

Algoritma Differential Evolution (DE) Dalam Optimalisasi Rute Distribusi Produk Nestle (Studi Kasus: PT. Paris Jaya Mandiri)

Daniel B. Paillin^{*1)}, Johan M Tupan²⁾, Rizki Anggraeni Utami Putri S³⁾

¹⁾Jurusan, Teknik Industri, Universitas Pattimura, Kampus Poka, Ambon, 97233, Indonesia

Email: daniel.paillin@fatek.unpatti.ac.id, johan.tupan@fatek.unpatti.ac.id, aini.utami15@gmail.com

ABSTRAK

Tujuan penelitian ini adalah menyelesaikan *vehicle routing problem* menggunakan algoritma *differential evolution* untuk mendapatkan rute optimum berdasarkan total jarak dan total waktu tempuh yang minimal. Contoh kasus nyata persoalan ini yaitu menentukan rute kendaraan pendistribusian produk Nestle pada PT. Paris Jaya Mandiri yang ada di kota Ambon. Hasil penelitian memperlihatkan bahwa algoritma *differential evolution* mampu memberikan solusi yang terbaik dengan penghematan jarak sebesar 19,68% dan penghematan total waktu sebesar 48,22% dari rute awal perusahaan.

Kata kunci : *Differential Evolution*, Rute Optimum, *Vehicle Routing Problem*

1. Pendahuluan

Transportasi merupakan salah satu faktor yang memegang peranan penting dalam aktifitas logistik, terutama dalam hal kegiatan distribusi barang dari produsen ke konsumen akhir. Bowersox (2002) menyatakan bahwa pengeluaran terbesar dalam unsur-unsur biaya dari operasi logistik adalah biaya transportasi. Untuk menurunkan biaya, perusahaan akan berusaha sebisa mungkin agar dapat melakukan proses pendistribusian dengan optimal dan seefisien mungkin, karena dengan demikian perusahaan dapat meminimumkan biaya distribusi, jarak tempuh maupun waktu yang diperlukan dalam melakukan proses distribusi.

PT. Paris Jaya Mandiri adalah salah satu perusahaan distributor produk Nestle di Ambon dimana wilayah pendistribusian meliputi kecamatan Sirimau dan kecamatan Nusaniwe dengan jumlah outlet sebanyak 67 yang tersebar pada 39 titik lokasi yang berbeda. Banyaknya jumlah outlet menyebabkan munculnya banyak pilihan rute yang digunakan oleh perusahaan. Dalam melakukan proses distribusi, perusahaan belumlah bisa menemukan rute yang pasti untuk meminimumkan total jarak tempuh untuk 67 outlet tersebut. Padahal jika total jarak tempuh diminimumkan maka proses distribusi produk tersebut dapat dilakukan dengan optimal dan efisien. Hal tersebut juga pastinya dapat meminimumkan biaya dan waktu dalam melakukan proses pendistribusian.

Permasalahan diatas lebih dikenal dengan nama *Vehicle Routing Problem* (VRP) yang didefinisikan sebagai penentuan rute optimal dengan ongkos optimal dari satu depot ke pelanggan yang lokasinya tersebar dengan jumlah permintaan diketahui diketahui (Bräysy, 2003). Lawler et,al (1983) VRP merupakan permasalahan yang sulit karena termasuk *hard combinatorial problem*, sehingga banyak teknik baik optimal maupun heuristik yang dikembangkan untuk memecahkan permasalahan ini. Paillin dan Tupan (2018) menggunakan algoritma *branch and bound* dan *cheapest insertion* untuk pemecahan penentuan rute optimal distribusi produk nestle dikota ambon dengan cara membandingkan performansi solusi dari masing-masing teknik. Paillin dan Watimena (2015) menyelesaikan VRP menggunakan algoritma *sequential insertion* untuk mendapatkan rute optimum dalam pendistribusian Bahan Bakar Minyak (BBM) pada kawasan timur Indonesia. Fradina dan Saptaningtyas (2017) membandingkan performansi dari *sweep algorithm* dan *genetic algorithm* dalam penyelesaian VRP untuk optimalisasi kegiatan distribusi gula di Yogyakarta.

Banyak metode dan algoritma yang dapat digunakan untuk menyelesaikan masalah VRP, dimana setiap algoritma mempunyai keunggulan dan kekurangan tersendiri. Suatu algoritma memiliki hasil yang berbeda-beda untuk kasus yang berbeda. Dengan kata lain bahwa suatu

algoritma yang memiliki nilai optimasi yang tinggi untuk suatu kasus, belum tentu akan memiliki nilai optimasi yang tinggi juga untuk kasus yang berbeda (Paillin dan Tupan, 2018). Sehingga tujuan penelitian ini adalah mendapatkan rute yang optimum untuk pendistribusian produk nestle di kota Ambon berdasarkan jarak tempuh yang optimal dan waktu tempuh yang optimal.

Berdasarkan masalah tersebut, penulis tertarik untuk membahas optimalisasi rute distribusi produk nestle dengan algoritma *Differential Evolution* (DE) sehingga distribusi produk dapat menjadi lebih efisien.

2. Metode

Vehicle Routing Problem (VRP)

VRP pertamakali diperkenalkan oleh Dantzig dan Ramser tahun 1959 dalam bentuk rute dan penjadwalan truk. Clark dan Wright tahun 1964 kemudian mengembangkan penelitian VRP dengan memperkenalkan istilah depot sebagai tempat keberangkatan dan kembalinya kendaraan. Clarke dan Wright menggunakan *saving algorithm*, sejak saat itu penelitian VRP terus berkembang karena peran VRP yang penting dalam distribusi dunia industri. VRP standar dapat dijelaskan sebagai berikut: terdapat sebuah depot dan beberapa pelanggan dengan lokasi dan permintaan diketahui. VRP bertujuan untuk menentukan beberapa rute yang meminimumkan fungsi tujuan dengan tetap memenuhi seluruh permintaan pelanggan. Sebuah rute mencakup urutan mengunjungi pelanggan dengan kendaraan yang berangkat dan berakhir ke depot. Total permintaan semua pelanggan dalam satu rute tidak boleh melebihi kapasitas kendaraan yang digunakan. Setiap rute ditunjukkan oleh satu kendaraan yang mengunjungi pelanggan sebanyak satu kali. Karena keterbatasan pada kapasitas kendaraan, VRP standar sering disebut juga sebagai *capacitated vehicle routing problem*.

Diketahui sebuah jaringan $G = (N,L)$ diman N menunjukkan sekumpulan *node* $N = \{0,1,...,67\}$ dan $L = \{(i,j); i,j \in N, i \neq j\}$ menunjukkan himpunan *arc* (*link*). *Node* 0 menunjukkan depot dimana terdapat sejumlah NV kendaraan. Matriks jarak $D = d_{ij}$ didefinisikan pada L . jika $d_{ij} = d_{ji}$ untuk semua (i,j) maka permasalahan dapat dikatakan simetri dan *arc* merepresentasikan busur yang tidak berarah (*undirected arcs / edge*). Permintaan pelanggan i dinyatakan dengan q_i dan jumlah *demand* pelanggan pada satu rute tidak melebihi kapasitas kendaraan Q^k . Tujuan dari VRP dasar ini adalah penentuan rute NV kendaraan yang memberikan total jarak tempuh minimal.

Berikut adalah formulasi model matematik CVRP dinyatakan sebagai berikut:

$$\text{Minimasi } \sum_i \sum_j \sum_k d_{ij} x_{ij}^k \quad (1)$$

Dengan batasan

$$\sum_i \sum_j x_{ij}^k = 1, \text{ untuk semua } j \quad (2)$$

$$\sum_i x_{ip}^k - \sum_j x_{pj}^k = 0, \text{ untuk semua } p, k \quad (3)$$

$$\sum_i q_i - \left(\sum_j x_{ij}^k \right) \leq Q^k, \text{ untuk semua } k \quad (4)$$

$$\sum_{j=1}^n x_{oj}^k \leq 1, \text{ untuk semua } k \quad (5)$$

$$y_i - y_j + n \sum_{k=1}^{NV} x_{oj}^k \leq n-1, \quad i \neq j, \quad i \neq 0, j \neq 0 \quad (6)$$

$$x_{ij}^k \in \{0,1\}, \text{ untuk semua } i, j \text{ dan } k \quad (7)$$

$$y_i \text{ arbitrary} \quad (8)$$

Fungsi tujuan yang meminimasi total jarak ditunjukkan pada persamaan (1). Pembatas (2) menjamin bahwa setiap node hanya dilalui oleh satu kendaraan. Pembatas (3) menjamin bahwa kendaraan yang meninggalkan node i telah melayani node i tersebut. Pembatas kapasitas kendaraan dinyatakan dengan pertidaksamaan (4). Pembatas (5) menjamin bahwa kendaraan yang ditugaskan tidak melebihi NV kendaraan. Pembatas (6) berperan sebagai eliminasi subtur sehingga rute kendaraan selalu dimulai dari depot dan berakhir di depot.

Differential Evolution

Differential evolution adalah satu dari beberapa metode metaheuristik yang pemakaiannya cukup luas dalam bidang rekayasa (Santosa dan Jin Ai, 2017). DE termasuk metode pencarian stokhastik dan berdasarkan populasi. DE mempunyai kesamaan dengan *evolutionary algorithms* (EA) yang lain, tetapi berbeda dalam hal informasi jarak dan arah dari populasi yang sekarang digunakan untuk memandu proses pencarian solusi yang lebih baik. Seperti halnya semua keluarga Evolutionary Algorithm (EA), DE juga didasarkan pada pembangkitan populasi titik-titik untuk mencapai minimum suatu fungsi. DE agak berbeda dengan algoritma EA dalam hal:

- Mutasi diterapkan dulu untuk membangkitkan vector percobaan (trial vector), yang kemusian akan digunakan dalam proses crossover untuk menghasilkan satu turunan (offspring)
- Step size dalam mutasi tidak disampel dari distribusi populasi yang sudah diketahui.

Dalam DE, step size dalam mutase dipengaruhi oleh perbedaan diantara individu dalam populasi sekarang. Titik-titik ini akan disampling secara acak sebagai titik awal. Kemudian perlu dilakukan pembatasan nilai-nilai (batas atas maupun batas bawah) dimana kira-kira nilai variable yang dicari akan berada. Pembangkitan populasi awal tentu saja harus memperhatikan batas bawah dan batas atas ini. Setiap vector (dari titik-titik dalam populasi) diberi indeks untuk menandai posisinya. DE membangkitkan titik baru berdasarkan titik yang ada dan selisih antara dua titik dalam populasi. Ini berbeda dengan metoda direct search seperti Nelder Mead yang membangkitkan titik baru dengan cara refleksi atau melakukan sampling dari suatu distribusi tertentu seperti dalam ES.

Berikut ini adalah langkah-langkah dalam DE (Nurdiansyah R, 2016).

1. Inisialisasi

Sebelum dilakukan inisialisasi vector variable yang dicari, perlu dilakukan penentuan batas atas (ub) dan batas bawah (lb). Batas bawah dan batas atas akan digunakan sebagai langkah awal pembangkitan nilai variable yang dicari. Untuk pembangkitan nilai awal variable generasi ke 0, variable ke j dan vector i bias diwakili dengan notasi berikut

$$x_{j,i,0} = lb_j + rand_j(1)(ub_j - lb_j) \quad (9)$$

Bilangan random dibangkitkan dengan fungsi rand, dimana bilangan yang dihasilkan terletak antara (0, 1). Indeks j menunjukkan variable ke j . Dalam kasus minimasi fungsi dengan 2 variabel, maka j akan bernilai 1 dan 2. Penentuan batas atas dan batas bawah sangat tergantung pada masalahnya dan pertimbangan user. Jika nilai yang dicari sulit ditentukan posisinya maka rentang batas atas dan batas bawah bias dibuat lebih lebar.

2. Mutasi

Setelah diinisialisasi, DE akan memutasi dan mengkombinasi populasi awal untuk menghasilkan populasi dengan ukuran N vector percobaan. Dalam DE, mutase dilakukan dengan cara menambahkan perbedaan dua vector terhadap vector ketiga dengan cara

$$v_{i,g} = x_{r0,g} + f(x_{r1,g} - x_{r2,g}) \quad (10)$$

Kita lihat bahwa perbedaan dua vector yang dipilih secara random perlu diskala dulu sebelum ditambahkan ke vector ketiga, $x_{r0,g}$. Faktor skala $F \in (0,1+)$ bernilai riil positif untuk mengendalikan tingkat pertumbuhan populasi. Biasanya, walaupun nilai F berupa bilangan riil positif berapapun, menurut pengalaman nilai (0,1) menghasilkan kinerja yang bagus. Indeks

vector basis, r_0 , dapat ditentukan dengan berbagai cara. Tetapi di sini digunakan cara random yang berbeda dengan indeks untuk vector target, i . Selain berbeda satu sama lain dan berbeda dengan indeks untuk vector basis dan vector target, indeks vector selisih r_1 dan r_2 , juga dipilih sekali permutan.

3. Crossover

Crossover yang dimaksud di sini berbeda dengan crossover dalam Genetic Algorithm. Pada tahap ini DE menyilangkan setiap vector, $x_{i,g}$, dengan mutan, $v_{i,g}$, untuk membentuk vector hasil persilangan, $u_{i,g}$ dengan formula.

$$u_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g}, & \text{jika } rand \leq Cr \\ v_{j,i,g}, & \text{untuk lainnya} \end{cases} \quad (11)$$

Probabilitas crossover, $Cr \in (0,1)$ adalah nilai yang didefinisikan untuk mengendalikan fraksi nilai variable yang disalin dari mutan.

4. Seleksi

Seleksi di sini adalah seleksi antara dua vector. Jika trial vector, $u_{i,g}$, mempunyai nilai fungsi tujuan yang lebih kecil dari fungsi tujuan vektor targetnya, $x_{i,g}$, maka $u_{i,g}$ akan menggantikan posisi $x_{i,g}$ dalam populasi pada generasi berikutnya. Jika terjadi sebaliknya, vector target akan tetap pada posisinya dalam populasi.

Berikut ini adalah Pseudocode dari algoritma DE

Input: N, D, tmax, LB, UB, Cr

Output: Nilai $x, f(x)$

foreach $I = 1 : N$ **do**

$x(i) = LB + rand(UB-LB)$

Hitung nilai fungsi tujuan $f(x_i)$

end

while $t < tmax$ **do**

foreach $i = 1 : N$ **do**

$r_0 = rand, r_1 = rand$

if $r_0 = i \ \& \ r_2 = r_1 = r_0$ **then**

break;

else

$i = x_{r_0} + F(x_{r_1} - x_{r_2})$

end

if $v_i < LB$ **then**

$v_i = LB$

end

if $v_i > UB$ **then**

$v_i = UB$

end

if $rand < Cr$ **then**

$u_i = v_i$

else

$u_i = x_i$

end

if $f(v_i) < f(x_i)$ **then**

$x_i = v_i$

else

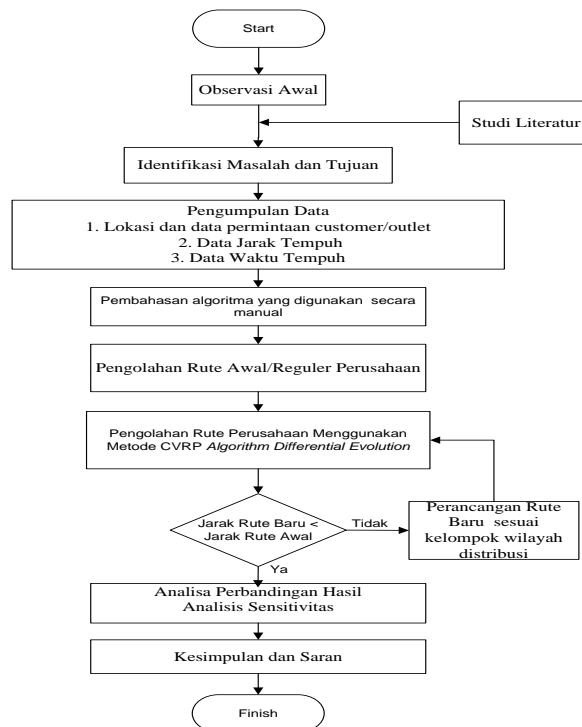
$x_i = x_i$

end

end

end

Berikut ini adalah tahapan penelitian yang dilakukan dapat dilihat pada gambar 1. berikut ini:

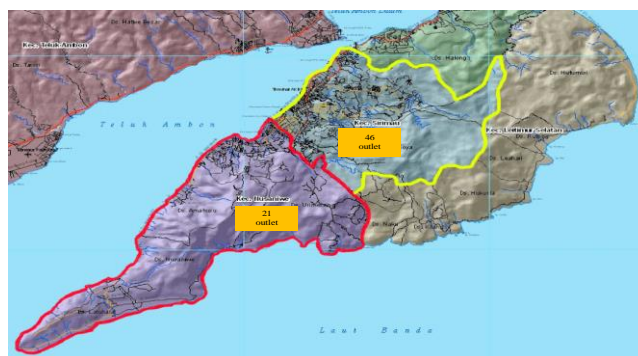


Gambar 1. Diagram Alir Penelitian

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 2. 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	Blitar	Pertokoan Batu Merah	31	C39	Berkat	Trikora	9
C6	Manna Toko	Galala	26	C40	Maluku Jaya Abadi Toko	Diponegoro (Kel. 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 Ivana	Jl. Pantai Mardika	16	C51	Darren	Dr. Kayadoe	39
C18	Apotek	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	Lampat Jaya	Jl. Pantai Mardika	19	C55	Mikael Toko	Gudang Arang	11
C22	Teki Toko	Karpan	15	C56	Annyong 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	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*.

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)

Rute Reguler/ Rute Awal PT. Paris Jaya Mandiri

Rute awal perusahaan merupakan rute yang biasanya dilalui oleh petugas distribusi. Untuk pengolahan jarak pada rute tersebut dilakukan berdasarkan rute perusahaan yang telah ada dimulai dari gudang menuju outlet pertama, dan dilanjutkan ke outlet berikutnya sampai ke outlet yang terakhir sesuai dengan urutan yang telah ditetapkan oleh perusahaan. Saat ini perusahaan memiliki 6 rute distribusi yang biasanya dilalui oleh petugas distribusi. Hasil pengolahan data rute reguler perusahaan diperoleh dengan menjumlahkan jarak tempuh dan

menjumlahkan waktu tempuh yang dialui oleh petugas distribusi berdasarkan data jarak dan waktu tempuh . Sedangkan untuk waktu total (Wt) didapat dari jumlah waktu tempuh dengan waktu bongkar muat barang pada setiap outlet.

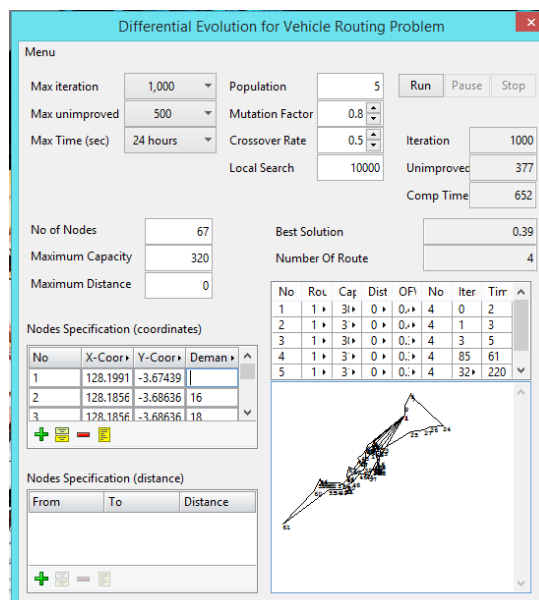
Tabel 2. Hasil Data Reguler

Rute	Destinasi Outlet	Jumlah		
		Jarak (Km)	Waktu (Menit)	Total Waktu (Menit)
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	13.09	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

Hasil dan Analisa Algorithm Differential Evolusion Dalam Penentuan Rute Distribusi Produk Nestle Pada PT. Paris Jaya Mandiri.

VRP merupakan permasalahan penentuan rute optimal dengan ongkos optimal dari satu depot ke pelanggan yang lokasinya tersebar dengan jumlah permintaan diketahui. Pada permasalahan ini, ada sebuah depot (PT. PJM) dan sejumlah pelanggan (outlet) untuk dikunjungi dengan demand yang berbeda-beda. Kendaraan diharapkan untuk mengunjungi outlet dan memenuhinya dari depot

Algoritma DE bertujuan untuk membantu pengambilan keputusan dalam merancang rute kendaraan yang optimum. Berikut merupakan hasil olahan rute distribusi dengan menggunakan *Algorithm Differential Evolution*.



Gambar 3. Pengolahan Data Menggunakan Algoritma DE

Secara ringkas hasil alternatif terbaik yang didapat menggunakan *algorithm differential evolution* ditunjukkan pada tabel berikut :

Tabel 3. Hasil Pengolahan Data dengan Algoritma *Differential Evolution*

Rute	Destinasi Outlet	Jumlah		
		Jarak (Km)	Waktu (Menit)	Total Waktu (Menit)
1	PJM-C2-C3-C5-C13-C14-C12-C11-C42-C29-C31-C35-C36-C38-C34-C32-C33-C28-C23-PJM	9.93	14.81	37.24
2	PJM-C22-C19-C21-C39-C40-C65-C66-C67-C68-C43-C47-C44-C45-C20-C18-C10-PJM	12.66	17.98	38.99
3	PJM-C17-C64-C57-C62-C63-C58-C56-C59-C60-C61-C55-C54-C49-C52-C53-C51-C50-C48-C46-C41-C37-C30-C16-PJM	24.74	37.11	49.48
4	PJM-C4-C6-C15-C25-C27-C26-C24-C7-C8-C9-PJM	9.5	14.25	21.59

Analisa Perbandingan Rute Awal dengan Hasil Perhitungan Rute Menggunakan , Algoritma DE dan Metode *Traveling Salesman Problem*

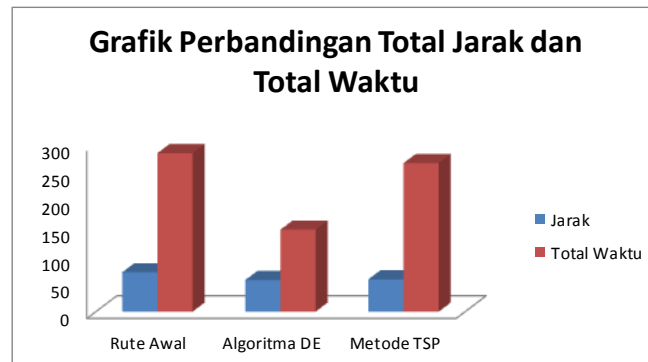
Untuk memperoleh jarak tempuh yang paling optimal bagi perusahaan, maka dilakukan perhitungan dengan menggunakan algoritma *Differential Evolution*. Berikut merupakan perbandingan hasil jarak tempuh dari rute awal, Algoritma DE dan Metode *Traveling Salesman Problem* yang diteliti sebelumnya oleh Paillin dan Tupan (2018)

Tabel 4. Perbandingan Rute Awal dengan Hasil Algoritma DE dan Metode TSP

	Rute	Destinasi Outlet	Total		
			Jarak (Km)	Waktu	Total Waktu
Rute Awal	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	13.09	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
Algoritma DE	1	PJM-C2-C3-C5-C13-C14-C12-C11-C42-C29-C31-C35-C36-C38-C34-C32-C33-C28-C23-PJM	9.93	14.81	37.24
	2	PJM-C22-C19-C21-C39-C40-C65-C66-C67-C68-C43-C47-C44-C45-C20-C18-C10-PJM	12.66	17.98	38.99
	3	PJM-C17-C64-C57-C62-C63-C58-C56-C59-C60-C61-C55-C54-C49-C52-C53-C51-C50-C48-C46-C41-C37-C30-C16-PJM	24.74	37.11	49.48
	4	PJM-C4-C6-C15-C25-C27-C26-C24-C7-C8-C9-PJM	9.5	14.25	21.59
	Total			56.83	84.15
Metode TSP	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
	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			58.05	87.01

Dari table 4. diatas menunjukkan perbedaan jarak antara rute awal perusahaan di bandingkan dengan rute hasil algoritma DE dan rute hasil metode TSP. Algoritma DE memberikan hasil terbaik yaitu menghasilkan 4 rute dengan total jarak 56.83 km dan total waktu sebesar 147 menit dibandingkan dengan metode TSP yang menghasilkan 5 rute dengan total jarak 58.05 km dan total waktu sebesar 266.71 menit.

Untuk lebih memperjelas hasil perbandingan yang diperoleh dari rute awal dengan algoritma DE dan metode TSP maka dapat dilihat pada gambar grafik jarak berikut ini.



Gambar 4. Grafik Perbandingan Total Jarak Tempuh dan Total Waktu

Sesuai dengan grafik tersebut hasil olahan data dengan algoritma DE memberikan hasil jarak yang paling minimum dan total waktu yang paling minimum, dimana algoritma ini mengalami pengurangan jarak tempuh yang lebih hemat dari rute awal dan metode lainnya yaitu selisih 13.93 Km dari jarak rute awal dan 1.22 km dari rute hasilmetode TSP. Telah diketahui sebelumnya bahwa jarak mempengaruhi waktu sehingga berdasarkan hasil tersebut dapat dilihat bahwa selisih waktu tempuh yang diperoleh adalah sebesar 137.21 menit dari rute awal dan 119.41 menit dari rute hasil metode TSP

Untuk lebih memrperjelas penghematan solusi yang diperoleh maka dilakukan perhitungan persentase penghematan jarak, waktu, dan total waktu yang diperoleh setelah diolah. Hasil perhitungan persentase ini diperoleh dengan cara mengurangi jarak regular dengan jarak metode yang digunakan kemudian dibagi dengan jarak regular dan dikalikan dengan 100%.

Tabel.5. Persentase Penghematan Hasil Olahan Data Dengan Rute Regular Perusahaan

Metode	Penghematan Jarak	Penghematan Total Waktu
<i>Algoritma Differential Evolution</i>	19.68%	48.22%
<i>Traveling Salesman Problem (B&B)</i>	17.96%	6.25%

Jika dibandingkan dengan rute regular/awal perusahaan dapat dilihat bahwa hasil perhitungan persentase penghematan lagoritma DE dan metode TSP tersebut sama – sama memberikan solusi optimal terhadap rute tempuh perusahaan. Tetapi hasil persentase penghematan terbesar adalah dengan penggunaan algoritma DE yang memiliki output efisiensi rute tempuh baik dari segi jarak sebesar 19.68%, dan waktu total 48.22%. Persentase penghematan ini juga menunjukkan terjadi penghematan jarak sebesar 13.93 Km dengan penghematan total waktu tempuh selama 137.21 menit dari rute awal perusahaan.

4. Simpulan

Berdasarkan hasil pengolahan dan analisis data maka dapat ditarik kesimpulan yaitu:

1. Dari hasil membandingkan pengolahan data menggunakan metode Banch and Bound dan hasil pengolahan dengan Algorithm Differential Evolution didapatkan hasil rute minimum adalah dengan menggunakan Algorithm Differential yaitu rute 1: PJM-C2-C3-C5-C13-C14-C12-C11-C42-C29-C31-C35-C36-C38-C34-C32-C33-C28-C23-PJM; rute 2: PJM-

- C22-C19-C21-C39-C40-C65-C66-C67-C68-C43-C47-C44-C45-C20-C18-C10-PJM; rute 3: PJM-C17-C64-C57-C62-C63-C58-C56-C59-C60-C61-C55-C54-C49-C52-C53-C51-C50-C48-C46-C41-C37-C30-C16-PJM; rute 4: PJM-C4-C6-C15-C25-C27-C26-C24-C7-C8-C9-PJM.
2. Algoritma DE memberikan hasil terbaik yaitu menghasilkan 4 rute dengan total jarak 56.83 km dan total waktu sebesar 147 menit dibandingkan dengan metode TSP yang menghasilkan 5 rute dengan total jarak 58.05 km dan total waktu sebesar 266.71 menit.
 3. Jika dibandingkan dengan rute regular/awal perusahaan dapat dilihat bahwa hasil perhitungan persentase penghematan algoritma DE dan metode TSP tersebut sama – sama memberikan solusi optimal terhadap rute tempuh perusahaan. Tetapi hasil persentase penghematan terbesar adalah dengan penggunaan algoritma DE yang memiliki output efisiensi rute tempuh baik dari segi jarak sebesar 19.68%, dan waktu total 48.22%.

Daftar Pustaka

- Bowersox, D.J (2002). *Logistical Management*, Macmillan Publishing coInc
- Bräysy, O. (2003). *A Reactive Variable Neighborhood Search for the Vehicle Routing Problem with Time Windows*, INFORMS Journal on Computing, Vol 15 No 4.
- Fradina. S.E dan Saptaningtyas. F. Y, (2017). Penerapan Algoritma Sweep dan Algoritma Genetika pada Penyelesaian Capacitated Vehicle Routing Problem (CVRP) untuk Optimasi Pendistribusian Gula, *Jurnal Matematika*, Vol 6, No 2, pp 63-65.
- Lawler.E.L, Lenstra.J.K, Rinooy Kan.A.H.G, and Shimoys.D.B (1983). *The Traveling Salesman Problem (a Guided Tour of Combinatorial Optimization)*, Jhon Wiley&Sons, Chichester.
- Nurdiansyah, R (2016). Optimasi Penjadwalan Flow Shop Menggunakan Algoritma *Hybrid Differential Evolution*, *Rekayasa Energi Manufaktur*, Vol 1 No 2 pp 43-48.
- Paillin. D.B dan Tupan. J.M (2018). Pemecahan Traveling Salesman Problem Menggunakan Teknik Branch and Bound dan Cheapest Insertion Heuristic (Studi Kasus: PT. Paris Jaya Mandiri–Ambon). *Prosiding Seminar dan Konfrensi Nasional IDEC*. 110-120. (Surakarta, 7-8 Mei 2018).
- Paillin, D. B. dan Wattimena, E.(2015). Penerapan Algoritma Sequential Insertion dalam Pendistribusian BBM di Kawasan Timur Indonesia (Studi Kasus pada PT Pertamina UPMS VIII Terminal Transit Wayame-Ambon), *ARIKA*, Vol 9, No 1 pp. 53-62.
- Santosa. B dan The Jin Ai, (2017). Pengantar Metaheuristik (Implementasi dengan Matlab), ITS Tekno Sains, Surabaya .