

## BAB III METODOLOGI PENELITIAN

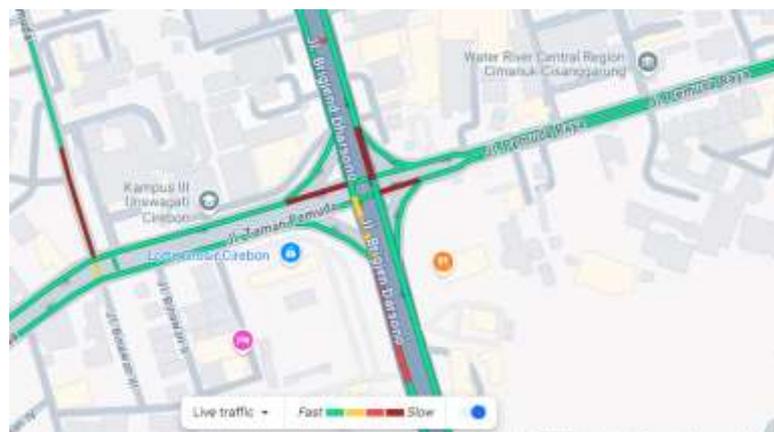
Bab ini membahas metodologi yang digunakan untuk pengendalian durasi dan fase lalu lintas di Simpang Jalan Bypass Kota Cirebon dengan menerapkan Metode *Hybrid Fuzzy Multi-Stage* dan *Ant Colony Optimization*.

### 3.1 Deskripsi Masalah

Penelitian ini membahas masalah optimalisasi durasi dan fase lalu lintas untuk mengatasi masalah kemacetan pada jalur tersebut. Adapun lokasi pada penelitian ini dilakukan di salah satu jalan di Kota Cirebon, yaitu di Simpang Jalan Bypass Kota Cirebon. Persimpangan ini dikelilingi oleh 4 jalan utama dengan 4 lampu lalu lintas, yaitu:

- *Jln. Pemuda*
- *Jln. Bridgen Darsono (dari Jawa)*
- *Jln. Bridgen Darsono (dari Jakarta)*
- *Jln. Terusan Pemuda*

Lokasi penelitian merupakan jalur pantura sehingga jalur ini dilewati oleh berbagai macam kendaraan. Kendaraan yang melewati persimpangan ini diantaranya terdiri dari kendaraan roda dua sampai kendaraan roda empat seperti mobil pribadi dan truk logistik. Kondisi lokasi penelitian pada persimpangan ini dapat lihat pada Gambar 3.1.



Gambar 3.1 Gmaps Persimpangan Jalan Pemuda Kota Cirebon

Perhitungan durasi lalu lintas pada persimpangan ini menggunakan sistem perhitungan tradisional berdasarkan penjumlahan dari durasi lampu merah. Pengaturan lalu lintas pada alokasi persimpangan ini terdiri beberapa sesi berikut:

- Sesi 1: Pukul 06:00-09:30
- Sesi 2: Pukul 09:30-17:30
- Sesi 3: Pukul 17:30-23:59
- Sesi 4: Pukul 23:59-06:00

Lokasi penelitian ini memiliki permasalahan utama dalam penentuan durasi dan fase lampu lalu lintas yang belum optimal. Saat ini, perhitungan durasi dilakukan dengan metode sederhana berupa penjumlahan dan pengurangan untuk menentukan waktu tiap fase lalu lintas. Meskipun metode ini mempertimbangkan kapasitas antrian kendaraan dan kondisi jalur, perhitungannya masih didasarkan pada data lama dan perkiraan kondisi persimpangan, sehingga kurang akurat dan belum optimal.

Untuk mengatasi permasalahan tersebut, penelitian ini menggunakan data terbaru dari observasi lapangan untuk menentukan fase dan durasi lampu lalu lintas yang lebih optimal. Sistem pengaturan lalu lintas yang dikembangkan bertujuan untuk mengurangi kemacetan dengan mengoptimalkan fase dan durasi lampu hijau di tiap ruas jalan. Penentuan fase lalu lintas dilakukan menggunakan logika *Fuzzy Mamdani Multi-Stage*, sedangkan *Improved Ant Colony Optimization* digunakan untuk menentukan durasi waktu tunggu yang optimal. Durasi optimal didefinisikan sebagai waktu nyala tiap fase sinyal lalu lintas yang ditentukan secara terkendali berdasarkan kondisi lokasi, volume kendaraan, dan karakteristik persimpangan, sehingga dapat meminimalkan antrian, mengurangi waktu tunggu, serta meningkatkan kelancaran arus lalu lintas.

### 3.2 Tahapan Penelitian

Tahapan penelitian ini adalah sebagai berikut.

#### 1. Studi Pustaka

Pada tahapan ini, dilakukan studi pustaka dengan cara mempelajari teori-teori yang berkaitan dengan penelitian ini dari buku, jurnal, maupun sumber lainnya. Untuk melakukan penelitian ini, beberapa konsep pembelajaran yang dipelajari meliputi pengenalan logika *fuzzy* dan ACO. Kemudian, dilakukan

pengumpulan informasi yang relevan dan sumber kajian ilmiah untuk pemahaman lebih lanjut mengenai Logika *Fuzzy Multi-Stage* dan *Fuzzy Mamdani*.

## 2. Pengumpulan Data

Pada tahapan ini, akan dilakukan pengumpulan data yang dibutuhkan untuk analisis. Data yang diperlukan pada penelitian ini adalah data yang berhubungan dengan durasi dan fase pada lalu lintas.

## 3. Penyelesaian Masalah

Pada tahapan ini akan diselesaikan masalah optimasi durasi dan fase lalu lintas di persimpangan dengan menggunakan metode *hybrid Logika Fuzzy Multi-Stage* dan *Improved Ant Colony Optimization*.

## 4. Penarikan Kesimpulan

Pada tahapan ini, akan ditarik kesimpulan berdasarkan analisis hasil yang didapatkan.

### 3.3 Asumsi dan Istilah

Asumsi yang ditetapkan pada penelitian ini adalah sebagai berikut:

#### 1. Kecepatan kendaraan konstan.

Kendaraan diasumsikan bergerak dengan kecepatan rata-rata yang tidak banyak berubah selama fase hijau.

#### 2. Siklus lampu tetap.

Total waktu satu siklus lampu lalu lintas (merah, kuning, hijau) tetap konstan, meskipun durasi masing-masing fase bisa berubah.

#### 3. Antrian kendaraan bersifat linear.

Kendaraan di ruas jalan dianggap membentuk antrian linear dengan waktu tunggu yang proporsional terhadap panjang antrian.

Adapun istilah-istilah yang digunakan dalam model optimasi penelitian terkait pengaturan lampu lalu lintas menggunakan metode *hybrid fuzzy mamdani multistage* dan IACO adalah sebagai berikut:

### 1. *Headway*

*Headway* merupakan jarak waktu rata-rata antara kendaraan yang keluar dari antrian, dihitung dari kendaraan pertama hingga kendaraan berikutnya. Satuan yang digunakan yaitu detik per kendaraan. *Headway* digunakan untuk menghitung jumlah kendaraan yang dapat keluar selama durasi lampu hijau ( $T_h$ ). Adapun rumus umum untuk menghitung jumlah kendaraan yang keluar adalah sebagai berikut:

$$Q_{out} = \frac{T_h}{h_w}$$

di mana:

- $Q_{out}$  : Jumlah kendaraan yang keluar.
- $h_w$  : *Headway* (detik per kendaraan).
- $T_h$  : Durasi lampu hijau (detik)

### 2. Fase

Fase merupakan siklus lampu lalu lintas untuk satu kelompok kendaraan yang memiliki prioritas jalan yang sama. Jenis fase dalam persimpangan terbagi menjadi 3 yaitu:

- a. Fase Hijau : Kendaraan pada arah tertentu diberi izin untuk bergerak
- b. Fase Merah : Kendaraan pada ruas tertentu wajib berhenti
- c. Fase Kuning : Peringatan siap-siap masa transisi sebelum lampu berubah.

### 3. Panjang Antrian

Panjang antrian merupakan jumlah kendaraan yang mengantri pada suatu jalur tertentu sebelum lampu berubah menjadi hijau. Satuan yang digunakan dalam panjang antrian ini adalah kendaraan. Panjang antrian ini akan dijadikan salah satu input utama dalam sistem *fuzzy* untuk menentukan tingkat urgensi dari fase hijau.

### 4. Waktu Tunggu

Waktu tunggu adalah rata-rata waktu yang dibutuhkan kendaraan dalam antrian sebelum dapat melintas pada fase hijau. Waktu tunggu diperlukan dalam menentukan prioritas fase hijau. Semakin tinggi waktu tunggu, semakin tinggi tingkat urgensi.

### 5. Siklus Lampu Lalu Lintas

Siklus adalah waktu total yang dibutuhkan untuk semua fase lampu lalu lintas pada suatu persimpangan. Siklus memiliki satuan detik. Adapun rumus umum dari siklus yaitu:

$$T_{\text{cycle}} = \sum_{i=1}^N T_h^i + T_{\text{kuning}} + T_{\text{merah}}$$

di mana:

- $T_h^i$  : Durasi hijau untuk jalur  $i$ .
- $T_{\text{kuning}}$  : Durasi fase kuning
- $T_{\text{merah}}$  : Durasi fase merah

#### 6. Tingkat Urgensi

Tingkat urgensi merupakan output dari sistem fuzzy bertingkat yang menunjukkan prioritas jalur tertentu untuk mendapatkan lampu hijau lebih cepat. Urgensi dibagi menjadi 3 kategori yaitu “*Kurang Urgen*”, “*Urgen*”, dan “*Sangat Urgen*”. Tingkat urgensi dalam penelitian ini digunakan untuk menentukan tambahan waktu hijau ( $\Delta T_h$ )

#### 7. Tambah Waktu Hijau

Tambah waktu hijau adalah durasi tambahan yang diberikan pada jalur tertentu untuk memperpanjang fase hijau berdasarkan jumlah kendaraan yang masih tersisa dalam antrian. Satuan dari tambah waktu hijau adalah detik. Tambahan waktu hijau dihitung berdasarkan output *fuzzy* dan digunakan dalam optimasi IACO.

#### 8. Pheromone

Pheromone adalah jejak virtual dalam IACO yang digunakan untuk menilai seberapa baik suatu solusi. Jika solusi memiliki kualitas yang baik (seperti waktu tunggu kendaraan yang lebih singkat), jejak pheromone pada solusi tersebut akan semakin banyak.

## 9. Visibilitas

Visibilitas adalah parameter dalam IACO yang menggambarkan kualitas solusi secara langsung. Visibilitas biasanya merupakan kebalikan dari total waktu tunggu. Adapun rumus umum dari visibilitas adalah:

$$\eta = \frac{1}{Z}$$

## 10. Probabilitas Pemilihan Solusi

Probabilitas adalah kemungkinan semut virtual memilih solusi tertentu dalam IACO, yang dihitung berdasarkan pheromone ( $\tau$ ) dan visibilitas ( $\eta$ ). Probabilitas menentukan arah semut dalam menemukan solusi terbaik. Adapun rumus umumnya yaitu sebagai berikut:

$$p_j = \frac{\tau_j^\alpha \cdot \eta_j^\beta}{\sum (\tau_k^\alpha \cdot \eta_k^\beta)}$$

di mana:

- $\alpha$  : Pengaruh jejak feromon
- $\beta$  : Pengaruh visibilitas

### 3.4 Model Optimasi

Fungsi tujuan dalam optimasi ini adalah untuk meminimalkan durasi waktu tunggu kendaraan di semua ruas jalan, yang dapat dinyatakan sebagai berikut:

$$Z = \sum_{i=1}^N J_{\text{hijau},i}$$

di mana:

- $Z$  : Total waktu tunggu kendaraan
- $J_{\text{hijau},i}$  : Waktu tunggu kendaraan pada ruas jalan  $i$  selama fase hijau
- $N$  : Jumlah ruas jalan

Adapun fungsi kendala dari model terdiri dari:

#### a. Batas Durasi Lampu Hijau

Durasi lampu hijau pada tiap ruas jalan harus berada dalam rentang minimum dan maksimum yang telah ditentukan, yaitu:

$$T_{\min} \leq T_h \leq T_{\max}$$

di mana:

- $T_{\min}$ : Batas minimum durasi lampu hijau.
  - $T_{\max}$ : Batas maksimum durasi lampu hijau.
- b. Total Waktu Siklus Tetap

Total durasi lampu hijau pada semua ruas jalan tidak boleh melebihi waktu siklus lampu lalu lintas  $T_{\text{cycle}}$ , yaitu:

$$\sum_{i=1}^N T_h \leq T_{\text{cycle}}$$

dengan  $T_{\text{cycle}}$  adalah waktu total satu siklus lampu lalu lintas, termasuk lampu merah, kuning, dan hijau.

- c. *Headway* kendaraan

Berdasarkan situasi studi kasus di Simpang Jalan Bypass Kota Cirebon, situasi *headway* yang digunakan adalah tidak semua kendaraan keluar selama lampu hijau. Pada situasi ini, jumlah kendaraan yang keluar ( $Q_{\text{out}}$ ) lebih sedikit dibandingkan jumlah kendaraan dalam antrian ( $Q$ ), karena durasi lampu hijau ( $T_h$ ) tidak cukup untuk mengosongkan antrian. Adapun rumus umum untuk menghitung kendaraan yang keluar sebagai berikut:

$$Q_{\text{out}} = \min\left(Q, \frac{T_h}{h_w}\right)$$

Sedangkan, kendaraan yang tersisa dapat dihitung sebagai berikut:

$$Q_{\text{remaining}} = Q - Q_{\text{out}}$$

Berdasarkan perhitungan sebelumnya, maka untuk menghitung total waktu tunggu ketika seluruh kendaraan dalam antrian akan keluar maka  $T_h \geq (Q - 1)h_w$ , total harapan waktu tunggu dari kendaraan selama waktu hijau dapat dinyatakan dalam persamaan (1).

$$J_{\text{hijau}} = \frac{Q(Q-1)h_w}{2} + \frac{\lambda((Q-1)h_w)^2}{2} + \frac{\lambda((Q-1)h_w)[\lambda((Q-1)h_w)-1]h_w}{2} \quad (1)$$

Namun, ketika  $T_h < (Q - 1)h_w$  maka tidak semua kendaraan dalam antrian awal dapat keluar dalam antrian. Total waktu tunggu ini dapat dinyatakan dalam persamaan (2).

$$J_{\text{hijau}} = \frac{Q_{\text{out}}(Q_{\text{out}}-1)h_w}{2} + (Q - Q_{\text{out}})(t_2 - t_1) + \frac{\lambda(T_h)^2}{2} \quad (2)$$

di mana:

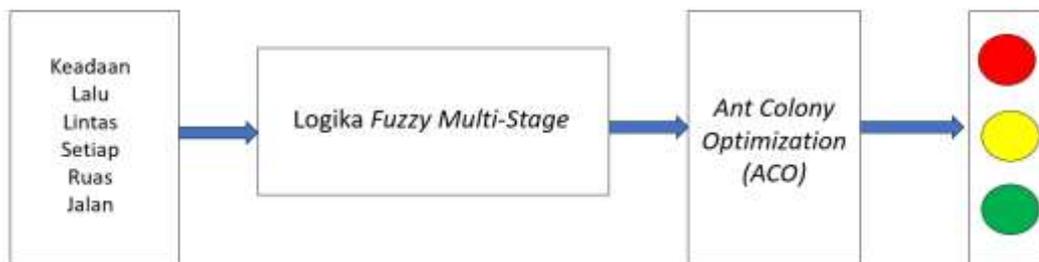
- $Q$  : Jumlah kendaraan dalam antrian.
- $Q_{out}$  : Jumlah kendaraan yang berhasil keluar selama fase hijau.
- $h_w$  : Jarak waktu antar kendaraan (headway).
- $T_h$  : Durasi waktu hijau.
- $\lambda$  : Kapasitas Jalan (Banyak Kendaraan/detik)

d. Hubungan *Fuzzy* dan *IACO*

*Fuzzy* pada penelitian ini berfungsi untuk memberikan keputusan fase mana yang akan mendapatkan durasi lampu hijau terlebih dahulu berdasarkan output dari *Fuzzy* Urgensi dan *Fuzzy* Tambah Hijau. *Fuzzy* Urgensi dalam penelitian dilakukan untuk menentukan prioritas jalur berdasarkan input dari lama antrian dan panjang antrian. Sedangkan *Fuzzy* Tambah Hijau dilakukan untuk menentukan fase mana yang akan mendapatkan tambahan durasi lampu hijau berdasarkan kendaraan yang tersisa dan lamanya lampu hijau.

*IACO* dalam penelitian bertujuan untuk mengoptimalkan durasi lampu hijau dengan meminimalkan total waktu tunggu kendaraan disetiap arah berdasarkan volume lalu lintas. Metode ini juga membantu untuk menentukan siklus sehingga mampu mengurangi antrian dalam lalu lintas.

Model sistem yang akan dibangun dapat dilihat pada Gambar 3.2.



Gambar 3.2 Skema Optimasi

### 3.5 Teknik Penyelesaian

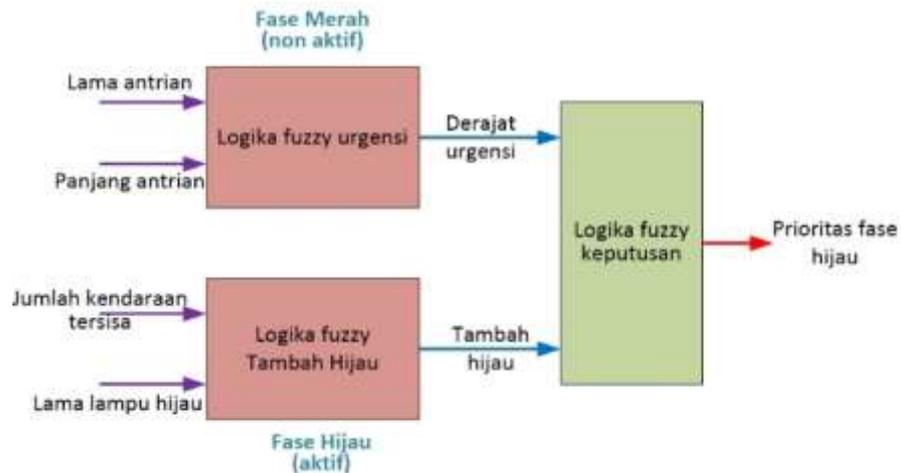
Penyelesaian permasalahan pada penelitian ini menggunakan metode *hybrid* antara *Logika Fuzzy Mamdani Multi-Stage* dan *Improve Ant Colony Optimization (IACO)* untuk mendapatkan fase dan waktu siklus yang optimum sesuai dengan data

Devi Dwi Andriyani, 2025

IMPLEMENTASI METODE *HYBRID FUZZY MAMDANI MULTI-STAGE* DAN *IMPROVED ANT COLONY OPTIMIZATION* PADA PENGATURAN LAMPU LALU LINTAS

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

lalu lintas terbaru. Skema model logika *fuzzy* bertingkat yang diusulkan dapat dilihat pada Gambar 3.3



Gambar 3. 3 Skema Logika *Fuzzy*

### 3.5.1 Logika *Fuzzy Mamdani Multi-Stage*

Logika *Fuzzy* yang digunakan dalam penelitian ini adalah Logika *Fuzzy Mamdani Multi-Stage* merujuk pada penelitian (Ridwan & dkk, 2022). Metode ini memiliki tiga tahapan logika *Fuzzy*, yaitu tahap pertama Logika *Fuzzy Urgensi* dan tahap kedua Logika *Fuzzy Tambah Hijau* kemudian, dari output pada tahap pertama dan kedua akan dilanjutkan pada tahap ketiga yaitu Logika *Fuzzy Keputusan*. Logika *Fuzzy urgensi* adalah logika *fuzzy* yang nilai inputnya berasal dari jalan pada saat fase merah. Nilai input dari fase merah ini digunakan untuk menentukan fase berdasarkan tingkat urgensinya. Sedangkan nilai input logika *fuzzy TambahHijau* berasal dari ruas jalan yang mengalami fase hijau. Nilai input pada tahap kedua ini berfungsi untuk menentukan fase mana yang memerlukan tambahan waktu hijau. Kemudian Setelah mendapatkan output dari *fuzzy* tahap 1 dan *fuzzy* tahap 2 maka akan dilanjutkan pada *fuzzy* keputusan untuk menentukan fase dalam lalu lintas tersebut berdasarkan kedua tahap logika *fuzzy* sebelumnya. Berikut ini merupakan langkah-langkah dari Logika *Fuzzy Mamdani Multi-Stage*:

#### 1. Penentuan Variabel: Masukan dan Keluaran

##### 1) Tahap 1: *Fuzzy Urgensi*

Input:

- Panjang Antrian: Kondisi panjang kendaraan di ruas jalan (contoh: Tidak Padat, Cukup Padat, Padat).
- Lama Antrian: Waktu kendaraan menunggu dalam antrian (contoh: Tidak Lama, Cukup Lama, Lama).

Output:

- *Derajat Urgensi*: Tingkat kepentingan untuk memberikan prioritas lampu hijau (contoh: *Kurang Urgen, Urgen, Sangat Urgen*).

## 2) Tahap 2: *Fuzzy* Tambah Hijau

Input:

- Kendaraan Tersisa: Jumlah kendaraan yang belum keluar dari antrian saat lampu hijau (contoh: Sedikit, Sedang, Banyak).
- Lama Hijau: Durasi lampu hijau yang sudah menyala (contoh: Tidak Lama, Cukup Lama, Lama).

Output:

- *Tambahan Hijau*: Durasi tambahan waktu hijau (contoh: Sedikit Lama, Cukup Lama, Lama).

## 3) Tahap 3: *Fuzzy* Keputusan

Input:

- *Fuzzy* Urgensi
- *Fuzzy* Tambah Hjaiu

Output:

- Skala prioritas (*Tidak Berlaih, Sedikit Beralih, dan Beralih*)

## 2. *Fuzzifikasi*

Pada tahap ini, data masukan dari variabel yang telah ditentukan (misalnya, panjang antrian) diubah menjadi nilai *fuzzy* berdasarkan fungsi keanggotaan. Fungsi keanggotaan ini menunjukkan seberapa besar nilai input termasuk dalam kategori tertentu, seperti "Tidak Padat," "Normal," atau "Sangat Padat." Dengan menggunakan fungsi keanggotaan segitiga untuk menentukan derajat keanggotaan input terhadap nilai *fuzzy*, fungsi ini didefinisikan dengan tiga parameter utama yaitu:

- 1)  $a$  : Batas awal, di mana nilai derajat keanggotaan mulai naik dari 1.

- 2)  $b$  : Puncak segitiga, di mana nilai derajat keanggotaan mencapai 1.
- 3)  $c$  : Batas akhir, di mana nilai derajat keanggotaan mulai turun ke 0.

Fungsi keanggotaan segitiga dirumuskan sebagai berikut:

$$\mu(x) = \begin{cases} 0, & \text{jika } x \leq a \text{ atau } x \geq c, \\ \frac{x-a}{b-a}, & \text{jika } a < x < b, \\ 1, & \text{jika } x = b \\ \frac{c-x}{c-b}, & \text{jika } b < x < c. \end{cases}$$

### 3. Pembentukan Aturan Fuzzy (*Fuzzy Rule Base*)

Dalam sistem pengendalian lampu lalu lintas berbasis *fuzzy*, aturan *fuzzy* berfungsi sebagai dasar pengambilan keputusan dalam menentukan output berdasarkan variabel input yang diberikan. Aturan ini dibentuk menggunakan aturan berbasis *IF-THEN*, yang menghubungkan setiap kondisi input dan output yang sesuai. Dalam penelitian ini, aturan *fuzzy* dikategorikan ke dalam dua tahap utama, yaitu *Fuzzy Urgensi* dan *Fuzzy Tambah Hijau*.

Pembentukan aturan *fuzzy* dalam penelitian ini dilakukan berdasarkan data kondisi pada lalu lintas lokasi penelitian. Aturan-aturan *fuzzy* disusun dengan mempertimbangkan variabel-variabel yang paling berpengaruh terhadap kebutuhan penentuan tingkat urgensi lalu lintas, yakni pada tahap1 *Fuzzy Urgensi* terdiri dari variabel panjang antrian kendaraan dan lama antrian yang dialami tiap jalur. Kedua variabel tersebut akan dijadikan output utama dalam penentuan *fuzzy* urgensi. Aturan *fuzzy* dibentuk berdasarkan pola hubungan logi yaitu, semakin banyak kendaraan menantri dan semakin lama waktu tunggu kendaraan, maka semakin tinggi urgensinya.

Selanjutnya, pada tahap 2 *Fuzzy Tambah Hijau* terdiri dari variabel input berupa banyak kendaraan tersisa dan durasi lampu hijau. Aturan *fuzzy* tahap 2 dibentuk dengan pertimbangan jika kendaraan yang tersisa banyak dan lama hijau sebentar maka akan memerlukan tambahan durasi lampu hijau, begitupun dengan aturan selanjutnya. Pembentukan aturan-aturan ini bertujuan untuk meningkatkan kelancaran arus lalu lintas secara menyeluruh. Dengan kata lain, aturan *fuzzy* yang digunakan dalam penelitian ini bersifat adaptif terhadap kondisi lalu lintas secara realistis pada data di lapangan.

Aturan *fuzzy* dalam setiap tahap ditentukan menggunakan operator *AND*, sehingga setiap kombinasi dari kondisi input menghasilkan suatu keputusan output tertentu. Adapun aturan *fuzzy* yang diperoleh sebagai berikut:

**a.** Tahap 1: Fuzzy Urgensi

[R1] ***IF*** Panjang Antrian ***IS*** *TidakPadat* ***AND*** Lama Antrian ***IS*** *TidakLama* ***THEN*** Derajat Urgensi ***IS*** *Kurang Urgen*.

[R2] ***IF*** Panjang Antrian ***IS*** *TidakPadat* ***AND*** Lama Antrian ***IS*** *CukupLama* ***THEN*** Derajat Urgensi ***IS*** *Kurang Urgen*.

[R3] ***IF*** Panjang Antrian ***IS*** *TidakPadat* ***AND*** Lama Antrian ***IS*** *Lama* ***THEN*** Derajat Urgensi ***IS*** *Kurang Urgen*.

[R4] ***IF*** Panjang Antrian ***IS*** *CukupPadat* ***AND*** Lama Antrian ***IS*** *TidakLama* ***THEN*** Derajat Urgensi ***IS*** *Kurang Urgen*.

[R5] ***IF*** Panjang Antrian ***IS*** *CukupPadat* ***AND*** Lama Antrian ***IS*** *CukupLama* ***THEN*** Derajat Urgensi ***IS*** *Kurang Urgen*.

[R6] ***IF*** Panjang Antrian ***IS*** *CukupPadat* ***AND*** Lama Antrian ***IS*** *Lama* ***THEN*** Derajat Urgensi ***IS*** *Urgen*.

[R7] ***IF*** Panjang Antrian ***IS*** *Padat* ***AND*** Lama Antrian ***IS*** *TidakLama* ***THEN*** Derajat Urgensi ***IS*** *Urgen*.

[R8] ***IF*** Panjang Antrian ***IS*** *Padat* ***AND*** Lama Antrian ***IS*** *CukupLama* ***THEN*** Derajat Urgensi ***IS*** *Urgen*.

[R9] ***IF*** Panjang Antrian ***IS*** *Padat* ***AND*** Lama Antrian ***IS*** *Lama* ***THEN*** Derajat Urgensi ***IS*** *Sangat Urgen*.

**b.** Tahap 2: Fuzzy Tambah Hijau

[R1] ***IF*** Kendaraan Tersisa ***IS*** *Sedikit* ***AND*** Lama Hijau ***IS*** *TidakLama* ***THEN*** Tambahan Hijau ***IS*** *Sedikit Lama*.

[R2] ***IF*** Kendaraan Tersisa ***IS*** *Sedikit* ***AND*** Lama Hijau ***IS*** *CukupLama* ***THEN*** Tambahan Hijau ***IS*** *Sedikit Lama*.

[R3] ***IF*** Kendaraan Tersisa ***IS*** *Sedikit* ***AND*** Lama Hijau ***IS*** *Lama* ***THEN*** Tambahan Hijau ***IS*** *Sedikit Lama*.

[R4] ***IF*** Kendaraan Tersisa ***IS*** *Sedang* ***AND*** Lama Hijau ***IS*** *TidakLama* ***THEN*** Tambahan Hijau ***IS*** *Cukup Lama*.

- [R5] **IF** Kendaraan Tersisa **IS** Sedang **AND** Lama Hijau **IS** CukupLama **THEN** Tambahan Hijau **IS** Cukup Lama.
- [R6] **IF** Kendaraan Tersisa **IS** Sedang **AND** Lama Hijau **IS** Lama **THEN** Tambahan Hijau **IS** Lama.
- [R7] **IF** Kendaraan Tersisa **IS** Banyak **AND** Lama Hijau **IS** TidakLama **THEN** Tambahan Hijau **IS** Lama.
- [R8] **IF** Kendaraan Tersisa **IS** Banyak **AND** Lama Hijau **IS** CukupLama **THEN** Tambahan Hijau **IS** Lama.
- [R9] **IF** Kendaraan Tersisa **IS** Banyak **AND** Lama Hijau **IS** Lama **THEN** Tambahan Hijau **IS** Lama.

#### 4. Inferensi *Fuzzy Mamdani*

Inferensi *fuzzy* Mamdani digunakan untuk memperoleh keluaran *fuzzy* berdasarkan kombinasi aturan IF-THEN dan derajat keanggotaan input. Metode ini populer karena perseptif dan mudah diinterpretasikan, terutama dalam sistem kendali lalu lintas yang kompleks. Metode Mamdani sering juga dikenal dengan nama Metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan output, diperlukan 4 tahapan (J & Langi, 2017):

##### 1. Pembentukan Himpunan *Fuzzy* (Fuzzifikasi)

Dalam melakukan *fuzzifikasi*, penulis menenrukan fungsi keanggotaan beserta satuannya untuk *input* dan *output*. Fungsi keanggotaan berdasarkan faktor-faktor yang penulis gunakan untuk mengoptimasi durasi dan fase lalu lintas sebagai fungsi keanggotaan untuk input yaitu lama antrian, panjang antrian, dan sisa kendaraan. Sementara derajat urgensi dan derajat tambahan hijau dijadikan sebagai fungsi keanggotaan untuk *output*. Adapun fungsi keanggotaan yang digunakan dalam penelitian ini adalah fungsi keanggotaan segitiga, dimana dalam menentukan batas-batasnya ialah dengan dibagi 3 dari domain semestanya. Adapun rumusnya yaitu sebagai berikut:

$$\mu_{kondisi}(x) = \begin{cases} 0, & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{b-x}{c-b}, & b \leq x \leq c \end{cases}$$

## 2. Aplikasi Fungsi Implikasi (Aturan)

Setiap aturan *fuzzy* berbentuk logika IF-THEN. Pada tahap ini, digunakan operator minimum (min) untuk menentukan seberapa kuat sebuah aturan aktif, berdasarkan derajat keanggotaan dari input. Contohnya dapat dilihat pada Tabel 3.1 kasus lalu lintas berikut.

Tabel 3. 1 Aturan *Fuzzy*

		Lama Antrian		
		Tidak Lama	Cukup Lama	Lama
Panjang Antrian	Tidak Padat	Kurang Urgen	Kurang Urgen	Kurang Urgen
	Cukup Padat	Kurang Urgen	Urgen	Urgen
	Padat	Urgen	Sangat Urgen	Sangat Urgen

## 3. Komposisi Aturan (Agregasi)

Komposisi aturan dilakukan dengan melihat hasil dari fungsi implikasi dari setiap aturan dan menggunakan metode MAX dalam melakukan perhitungannya. Berikut merupakan rumus dari metode MAX:

$$\alpha = \text{MAX}(\mu_{k1}(x_i), \mu_{k2}(x_i))$$

Di mana,

$\mu_{k1}(x_i)$ : Nilai derajat keanggotaan berdasarkan kondisi ke-1.

$\mu_{k2}(x_i)$ : Nilai derajat keanggotaan berdasarkan kondisi ke-2

$\alpha$  :  $\alpha$ -Predicate

## 4. Defuzzifikasi

Setelah diperoleh hasil *fuzzy* gabungan, langkah terakhir adalah defuzzifikasi, yaitu mengubah nilai *fuzzy* menjadi nilai tegas (*crisp*). Metode defuzzifikasi yang umum digunakan dalam mamdani adalah metode centroid, yang menghitung rata-rata tertimbang dari seluruh nilai dalam fungsi keanggotaan hasil inferensi. Berikut adalah rumus metode centroid yang digunakan dalam proses defuzzifikasi pada metode *Fuzzy Mamdani*:

$$Z = \frac{\int z \cdot \mu(z) dz}{\int \mu(z) dz}$$

Di mana,

- $Z$  : nilai output tegas (*crisp value*) hasil defuzzifikasi
- $z$  : variabel output
- $\mu(z)$  : derajat keanggotaan *fuzzy* output terhadap  $z$
- $\int$  : proses integral pada seluruh domain dari fungsi output *fuzzy*.

### 5. Agregasi dan Hasil Tahap ke Tahap

Hasil dari satu tahap menjadi masukan untuk tahap berikutnya. Berdasarkan teknik penyelesaian sebelumnya terkait *fuzzy* bertingkat maka output yang di dapat ialah:

- Output Derajat Urgensi dari tahap 1: Hasil dari *fuzzy* tahap pertama adalah tingkat urgensi jalur tertentu berdasarkan input dari lama antrian dan lama durasi lampu merah.
- Output Tambahan Hijau dari tahap 2: *Fuzzy* tahap 2 akan memberikan keluaran berupa tambahan durasi hijau pada setiap jalur berdasarkan sisa kendaraan dalam antrian dan durasi lampu hijau.
- Output Keputusan dari tahap 3: *Fuzzy* tahap 3 akan memberikan output terkait fase lalu lintas pada lokasi penelitian berdasarkan output dari tahap pertama dan tahap kedua.

### 3.5.2 Algoritma *Improved Ant Colony Optimization*

Dalam penelitian ini *IACO* digunakan sebagai metode optimasi untuk menentukan durasi waktu merah yang optimal pada lampu lalu lintas, sehingga total waktu tunggu kendaraan dapat diminimalkan. *IACO* dikonstruksi oleh kombinasi algoritma *ACO* biasa dengan metode operasi mutasi dan perubahan dua operator. Asumsi dalam konteks ini yaitu,

- Semut dianggap sebagai agen yang mencari solusi optimal.
- Jalur atau rute diwakili oleh durasi waktu hijau yang diuji.
- Pheromone menunjukkan kualitas solusi (seberapa optimal durasi waktu hijau tersebut).

Fungsi tujuan model adalah **meminimumkan** karena penelitian ini bertujuan untuk meningkatkan efisiensi lalu lintas dengan mengurangi dampak negatif kemacetan dengan cara meminimalkan waktu tunggu total, sistem lampu

lalu lintas dioptimalkan untuk memberikan aliran kendaraan yang lebih lancar di semua ruas jalan. Tujuan optimasi *IACO* pada penelitian ini adalah:

$$\text{Minimize } Z = \sum_{i=1}^N J_{hijau}$$

dimana:

- $Z$  : Total waktu tunggu kendaraan di semua ruas jalan
- $N$  : Jumlah ruas jalan pada persimpangan
- $J_{hijau}$  : Total waktu tunggu kendaraan selama lampu hijau

Berikut merupakan langkah utama dalam Algoritma *Improved Ant Colony Optimization*:

#### a. Inisialisasi Semut dan Parameter

Dalam penelitian ini terdapat beberapa parameter yaitu:

- 1) Jumlah kendaraan ( $Q$ ): Banyaknya kendaraan dalam antrian
- 2) Durasi lampu hijau ( $T_h$ ): Waktu hijau yang akan di optimalkan
- 3) *Headway* ( $h_w$ ): Jarak waktu antar kendaraan keluar dari lampu hijau ( $h_w = 2$  detik).
- 4) Jumlah iterasi ( $t_{max}$ ): Iterasi maksimum untuk algoritma ACO
- 5) Feromon ( $\rho$ ): Besarnya feromon pada setiap iterasi ( $0 < \rho < 1$ ).
- 6) Parameter  $\alpha$  dan  $\beta$  : Mengontrol bobot pengaruh feromon ( $\alpha$ ) dan visibilitas ( $\beta$ )

#### b. Perhitungan Visibilitas

Visibilitas ( $\eta_j$ ) didefinisikan sebagai kebalikan dari total waktu tunggu kendaraan ( $C_j$ ) yang dihasilkan oleh solusi  $j$ . Semakin kecil total waktu tunggu, semakin besar nilai visibilitas, sehingga solusi tersebut lebih menarik bagi semut. Berikut merupakan rumus dari perhitungan visibilitas:

$$\eta_j = \frac{1}{C_j}$$

### c. Pemilihan Solusi oleh Semut

Setiap semut memilih durasi waktu hijau ( $T_h$ ) berdasarkan probabilitas transisi, yang dihitung menggunakan rumus berikut:

$$p_j = \frac{[\tau_j]^\alpha \cdot [\eta_j]^\beta}{\sum_{k \in S} [\tau_k]^\alpha \cdot [\eta_k]^\beta}$$

Keterangan:

$p_j$  : Probabilitas semut memilih solusi  $j$ .

$\tau_j$  : Jumlah pheromone pada solusi  $j$ .

$\eta_j$  : Visibilitas dari solusi  $j$ .

$\alpha$  : Tingkat pengaruh pheromone.

$\beta$  : Tingkat pengaruh visibilitas.

$\tau_k$  : Jumlah pheromone pada semua kemungkinan durasi hijau  $k$ .

$\eta_k$  : visibilitas dari kemungkinan durasi hijau  $k$ .

### d. Pembaruan Pheromone

Pheromone diperbaharui setiap iterasi untuk meningkatkan probabilitas solusi optimal:

$$\tau_j(t+1) = (1 - \rho) \cdot \tau_j(t) + Q \cdot \frac{1}{W_j}$$

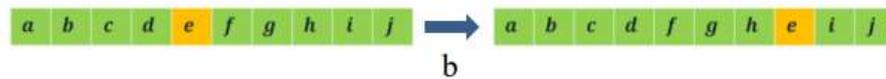
di mana:

- $\rho$  : Tingkat penguatan pheromone
- $Q$  : Banyaknya kendaraan dalam antrian.
- $W_j$  : Total waktu tunggu kendaraan untuk durasi  $T_h$  tertentu.

Solusi dengan waktu tunggu lebih rendah akan menghasilkan lebih banyak pheromone, sehingga lebih mungkin dipilih oleh semut untuk melanjutkan iterasi berikutnya.

### e. Operasi Mutasi

Mutasi merupakan proses perubahan sebagian sifat individu secara acak yang menghasilkan struktur genetik baru. Mutasi yang digunakan pada penelitian ini menggunakan Mutasi Sisipan (*Insertion Mutation*) yaitu memilih sebuah solusi terbaik secara acak dan memasukkannya pada posisi lain secara acak. Adapun pada Gambar 3.4 berikut menunjukkan jenis mutasi sisipan.



Gambar 3. 4 Gambar Mutasi Sisipan

Pada penelitian ini, mutasi dilakukan setelah proses konstruksi solusi oleh semut dan sebelum dilakukan pencarian lokal (*local search*), sesuai dengan alur tahapan IACO. Probabilitas mutasi menentukan seberapa besar kemungkinan solusi akan mengalami mutasi. Dalam penelitian ini, digunakan pendekatan probabilitas mutasi tetap (*fixed mutation probability*) sebesar  $P_m = 0,2$ . Artinya, setiap solusi pada siklus lampu lalu lintas memiliki peluang sebesar 20% untuk mengalami mutasi. Jika  $P_m$  terlalu kecil, mutasi tidak cukup efektif untuk menjelajah ruang solusi, sebaliknya jika terlalu besar proses pencarian menjadi tidak stabil. Berikut langkah-langkah mutasi sisipan yang diterapkan dalam penelitian ini:

1. Pilih Solusi yang Akan Dimutasi

Pilih satu kandidat solusi dari koloni semut yang telah dibentuk sebelumnya dan memenuhi syarat mutasi  $random \leq P_m(t)$ .

2. Tentukan Dua Posisi Fase (i dan j)

Dari solusi tersebut, pilih dua titik posisi secara acak, yaitu:

- Posisi  $i$ : elemen (fase lalu lintas) yang akan disisipkan.
- Posisi  $j$ : tempat penyisipan elemen dari posisi  $i$ .

3. Lakukan Proses Sisipan

Hapus elemen pada posisi  $i$ , kemudian sisipkan elemen tersebut pada posisi  $j$  tanpa menggandakan lain dalam solusi. Hal ini akan menghasilkan urutan fase lalu lintas baru.

4. Evaluasi Solusi Baru

Hitung nilai fitness dari solusi hasil mutasi sisipan. Jika nilai fitness membaik (misalnya waktu tunggu total menurun), maka solusi ini dapat dipertahankan. Jika tidak, bisa dipertimbangkan untuk tetap menggunakan solusi awal.

***f. Local Search***

Setelah proses mutasi selesai, dilakukan proses *local search* untuk menyempurnakan solusi dengan cara mengevaluasi tetangganya di sekitar solusi saat ini. Tujuan dari *local search* adalah memperbaiki solusi dengan menemukan alternatif yang memiliki total waktu tunggu lebih kecil. Strategi lokal ini memodifikasi solusi saat ini dengan melakukan pergeseran kecil pada durasi hijau setiap fase, misalnya menambah atau mengurangi 1-2 detik pada salah satu arah dengan tetap menjaga total durasi siklus konstan. Berikut merupakan langkah dari *local search*:

1. Input: Solusi hasil mutasi berupa kombinasi durasi hijau tiap fase simpang, misal [30, 35, 25, 30] detik.
2. Pilih Tetangga: Buat beberapa kandidat solusi tetangga dengan cara menambah 1-2 detik pada satu fase, dan mengurangi pada fase lain secara proporsional agar durasi siklus tetap. Contoh dari [30, 35, 25, 30] menjadi tetangga 1: [31, 34, 25, 30] dan tetangga 2: [30, 36, 24, 30], dst.
3. Evaluasi: Hitung total waktu tunggu dari setiap tetangga.
4. Pilih yang Terbaik: Jika ditemukan tetangga yang memiliki total waktu tunggu lebih kecil dari solusi saat ini, maka solusi saat ini diganti.
5. Iterasi: Langkah dapat diulang hingga menemukan solusi yang optimal.

**g. Kondisi Pemberhentian**

Iterasi dan perhitungan akan dihentikan ketika telah menemukan tingkat probabilitas tertinggi dan konvergen.