PENYELESAIAN MULTIPLE TRAVELLING SALESMAN PROBLEM (MULTI-TSP) DENGAN METODE ORDER CROSSOVER DALAM ALGORITMA GENETIKA

(Studi Kasus: Data Pelanggan Agen Surat Kabar di Kota Singkawang)

Synthia Wulandari, Helmi, Yudhi

INTISARI

Pendistribusian surat kabar oleh beberapa orang salesman pada suatu agen perlu meminimalisir rute yang ditempuh dari depot tertentu (agen surat kabar) menuju ke alamat pelanggan sehingga dapat diantar tepat waktu. Permasalahan dalam penentukan rute oleh beberapa orang salesman merupakan kasus dari Multiple Travelling Salesman Problem (Multi-TSP). Kasus Multi-TSP dapat diselesaikan dengan menggunakan Algoritma Genetika (AG). Algoritma genetika merupakan metode pencarian yang menggunakan tahapan operasi genetik dan seleksi alam. Pada proses penentukan rute terpendek dengan AG diperlukan langkah-langkah pembentukan populasi awal, penentuan nilai fitness, melakukan proses seleksi, melakukan operasi genetik (crossover dan mutasi), dan terbentuk individu baru. Proses AG menggunakan representasi permutasi untuk pendefinisian gen dalam pembentuk kromosom, pembentukan generasi awal menggunakan random generator, perhitungan nilai fitness. Proses seleksi dengan metode seleksi Roulette Wheel, operasi genetik (Order Crossover dan Swapping Mutation), sehingga diperoleh individu baru. Hasil simulasi dari agen surat kabar dengan probabilitas crossover sebesar 0,5 dan probabilitas mutasi sebesar 0,01 yang dilakukan oleh empat orang loper surat kabar ke 30 alamat pelanggan diperoleh rute terpendek dengan jarak 64,03 km pada generasi pertama.

Kata Kunci: Rute Terpendek, Pelanggan Surat Kabar, Optimasi Kombinatorial

PENDAHULUAN

Proses distribusi yang dilakukan dengan mengirimkan semua pesanan konsumen dalam satu kali pesanan disebut *Travelling Salesman Problem* (TSP). Optimasi kombinatorial merupakan suatu permasalahan pada TSP [1]. Permasalahan TSP dideskripsikan untuk menentukan rute terpendek yang harus dilalui oleh seorang *salesman*, yang berangkat dari sebuah kota asal dan menyinggahi setiap kota tepat satu kali kemudian kembali lagi ke kota semula. Dikalangan masyarakat sekarang, permasalahan yang dihadapi bukan hanya memerlukan satu orang *salesman* melainkan beberapa orang *salesman* untuk dapat menyelesaikan suatu pekerjaan. Permasalahan tersebut dapat dikategorikan sebagai perluasan TSP yang disebut *Multiple Travelling Salesman* (Multi-TSP). Contoh kasus Multi-TSP yang sering terjadi dikalangan masyarakat adalah pendistribusian surat kabar, pendistribusian air minum, pendistribusian barang. Pada kasus pendistribusian surat kabar permasalahan yang sering dihadapi adalah keterlambatan waktu penyampaian surat kabar kepada pelanggan. Permasalahan tersebut sangat diperlukan solusi yang konkrit agar tidak terjadi keterlambatan dalam pengantaran.

Pemecahan masalah dalam kasus Multi-TSP menggunakan metode heuristik. Metode heuristik mampu menyelesaikan masalah dengan mendekati global optimalnya namun untuk beberapa kasus tertentu mampu mencapai solusi global optimum. Metode heuristik yang dapat digunakan seperti Algoritma Genetika (AG), *Ant Colony, Tabu Search, Nearest Neighbor* [2]. Permasalahan optimasi yang mempunyai banyak sekali kemungkinan solusi dapat diselesaikan dengan algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis yaitu Algoritma Genetika (AG). Permasalahan ini sulit untuk dilakukan dengan metode-metode eksak [2].

Penelitian ini bertujuan untuk menentukan rute terpendek dari permasalahan Multi-TSP dengan menggunakan AG pada kasus pendistribusian surat kabar. Pencarian rute terpendek dilakukan proses inisialisasi populasi dengan merepresentasikan setiap gen dan membentuk populasi awal dengan *random generator*. Kromosom yang terbentuk dievaluasi dengan menghitung nilai *fitness*. Nilai *fitness*

digunakan untuk menyeleksi kromosom yang layak untuk digunakan sebagai *parent* pada proses *crossover*. Kromosom hasil *crossover* dilanjutkan keproses mutasi untuk mengubah susunan gen dalam suatu kromosom hingga diperoleh sebanyak satu generasi. Hasil pada generasi generasi pertama diteruskan ke generasi selanjutnya hingga terpenuhi kriteria pemberhentian dan diperoleh rute terpendek untuk permasalahan Multi-TSP.

MULTIPLE TRAVELLING SALESMAN PROBLEM (MULTI-TSP)

Multi-TSP adalah suatu metode yang digunakan untuk mecari rute terpendek yang harus dilewati oleh beberapa orang *sales* dalam mengunjungi beberapa tempat tujuan dari suatu tempat asal [3]. Secara sistematis, permasalahan Multi-TSP dapat dinyatakan melalui persamaan berikut:

$$Z = \min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}.x_{ij},$$

dengan kendala-kendala,

1.
$$\sum_{i=1}^{n} x_{ij} = 1, untuk j = 2, ..., n;$$
2.
$$\sum_{j=1}^{n} x_{ij} = 1, untuk i = 2, ..., n;$$
3.
$$\sum_{i=1}^{n} x_{i,1} \le m;$$
4.
$$\sum_{j=1}^{n} x_{1,j} \le m; dan$$

5.
$$x_{ij} = \begin{cases} 1, bila \ salesman \ bergerak \ dari \ i \ ke \ j \ ; \\ 0, bila \ salesman \ tidak \ bergerak. \end{cases}$$

Keterangan:

m = Banyaknya jumlah salesman

 c_{ij} = Jarak dari titik *i* menuju *j*

 x_{ii} = Jalan dari titik *i* menuju *j*

Z = Fungsi tujuan yang meminimumkan perjalanan yaitu dengan meminimumkan jarak (c_{ij}) dengan x_{ij} yang bernilai 0 atau 1, artinya 0 jalan tersebut tidak dikunjungi dan 1 jalan tersebut dikunjungi [4].

Adanya kendala 1 dan 2 menjamin setiap daerah hanya dikunjungi sekali oleh seorang *salesman* dan kendala 3 serta 4 menjaga agar setiap *salesman* yang bergerak berangkat dan kembali dari dan ke daerah awal [5].

ALGORITMA GENETIKA

Algortima genetika merupakan metode pencarian yang memanfaatkan mekanisme operasi genetika dan seleksi alam. Berbeda dengan metode pencarian konvensional lain, AG bekerja dengan sekumpulan kandidat solusi n (kromosom) yang dikenal dengan istilah populasi. Masing-masing kromosom terdiri dari sejumlah bilangan atau simbol yang merepresentasikan suatu solusi yang layak untuk persoalan yang diselesaikan [2].

Kemunculan AG diinspirasikan dari teori-teori dalam ilmu biologi, sehingga banyak istilah dan konsep biologi yang digunakan. Konsep penting dari teori evolusi adalah *fitness* dan *selection* untuk proses reproduksi. Pada proses evolusi didunia nyata terdapat dua cara reproduksi yaitu *sexual reproduction* dan *asexual reproduction*. Pada *sexual reproduction*, kromosom-kromosom dari dua individu (*parent*) dikombinasikan untuk menghasilkan individu baru. Hal ini disebut dengan *crossover* (pindah silang). Proses pengkopian gen pada *parent* tidak luput dari kesalahan. Kesalahan pengkopian gen dikenal dengan istilah *mutation* (mutasi). *Asexual reproduction* hanya satu individu saja yang diperhatikan sehingga tidak terjadi proses *crossover*, tetapi proses mutasi mungkin bisa terjadi [6].

Proses pembentukan generasi baru dengan operasi genetika dilakukan hingga terpenuhi kriteria pemberhentian. Ada beberapa kriteria pemberhentian yang sering digunakan diantaranya [2]:

- a. Berhenti pada generasi tertentu (maximum generation)
- b. Berhenti setelah dalam beberapa generasi berturut-turut didapatkan nilai *fitness* tertinggi tidak berubah.
- c. Berhenti bila dalam beberapa generasi berikutnya tidak didapatkan perubahan nilai *fitness* yang lebih tinggi.

Permasalahan Multi-TSP dapat diselesaikan dengan AG, dilakukan langkah-langkah berikut:

- 1. Penentuan paremeter AG. Parameter yang dimaksud adalah parameter kontrol algoritma genetika yaitu, ukuran populasi, probabilitas *crossover*, dan probabilitas mutasi. Parameter algoritma merupakan salah satu bagian penting dalam penerapan AG namun tidak mudah ditentukan secara pasti. Hal ini tidak terlepas dari prinsip AG yang mengandalkan bilangan acak hampir dalam setiap langkah [7].
- 2. Tahap inisialisasi populasi dilakukan dengan merepresentasikan kromosom dan membentuk populasi awal. Representasi permutasi sering digunakan dalam merepresentasikan kromosom pada permasalahan optimasi kombinatorial. Representasi kromosom permutasi terdiri atas dua segmen yaitu panjang string kromosom bagian pertama merupakan urutan angka dari pengkodean kota yang dituju dan panjang string kromosom bagian kedua merupakan urutan angka dari jumlah pesanan [8]. Representasi permutasi diilustrasikan pada Gambar 1.

Gen	3	1	6	8	7	4	9	2	5		3	2	4
Bagian 1											Ва	agian	2

Gambar 1. Representasi Kromosom Permutasi

Cara pembentukan generasi awal yaitu dengan penggunaan *random number* generator [9]. Inti dari *random number* generator adalah melibatkan bilangan acak untuk nilai setiap gen sesuai dengan representasi kromosom yang digunakan.

- 3. Metode yang digunakan untuk mengevaluasi kromosom adalah dengan menghitung nilai fungsi tujuan sebagai nilai *fitness*. Nilai *fitness* didapat dengan perhitungan jarak yang ditempuh setiap kromosom yang didapat dari proses inisialisasi z_i , $\forall i = 1,2,3,...,n$ dan dilanjutkan dengan perhitungan nilai *fitness* tiap rute (f_i) .
- 4. Salah satu hal penting pada proses AG adalah bagaimana memilih kromosom yang digunakan untuk generasi berikutnya. Metode seleksi yang umum digunakan adalah *Roulette Wheel* (roda roulette). Prosedur seleksi dengan metode *Roulette Wheel* yaitu perhitungan total nilai *fitness* setiap kromosom (f_k) , perhitungan probabilitas *fitness* setiap kromosom (p_i) , perhitungan probabilitas kumulatif setiap kromosom (q_i) [10].
- 5. Operasi genetika terdiri atas *crossover* dan mutasi. *Crossover* merupakan suatu proses pembentukan kromosom turunan (*offspring*) dengan menggabungkan elemen dari kromosom induk terpilih (*parent*) [11].

Metode *crossover* yang digunakan untuk merepresentasikan kromosom permutasi yaitu *Order Crossover* (OX). OX diperkenalkan oleh Syswerda pada tahun 1989. Berikut prosedur OX [2]:

Prosedur OX

Langkah 1. Pilih satu bagian dari kromosom secara acak

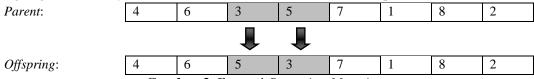


Langkah 2. Pindahkan substring terpilih, pada offspring sesuai pada posisinya

Langkah 3. Kemudian posisi yang masih kosong (*offspring* 1) diisi oleh gen-gen pada kromosom *parent* 2. Berlaku sebaliknya untuk *offspring* 2 yang diisi dengan gen pada kromosom *parent* 1.

Proses mutasi biasanya dilakukan dengan melakukan perubahan terhadap gen pada suatu kromosom. Metode mutasi yang biasanya digunakan untuk optimasi kombinatorial adalah metode

penukaran (*Swapping Mutation*). Proses mutasi dilakukan dengan cara menukarkan gen terpilih secara acak dengan gen sesudahnya [12]. Berikut ilustrasi *Swapping Mutation* pada Gambar 2.



Gambar 2. Ilustrasi Swapping Mutation

IMPLEMENTASI ALGORITMA GENETIKA PADA MULTI-TSP

Kasus Multi-TSP yang dibahas pada penelitian ini mengenai rute pendistribusian surat kabar kepada pelanggan di Kantor Biro Pontianak Pos Kota Singkawang. Sebanyak 4 loper koran yang disebar. Saat proses pendistribusian sangat diperlukan rute terpendek agar dapat mengefisiensikan waktu dan biaya pendistribusian. Berikut sampel lokasi pelanggan pendistribusian koran dapat dilihat pada Tabel 1.

Tabel 1 Lokasi Pelanggan dan Depot

Simpul	Lokasi	Simpul	Lokasi
0	Titik di Jalan Gunung Raya (Biro Pontianak Post)	16	Jalan Kridasana (Lenny Salon)
1	Titik di Jalan Alianyang (Kantor SSA)	17	Titik di Jalan Budi Utomo
2	Kantor Lurah Sungai Garam	18	Jalan Jend. Sudirman (PDAM Gunung Poteng)
3	Titik di Jalan Gunung Merapi	19	Titik di Jalan Pahlawan
4	Jalan Diponegoro (Hotel Khatulistiwa)	20	Titik di Jalan Antasari
5	Jalan Diponegoro (Bank Kalbar)	21	(Pratiwi Komputer)
6	Jalan Stasiun (SD Torsina)	22	Jalan Bambang Ismoyo (SDN 8 Singkawang Tengah)
7	Jalan Tsyafiuddin (SMK Mudita)	23	Jalan Nusantara (Pegadaian)
8	Titik di Jalan Yohana Godang	24	Titik di Jalan Dwi Tunggal
9	Titik di Jalan GM Situt	25	Jalan A. Yani (Rumah Makan Dangau)
10	Jalan Yos Sudarso (Honda Panca Motor)	26	Jalan Perwira (Rumah Sakit DKT)
11	Titik di Jalan Hermansyah	27	Titik di Jalan Tani
12	Jalan Sejahtera (Hotel Prapatan)	28	Jalan Tama (SMAN 1 Singkawang)
13	Titik di Jalan Suhada	29	(SDN 5 Singkawang Tengah)
14	Jalan Gusti Lelanang (KPP Pratama Singkawang)	30	Jalan Firdaus (KPPN)
15	Jalan Kalimantan (Gedung Sun Moon)		

Berdasarkan Tabel 1 dicari panjang jalan yang menghubungkan setiap simpul dengan menggunakan *Google Maps* pada Tabel 2 . Jalan yang dilalui dari simpul 0 ke simpul 4 sebesar 0,4 km tidak sama dengan jalan dari simpul 4 ke simpul 0 sebesar 1 km. Hal ini tergolong Multi-TSP asimetris, yaitu jarak antar simpul tidak sama apabila ditempuh dari arah yang berbeda.

Tabel 2. Data Jarak Antar Simpul

Simpul	0	1	2	3	4	5	6	7	8	9	10		30
0	0	2,1	6	1,2	0,4	2	1,3	1,4	0,85	1,7	1,9		0,18
1	2,1	0	3,9	3,1	2,2	0,95	1,7	1	2,8	1,7	0,65		2
2	6	3,9	0	7	6	4,8	5,3	4,9	6,6	5,6	4,6		5,8
3	1,2	3,1	7	0	1	2,4	1,6	2,3	0,4	1,6	2,6		1,2
4	1	2,4	6	1,7	0	1,6	0,95	1,4	1,5	1,4	1,7		0,85
30	0,18	2	5,8	1,2	0,23	1,6	1,2	1,2	0,8	1,6	1,7		0

Dari Tabel 2 menjelaskan bahwa jarak simpul 0 ke simpul 0 tidak memiliki jarak, maka bernilai 0. Hal tersebut berlaku untuk jarak antar simpul yang sama. Jarak simpul 0 ke simpul 1 sebesar 2,1 km, jarak simpul 0 ke simpul 2 sebesar 6 km, dan seterusnya.

Permasalahan kasus Multi-TSP dapat diselesaikan dengan menggunakan AG. Sebelum melakukan proses AG ditentukan parameter berupa Pop_size, Pc, dan Pm yang diperlukan dalam penentuan rute terpendek. Pop_size sebanyak 35 kromosom, Pc sebesar 0,5 ini berarti 50% dari total kromosom mengalami *crossover* dan Pm sebesar 0,01 ini berarti 1% dari total bit yang ada pada populasi mengalami mutasi.

Tahap inisialisasi populasi merupakan tahap merepresentasikan kromosom dan menentukan generasi awal. Representasi kromosom dinyatakan menggunakan representasi permutasi. Setiap gen pada kromosom berupa angka yang menyatakan nomor dari tiap simpul. Terdapat 30 simpul sebagai tempat tujuan dan 1 simpul sebagai depot. Tahap menentukan generasi awal pada permasalahan Multi-TSP menggunakan *random generator*. *Random generator* digunakan karena lebih mempercepat proses penentuan rute pilihan dan untuk menghindari kesamaan rute digunakan terdapat 35 kromosom yang dijadikan sebagai rute pilihan dengan 30 gen sebagai daerah tujuan. Terdapat 4 orang *salesman*. Untuk *salesman* pertama berangkat menuju 7 lokasi pelanggan, *salesman* kedua 9 lokasi pelanggan, *salesman* ketiga 6 lokasi pelanggan, dan *salesman* keempat 8 lokasi pelanggan.

Bilangan acak dibangkitkan sebanyak 30 angka dengan [0,1]. Setiap kromosom disimbolkan dengan Z_i untuk setiap n=1,2,3,...,35). Berikut bilangan acak yang digunakan untuk proses *random generator* kromosom 1 pada Tabel 3.

Tabel 3 Bilangan Acak Populasi Awal Kromosom 1

Lokasi	Bilangan	No.	Lokasi	Bilangan	No.	Lokasi	Bilangan	No.
Pelanggan	acak	Urut	Pelanggan	acak	Urut	Pelanggan	acak	Urut
Daerah 1	0,8147	20	Daerah 11	0,1576	5	Daerah 21	0,6557	14
Daerah 2	0,9058	22	Daerah 12	0,9706	30	Daerah 22	0,0357	1
Daerah 3	0,1270	3	Daerah 13	0,9572	26	Daerah 23	0,8491	21
Daerah 4	0,9134	23	Daerah 14	0,4854	10	Daerah 24	0,9340	25
Daerah 5	0,6324	12	Daerah 15	0,8003	19	Daerah 25	0,6787	15
Daerah 6	0,0975	2	Daerah 16	0,1419	4	Daerah 26	0,7577	17
Daerah 7	0,2785	7	Daerah 17	0,4218	9	Daerah 27	0,7431	16
Daerah 8	0,5469	11	Daerah 18	0,9157	24	Daerah 28	0,3922	8
Daerah 9	0,9575	27	Daerah 19	0,7922	18	Daerah 29	0,6555	13
Daerah 10	0,9649	29	Daerah 20	0,9595	28	Daerah 30	0,1712	6

Bilangan acak pada Tabel 3 diurutkan berdasarkan nilai terkecil. Bilangan acak terkecil adalah 0,0357 berada diurutan ke-22, maka daerah 22 menjadi daerah pertama. Proses berlanjut hingga angka terkecil terakhir adalah 0,9706 berada diurutan ke-12, maka daerah 12 menjadi daerah ke-30. Setelah daerah ke-30 maka kembali ke daerah asal. Rute yang didapat dari langkah tersebut adalah Z_1 = (22,6,3,16,11,30,7,28,17,14,8,5,29,21,25,27,26,19,15,1,23,2,4,18,24,13,9,20,10,12). Diperoleh populasi awal pada Tabel 4.

Tabel 4 Populasi Awal

No.	Kromosom
1	$Z_1 = (22,6,3,16,11,30,7,28,17,14,8,5,29,21,25,27,26,19,15,1,23,2,4,18,24,13,9,20,10,12)$
2	$Z_2 = (2,10,4,5,25,24,15,30,21,3,8,28,12,11,17,16,26,29,18,23,22,7,1,19,20,13,14,6,9,27)$
3	$Z_3 = (8,9,17,14,18,12,2,10,26,16,21,29,20,3,7,24,30,23,19,4,1,28,27,13,22,11,5,25,15,6)$
	•
35	$Z_{35} = (12,27,6,29,3,5,1,24,22,21,8,2,17,10,23,20,28,19,18,11,4,9,7,15,14,30,25,16,13,26)$

Sebelum memulai proses seleksi, kromosom yang sudah didapat dari proses inisialisasi dihitung nilai *fitness* setiap kromosom. Nilai *fitness* adalah nilai yang menyatakan baik tidaknya suatu solusi. Penentuan jarak yang ditempuh setiap kromosom (z_i) yang didapat dari proses inisialisasi z_i , $\forall i=1,2,3,...,35$ dengan memasukkan 4 salesman dalam pembagian daerah tujuan. Kromosom Z_1 terdiri atas salesman pertama (0,22,6,3,16,11,30,7,0)=3,1km+1,8km+...+1,4km dengan total jarak 12,9km, dengan cara yang sama untuk salesman kedua,ketiga dan keempat. Hasil dari masing-masing salesman ditotalkan maka didapat jarak tempuh untuk Z_1 . Perhitungan yang sama untuk Z_2 hingga Z_{35} . Nilai *fitness* tiap rute (f_i) diperoleh dari membagi satu dengan total jarak setiap kromosom. Nilai *fitness* pada Z_1 adalah $f_1 = \frac{1}{74} = 0,013513514$. Perhitungan yang sama untuk f_2 hingga f_{35} .

Proses seleksi merupakan proses memilih kromosom-kromosom yang dipertahankan dalam populasi selanjutnya. Proses seleksi yang paling mudah adalah proses seleksi dengan *Roulette Wheel selection*. Kromosom dengan nilai *fitness* yang besar menempati potongan lingkaran lebih besar daripada kromosom dengan nilai *fitness* kecil. Perhitungan total *fitness* dengan menjumlahkan nilai *fitness* setiap kromosom $f_k = 0.013513514 + 0.013655606 + \cdots + 0.0138217 = 0.4636$. Perhitungan probabilitas *fitness* setiap kromosom (p_i) , $p_1 = \frac{f_i}{f_k} = \frac{0.013513514}{0.4636} = 0.02914634$. Perhitungan yang sama untuk p_2 hingga p_{35} . Perhitungan probabilitas kumulatif tiap kromosom (q_i) $q_1 = 0.02914634$, $q_2 = 0.02914634 + 0.02945281 = 0.05859915$ hingga q_{35}

Proses *Roulette Wheel selection* dimulai dengan membangkitkan bilangan acak (r_k) pada range [0,1] sebanyak pop_size. Apabila $q_{i-1} < r_k < q_i$, $\forall k, i = 1,2,3,...,35$, maka kromosom ke-i terpilih sebagai induk. Nilai r_k dibangkitkan sebanyak 35 angka dalam [0,1] sebagai jumlah kromosom dalam satu populasi pada Tabel 5 untuk proses *Roulette Wheel Selection*.

Tabel 5 Bilangan Acak Untuk Roulette Wheel Selection

No.	Bilangan acak	No.	Bilangan acak	No.	Bilangan acak	No.	Bilangan acak
1	0,8235	10	0,1869	19	0,1626	28	0,5060
2	0,6948	11	0.4898	20	0,1190	29	0,6991
3	0,3171	12	0,4456	21	0,4984	30	0,8909
4	0,9502	13	0,6463	22	0,9597	31	0,9593
5	0,0344	14	0,7094	23	0,3404	32	0,5472
6	0,4387	15	0,7547	24	0,5853	33	0,1386
7	0,3816	16	0,2760	25	0,2238	34	0,1493
8	0,7655	17	0,6797	26	0,7513	35	0,2575
9	0,7952	18	0,6551	27	0,2551		

Nilai r_1 sebesar 0,8235 berada diantara q_{28} dan q_{29} yaitu 0,80466969 dan 0,83191958 maka kromosom 1 digantikan dengan kromosom 29. Dengan proses yang sama untuk bilangan acak r_2 hingga r_{35} . Populasi baru hasil seleksi terbentuk dan dihitung kembali total jarak setiap kromosom. Populasi hasil seleksi *Roulette Wheel Selection* diperoleh pada Tabel 6.

Tabel 6 Populasi Hasil Seleksi Roulette Wheel Selection

om
(3,0,16,4,24,10,21,7,0,28,8,17,27,2,15,12,5,0)
0,7,22,11,8,17,23,0,28,29,10,12,19,3,21,24,0)
5,0,27,30,14,29,11,2,0,9,15,24,8,18,23,7,13,0)
3,0,12,8,16,17,28,30,0,6,24,3,22,23,5,11,20,0)
2,(5

Proses *crossover* dilakukan agar diperoleh populasi dari setiap kromosom lebih baik dari proses sebelumnya. Proses *crossover* melibatkan kromosom-kromosom yang memiliki nilai acak pada [0,1].

Probabilitas crossover (Pc) yang digunakan adalah 0,5. Jika bilangan acak yang dihasilkan kurang dari 0,5 maka dipilih parent baru. Parent baru yang terpilih untuk mendapatkan kromosom-kromosom yang lebih baik. Bilangan acak (r_k) dibangkitkan pada [0,1] dalam Tabel 7 untuk proses Order Crossover.

Tabel 7 Bilangan Acak Untuk Proses *Order Crossover*

No.	Bilangan	No.	Bilangan	No.	Bilangan	No.	Bilangan
	acak		acak		acak		acak
1	0,7060	10	0,0344	19	0,7094	28	0,3404
2	0,0318	11	0,4387	20	0,7547	29	0,5853
3	0,2769	12	0,3816	21	0,2760	30	0,2238
4	0,0462	13	0,7655	22	0,6797	31	0,7513
5	0,0971	14	0,7952	23	0,6551	32	0,2551
6	0,8235	15	0,1869	24	0,1626	33	0,5060
7	0,6948	16	0,4898	25	0,1190	34	0,6991
8	0,3171	17	0,4456	26	0,4984	35	0,8909
9	0,9502	18	0,6463	27	0,9597		

Bilangan acak pada Tabel 7 yang memiliki nilai kurang dari sama dengan 0,5 (ditandai dengan warna merah) adalah r_2 , r_3 , r_4 , r_5 , r_8 , r_{10} , r_{11} , r_{12} , r_{15} , r_{16} , r_{17} , r_{21} , r_{24} , r_{25} , r_{26} , r_{28} , r_{30} , r_{32} . Kromosom Z_2 , Z_3 , Z_4 , Z_5 , Z_8 , Z_{10} , Z_{11} , Z_{12} , Z_{15} , Z_{16} , Z_{17} , Z_{21} , Z_{24} , Z_{25} , Z_{26} , Z_{28} , Z_{30} , Z_{32} yang menjadi *parent* baru untuk proses *crossover*. Berikut proses *crossover* dengan metode *Order Crossover*:

Crossover 1: $Z_3 > < Z_2$

Langkah 1. Pilih secara acak substring dari kedua kromosom

		_										_																					
Parent 1	3	1 2	5	4	19	6	20	10	2	1	22	26	28	17	1	6 1	12	5	27	30	14	- 29	9 1	1 :	2	9	15	24	8	18	23	7	13
Parent 2	9	2	15	30	20	1	1	6	25	5	4	18	6	13	2	6 1	4	27	7	22	11	8	17	23	28	-	29	10	12	19	3	21	24
	Langkah 2. Pindahkan substring terpilih, pada offspring sesuai pada posisinya																																
Offspring 1										21	22	2 2	.6	28	17	16	12	5	2	27	30	14	29	11	2	-	9						
Offspring 2											5	4	18	6	13	26	14	4	27	7	22	11	8	17	23	2	28						
	Langkah 3. Kemudian posisi yang masih kosong (offspring 1) diisi oleh gen-gen pada kromosom																																
	pa	rent	2,1	oeg	itu	set	alil	kny	a.																								
Offspring 1	15	20	1		25	4	18	6	13	21	22	2	6	28	17	16	12	5	2	7	30	14	29	11	2	9	7	8	23	10	19	3	24
Offspring 2	3	1	25	19)	20	10	21	16	5	4	18	6	1	3	26	14	27	7	22	11	8	17	23	3 2	28	12	30	29	2	9	15	24
	D.,	40 -		4:	400	o + 1		1 4	:						1 ~.	46161																	

Rute yang didapat hasil dari proses crossover 1 adalah

 $Z_3 = (0.15, 20, 1, 25, 4, 18, 6, 0, 13, 21, 22, 26, 28, 17, 16, 12, 5, 0, 27, 30, 14, 29, 11, 2, 0, 9, 7, 8, 23, 10, 19, 3, 24, 0)$

 $Z_2 = (0.3, 1, 25, 19, 20, 10, 21, 0, 16, 5, 4, 18, 6, 13, 26, 14, 27, 0, 7, 22, 11, 8, 17, 23, 0, 28, 12, 30, 29, 2, 9, 15, 24, 0)$

Setelah proses *crossover* diperoleh sebanyak 306 kromosom, kemudian dipilih kromosom yang memiliki jarak paling minimum dari proses *crossover*. Kromosom yang terpilih digunakan untuk proses selanjutnya. Berikut kromosom yang diperoleh pada Tabel 8 setelah dilakukan proses *Order Crossover* dengan total jarak minimum.

Tabel 8 Populasi Setelah Proses Order Crossover dengan Total Jarak Minimum

No.	Kromosom									
1	$Z_1 = (0,20,25,11,1,29,30,13,0,14,19,3,26,18,22,6,9,23,0,16,4,24,10,21,7,0,28,8,17,27,2,15,12,5,0)$	79,15								
2	$Z_2 = (0,19,9,21,29,1,12,20,0,16,5,4,18,6,13,26,14,27,0,7,22,11,8,17,23,0,28,15,25,3,2,24,10,30,0)$	75,03								
3	$Z_3 = (0,24,1,10,8,25,3,13,0,20,21,22,26,28,17,16,12,5,0,27,30,14,29,11,2,0,9,4,19,18,23,7,6,15,0)$	67,25								
	•									
	•									
35	$Z_{35} = (0,4,21,1,25,29,14,9,0,26,2,27,19,18,10,7,15,13,0,12,8,16,17,28,30,0,6,24,3,22,23,5,11,20,0)$	75,03								

Proses mutasi bertujuan meningkatkan keragaman kromosom yang ada pada populasi sehingga tidak terbawa pada solusi *optimum local*. Probabilitas mutasi yang digunakan adalah 1% dikarenakan menyeimbangkan dengan total panjang gen yang besar. Proses mutasi yang digunakan adalah dengan metode penukaran (*Swapping Mutation*). Banyak gen dalam 1 kromosom sebanyak 30 dan jumlah kromosom sebanyak 35. Maka panjang total gen adalah $30 \times 35 = 1050$ buah. Jumlah gen yang dimutasi adalah $1\% \times 1050 = 10,5 \approx 11$. Dengan demikian dibangkitkan bilangan acak sebanyak 11 bilangan bulat antara 1 sampai 1050. Bilangan acak yang digunakan pada Tabel 9 untuk proses *Swapping Mutation*.

Tabel 9 Bilangan Acak untuk Proses *Swapping Mutation*

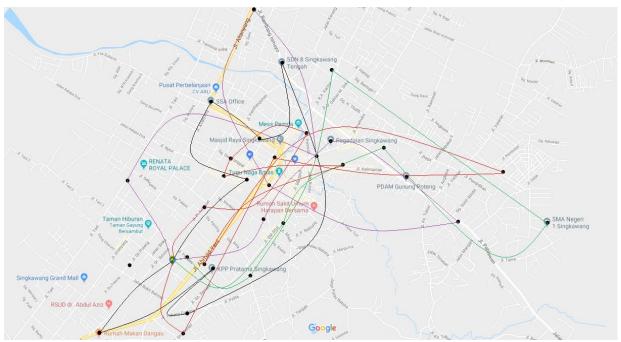
	2000 2 5 Brianguir Front Oritori Front Street Color											
No.	Bilangan acak	No.	Bilangan acak	No.	Bilangan acak	No.	Bilangan acak					
1	856	4	960	7	293	10	1014					
2	952	5	664	8	575	11	166					
3	134	6	103	9	1006							

Dari 11 bilangan acak yang digunakan untuk menukar posisi gen dengan gen selanjutnya (bersebelahan) sesuai dari urutan gen pertama dengan menggunakan metode *Swapping Mutation* (penukaran mutasi). Pada kromosom ke-29 gen ke-16 ditukar dengan gen ke-17, kromosom ke-32 gen ke-22 ditukar dengan gen ke-23, kromosom ke-5 gen ke-14 ditukar dengan gen ke-15, hingga pada kromosom ke-6 gen ke-16 ditukar dengan gen ke-17. Kromosom hasil proses *Swapping Mutation* diperoleh pada Tabel 10.

Tabel 10 Populasi Sesudah Proses Swapping Mutation

No.	Kromosom									
		Jarak								
1	$Z_1 = (0,20,25,11,1,29,30,13,0,14,19,3,26,18,22,6,9,23,0,16,4,24,10,21,7,0,28,8,17,27,2,15,12,5,0)$	79,15								
2	$Z_2 = (0,19,9,21,29,1,12,20,0,16,5,4,18,6,13,26,14,27,0,7,22,11,8,17,23,0,28,15,25,3,2,24,10,30,0)$	75,03								
3	$Z_3 = (0.24,1.10,8.25,3.13,0.20,21,22,26,28,17,16,12,5,0.27,30,14,29,11,2,0.9,4,19,18,23,7,6,15,0)$	67,25								
4	$Z_4 = (0,30,18,14,1,21,9,4,0,24,12,23,17,20,5,8,26,16,0,11,6,15,22,25,3,0,2,28,29,13,10,19,7,27,0)$	66,28								
5	$Z_5 = (0,10,9,20,19,13,27,30,0,5,21,3,8,28,12,17,11,16,0,26,29,18,23,22,7,0,1,24,15,2,4,14,25,6,0)$	67,48								
	·									
	·									
	•									
35	$Z_{35} = (0,4,21,1,25,29,14,9,0,26,2,27,19,18,10,7,15,13,0,12,8,16,17,28,30,0,6,24,3,22,23,5,11,20,0)$	75,03								

Proses mutasi merupakan tahap akhir dari perhitungan mencari rute terpendek dengan algoritma genetika. Tanda merah merupakan gen yang mengalami $Swapping\ Mutation$. Generasi pertama kromosom yang memiliki rute terpendek yaitu pada Z_{12} dengan total jarak tempuh sebesar 64,03 km. Urutan tempat tujuan setiap salesman dan total jarak tempuh setiap salesman dari kromosom $Z_{12}=(0.11,1.10,22,8.14,25,0.20,21,24,7.15,4.29,5,3,0.9,13,28,26,16,30,0.27,17,2,12,19,18,23,6,0)$ yaitu untuk salesman pertama rute yang dilalui adalah (0.11,1.10,22,8.14,25,0) dengan jarak tempuh sebesar 10,65 km, untuk salesman kedua rute yang dilalui adalah (0.20,21,24,7.15,4,29,5,3,0) dengan jarak tempuh sebesar 19,5 km, untuk salesman ketiga rute yang dilalui adalah (0.9,13,28,26,16,30,0) dengan jarak tempuh sebesar 12,53 km, dan untuk salesman keempat rute yang dilalui adalah (0.27,17,2,12,19,18,23,6,0) dengan jarak tempuh sebesar 21,35 km. Dari hasil tersebut diperoleh rute masing-masing salesman dimulai dan diakhiri dari daerah asal yaitu Biro Pontianak Post. Berikut ilustrasi dari rute yang ditempuh setiap salesman pada Gambar 3.



Gambar 3 Ilustrasi Multi-TSP Pendistribusian Surat Kabar

Garis berwarna hitam merupakan salesman pertama, garis berwarna merah merupakan salesman kedua, garis berwarna hijau merupakan salesman ketiga, dan garis berwarna ungu merupakan salesman keempat. Untuk memastikan rute pada Z_{12} sudah minimun, maka dilanjutkan penyelesaian dengan AG pada generasi kedua. Jika nilai fitness pada generasi kedua sama atau lebih kecil dari generasi pertama maka hasil jarak tempuh yang digunakan yaitu pada generasi pertama. Jika nilai fitness yang diperoleh pada generasi kedua lebih besar dari generasi pertama maka dilanjutkan dengan proses AG pada generasi selanjutnya.

Populasi awal dari generasi kedua diperoleh dari hasi mutasi kromosom pertama. Dilanjutkan dengan proses seleksi, crossover, dan mutasi dengan langkah yang sama pada generasi pertama. Nilai fitness yang diperoleh pada generasi kedua sama dengan generasi pertama yaitu pada Z_6 dan Z_{30} . Nilai fitness pada generasi pertama dan kedua sama, maka Z_{12} pada generasi pertama sudah mewakili solusi optimum dari proses penentuan rute terpendek. Jarak tempuh pada Z_{12} sebesar 64,03km merupakan solusi optimum dari proses AG.

PENUTUP

Penyelesaian kasus Multi-TSP dengan menggunakan Algoritma Genetika sebanyak dua generasi memperoleh hasil optimum sebesar 64,03 km pada generasi pertama dalam penentuan rute terpendek dari 4 orang *salesman* surat kabar yang ada di Kota Singkawang. Rute terpendek dimulai dan diakhiri pada Biro Pontianak Post sebagai depot. Berikut total jarak tujuan masing-masing *salesman*:

- 1. Untuk salesman pertama sebanyak 7 lokasi tujuan dengan jarak tempuh sebesar 10,65 km.
- 2. Untuk salesman kedua sebanyak 9 lokasi tujuan dengan jarak tempuh sebesar 19,5 km.
- 3. Untuk salesman ketiga sebanyak 6 lokasi tujuan dengan jarak tempuh sebesar 12,53 km.
- 4. Untuk salesman keempat sebanyak 8 lokasi tujuan dengan jarak tempuh sebesar 21,35 km.

Penelitian pencarian rute terpendek dengan algoritma genetika pada penelitian ini dapat dilanjutkan dengan perbandingan antara metode *Partiall Mapped Crossover* (PMX) dan *Order Crossover* (OX) pada proses *crossover*. Penelitian selanjutnya juga dapat menggunakan aplikasi untuk generasi yang lebih banyak seperti 100 generasi.

DAFTAR PUSTAKA

- [1] Farisi OIR, Pratamasunu GQO. Penyelesaian Multi_Depot Multiple Travelling Salesman Problem Menggunakan K-Means dan Ant colony Optimization. *Nusantara Journal of Computers adn its Aplications*. 2016 (02); 1-9
- [2] Syarif A. Algoritma Genetika Teori dan Aplikasi. Edisi Kedua. Yogyakarta: Graha Ilmu; 2014
- [3] Sari RN, Mahmudy WF. Penyelesaian Multiple Travelling Salesperson Problem (M-TSP) Dengan Algoritma Genetika: Studi Kasus Pendistribusian Air Mineral. *Jurnal Mahasiswa PTIIK Universitas Brawijaya*; 2015 (05).
- [4] Saptaningtyas FY. Multi Travelling Salesman Problem (MTSP) Dengan Algoritma Genetika Untuk Menentukan Rute Loper Koran Di Agen Surat Kabar. *Pythagoras*. 2012 (07); 55-64.
- [5] Mayuliana NK, Kencana EN, Harini LPI. Penyelesaian Multi Travelling Salesman Problem Dengan Algoritma Genetika. *E-Jurnal Matematika*. 2017 (06); 1-6.
- [6] Suyanto. Algoritma Genetika dalam Matlab. Edisi Pertama. Yogyakarta: Penerbit Andi; 2005.
- [7] Zukhri Z. Algoritma Genetika Metode Komputasi Evolusioner untuk Menyelesaikan Masalah Optimasi. Edisi Pertama. Yogyakarta Penerbit Andi; 2014
- [8] Karimah S, Widodo AW, Cholissodin I. Optimasi Multiple Travelling Salesman Problem Pada Pendistribusian Air Minum Menggunakan Algoritma Genetika (Studi Kasus:UD. Tosa Malang), *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. 2017 (01); 849-858.
- [9] Fauziah, Agustina I, Andryana S. Analisis Implementasi Random Number Generate(RNG) Pada Simulasi Antrian Menggunakan Aplikasi Berbasis Net Framework, *Seminar Nasional Informatika*; 2012.
- [10] Kusumadewi S, Purnomo H. *Penyelesaian Masalah Optimasi dengan Teknik-teknik Heuristik*. Edisi Pertama. Yogyakarta: Penerbit Graha Ilmu; 2005.
- [11] Fanggidae A, Lado FR. *Algortima Genetika dan Penerapannya*. Edisi Pertama. Yogyakarta: Teknosain; 2015
- [12] Joni DMAB, Nurcahyawati V. Penentuan Jarak Terpendek Pada Jalur Distribusi Barang Di Pulau Jawa Dengan Menggunakan Algoritma Genetika. *Seminar Nasional Pendidikan Teknik Informatika*; 2012

SYNTHIA WULANDARI : Jurusan Matematika FMIPA Untan, Pontianak,

synthiawulandari6@gmail.com

HELMI : Jurusan Matematika FMIPA Untan, Pontianak,

Helmi132205@yahoo.co.id

YUDHI : Jurusan Matematika FMIPA Untan, Pontianak,

dhye_dhoank@yahoo.co.uk