

BAB III

MODEL CVRP DAN PENYELESAIANNYA DENGAN MENGGUNAKAN ALGORITMA GENETIKA

3.1 Deskripsi Masalah

Pada bagian ini membahas deskripsi *Capacitated Vehicle Routing Problem* (CVRP) sebagai berikut. Terdapat sejumlah kendaraan yang akan mengirim barang ke sejumlah konsumen, dimana setiap kendaraan memiliki batasan kapasitas. Untuk mengangkut barang, setiap kendaraan memulai perjalanannya dari sebuah tempat, yang disebut depot, lalu melakukan perjalanan untuk memenuhi sejumlah permintaan setiap konsumen. Pada akhir perjalanan setiap harinya, setiap kendaraan harus kembali lagi ke depot.

Tujuan dari penyelesaian CVRP adalah menentukan rute setiap kendaraan dari depot ke konsumen dengan membawa muatan barang agar permintaan setiap konsumen terpenuhi dan total biaya perjalanan yang dibutuhkan oleh semua kendaraan adalah seminimum mungkin. Untuk menyelesaikan CVRP diperlukan metode penyelesaian yang tepat agar mendapatkan solusi optimal dengan waktu yang cepat.

Pada bab ini diturunkan model CVRP. Model tersebut dibangun untuk meminimumkan total biaya perjalanan bagi semua kendaraan. Kendala-kendala dari model tersebut menentukan aturan-aturan yang wajib dipenuhi pada CVRP. Pada bab ini juga menjelaskan cara kerja Algoritma Genetika untuk menyelesaikan CVRP. Penjelasan selengkapnya mengenai model CVRP dan cara kerja Algoritma Genetika dijelaskan pada Subbab 3.2 dan Subbab 3.3.

3.2 Model *Capacitated Vehicle Routing Problem* (CVRP)

Pada bagian ini diturunkan model optimisasi dari CVRP. Adapun asumsi-asumsi yang diambil pada pemodelan ini adalah sebagai berikut:

1. Hanya terdapat satu depot.
2. Setiap konsumen hanya dikunjungi tepat satu kali oleh satu kendaraan saja.
3. Jumlah permintaan konsumen dan kapasitas kendaraan diketahui.
4. Setiap kendaraan mempunyai batasan kapasitas yang sama.

5. Kendaraan yang tersedia cukup untuk mengirim semua permintaan konsumen.
6. Setiap konsumen terhubung satu sama lain dan jarak antar konsumen simetris, yaitu jarak dari konsumen i ke j sama dengan jarak konsumen j ke i .

Tahapan pertama pemodelan adalah mendefinisikan himpunan, parameter dan variabel keputusan yang digunakan dalam model ini. Berikut adalah himpunan-himpunan yang digunakan sebagai berikut:

V : himpunan konsumen

D : himpunan depot

K : himpunan kendaraan

Adapun parameter-parameter model didefinisikan sebagai berikut:

C_k : kapasitas kendaraan ke k

d_i : permintaan konsumen ke i

c_{ij} : biaya perjalanan dari i ke j , dimana $i, j \in V \cup D$

Variabel keputusan model CVRP didefinisikan untuk menentukan rute dari setiap kendaraan, dimana setiap rute kendaraan harus berawal dan berakhir di depot yang ditentukan. Variabel-variabel keputusan tersebut didefinisikan sebagai berikut:

$$x_{ij}^k = \begin{cases} 1, & \text{jika terdapat perjalanan dari } i \text{ ke } j \text{ dengan kendaraan } k \\ 0, & \text{yang lainnya} \end{cases}$$

$$y_i^k = \begin{cases} 1, & \text{jika kendaraan } k \text{ mengunjungi konsumen } i \\ 0, & \text{yang lainnya} \end{cases}$$

Selain himpunan, parameter dan variabel keputusan pada penelitian ini, didefinisikan kendala-kendala model CVRP sebagai berikut :

1. Setiap konsumen hanya dikunjungi tepat satu kali oleh satu kendaraan.

Kendala ini diekspresikan sebagai berikut:

$$\sum_{k \in K} y_i^k = 1, \forall i \in V$$

2. Rute dari setiap kendaraan berawal dari depot yang sama. Ini berarti, banyaknya kendaraan yang meninggalkan depot sama dengan banyaknya kendaraan yang tersedia. Kendala ini diekspresikan sebagai berikut:

$$\sum_{k \in K} y_i^k = |K|, \forall i \in D$$

3. Setiap kendaraan akan mengunjungi satu konsumen. Setelah itu kendaraan akan meninggalkan konsumen tersebut untuk melanjutkan perjalanan menuju konsumen selanjutnya dan akhir perjalanan menuju depot. Ini berarti, kendaraan yang menuju konsumen i akan meninggalkan konsumen i . Kendala ini diekspresikan sebagai berikut:

$$\sum_{j \in V} x_{ij}^k = \sum_{j \in V} x_{jl}^k = y_i^k, \forall i, l \in V \cup D, \forall k \in K$$

4. Setiap kendaraan memiliki batasan kapasitas sebesar C_k . Oleh karena itu total permintaan konsumen yang harus dipenuhi dalam satu rute oleh setiap kendaraan tidak boleh melebihi kapasitas kendaraan. Kendala ini diekspresikan sebagai berikut:

$$\sum_{i \in V} d_i y_i^k \leq C_k, \forall k \in K$$

5. Rute setiap kendaraan harus terhubung. Misalkan S adalah himpunan bagian dari V dan $|S|$ adalah banyaknya anggota di S . Jika suatu kendaraan mengunjungi $i \in S$ dan jika terdapat rute menuju $j \in S$, maka banyak rute yang dilalui oleh kendaraan tersebut di S tidak lebih dari $|S| - 1$. Kendala ini diekspresikan sebagai berikut:

$$\sum_{i \in S} \sum_{j \in S} x_{ij}^k \leq |S| - 1, \forall S \subseteq V, |S| \geq 2, \forall k \in K$$

Fungsi tujuan dari model CVRP adalah meminumkan total biaya perjalanan bagi semua kendaraan di K . Fungsi tujuan tersebut dituliskan sebagai berikut:

$$z = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}^k$$

Selengkapnya model CVRP dituliskan sebagai model optimisasi berikut:

Meminimumkan:

$$z = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}^k \quad (3.1)$$

berdasarkan

$$\sum_{k \in K} y_i^k = 1, \forall i \in V \quad (3.2)$$

$$\sum_{k \in K} y_i^k = |K|, \forall k \in K, \forall i \in D \quad (3.3)$$

$$\sum_{j \in V} x_{ij}^k = \sum_{j \in V} x_{ji}^k = y_i^k, \forall i \in V, \forall k \in K \quad (3.4)$$

$$\sum_{i \in V} d_i y_i^k \leq C, \forall k \in K \quad (3.5)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij}^k \leq |S| - 1, \forall S \subseteq V, |S| \geq 2, \forall k \in K \quad (3.6)$$

$$x_{ij}^k \in \{0,1\}, \forall i,j \in V, k \in K \quad (3.7)$$

$$y_i^k \in \{0,1\}, \forall i \in V, k \in K \quad (3.8)$$

3.3 Implementasi Algoritma Genetika pada CVRP

CVRP merupakan jenis permasalahan kombinatorial yang di klasifikasikan sebagai *NP-Hard Problem*. Ini berarti bahwa solusi *feasible* dari masalah tersebut sangat banyak. Oleh karena itu metode-metode penyelesaian analitik membutuhkan waktu yang sangat lama untuk menemukan solusi optimal. Metode-metode penyelesaian alternatif sangat dibutuhkan agar solusi optimal dapat diperoleh dengan cepat.

Pada penelitian ini, model CVRP pada Persamaan 3.1 sampai persamaan dengan Persamaan 3.8 diselesaikan menggunakan Algoritma Genetika. Algoritma Genetika adalah salah satu algoritma yang telah terbukti berhasil menyelesaikan masalah kombinatorial yang masuk dalam klasifikasi *NP-Hard Problem*. Pada Algoritma Genetika, solusi dari CVRP direpresentasikan sebagai kromosom-kromosom. Sebuah kromosom terdiri dari sekumpulan gen yang merepresentasikan konsumen. Pada tahap awal, kromosom-kromosom ini dibangkitkan secara acak dalam sebuah populasi. Kromosom-kromosom dalam suatu populasi dianalogikan sebagai himpunan-himpunan solusi CVRP. Kemudian himpunan-himpunan solusi dievaluasi tingkat penyesuaian diri terhadap lingkungannya dengan sebuah nilai yang dinamakan sebagai nilai *fitness*. Nilai *fitness* ini biasanya berhubungan erat dengan fungsi tujuan dari permasalahan yang dibahas. Semakin besar nilai *fitness* maka himpunan solusi tersebut akan memiliki peluang yang besar untuk menuju solusi optimum. Himpunan yang mempunyai nilai *fitness* rendah akan dihapus dan diganti dengan himpunan solusi baru yang mempunyai nilai *fitness* paling tinggi.

Nilai *fitness* yang tinggi dari sebuah solusi memberikan kesempatan untuk terseleksi menjadi induk dalam tahap reproduksi melalui perkawinan silang dengan solusi lain dalam populasi tersebut. Dalam tahap reproduksi perkawinan silang akan menghasilkan anak sebagai solusi baru, yang membawa beberapa sifat dari induknya. Sedangkan solusi dalam populasi yang tidak terseleksi dalam reproduksi tidak digunakan kembali. Dengan cara seperti ini, beberapa generasi dengan karakteristik yang bagus akan bermunculan dalam populasi tersebut. Semakin banyak perkawinan silang, maka akan semakin banyak kemungkinan solusi terbaik yang dapat diperoleh.

Melalui semua proses seleksi alam genetika, diharapkan menghasilkan solusi baru dengan nilai *fitness* yang lebih tinggi dari generasi sebelumnya sebagai generasi baru. Setelah didapat beberapa generasi dilakukan maka dihasilkan solusi optimal.

3.3.1 Representasi Kromosom

Satu kromosom merepresentasikan satu solusi. Pada kasus CVRP satu kromosom memuat sejumlah gen, dimana satu gen mewakili satu konsumen. Jadi panjang kromosom sama dengan banyaknya konsumen. Peneliti merepresentasikan kromosom dengan menggunakan *permutation encoding*. Representasi kromosom dengan menggunakan *permutation encoding* menggunakan kumpulan urutan nilai integer.

Misal jika terdapat 9 konsumen maka satu kromosom memuat 9 gen. Gen tersebut dinyatakan dengan nilai integer acak dari 1 sampai 9. Representasi ini dapat dilihat pada Gambar 3.1. Setiap kendaraan (K) akan mengirimkan sejumlah *box* untuk memenuhi permintaan sejumlah konsumen tersebut. Satu kromosom akan menghasilkan rute bagi setiap kendaraan. Untuk mendapatkan rute yang dilalui oleh setiap kendaraan pada satu kromosom dilakukan langkah-langkah sebagai berikut. Misal elemen ke- i dari suatu kromosom dengan panjang n adalah g_i .

1. Inisialisasi rute pertama K_1 dengan memasukan gen urutan pertama, menjadi $K_1 = \{g_1\}$.
2. Hitung total permintaan gen di K_1 .

3. Jika total permintaan gen kurang atau sama dengan kapasitas K_i , $i=1,2,3,..$ maka tambahkan gen urutan selanjutnya g_{i+1} , jika tidak maka kurangi gen urutan terakhir yang dimasukkan dari K_i dan masukan gen tersebut ke dalam rute baru K_{i+1} .
4. Lakukan langkah 3 sampai semua gen dalam satu kromosom sudah terpilih semua.

Kromosom	3	5	7	4	2	1	6	9	8
----------	---	---	---	---	---	---	---	---	---

Gambar 3.1 Representasi Kromosom

Pada kromosom Gambar 3.1 dimisalkan bahwa batasan kapasitas setiap kendaraan adalah 25 *box* dengan permintaan konsumen pertama sampai ke-9 masing-masing adalah 1,2,3,..9 *box*. Untuk menentukan rute setiap kendaraan dilakukan langkah 1 sampai langkah 4 seperti di atas. Untuk rute pertama masukan urutan konsumen pertama adalah 3 karena permintaannya $3 < 25$, maka tambahkan konsumen urutan selanjutnya dalam rute tersebut. Setelah langkah 3 diulang, rute pertama berisikan rute konsumen 3-5-7-4-2-1 dengan total permintaannya 22 *box*. Urutan konsumen selanjutnya yaitu 6 tidak bisa ditambahkan ke dalam rute pertama karena jika ditambahkan membuat total permintaan melebihi kapasitas kendaraan sehingga dimasukkan ke dalam rute kedua. Untuk rute kedua ulangi lagi langkah 3 sehingga rute kedua berisikan rute konsumen 6-9-8 dengan total permintaan 23 *box*. Semua konsumen dalam satu kromosom sudah terpilih semua sehingga satu kromosom menghasilkan rute pertama dan rute kedua. Setiap kendaraan akan memulai dan mengakhiri perjalanan dari sebuah tempat yaitu depot yang dimisalkan dengan nilai 0. Jadi rute pertama untuk kendaraan ke-1 adalah 0-3-5-7-4-2-1-0 dengan total permintaan 22 *box* dan rute kedua untuk kendaraan ke-2 adalah 0-6-9-8-0 dengan total permintaan 23 *box*.

3.3.2 Pembangkitan Populasi Awal

Proses pembangkitan populasi awal dilakukan secara acak. Populasi tersebut berisi beberapa kromosom yang telah didefinisikan sebelumnya. Banyaknya populasi awal yang dibangkitkan ditentukan oleh parameter *population_rate* (*pp*).

3.3.3 Nilai *Fitness*

Nilai *fitness* merepresentasikan kualitas sebuah kromosom sebagai solusi. Semakin tinggi nilai *fitness* maka solusi semakin baik. Tujuan dari penyelesaian masalah ini adalah untuk meminimumkan total biaya perjalanan semua kendaraan maka fungsi *fitness* yang digunakan adalah

$$F = \frac{1}{z}$$

dimana F adalah fungsi *fitness* dan z adalah fungsi tujuan dari CVRP . Nilai z semakin kecil akan membuat fungsi *fitness* semakin besar dan sebaliknya

3.3.4 Seleksi

Setelah nilai *fitness* dari setiap kromosom ditentukan, tahapan selanjutnya adalah seleksi. Seleksi merupakan proses untuk memilih kromosom yang akan menjadi induk pada proses reproduksi. Proses seleksi yang digunakan dalam penelitian ini *roulette wheel*. Proses *roulette wheel* merupakan metode seleksi dengan cara memilih calon induk berdasarkan nilai *fitness* yang dimilikinya. Semakin besar nilai *fitness*, maka semakin baik kromosom tersebut dan semakin besar pula kemungkinan untuk terpilih sebagai induk.

3.3.5 Crossover

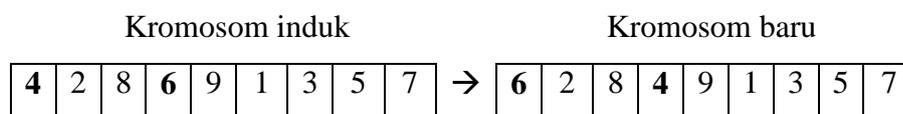
Crossover adalah operator yang membutuhkan dua kromosom yang akan menjadi induk untuk menghasilkan kromosom baru sebagai anak. Untuk masalah ini proses *crossover* yang digunakan adalah *order crossover*. Proses ini ditentukan oleh parameter *crossover_rate* (ρc) yang telah ditentukan dan akan dibangkitkan bilangan acak 0 sampai 1. Jika nilai bilangan acak kurang dari ρc maka dilakukan *crossover* dan jika tidak maka sebaliknya. Langkah-langkah dalam proses *order crossover* adalah sebagai berikut:

1. Tentukan dua kromosom yang akan dijadikan induk 1 dan induk 2 melalui proses seleksi.
2. Tentukan titik *crossover* caranya dengan pilih dua nilai integer secara acak dari 1 sampai n banyaknya konsumen. Misal terpilih titik *crossover* pada posisi konsumen urutan ke 6 dan 8 dari kromosom pada Gambar 3.1.

3.3.6 Mutasi

Mutasi yang digunakan pada penelitian ini adalah *swapping mutation*. Jumlah kromosom induk yang mengalami mutasi adalah seluruh kromosom baru hasil *crossover*. Gen yang akan dimutasi ditentukan oleh parameter yang dinamakan *mutation_rate* (ρm). Proses mutasi dilakukan dengan cara mengganti posisi gen yang terpilih dengan posisi gen lainnya yang dipilih secara acak.

Misal pilih ρm sebesar 0,1 dan gen menyatakan konsumen. Untuk setiap konsumen yang ada akan di bangkitkan nilai integer acak dari 0 sampai 1. Jika pada urutan konsumen pertama terpilih nilai integer acak 0,05 berarti nilai tersebut kurang dari 0,1 maka urutan konsumen pertama akan ditukarkan dengan konsumen urutan yang lain yang dipilih secara acak misal urutan ke-4 maka urutan konsumen pertama ditukar dengan urutan konsumen ke-4. Untuk urutan konsumen ke-2 sampai ke-9 terpilih nilai integer acak lebih dari 0,1 maka tidak terjadi proses mutasi. Tujuan dari mutasi ini untuk mendapat nilai *fitness* yang lebih baik sehingga mendapat rute perjalanan terpendek dalam penelitian ini. Hasil dari mutasi diilustrasikan pada Gambar 3.3



Gambar 3.3 *Swapping Mutation*

Setelah menyelesaikan proses mutasi maka satu generasi atau iterasi telah terselesaikan sehingga diperoleh kromosom baru hasil seleksi, *crossover* dan mutasi. Dengan adanya kromosom baru maka jumlah populasi menjadi dua kali lipat. Setiap kromosom tersebut akan dihitung kembali nilai *fitness*-nya untuk dipilih sebagai populasi baru generasi selanjutnya. Populasi yang terpilih untuk generasi selanjutnya sebanyak populasi awal berdasarkan nilai *fitness* tertinggi yang diurutkan. Untuk mendapatkan nilai *fitness* yang lebih baik maka tahapan generasi akan terus diulang sampai banyaknya generasi dan ukuran populasi yang ditentukan.