

OPTIMASI VEHICLE ROUTING PROBLEM WITH TIME WINDOW (VRPTW) PADA DISTRIBUSI PRODUK PANGAN MENGGUNAKAN ALGORITMA GENETIKA

Rayandra Yala Pratama¹, Wayan Firdaus Mahmudy²

¹Mahasiswa Program Studi Informatika/Ilmu Komputer Universitas Brawijaya

²Dosen Program Studi Informatika/Ilmu Komputer Universitas Brawijaya

Program Studi Informatika/Ilmu Komputer

Program Teknologi Informasi dan Ilmu Komputer

Universitas Brawijaya

Jalan Veteran Malang 65145, Indonesia

Email: ¹rayandra.pratama@gmail.com, ²wayanfm@ub.ac.id

ABSTRAK

Dalam industri produk pangan, proses pengiriman atau proses distribusi merupakan proses yang penting karena produk pangan tidak dapat bertahan lama saat melakukan pengiriman sehingga membutuhkan waktu dan rute tercepat. Jauh tidaknya rute yang diambil menentukan besar kecilnya pengeluaran untuk proses distribusi karena semakin jauh rute yang ditempuh maka semakin besar biaya yang harus dikeluarkan. Sebaliknya, jika rute yang diambil adalah rute terpendek maka proses pengiriman akan menjadi efisien dan dapat menekan biaya pengiriman. Menentukan rute distribusi menjadi semakin sulit jika terdapat banyak pelanggan yang harus dikunjungi dan setiap pelanggan mempunyai kebijakan waktu tertentu dalam menerima pengiriman. Permasalahan ini dikenal dengan *Vehicle Routing Problem with Time Windows* (VRPTW). Permasalahan VRPTW dapat diselesaikan menggunakan algoritma genetika karena algoritma genetika menghasilkan beberapa solusi. Dalam memecahkan solusi, algoritma genetika membuat kromosom yang terdiri dari nomor-nomor yang merepresentasikan pelanggan yang harus dikunjungi. Kromosom ini yang selanjutnya digunakan dalam proses perhitungan bersama dengan operator genetika lainnya seperti ukuran populasi, banyaknya generasi, *crossover* dan *mutation rate*. Setelah mendapatkan hasil maka langkah selanjutnya adalah melakukan pengujian. Pengujian ini berfungsi untuk mencari parameter yang menghasilkan nilai *fitness* terbaik. Hasil dari pengujian didapatkan bahwa ukuran populasi terbaik sebesar 300 dengan generasi sebanyak 3000 serta kombinasi *crossover* dan *mutation rate* masing-masing 0.4 dan 0.6. Pengujian ini juga didapatkan seleksi terbaik yaitu seleksi elitis. Setelah mendapatkan parameter-parameter terbaik, didapatkan hasil nilai *fitness* dari parameter-parameter terbaik sebesar 0.000788.

Kata kunci: Algoritma Genetika, VRPTW, distribusi, *Fitness*, *Time Window*, produk pangan

ABSTRACT

In the food products industry, the distribution process is very important because the food product can expired during distribution so fastest route is needed. Distribution expenses are depend on the route it takes since the further distance the greater the costs. On the other hand, if the route taken is the shortest one, the delivery process will be efficient and can reduce the cost of distribution. Determining the distribution becomes more difficult if there are many customers must visited and every customer has a time window for serving the delivery. This problem is known as the *Vehicle Routing Problem with Time Windows* (VRPTW). VRPTW problems can be solved using genetic algorithms because genetic algorithms generate multiple solutions at once. Genetic algorithms make chromosomes from serial numbers that represent the customer to visit. These chromosomes are used in the calculation process together with other genetic operators such as population size, number of generations, *crossover* and *mutation rate*. After getting the results, the next step is to test the program. This test is used to find the parameters that produce the best fitness value. The results show that the best population size is 300, 3,000 generations and the combination of *crossover* and *mutation rate* is 0.4 and 0.6. The result also showing that the best selection method is elitist selection. The fitness value from the best parameters is 0.000788.

Keywords: Genetic Algorithm, VRPTW, distribution, *Fitness*, *Time Window*, food products

1. PENDAHULUAN

1.1 Latar Belakang

Distribusi merupakan kegiatan yang dilakukan untuk memenuhi permintaan pelanggan. Pelanggan disini bisa berupa toko milik perseorangan maupun cabang dari pabrik yang melakukan proses distribusi. Jika jumlah pelanggan yang harus dipenuhi sedikit, maka proses distribusi akan menjadi mudah karena tempat yang harus dikunjungi sedikit dan mudah untuk menentukan rute tercepat. Namun, akan berbeda jika jumlah pelanggan yang harus dikunjungi lebih dari 50. Pencarian rute tercepat akan menjadi lama dan sulit jika ditambahkan waktu untuk menerima barang pada masing-masing pelanggan. Permasalahan ini dapat disebut juga *vehicle routing problem with time windows* (VRPTW). Permasalahan VRPTW ini berbeda dengan *Traveling Salesman Problem* yang hanya mencari solusi berdasarkan rute saja tanpa ada batasan waktu atau *time window* (Hlaing dan Khine, 2011).

Terdapat banyak metode yang dapat digunakan untuk menyelesaikan permasalahan VRPTW. *Improved Simulated Annealing* digunakan untuk mengatasi masalah VRPTW pada dataset permasalahan *Solomon Benchmark Problems*. Penelitian menyimpulkan bahwa *Improved Simulated Annealing* dapat digunakan untuk menyelesaikan permasalahan VRPTW dan menghasilkan solusi dalam waktu rata-rata 82.29 detik (Mahmudy, 2014). *Distributed evolutionary algorithms* digunakan untuk menyelesaikan *vehicle routing problem* yang memberikan hasil bahwa dengan menambahkan algoritma distribusi kepada *evolutionary algorithms* mempunyai hasil yang lebih baik meskipun dengan jumlah perhitungan yang sama (Puljic dan Manger, 2012). Pada permasalahan *Vehicle Routing Problem with Soft Time Windows* juga dapat diselesaikan dengan *Improved Genetic Algorithms* yang menunjukkan hasil yang lebih baik daripada metode pola *Group-based*, CW dan tipe persilangan O-X (Li dkk, 2015).

Optimized Crossover Genetic Algorithm dapat digunakan untuk menyelesaikan permasalahan *vehicle routing problem with time windows* dengan hasil kualitas yang cukup baik pada total jarak terjauh (Nazif dan Lee, 2010). Permasalahan yang sama juga ditemukan pada penelitian sebelumnya dengan objek distribusi minuman bersoda. Penelitian sebelumnya mengungkapkan bahwa masalah VRPTW dapat diselesaikan dengan menggunakan *evolution strategies* yang disimpulkan dari tidak adanya waktu *tardy* dan semua pelanggan terlayani (Harun, Mahmudy dan Yudistira, 2014).

Penelitian yang sama juga dilakukan pada permasalahan pencarian rute optimal untuk distribusi

air minum dapat menerapkan algoritma genetika sebagai solusi dengan kendala *time window*. Penelitian tersebut mengungkapkan semakin banyak generasi belum tentu akan mendapatkan hasil yang optimal (Sundarningsih, Mahmudy dan Sutrisno, 2015). Algoritma genetika juga dapat menyelesaikan VRPTW untuk distribusi barang dengan objek mie instan. Pada penelitian tersebut disimpulkan bahwa ukuran generasi paling optimal yaitu berada pada generasi 2000 dan populasi optimal berada pada 140 populasi (Saputri, Mahmudy dan Ratnawati, 2015).

VRPTW merupakan permasalahan yang sepele jika jumlah pelanggan yang harus dilayani sedikit, akan tetapi menjadi permasalahan yang kompleks jika jumlah pelanggan yang harus dilayani lebih dari 50 pelanggan. Sedangkan, produk pangan merupakan barang yang dapat kadaluarsa atau membusuk jika proses pengiriman/distribusi membutuhkan waktu yang sangat lama. Pemilihan rute yang tercepat akan sangat membantu dalam proses distribusi karena akan menghemat waktu dan juga pengeluaran. Dengan pertimbangan tersebut, maka permasalahan akan diselesaikan menggunakan algoritma genetika karena algoritma genetika merupakan algoritma yang fleksibel. Algoritma ini membutuhkan input berupa fungsi tujuan dan menghasilkan lebih dari satu solusi.

1.2 Rumusan Masalah

Rumusan masalah yang terdapat dalam penelitian ini antara lain:

1. Bagaimana menerapkan algoritma genetika dengan *Vehicle Routing Problem With Time Windows* (VRPTW) pada pencarian rute terpendek?
2. Bagaimana nilai *fitness* yang dihasilkan dari seleksi Elitis dan *Binary Tournament*?
3. Bagaimana pengaruh parameter genetika terhadap nilai *fitness* yang didapatkan?
4. Bagaimana pengaruh *crossover rate* dan *mutation rate* terhadap nilai *fitness*?
5. Bagaimana hasil nilai *fitness* jika menggunakan parameter terbaik?

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Dataset diambil dari website VRPTW Benchmark Problems.
<http://web.cba.neu.edu/~msolomon/problems.htm>.
2. Dataset yang dipakai dalam penelitian ini adalah tipe R1 dan nomor R101.

3. Jumlah kendaraan yang digunakan sejumlah 25 dan mempunyai kapasitas masing-masing 200.
4. Kolom dataset yang digunakan adalah Cust No., XCoord, YCoord, Demand, Ready Time dan Service Time.
5. Kendaraan diberangkatkan secara urut dan bergantian dengan selang waktu 5 satuan.
6. Kecepatan kendaraan konstan yaitu 2 satuan.
7. Kendaraan yang tiba sebelum waktu buka (Service Time) dikenakan pinalti waktu.
8. Kendaraan yang tidak memiliki cukup suplai saat melayani pelanggan dikenakan pinalti kapasitas.
9. Dataset diambil dari website VRPTW Benchmark Problems

1.4 Tujuan Penelitian

Tujuan Penelitian ini adalah sebagai berikut :

1. Menyelesaikan permasalahan VRPTW menggunakan algoritma genetika.
2. Mengetahui hasil fitness untuk setiap tahapan pengujian.
3. Mengetahui hasil fitness saat diuji menggunakan seleksi Elitis dan *Binary Tournament*.

2. LANDASAN KEPUSTAKAAN

2.1 *Vehicle Routing Problem With Time Windows* (VRPTW)

Kegiatan distribusi merupakan salah satu faktor penting dalam perusahaan karena terdapat banyak toko yang harus dikunjungi dalam mendistribusikan barangnya. Banyak sekali toko yang harus dikunjungi membuat perusahaan harus memilih rute distribusi yang paling tepat sehingga biaya yang dikeluarkan dalam kegiatan distribusi semakin minimum. Jika perusahaan salah memilih rute yang akan ditempuh, maka perusahaan tersebut harus merugi karena biaya yang dikeluarkan semakin besar. Permasalahan ini dikenal sebagai *Vehicle Routing Problem* (VRP). VRP merupakan permasalahan yang didesain untuk mencari rute kendaraan yang memiliki biaya paling kecil dengan depot sebagai tempat awal mulai dan selesai melakukan distribusi. Kendaraan pengangkut barang hanya membawa barang sesuai dengan kapasitas kendaraan dan hanya mengunjungi toko sekali selama proses distribusi. Permasalahan VRP yang dipengaruhi oleh jam buka toko atau waktu-waktu tertentu disebut *Vehicle Routing Problem With Time Windows* (VRPTW). VRPTW membuat kendaraan pengangkut barang mengunjungi toko pada jam yang telah ditentukan oleh toko tersebut atau dengan kata lain kendaraan harus berkunjung pada waktu toko tersebut buka.

2.1 Algoritma Genetika

Algoritma genetika merupakan tipe algoritma evolusi yang banyak diaplikasikan dalam berbagai bidang seperti biologi, fisika, sosial dan lain-lain. Hal ini terjadi karena kemampuan dari algoritma genetika yang dapat menyelesaikan berbagai permasalahan yang kompleks dan berkembang dengan pesat seiring perkembangan teknologi (Mahmudy, 2013).

Tahap awal dari algoritma genetika yaitu membangkitkan populasi awal secara acak. Populasi awal ini merupakan calon solusi yang terdiri dari kromosom-kromosom yang akan diregenerasi tiap iterasi. Pada setiap iterasi, kromosom tersebut dihitung nilai *fitness*-nya. Setelah iterasi, dihasilkan kromosom baru yang disebut *offspring*. *Offspring* dihasilkan dari penggabungan dua kromosom yang telah mengalami mutasi sebelumnya. Penggabungan dua kromosom ini menggunakan operator *crossover* atau penyilangan. Proses iterasi akan dilakukan sampai ditemukan solusi yang paling optimal atau sampai batas iterasi yang ditentukan sistem.

Proses dalam algoritma genetika adalah sebagai berikut :

1. [Inisialisasi] Membuat individu-individu acak yang terdiri dari kromosom tertentu. Kromosom merupakan solusi dari permasalahan
2. [Iterasi/looping] Melakukan proses tertentu selama belum menemui kondisi berhenti
 - a. [Reproduksi] Proses untuk menghasilkan keturunan dari individu-individu dalam populasi sebelumnya
 - b. [Evaluasi] Menghitung nilai *fitness* pada setiap kromosom. Kromosom dengan nilai *fitness* yang tinggi akan dipilih untuk menjadi calon solusi
 - c. [Seleksi] memilih individu dari kumpulan populasi dan *offspring*. Semakin tinggi nilai *fitness* dari suatu kromosom, semakin besar peluang kromosom tersebut untuk dipilih.

2.1.1 Representasi Kromosom

Dalam algoritma genetika, masalah riil harus diterjemahkan kedalam terminology biologi atau yang disebut juga representasi *chromosome*. Pengkodean (*encoding*) merupakan cara yang dapat digunakan untuk merepresentasikan *chromosome*. Jenis pengkodean bergantung dengan permasalahan yang terjadi.

Menurut sturktur dari pengkodean dibagi menjadi dua bagian yaitu pengkodean 1 dimensi dan 2 dimensi. Pengkodean 1 dimensi terdiri dari pengkodean biner,

oktal, heksadesimal, permutasi dan nilai sedangkan pengkodean 2 dimensi adalah pengkodean *tree* (Kumar, 2013).

a. Pengkodean Biner

Pengkodean biner merupakan bentuk yang paling umum yang setiap kromosom direpresentasikan dengan bilangan biner 0 dan 1. Pengkodean ini dapat digunakan untuk menyelesaikan permasalahan *knapsack* dengan 0 menandakan barang tersebut tidak ada dan sebaliknya. Contoh pengkodean biner dapat dilihat dalam Tabel 2.2 berikut :

Tabel 2.1 Pengkodean Biner

Kromosom1	001101011101
Kromosom2	1110010111001

b. Pengkodean Oktal

Pengkodean oktal merupakan pengkodean yang merepresentasikan kromosom dengan bilangan oktal 0-7. Contoh pengkodean oktal terdapat dalam Tabel 2.3 berikut :

Tabel 2.2 Pengkodean Oktal

Kromosom1	726141046512
Kromosom2	047112630421

c. Pengkodean Heksadesimal

Pengkodean heksadesimal menggunakan bilangan heksadesimal (0-9, A-F) sebagai representasi kromosomnya. Contoh pengkodean ini terdapat dalam Tabel 2.4 :

Tabel 2.3 Pengkodean Heksadesimal

Kromosom1	3B2104C
Kromosom2	AA523B9

d. Pengkodean Permutasi

Pengkodean permutasi dapat digunakan dalam menyelesaikan permasalahan dalam hal pengurutan seperti *travelling salesman problem* (TSP). Dalam pengkodean permutasi, kromosom direpresentasikan dengan nomor urut dari tujuan-tujuan yang akan dilalui. Tabel 2.5 merupakan contoh pengkodean permutasi.

Tabel 2.4 Pengkodean Permutasi

Kromosom1	5 7 2 1 4 9 3 6 10 8
Kromosom2	10 5 9 2 8 7 1 6 3 4

Terdapat beberapa operator *crossover* yang dapat digunakan pada pengkodean permutasi antara lain *partially mapped*

crossover (PMX), *cycle crossover* (OCX) dan *order crossover* (OX).

e. Pengkodean Nilai (*Value Encoding*)

Dalam pengkodean ini, masing-masing kromosom direpresentasikan dengan beberapa tipe nilai seperti integer, bilangan riil, karakter abjad, maupun berupa objek. Jika menggunakan bilangan tipe integer, maka dapat mengaplikasikan *crossover* yang sama dengan pengkodean biner. Pengkodean ini dapat digunakan pada permasalahan jaringan neural yang menggunakan pembobotan untuk input. Contoh pengkodean nilai dapat dilihat dalam Tabel 2.6.

Tabel 2.5 Pengkodean Nilai (*Value Encoding*)

Kromosom1	AHDGBCDCHDFKG
Kromosom2	1.34, 2.12, 0.11, 4.21

f. Pengkodean *Tree*

Pengkodean *tree* digunakan dalam pemrograman yang berhubungan dengan evolusi atau genetika. Kromosom direpresentasikan dengan gambar pohon.

2.1.2 Seleksi

Proses seleksi merupakan proses yang penting dalam algoritma genetika dimana proses ini menentukan kromosom yang akan digunakan dalam generasi selanjutnya. Terdapat beberapa metode seleksi yang dapat digunakan dalam algoritma genetika, antara lain:

a. Seleksi *Binary Tournament*

Seleksi ini bekerja seperti sistem turnamen yang mencari satu pemenang dari dua individu. Proses seleksi sebagai berikut:

1. Menentukan nilai n
2. Untuk setiap n , melakukan:
 - a. Memilih dua individu secara acak
 - b. Memilih individu dengan nilai *fitness* yang paling tinggi diantara dua individu yang dipilih

b. Seleksi Elitis

Seleksi elitis merupakan metode untuk mengambil individu dengan nilai *fitness* tertinggi yang selanjutnya akan digunakan menjadi generasi baru. Metode ini yaitu merangkai kromosom dengan urutan dari nilai *fitness* tertinggi hingga terendah lalu

memilih masing-masing n-kromosom teratas dari rangkaian tersebut.

c. Seleksi *Roulette Wheel*

Metode seleksi *roulette wheel* merupakan metode seleksi yang mengadopsi permainan *roulette wheel* dengan menempatkan kromosom pada bagian-bagian dari roda sesuai dengan nilai probabilitas. Nilai probabilitas didapatkan dari perhitungan nilai *fitness*. Semakin besar nilai probabilitas suatu kromosom maka bagian yang didapatkan dalam roda tersebut akan semakin besar. Proses seleksi *roulette wheel* adalah sebagai berikut:

1. Menghitung nilai *fitness* (fv) masing-masing individu.
2. Menghitung total *fitness* (Sf) dari seluruh individu.
3. Menghitung rata-rata *fitness* (Af) dari semua individu.
4. Menghitung *expected fitness* (Ef).
5. Menghitung probabilitas masing-masing individu berdasarkan nilai dari (Ef).
6. Menghitung nilai probabilitas kumulatif (*Probcum*).
7. Membangkitkan angka random (G) 0-*Probcum*.
8. Menyeleksi individu dengan jumlah nilai Ef individu sebelumnya dan juga individu tersebut (*probcum*) lebih besar dari nilai (G).
9. Kembali ke langkah ke-6 untuk melakukan perulangan sebanyak *popsi*.

d. Seleksi Ranking / *Rank Selection*

Metode seleksi ini bekerja berdasarkan perangkian nilai *fitness*. Setelah didapatkan ranking setiap kromosom, maka melakukan perhitungan nilai *fitness* menggunakan rumus baru. Setelah itu, menyeleksi kromosom menggunakan metode seleksi *roulette wheel*. Berikut merupakan proses *rank selection*:

1. Mengurutkan kromosom berdasarkan nilai *fitness* dari tertinggi hingga terendah.
2. Memberi ranking pada masing-masing kromosom.
3. Menghitung nilai *fitness* baru menggunakan rumus dibawah ini:

$$F = \max - (\max - \min) * \frac{\text{rank} - 1}{N_{pop} - 1}$$

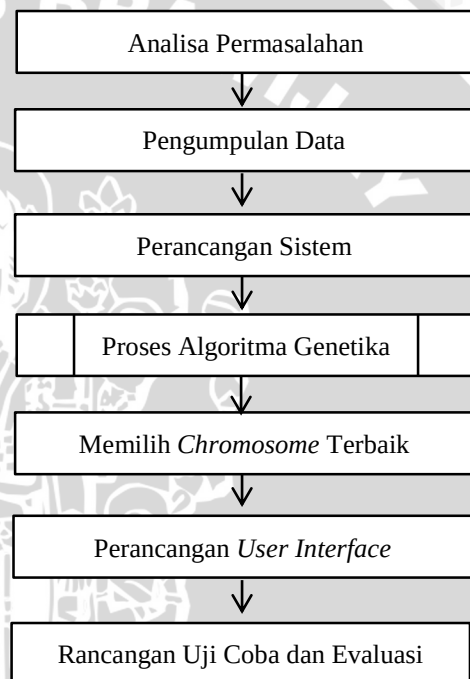
Dimana $1 < \max \leq 2$ & $\min = 2 - \max$

3. METODOLOGI

Bab metodologi membahas tentang metode yang digunakan dalam penelitian ini.

3.1 Tahapan Penelitian

Penelitian ini menggunakan pendekatan perancangan (*design*) yang mempunyai tahapan penelitian seperti yang digambarkan pada Gambar 3.1 berikut:



Gambar 3. 1 Diagram Alir Metodologi

3.1 Data Uji

Data yang digunakan dalam penelitian ini merupakan data yang berasal dari *website* VRPTW Benchmark Problems (<http://web.cba.neu.edu/~msolomon/problems.htm>). Dalam *website* tersebut terdapat 6 macam data yang berbeda. Data berdasarkan geografis yang acak terdapat dalam data R1 dan R2, data yang terkelompok terdapat dalam data C1 dan C2 sedangkan RC1 dan RC2 merupakan data campuran dari kedua dataset. R1, C1 dan RC1 adalah data yang menyediakan kapasitas kendaraan 200 sehingga hanya dapat melayani 5-10 pelanggan dalam sekali distribusi. Sedangkan R2, C2 dan RC2 menyediakan kapasitas lebih dari 600 sehingga dapat melayani lebih dari 30 pelanggan. Data yang

digunakan dalam penelitian ini adalah data dengan nomor R101. Dalam data R101, kolom *due date* tidak digunakan dan menggunakan kolom selain *due date* dalam perhitungan.

3.2 Algoritma yang digunakan

Penelitian ini menggunakan algoritma genetika yang menghasilkan hasil akhir berupa kromosom dengan nilai *fitness* terbaik. Kromosom dengan nilai *fitness* terbaik merupakan representasi dari urutan pelanggan yang dikunjungi dalam proses distribusi.

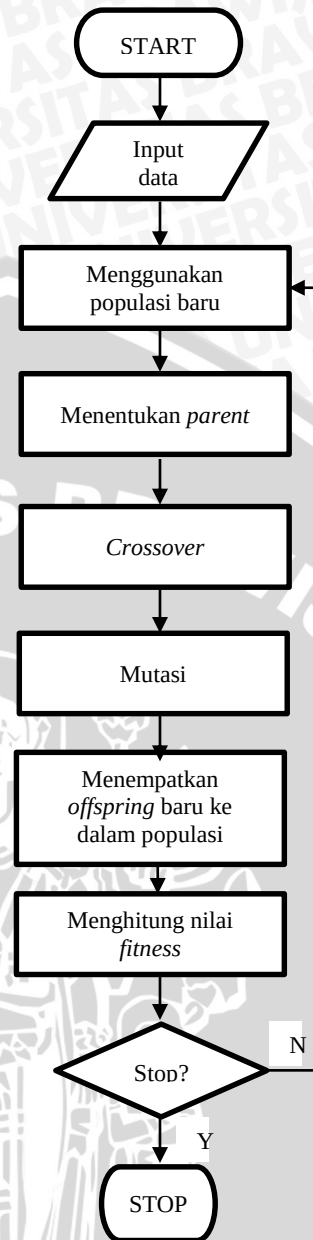
3.3 Kebutuhan Sistem

Kebutuhan sistem dalam bab ini digunakan untuk mengetahui kebutuhan yang digunakan peneliti dalam menyelesaikan pengembangan sistem. Kebutuhan ini meliputi *software* dan *hardware*. Untuk menunjang penelitian maka spesifikasi yang dibutuhkan adalah sebagai berikut:

1. Kebutuhan Hardware, yaitu:
 - Komputer/laptop
 - RAM 4/8 GB
2. Kebutuhan Software, yaitu:
 - Microsoft Office 2013
 - Netbean

4. PERANCANGAN

Penelitian ini menggunakan algoritma genetika untuk menyelesaikan permasalahan VRPTW pada distribusi produk pangan. Berikut merupakan siklus algoritma genetika yang digambarkan dalam Gambar 4.1.

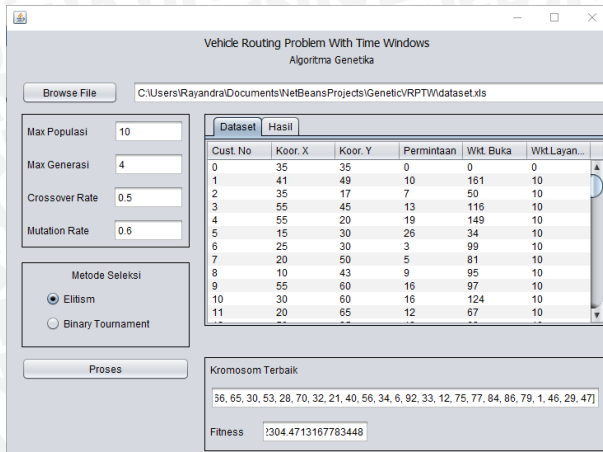


Gambar 4. 2 Diagram Alir Metodologi

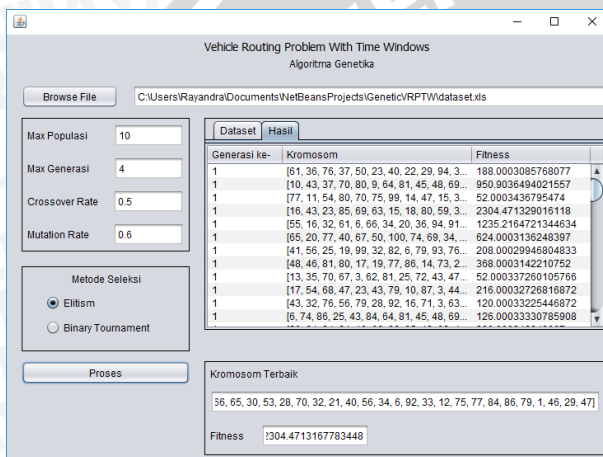
5. IMPLEMENTASI

Hasil dari implementasi optimasi *Vehicle Routing Problem with Time Window* (VRPTW) pada distribusi produk pangan menggunakan algoritma genetika berupa antar muka program yang terdiri dari dua bagian yaitu *input* dan *output* (hasil). Pada bagian *input*, pengguna dapat memasukkan ukuran populasi, banyaknya generasi, dan kombinasi Cr dan Mr. Sedangkan, pada bagian hasil dibagi menjadi bagian dataset dan bagian hasil. Bagian dataset akan muncul dataset yang digunakan dalam program, seperti pada Gambar 5.1 berikut. Sedangkan Gambar 5.2 merupakan bagian hasil, terdapat daftar kromosom yang ada dalam

proses algoritma genetika beserta nilai *fitness* dan generasi.



Gambar 5. 3 Tampilan Dataset



Gambar 5. 2 Tampilan Hasil

6. PENGUJIAN DAN ANALISA

Pada perancangan pengujian terdapat beberapa mekanisme pengujian untuk membandingkan beberapa kondisi yang menghasilkan nilai *fitness* yang terbesar. Beberapa mekanisme pengujian adalah sebagai berikut:

1. Pengujian untuk membandingkan nilai *fitness* menggunakan seleksi elitis dan seleksi *binary tournament*.
2. Pengujian untuk mengetahui ukuran populasi yang optimal.
3. Pengujian untuk mengetahui banyaknya generasi yang optimal.

4. Pengujian untuk mengetahui kombinasi *crossover* dan *mutation rate* yang optimal.
5. Pengujian untuk mengetahui besar nilai *fitness* jika menggunakan parameter terbaik.

6.1 Hasil dan Pembahasan Pengujian Perbandingan Metode Seleksi

Pengujian yang dilakukan pertama kali adalah menguji metode seleksi yang lebih baik sehingga akan digunakan untuk pengujian selanjutnya. Pengujian ini dilakukan untuk membandingkan metode seleksi elitis dan *binary tournament* dengan menggunakan ukuran populasi sebesar 50, banyaknya generasi sebesar 500, crossover dan *mutation rate* masing-masing sebesar 0.5. Sebanyak 100 data pelanggan digunakan dalam proses pengujian. Proses pengujian dilakukan sebanyak 10 kali untuk masing-masing metode seleksi. Hasil pengujian terdapat pada Gambar 6.1.

Grafik dari hasil pengujian perbandingan metode seleksi yang masing-masing dilakukan sebanyak 10 kali percobaan menunjukkan bahwa nilai *fitness* yang dihasilkan melalui seleksi elitis lebih stabil jika dibandingkan dengan nilai *fitness* yang dihasilkan melalui seleksi *binary tournament*. Dalam seleksi elitis, individu-individu dipilih berdasarkan nilai *fitness* tertinggi yang sebelumnya telah dilakukan *sorting*. Hal ini membuat seleksi elitis mempertahankan individu-individu dengan nilai *fitness* tertinggi untuk digunakan dalam generasi selanjutnya. Berbeda dengan seleksi elitis, metode seleksi *binary tournament* bekerja dengan cara membandingkan nilai *fitness* dari dua individu yang dipilih secara acak sehingga individu dengan nilai *fitness* yang rendah dapat lolos untuk generasi selanjutnya. Rata-rata *fitness* dari seleksi elitis juga lebih tinggi dibanding dengan rata-rata *fitness* yang dihasilkan oleh seleksi *binary tournament* yaitu 0.00059325420 untuk seleksi elitis dan 0.00027986203 untuk seleksi *binary tournament*. Dalam pengujian untuk membandingkan metode seleksi, diketahui bahwa metode seleksi elitis lebih baik daripada metode seleksi *binary tournament* sehingga pada pengujian selanjutnya akan menggunakan metode seleksi elitis.



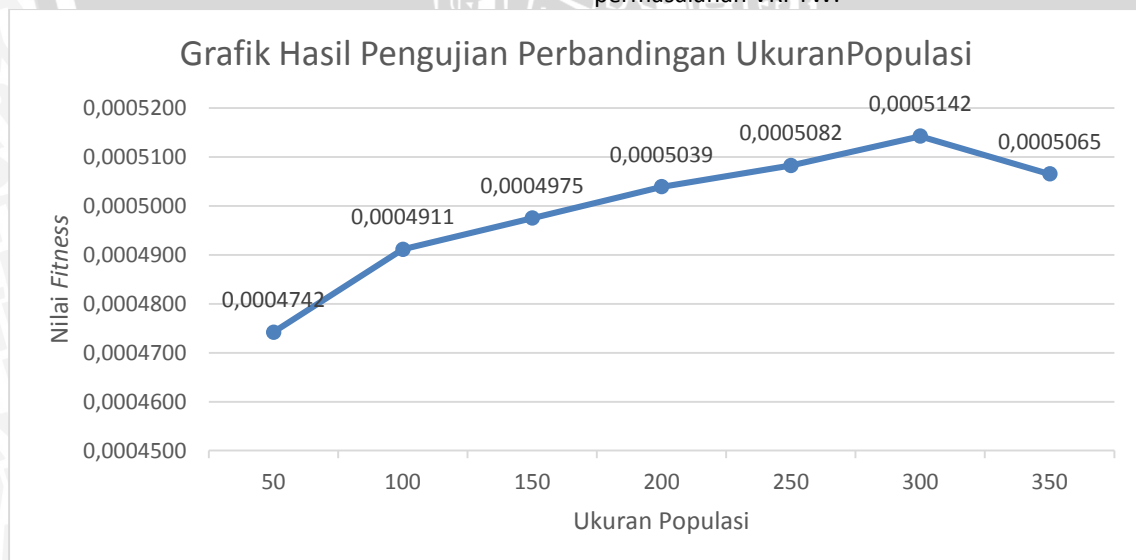
Gambar 6. 4 Grafik Hasil Pengujian Perbandingan Metode Seleksi

6.2 Hasil dan Pembahasan Pengujian Ukuran Populasi Optimal

Pengujian selanjutnya adalah pengujian untuk mengetahui ukuran populasi optimal yang dilakukan pada 100 pelanggan. Pengujian ini menggunakan jumlah generasi sebanyak 100 dengan crossover dan *mutation rate* masing-masing sebesar 0.5. Ukuran populasi yang digunakan adalah 50-350 dengan kelipatan 50. Percobaan ini dilakukan sebanyak 10 kali untuk setiap

ukuran populasi. Hasil pengujian terdapat pada Gambar 6.2.

Grafik menunjukkan hasil nilai *fitness* dari ukuran populasi 50 sampai populasi 350. Dapat dilihat bahwa nilai *fitness* naik secara signifikan dimulai dari populasi berjumlah 50 dan mencapai puncak pada populasi berjumlah 300 dengan nilai *fitness* 0.0005142. Tetapi nilai *fitness* mengalami penurunan pada populasi berjumlah 350 dengan nilai *fitness* 0.0005065. Dapat ditarik kesimpulan bahwa ukuran populasi 300 merupakan ukuran populasi yang paling optimal untuk permasalahan VRPTW.

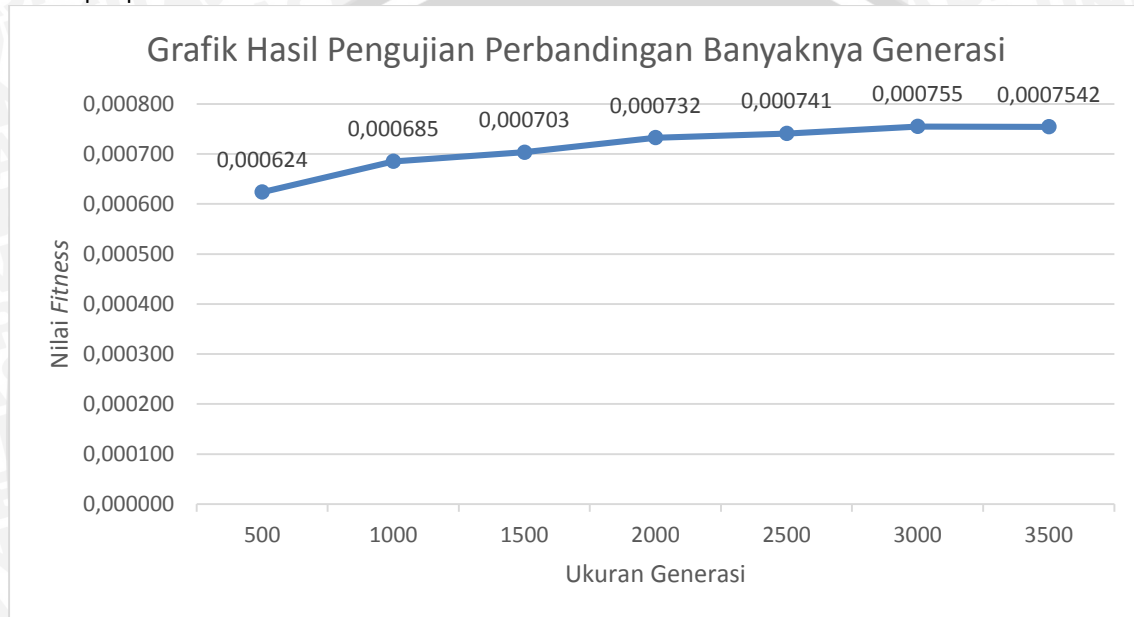


Gambar 6.1 Grafik Hasil Pengujian Perbandingan Ukuran Populasi

6.3 Hasil dan Pembahasan Pengujian Banyaknya Generasi Optimal

Pengujian ini dilakukan untuk mengetahui banyaknya generasi yang paling optimal dan menggunakan 100 data pelanggan. Pengujian ini menggunakan populasi sebesar 100 dengan *crossover* dan *mutation rate* masing-masing sebesar 0.5. Banyaknya generasi yang akan diuji adalah 500-3500 dengan kelipatan 500. Pengujian dilakukan sebanyak 10 kali pada masing-masing jumlah generasi. Hasil pengujian terdapat pada Gambar 6.3.

Berdasarkan grafik, diketahui bahwa nilai *fitness* mengalami kenaikan yang cukup signifikan dari jumlah generasi 500 ke 1000 dan terus mengalami kenaikan sampai jumlah generasi 2000. Akan tetapi, nilai *fitness* mulai stabil pada jumlah generasi 2500 dan mencapai puncak pada generasi 3000 dengan nilai *fitness* sebesar 0.0007550. Setelah generasi 3000, nilai *fitness* mengalami penurunan nilai. Sehingga, banyaknya generasi yang paling optimal dalam permasalahan VRPTW adalah 3000.

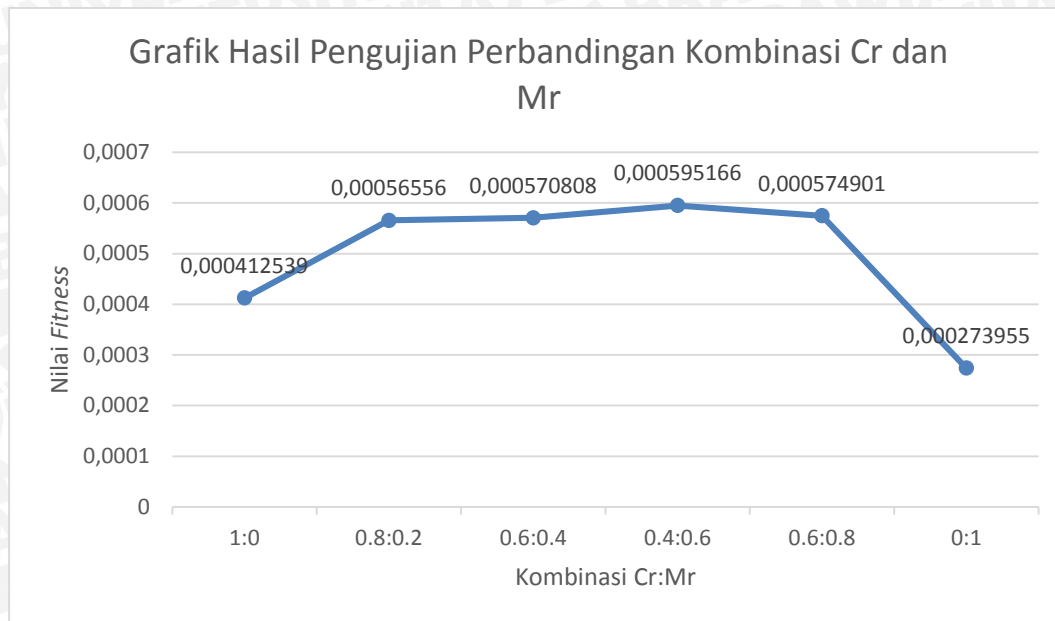


Gambar 6.2 Grafik Hasil Pengujian Perbandingan Banyaknya Generasi

6.4 Hasil dan Pembahasan Pengujian Kombinasi *Crossover* dan *Mutation Rate*

Pengujian ini dilakukan untuk mengetahui kombinasi *crossover rate* dan *mutation rate* untuk menghasilkan nilai *fitness* yang terbesar. Dalam pengujian ini, terdapat 100 data pelanggan yang akan dilakukan dalam proses pengujian. Jumlah populasi yang digunakan sebesar 50 dan jumlah generasi sebesar 500. Pengujian dilakukan 10 kali untuk setiap kombinasi. Hasil pengujian terdapat pada Gambar 6.4.

Grafik hasil pengujian kombinasi Cr dan Mr menunjukkan bahwa nilai *fitness* mengalami kenaikan yang signifikan dari kombinasi Cr:Mr 1:0 ke 0.8:0.2 tetapi setelah kombinasi kedua tidak menunjukkan kenaikan yang signifikan dan memuncak pada kombinasi 0.4:0.6. Nilai *fitness* justru turun pada kombinasi 0.6:0.4 dan mengalami penurunan yang signifikan pada kombinasi Cr:Mr 0:1. Hal ini menunjukkan bahwa algoritma genetika tidak dapat menghasilkan *fitness* yang optimal jika salah satu *crossover rate* atau *mutation rate* diberikan nilai 0. Kombinasi Cr:Mr yang paling optimal terletak pada kombinasi 0.4:0.6 dengan nilai *fitness* sebesar 0.00059516550.



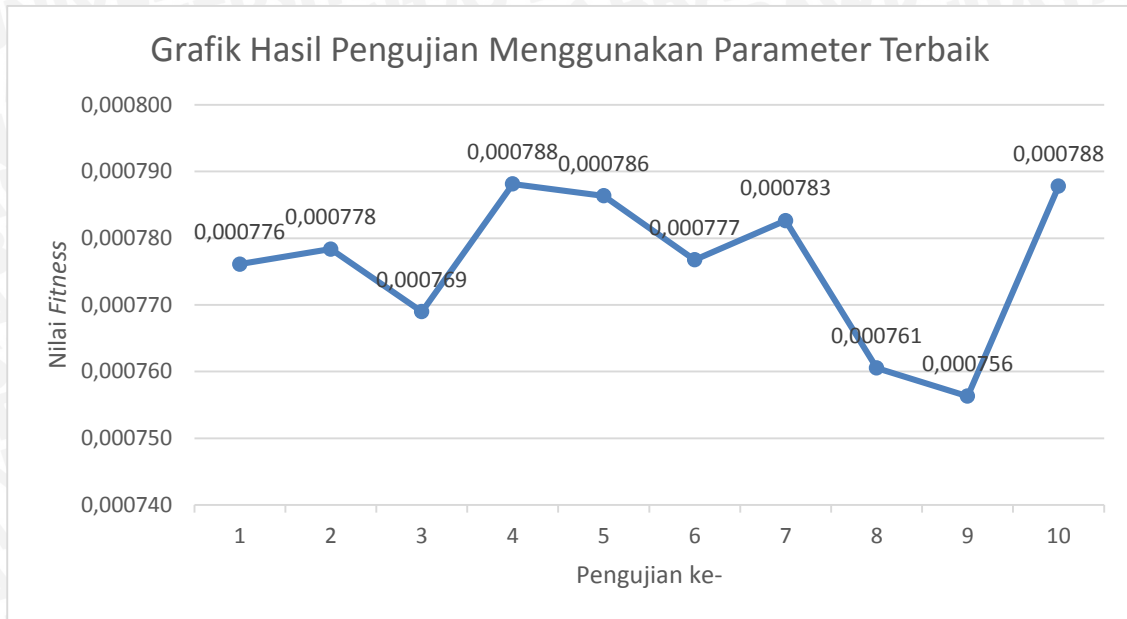
Gambar 6.3 Grafik Hasil Pengujian Perbandingan Kombinasi Cr dan Mr

6.5 Hasil dan Pembahasan Pengujian Menggunakan Parameter Terbaik

Pengujian ini menggunakan nilai terbaik dari parameter-parameter yang telah diuji sebelumnya. Pada pengujian ini, ukuran populasi yang digunakan sebesar 300 dengan generasi sebanyak 3000 generasi. Kombinasi *crossover rate* dan *mutation rate* sebesar 0.4:0.6 dan metode seleksi yang digunakan adalah metode seleksi elitis. Pengujian dilakukan sebanyak 10 kali dan diambil satu hasil dengan nilai *fitness* terbaik.

Nilai *fitness* yang didapatkan merupakan nilai *fitness* dari masing-masing percobaan saat melakukan pengujian. Dari 10 kali percobaan, nilai *fitness* mengalami kenaikan dan penurunan. Nilai *fitness* terbesar dihasilkan dari pengujian ke-4 dan ke-10 dengan nilai *fitness* sebesar 0.000788. Setelah pengujian ke-4, nilai *fitness* mengalami penurunan dan kenaikan sampai pengujian ke-10. Dalam pengujian ini, *fitness*

mengalami naik turun tetapi tidak melebihi 0.000788 sehingga dapat disimpulkan bahwa nilai *fitness* terbaik berada pada nilai 0.000788. Jalur yang dihasilkan pada percobaan ke-4 adalah 23, 81, 29, 35, 48, 43, 75, 49, 69, 27, 53, 26, 19, 42, 73, 67, 10, 8, 14, 16, 70, 28, 55, 71, 58, 52, 13, 47, 90, 89, 6, 56, 32, 4, 51, 68, 22, 12, 77, 79, 2, 18, 98, 17, 7, 37, 87, 72, 15, 83, 59, 99, 41, 39, 66, 3, 57, 21, 76, 24, 40, 54, 9, 33, 20, 1, 82, 91, 88, 96, 92, 46, 11, 93, 100, 36, 62, 5, 84, 44, 31, 60, 85, 45, 74, 50, 80, 78, 30, 25, 65, 34, 63, 94, 97, 95, 64, 61, 86, 38 dan jalur distribusi yang dihasilkan pada percobaan ke-10 adalah 87, 13, 21, 68, 64, 42, 95, 36, 58, 67, 65, 34, 19, 100, 85, 61, 82, 59, 99, 98, 74, 76, 80, 78, 88, 83, 17, 38, 43, 96, 92, 8, 41, 4, 81, 20, 70, 26, 55, 71, 2, 53, 27, 49, 90, 23, 51, 24, 57, 40, 28, 25, 30, 12, 29, 35, 62, 5, 91, 46, 10, 89, 72, 50, 63, 94, 6, 56, 31, 73, 77, 33, 48, 44, 16, 86, 15, 18, 37, 93, 11, 84, 14, 45, 22, 1, 52, 47, 75, 3, 79, 9, 32, 7, 97, 60, 69, 39, 66, 54.



Gambar 6.4 Grafik Hasil Pengujian Menggunakan Parameter Terbaik

7. PENUTUP

Dari hasil pengujian terhadap program optimasi VRPTW pada distribusi produk pangan menggunakan algoritma genetika dapat disimpulkan bahwa:

7.1 Kesimpulan

1. Permasalahan VRPTW dapat diselesaikan dengan menerapkan algoritma genetika untuk mencari rute distribusi yang optimal. Algoritma genetika membangkitkan individu-individu yang terdiri dari *node-node* pelanggan secara acak sehingga didapatkan rute-rute distribusi. Setelah membangkitkan individu-individu secara acak, maka algoritma genetika menghitung nilai *fitness* setiap individu dan menyeleksi individu-individu tersebut sehingga didapatkan rute distribusi yang paling optimal berdasarkan nilai *fitness*.
2. Berdasarkan pengujian terhadap metode seleksi, metode seleksi elitis lebih unggul dengan rata-rata *fitness* sebesar 0.00059325420 jika dibanding dengan metode seleksi *binary tournament* yang hanya menghasilkan rata-rata *fitness* sebesar 0.00027986203. Metode seleksi elitis lebih unggul dikarenakan metode ini mempertahankan individu-individu yang mempunyai nilai *fitness* tertinggi untuk lolos ke generasi selanjutnya. Sedangkan, metode seleksi *binary tournament* membandingkan nilai *fitness* dua individu yang dipilih secara acak sehingga memungkinkan untuk individu dengan nilai *fitness* rendah untuk lolos ke generasi selanjutnya atau individu dengan nilai

fitness tinggi tetapi tidak terpilih untuk mengikuti seleksi.

3. Nilai *fitness* yang dihasilkan dalam proses algoritma genetika bergantung pada parameter genetika yang diinputkan oleh pengguna. Pengujian dilakukan untuk mengetahui pengaruh tiap parameter terhadap nilai *fitness* yang dihasilkan. Pada pengujian ukuran populasi didapatkan bahwa nilai *fitness* mengalami kenaikan dari ukuran populasi 50 dan berhenti pada puncaknya di 300 lalu menurun di ukuran populasi 350. Nilai *fitness* yang didapatkan pada ukuran populasi 300 sebesar 0.0005065. Pengujian juga dilakukan untuk mencari banyaknya generasi yang menghasilkan nilai *fitness* terbesar. Nilai *fitness* mengalami kenaikan dari generasi 500 sampai generasi 3000 dan mengalami penurunan pada generasi 3500 sehingga banyaknya generasi yang optimal adalah 3000 dengan nilai *fitness* sebesar 0.0007550.
4. Kombinasi parameter genetika *crossover rate* dan *mutation rate* mempengaruhi hasil nilai *fitness*. Pada pengujian yang dilakukan dengan mengkombinasikan nilai Cr dan Mr didapatkan bahwa kombinasi terbaik adalah *crossover rate* sebesar 0.4 dan *mutation rate* sebesar 0.6 yang menghasilkan nilai *fitness* sebesar 0.0005951655.
5. Pada pengujian menggunakan parameter terbaik menghasilkan nilai *fitness* 0.000788. Nilai *fitness* yang dihasilkan merupakan nilai yang tertinggi jika dibandingkan dengan pengujian-pengujian sebelumnya, hal ini terjadi karena pengujian ini

menggunakan parameter-parameter terbaik yang didapatkan melalui pengujian sebelumnya

7.2 Saran

Aplikasi ini dapat dikembangkan lebih lanjut dengan menambahkan jumlah pelanggan maupun memperlebar *time window* dan dapat dikembangkan dengan membandingkan lebih banyak metode seleksi sehingga hasil yang didapatkan dapat lebih bervariasi.

8. DAFTAR PUSTAKA

- Harun, I.A, Mahmudy, W.F, dan Yudistira, N., Implementasi *Evolution Strategies* untuk Penyelesaian *Vehicle Routing Problem With Time Windows* pada Distribusi Minuman Soda XYZ, 2014. Jurnal Skripsi Mahasiswa PTIIK Universitas Brawijaya.
- Sundarningsih, D., Mahmudy, W.F, & Sutrisno, Penerapan Algoritma Genetika Untuk Optimasi *Vehicle Routing Problem With Time Window* (VRPTW) : Studi Kasus Air Minum Kemasan, 2015. Jurnal Skripsi Mahasiswa PTIIK Universitas Brawijaya.
- Saputri, M.W, Mahmudy, W.F & Ratnawati, D.E, Optimasi *Vehicle Routing Problem with Time Window* (VRPTW) Menggunakan Algoritma Genetika Pada Distribusi Barang, 2015. Jurnal Skripsi Mahasiswa PTIIK Universitas Brawijaya.
- Mahmudy, W.F, Modul Algoritma Evolusi, 2013. Modul Kulia Semester Ganjil 2013-2014 PTIIK Universitas Brawijaya.
- Gandhi, S., Khan, D., & Solanki, V.S, *A Comparative Analysis of Selection Scheme*, 2012. International Journal of Soft Computing and Engineering (IJSCE).
- Li, P. dkk, *Vehicle Routing Problem with Soft Time Windows Based on Improved Genetic Algorithm for Fruits and Vegetables Distribution*, 2015. Hindawi Publishing Corporation.
- Nazif, H., Lee, L.S, *Optimized Crossover Genetic Algorithm for Vehicle Routing Problem with Time Windows*, 2010. American Journal of Applied Sciences.
- Puljic, K., Manger, R., *A Distributed Evolutionary Algorithm With A Superlinear Speedup For Solving The Vehicle Routing Problem*, 2012. Computing and informatics.
- Hlaing, Z.C.S.S, Khine, M.A., *Solving Traveling Salesman Problem by Using Improved And Colony Optimization Algorithm*, 2011. International Journal of Information and Education Technology.
- Kumar, A., *Encoding Schemes in Genetic Algorithm*, 2013. International Journal of Advanced Research in IT and Engineering.
- Lukas, S., Anwar. T., Yuliani, W., Penerapan Algoritma Genetika Untuk *Traveling Salesman Problem* Dengan Menggunakan Metode *Order Crossover* dan *Insertion Mutation*, 2005. Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005).
- Alabsi, F., Naoum, R., *Comparison of Selection Methods and Crossover Operations using Steady State Genetic Based Intrusion Detection System*, 2012. Journal of Emerging Trends in Computing and Information Sciences.
- Mahmudy, Wayan Firdaus, *Improved Simulated Annealing For Optimization Of Vehicle Routing Problem With Time Windows (VRPTW)*, 2014. *Kursor Journal*, vol. 7, no. 3, page 109-116.