Hasil dari tiap prosedur kemudian dibandingkan dan dipilih hasil yang terbaik. Algoritma/langkah-langkah untuk pencarian solusi sebagai berikut

1. Tentukan koordinat depot dan konsumen.
3. Hitung jarak antara konsumen dan jarak antara konsumen dan depot.
4. Gunakan algoritma *Sweep* dan algoritma *Nearest Neighbor* untuk mendapatkan solusi, *s*.
5. Gunakan masing-masing local search untuk memperbaiki solusi, *s*.
6. Bandingkan solusi yang didapatkan oleh masing-masing *local search*. Pilih yang terbaik.
7. Jika solusi yang dipilih, s' , lebih baik dari *s*, maka solusi yang digunakan adalah s' Jika tidak, gunakan *s* sebagai solusi.

4. Pembahasan

Pada penelitian ini, dirancang program lunak menggunakan *Software* Microsoft Excel dengan *tools Visual Basic*. Program ini berfungsi untuk mencari rute terbaik dan mengetahui total jarak yang dibutuhkan dengan input yang telah disiapkan sebelumnya. Dengan program ini dapat terlihat perbedaan untuk setiap *output* yang dihasilkannya dan mempersingkat waktu pengerjaan yang dibutuhkan. Contoh *user interface* perangkat lunak ini dapat dilihat pada Gambar 2.



Gambar 2. User Interface

Dari data input yang didapat dilakukan pengolahan data menggunakan Algoritma *Sweep* dapat dilihat pada Tabel 1.

Tabel 1. Hasil perhitungan Menggunakan Algoritma *Sweep*

Urutan Rute	Total Jarak (km)	Waktu Tempuh (menit)	Muatan (mangkok)
0-20-03-23-04-10-0	21.190	74.87	100
0-12-17-34-21-05-11-08-0	15.269	59.32	95
0-09-07-35-13-16-0	26.024	86.62	100
0-28-27-30-24-19-0	14.497	58.62	100
0-18-06-02-14-0	19.440	69.45	95
0-25-31-15-01-22-0	29.050	93.97	100
0-26-32-33-29-0	31.199	98.02	95
Total	156.669	540.862	685

Berdasarkan hasil tersebut terdapat tujuh rute dengan total jarak 156,669. Hasil ini kemudian diperbaiki menggunakan algoritma *Local Search*. Hasilnya dapat dilihat pada Tabel 2. Dari Tabel 2 dapat dilihat bahwa hasil terbaik diberikan oleh prosedur *1-Insertion Intra Route* dengan jarak 153,247.

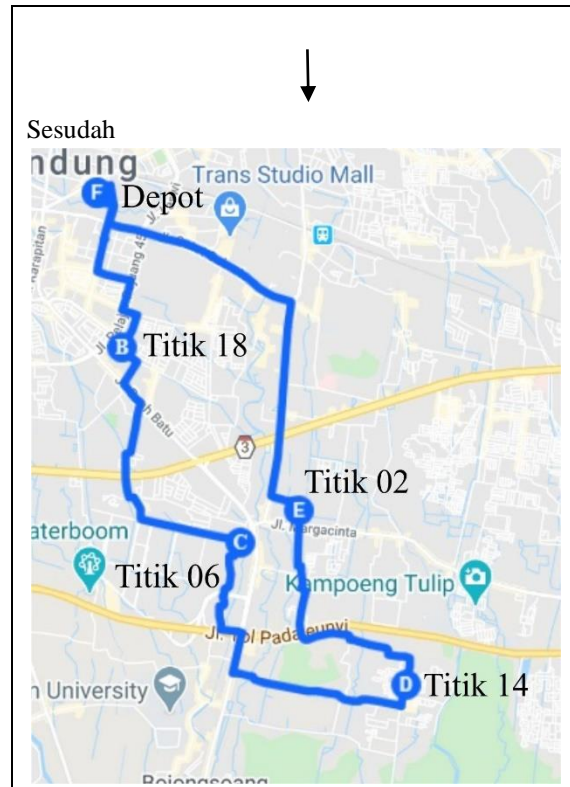
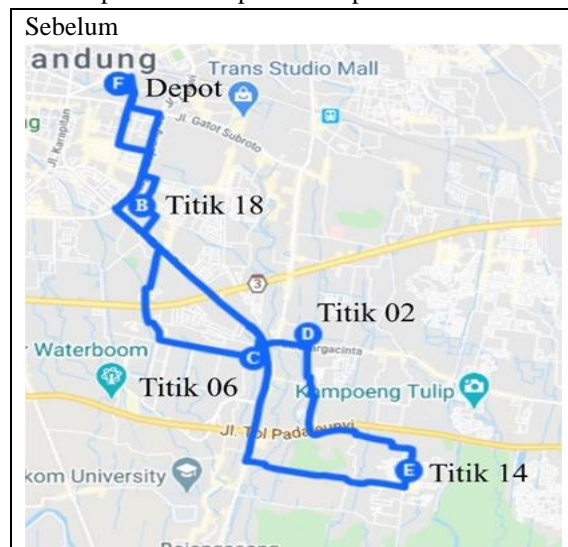
Tabel 2. Hasil Perbaikan Menggunakan Algoritma *Local Search*

<i>Local Search</i>	Total Jarak (km)
<i>Insertion Intra Route</i>	153.247
<i>Swap Intra Route</i>	153.658
<i>Insertion Inter Route</i>	156.669
<i>Swap Inter Route</i>	155.029

Tabel 3. Rute Setelah Diperbaiki

Urutan Rute	Total Jarak (km)	Waktu Tempuh (menit)	Muatan (mangkok)
0-20-03-23-04-10-0	21.190	74.87	100
0-12-17-34-21-05-08-11-0	15.268	59.32	95
0-09-35-13-16-07-0	25.540	85.44	100
0-28-30-24-19-27-0	12.690	54.23	100
0-18-06-14-02-0	18.310	66.71	95
0-25-31-15-01-22-0	29.050	93.97	100
0-26-32-33-29-0	31.199	98.02	95
Total	153.247	532.55	685

Gambar perbandingan salah satu rute sebelum dan sesudah perbaikan dapat dilihat pada Gambar 3.



Gambar 3. Perbandingan Rute

Perbaikan yang dilakukan menggunakan *Insertion Intra Route* menghasilkan jarak 153,347 km.

5. Kesimpulan

Penelitian ini bertujuan untuk meminimasi total jarak yang harus ditempuh. Algoritma Sweep, algoritma Nearest Neighbor dan local search diaplikasikan untuk mendapatkan solusi. Total jarak yang didapat oleh perusahaan saat ini adalah sebesar 181,17 km. Sedangkan total jarak yang dihasilkan menggunakan algoritma usulan kemudian diperbaiki menjadi 153,247 km. Hal ini menunjukkan bahwa algoritma yang diusulkan dapat menyelesaikan permasalahan yang terdapat pada perusahaan yaitu mengurangi total jarak sebesar 27,923 km (15%).

Daftar Referensi

G. Barbazoglu & D. Ozgur. (1999). A Tabu Search for the Vehicle Routing Problem. *Computers & Operations Research* 26, 255-270.

K. Braekers, K. Ramaekers & I.V. Nieuwenhuyse. (2016). The Vehicle Routing Problem: State of the Art Classification and Review. *Computers & Industrial Engineering*, 99, 300-313.

A. Gomez, A. Imran & S. Salhi. (2013). Solution of

- classical transport problems with bee algorithms. *International Journal of Logistics Systems and Management*, 15 (2-3), 160-170
- A. Imran. (2013). A variable neighborhood search-based heuristic for the multi-depot vehicle routing problem. *Jurnal Teknik Industri*, 15 (2), 95-102.
- A. Imran & L. Okdinawati. (2012). Adaptation of the variable Neighborhood search heuristic to solve the vehicle routing problem. *Jurnal Teknik Industri* 12 (1), 2012, pp.10-15
- A. Imran, M. Luis & L. Okdinawati. (2016). A Variable Neighborhood Search for the Heterogeneous Fixed Fleet Vehicle Routing Problem. *Jurnal Teknologi* ,78 (9), 53-58, 2016
- I.N. Pujawan & M. Er. (2017).”Supply Chain Management”, Andi, *Edisi Ketiga*. Yogyakarta. 216
- S. Salhi, A. Imran & N.A. Wassan. (2014). The multi-depot vehicle routing problem with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research* 52, 315-325.
- R. Saraswati, M. Hisjam & W. Sutopo. (2017). Penyelesaian *Capacitated Vehicle Routing Problem* dengan Menggunakan Algoritma *Sweep* untuk Penentuan Rute Distribusi Koran : Studi Kasus. *Jurnal Manajemen Pemasaran*, 11(2)
- J.F. Sze, S. Salhi, and N.A. Wassan. (2016). A hybridization of adaptive variable neighborhood search and large neighborhood search: Application to the vehicle routing problem," *Expert Systems & Applications*, 65, 383-397.
- C.D. Tarantilis, C.T. Kiranoudis & V.S. Vassiliadis. (2002). A Backtracking Adaptive Threshold Accepting Metaheuristic Method for the Vehicle Routing Problem. *System Analysis Modelling Simulation*, 42, 631-664.
- P. Toth & D. Vigo. (2002). “The Vehicle Routing Problem”, Society for Industrial and Applied Mathematics, *Second Edition*. Philadelphia. 1-26.
- B. Yu, Z. Yang & B. Yao. (2009). An Improved Ant Colony Optimization for Vehicle Routing Problem. *European Journal of Operational Research*, 196,171-176.