

## BAB III

### METODOLOGI PENELITIAN

Bab ini membahas metodologi yang akan digunakan untuk menyelesaikan masalah *Capacitated Vehicle Routing Problem with Time Windows* (CVRPTW) menggunakan Algoritma *Differential Evolution* (DE).

#### 3.1 Deskripsi Masalah

Penelitian ini membahas tentang *Capacitated Vehicle Routing Problem with Time Window* (CVRPTW). CVRPTW merupakan salah satu varian dari VRP dengan kendala tambahan berupa depot yang digunakan sebagai pusat distribusi sekaligus titik awal berangkat kendaraan dan titik akhir kendaraan. Terdapat  $k$  kendaraan berkapasitas tertentu yang membawa barang. Barang tersebut akan didistribusikan kepada sejumlah pelanggan. Setiap pelanggan memiliki permintaan terhadap  $n$  barang. Terdapat periode waktu pelayanan yang diizinkan. Pelanggan memiliki jumlah permintaan yang tidak melebihi kapasitas kendaraan dan kendaraan tersebut dapat mengantarkan sejumlah  $n$  barang permintaan dengan hanya melakukan pengiriman satu kali pada pelanggan. Kendaraan tersebut harus mengantarkan sejumlah  $n$  barang permintaan kepada pelanggan sesuai jumlah permintaan dan waktu pelayanan yang diminta pelanggan. Apabila waktu pengantaran barang terlambat dari waktu yang diminta pelanggan, maka depot akan membayar penalti kepada pelanggan sesuai waktu per menit keterlambatan pengantaran barang.

Tujuan dari penyelesaian CVRPTW adalah untuk menentukan rute optimal, yaitu rute dengan biaya distribusi minimum untuk mengantarkan barang ke sejumlah pelanggan menggunakan beberapa  $k$  kendaraan yang memuat beberapa jenis  $p$  barang dengan mempertimbangkan total jarak tempuh dan waktu tempuh. Dalam penelitian ini, CVRPTW akan diselesaikan menggunakan Algoritma *Differential Evolution* (DE).

#### 3.2 Tahapan Penelitian

Tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Studi Pustaka

Pada tahap ini, dilakukan studi pustaka terkait penelitian, termasuk penelitian terdahulu tentang *Capacitated Vehicle Routing Problem with Time Windows*, Algoritma *Differential Evolution*, serta penelitian serupa dalam mencari rute minimum dari berbagai sumber berupa jurnal nasional, jurnal internasional, *proceeding*, dan skripsi.

## 2. Pengumpulan Data

Pada tahap selanjutnya, diperlukan pengumpulan data yang diperlukan untuk analisis. Data yang akan digunakan dalam penelitian ini bersifat sekunder yang diperoleh dari website resmi dan sebagian data merupakan data *dummy* yang menyerupai data asli. Data yang digunakan dalam penelitian ini mencakup data lokasi depot, data lokasi pelanggan, data permintaan pelanggan, data jarak depot ke setiap pelanggan, data waktu tempuh dari depot ke pelanggan, dan data kendaraan yang digunakan.

## 3. Pembangunan Model Optimisasi

Pada tahapan ini akan dibangun model optimisasi dengan mendefinisikan asumsi, indeks, dan parameter model.

## 4. Penyelesaian Model Optimisasi

Pada tahap ini, model optimisasi akan diselesaikan menggunakan Algoritma *Differential Evolution*.

## 5. Validasi

Validasi dilakukan untuk memastikan bahwa program menghasilkan *output* atau keluaran yang bernilai sama dengan perhitungan manual.

## 6. Implementasi

Selanjutnya, model optimisasi dan teknik penyelesaian yang telah valid akan diimplementasikan pada permasalahan distribusi bahan bakar bensin pada SPBU di Kota Bandung dengan menggunakan Algoritma *Differential Evolution* (DE).

## 7. Penarikan Kesimpulan

Pada tahapan ini akan ditarik kesimpulan berdasarkan analisis hasil implementasi.

### 3.3 Penyelesaian Model Optimisasi menggunakan DEA

Permasalahan CVRPTW termasuk ke dalam kategori *NP-Hard Problem* yang berarti bahwa waktu komputasi yang digunakan akan semakin sulit seiring dengan meningkatnya ruang lingkup permasalahan. Hal ini menunjukkan bahwa solusi yang dihasilkan sangat banyak. Apabila menggunakan solusi eksak untuk menyelesaikannya, maka akan membutuhkan waktu yang lama untuk memecahkan masalah tersebut. Oleh karena itu, penelitian ini akan menggunakan metode penyelesaian *metaheuristic* untuk memperoleh penyelesaian yang lebih cepat dengan menggunakan Algoritma *Differential Evolution* (DE).

Algoritma *Differential Evolution* (DE) pertama kali diperkenalkan oleh Price dan Storn pada tahun 1997 (Price & Storn, 1997). Tahapan yang akan dilakukan untuk menyelesaikan CVRPTW menggunakan Algoritma DE akan melalui 4 tahapan, yaitu tahap inialisasi, tahap mutasi, tahap rekombinasi, dan seleksi. Dalam kasus ini, kita akan menggunakan metode *metaheuristic* yang dapat digunakan untuk menemukan solusi optimal dan dapat diaplikasikan untuk masalah yang besar dan kompleks. Tahap-tahap penyusunan algoritma dalam penelitian ini adalah sebagai berikut:

#### 1) Menentukan parameter kontrol

Tahap awal dalam penyusunan algoritma (DE) adalah menentukan parameter kontrol yang akan digunakan. Parameter kontrol yang perlu ditetapkan adalah ukuran populasi (NP), kontrol mutasi (F), dan pindah silang (Cr). Parameter kontrol adalah parameter yang digunakan untuk menentukan jumlah solusi awal pada populasi awal. Parameter kontrol yang sering digunakan untuk ukuran populasi adalah  $NP = 10 \times d$  dengan  $d$  adalah jumlah dimensi/pelanggan. Parameter mutasi berkisar antara 0-2 dengan nilai efektif antara 0,4-1. Untuk menentukan nilai parameter mutasi dan pindah silang terbaik, maka perlu dilakukan percobaan dengan menggunakan nilai efektif F dan Cr. (Anniza Zahara, 2011).

#### 2) Menentukan populasi awal

Populasi awal merupakan sebuah matriks yang berisi jumlah individu awal. Untuk kasus CVRPTW, setiap individu awal akan merepresentasikan rute perjalanan dengan kapasitas yang dibawa dan urutan pengantaran di

setiap rute. Populasi awal ditetapkan secara acak. Setiap individu awal dapat dicari dengan rumus sebagai berikut:

$$\text{Individu awal} = \text{batas}_{\text{bawah}} + (\text{batas}_{\text{atas}} - \text{batas}_{\text{bawah}}) \times \text{bilangan acak}$$

Pada proses ini, bilangan acak yang digunakan besarnya antara 0-1. Batas bawah dan batas atas masing-masing ditetapkan sebesar -1 dan 1. Jumlah individu awal tergantung pada parameter kontrol ukuran populasi. Pada kasus ini, parameter ukuran populasi sebanyak dua kali jumlah pelanggan, misalkan terdapat 33 pelanggan maka jumlah individu awalnya adalah 66.

Populasi awal yang masih berupa bilangan acak yang memiliki besar antara -1 sampai 1 sehingga populasi awal belum merepresentasikan urutan pengiriman ke setiap pelanggan. Oleh sebab itu, perlu dilakukan proses penerjemahan bilangan acak ke dalam bilangan yang merepresentasikan urutan berdasarkan besar kecilnya bilangan sehingga terdapat suatu bilangan bulat yang dapat dibaca sebagai urutan. (Kresentia, 2009).

### 3) Inisialisasi

Jumlah sampel dalam populasi adalah ukuran populasi atau jumlah vektor yang akan digunakan dalam iterasi evolusi diferensial. Jumlah sampel dalam populasi akan dihasilkan secara acak pada iterasi pertama dan setiap vektor serupa. Semua vektor akan diubah menjadi vektor mutan dan percobaan menggunakan mutasi dan proses rekombinasi. Vektor terdiri dari pengiriman urutan pelanggan dan jenis kendaraan untuk pengiriman.

### 4) Mutasi

Operasi mutasi adalah proses kedua dari mekanisme Algoritma DE. Persamaan (3.1) digunakan untuk menggabungkan tiga vektor yang dipilih secara acak menjadi vektor mutan. Proses mutasi dilakukan dengan cara mengambil 3 vektor acak secara random. Faktor penskalaan adalah konstanta dari  $[0, 2]$ .  $V_{i,G}$  adalah vektor mutan di mana

$$V_{ji,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (3.1)$$

$V_{ji,G+1}$  = vektor mutan

$X_{r1,G}, X_{r2,G}, X_{r3,G}$  = vektor acak

$F$  = faktor penskalaan

### 5) Rekombinasi

Rekombinasi menggabungkan solusi sukses dari iterasi sebelumnya seperti yang ditunjukkan pada Gambar 3.1. Vektor percobaan  $U_{ji,G+1}$  diperoleh dari Persamaan (3.1). Nilai posisi vektor dapat berupa nilai posisi target atau vektor percobaan, tergantung pada angka acak yang dihasilkan untuk posisi itu dan nilai laju *crossover*. Nilai pada posisi yang ditentukan akan disalin dari nilai itu posisi di vektor target apabila angka acak untuk itu posisi lebih besar dari laju *crossover*. Jika angka acak yang dihasilkan untuk posisi tersebut lebih rendah dari tingkat *crossover* maka nilai posisi pada posisi itu akan menjadi sama dengan nilai vektor mutan sehingga menghasilkan sebuah solusi baru (solusi trial). Selanjutnya, solusi trial ini dipecah berdasarkan kapasitas maksimum kendaraan dan batasan waktu. Rumus yang mewakili mekanisme ini adalah Persamaan (3.2).

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{jika } (randb(j) \leq CR) \text{ atau } j = rnbr(i) \\ X_{ji,G} & \text{lainnya} \end{cases} \quad (3.2)$$

di mana

$U_{ji,G+1}$  = vektor percobaan

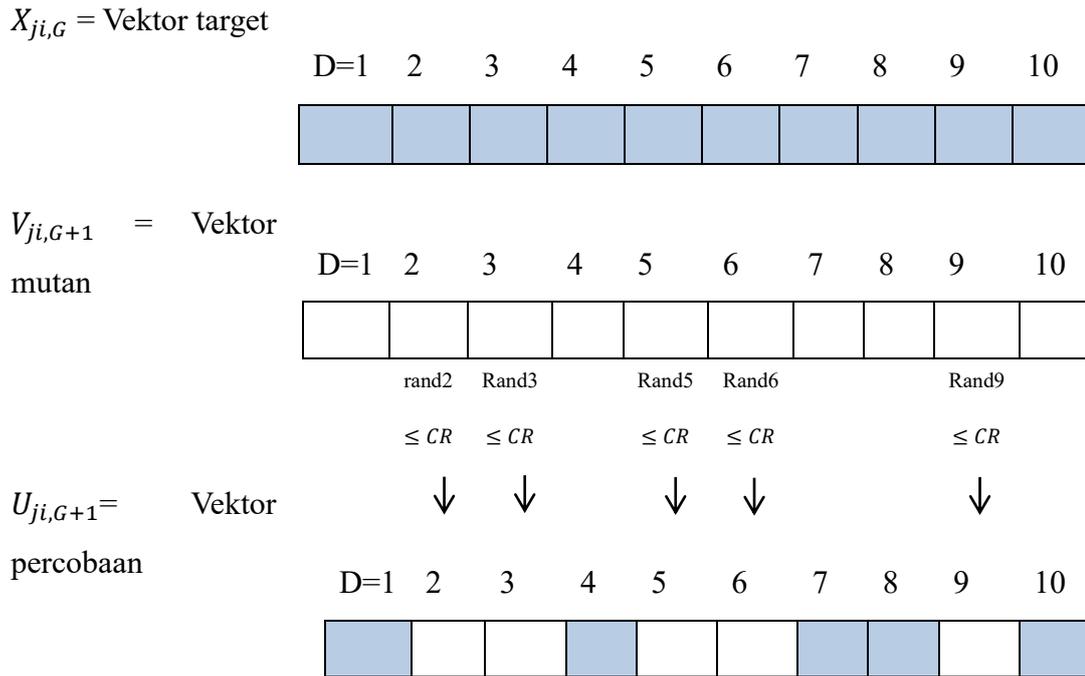
$V_{ji,G+1}$  = vektor mutan

$X_{ji,G}$  = vektor tujuan

$randb(j)$  = sebuah bilangan acak dari [0,1]

$rnbr(i)$  = sebuah bilangan bulat dari [1, 2, ..., D]

$Cr$  = tingkat persilangan [0, 1]



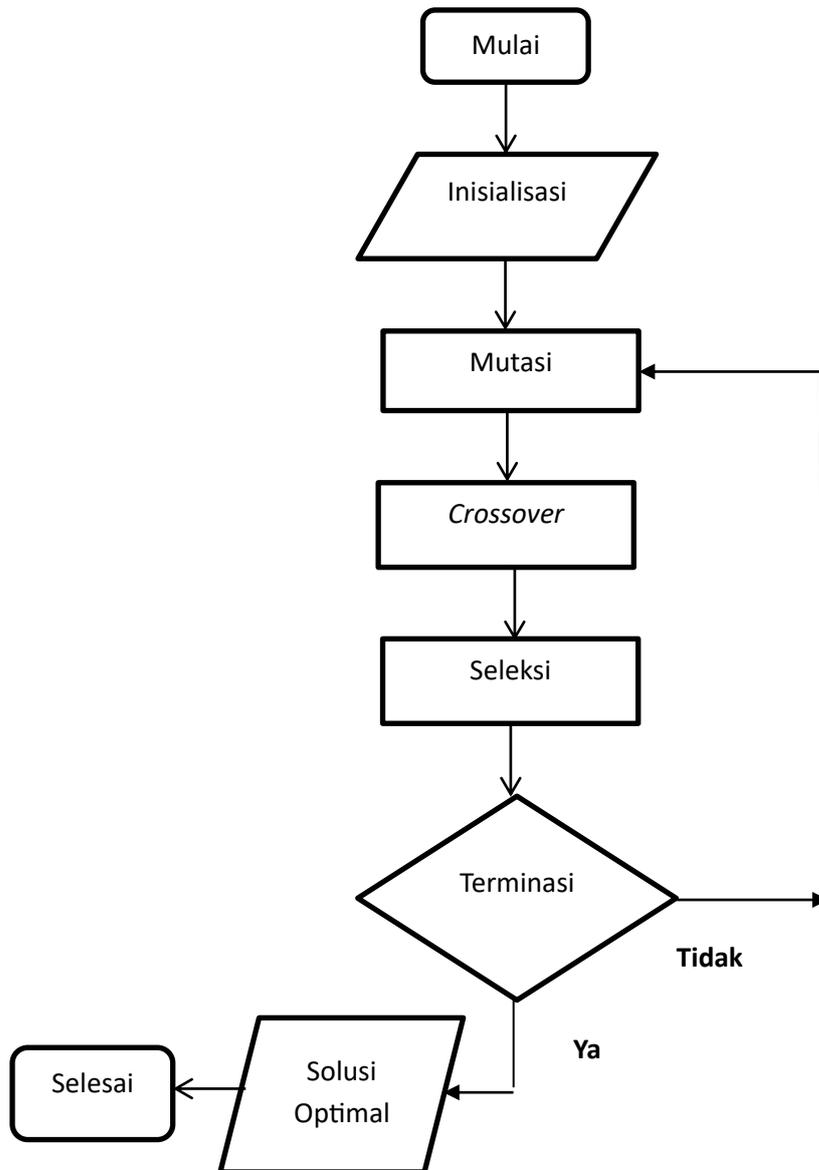
Gambar 3. 1. Contoh Rekombinasi

6) Seleksi

Pada tahap ini dilakukan perbandingan nilai *fitness* antara *parent* dan individu keturunan, maka yang paling unggul dipilih untuk masuk ke generasi berikutnya berdasarkan Persamaan (3.3).

$$x_{i,g+1} = \begin{cases} u_{i,g}, & \text{jika } f(u_{i,g}) \leq f(x_{i,g}) \\ x_{i,g}, & \text{lainnya} \end{cases} \quad (3.3)$$

Diagram alur implementasi algoritma DE dapat dilihat pada Gambar 3.2.



Gambar 3. 2. Alur Kerja Algoritma *Differential Evolution*