

PERHITUNGAN MANUAL SISTEM FUZZY MAMDANI UNTUK SIMULASI LAMPU LALU LINTAS ADAPTIF

TANGGAL DAN WAKTU PENYUSUNAN

- Tanggal: 29 Juni 2025
- Waktu: 12:33 PM WIB

PENDAHULUAN

Dokumen ini berisi perhitungan manual untuk sistem fuzzy Mamdani yang digunakan dalam simulasi lampu lalu lintas adaptif. Sistem ini menentukan durasi lampu hijau (`durasi_hijau`) berdasarkan jumlah kendaraan di sisi aktif (`sisi_aktif`) menggunakan logika fuzzy. Perhitungan mencakup fuzzification, rule evaluation, aggregation, dan defuzzification, dengan contoh input 70 kendaraan.

DEFINISI VARIABEL DAN FUNGSI KEANGGOTAAN

Input (`sisi_aktif`): Jumlah kendaraan, rentang 0-150.

- `sepi` : [0, 0, 50]
- `sedang` : [30, 75, 120]
- `padat` : [100, 150, 150]

Output (`durasi_hijau`): Durasi lampu hijau (detik), rentang 10-90.

- `pendek` : [10, 10, 35]
- `sedang` : [25, 50, 75]
- `lama` : [65, 90, 90]

Fungsi keanggotaan menggunakan model segitiga (`trimf`), dengan rumus:

- Jika $x \leq a$ atau $x \geq c$, maka $\mu = 0$.
- Jika $a < x \leq b$, maka $\mu = \frac{x-a}{b-a}$.
- Jika $b < x \leq c$, maka $\mu = \frac{c-x}{c-b}$.

PERHITUNGAN MANUAL UNTUK INPUT 70 KENDARAAN

LANGKAH 1: FUZZIFICATION

Hitung derajat keanggotaan (μ) untuk setiap fungsi keanggotaan berdasarkan input 70.

- Untuk **sepi** [0, 0, 50]: Karena $70 > 50$, di luar rentang kanan.
 $\mu_{\text{sepi}} = 0$.
- Untuk **sedang** [30, 75, 120]: $a = 30$, $b = 75$, $c = 120$. Karena $30 < 70 \leq 75$, gunakan: $\mu = \frac{x-a}{b-a}$. $\mu_{\text{sedang}} = \frac{70-30}{75-30} = \frac{40}{45} \approx 0.8889$.
- Untuk **padat** [100, 150, 150]: Karena $70 < 100$, di luar rentang kiri.
 $\mu_{\text{padat}} = 0$.

Hasil Fuzzification:

Kategori	Derajat Keanggotaan (μ)
sepi	0
sedang	0.8889
padat	0

LANGKAH 2: RULE EVALUATION

Terapkan aturan fuzzy:

- Rule 1: IF **sisi_aktif** IS **sepi** THEN **durasi_hijau** IS **pendek**.
 - $\mu_{\text{sepi}} = 0$, kontribusi untuk **pendek** = 0.
- Rule 2: IF **sisi_aktif** IS **sedang** THEN **durasi_hijau** IS **sedang**.
 - $\mu_{\text{sedang}} = 0.8889$, kontribusi untuk **sedang** = 0.8889.
- Rule 3: IF **sisi_aktif** IS **padat** THEN **durasi_hijau** IS **lama**.
 - $\mu_{\text{padat}} = 0$, kontribusi untuk **lama** = 0.

Hasil:

- Hanya Rule 2 aktif dengan $\mu = 0.8889$.

LANGKAH 3: AGGREGATION

Agregasi menggabungkan output aturan. Karena hanya Rule 2 aktif, kurva `durasi_hijau['sedang']` ([25, 50, 75]) dipotong pada ketinggian 0.8889.

- Dari 25 ke 50: Kenaikan linier dari 0 ke 0.8889.
- Dari 50 ke 75: Penurunan linier dari 0.8889 ke 0.

LANGKAH 4: DEFUZZIFICATION

Gunakan metode centroid untuk menghitung nilai numerik tunggal.

- Area Segitiga Kiri (25 ke 50):
 - Basis = $50 - 25 = 25$
 - Tinggi = 0.8889
 - Area = $\frac{1}{2} \times 25 \times 0.8889 \approx 11.11125$
 - Centroid $x = 25 + \frac{25}{3} \times 0.8889 \approx 41.48$
- Area Segitiga Kanan (50 ke 75):
 - Basis = $75 - 50 = 25$
 - Tinggi = 0.8889
 - Area = $\frac{1}{2} \times 25 \times 0.8889 \approx 11.11125$
 - Centroid $x = 50 + \frac{25}{3} \times 0.8889 \approx 57.41$
- Total Area = $11.11125 + 11.11125 = 22.2225$
- Centroid Keseluruhan:
 - Numerator = $(41.48 \times 11.11125) + (57.41 \times 11.11125) \approx 460.889 + 637.844 \approx 1098.733$
 - $x_{\text{centroid}} = \frac{1098.733}{22.2225} \approx 49.45$

Hasil Defuzzifikasi: Durasi hijau ≈ 49.45 detik.

VERIFIKASI

Jalankan kode dengan:

```
sim_sisi_aktif.input['sisi_aktif'] = 70
sim_sisi_aktif.compute()
print(sim_sisi_aktif.output['durasi_hijau']) # Harus ≈
49.45 detik
```

Hasilnya akan mendekati 49-50 detik, sesuai perhitungan manual.

PEMBUATAN GRAFIK

MENGGUNAKAN PYTHON (MATPLOTLIB)

1. Instal Matplotlib: `pip install matplotlib`.
2. Gunakan kode berikut:

```
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz

# Input universe (Jumlah Kendaraan)
x = np.arange(0, 151, 1)
# Output universe (Durasi Hijau)
y_durasi = np.arange(10, 91, 1)

# Fungsi Keanggotaan Input
sepi = fuzz.trimf(x, [0, 0, 50])
sedang = fuzz.trimf(x, [30, 75, 120])
padat = fuzz.trimf(x, [100, 150, 150])

# Fungsi Keanggotaan Output
pendek = fuzz.trimf(y_durasi, [10, 10, 35])
sedang_out = fuzz.trimf(y_durasi, [25, 50, 75])
lama = fuzz.trimf(y_durasi, [65, 90, 90])

# Plotting
plt.figure(figsize=(12, 6))

# Grafik Input
plt.subplot(1, 2, 1)
plt.plot(x, sepi, 'b', label='Sepi')
plt.plot(x, sedang, 'g', label='Sedang')
plt.plot(x, padat, 'r', label='Padat')
plt.title('Fungsi Keanggotaan Jumlah Kendaraan')
plt.xlabel('Jumlah Kendaraan')
plt.ylabel('Derajat Keanggotaan')
plt.legend()
plt.grid(True)
```

```
# Grafik Output
plt.subplot(1, 2, 2)
plt.plot(y_durasi, pendek, 'b', label='Pendek')
plt.plot(y_durasi, sedang_out, 'g', label='Sedang')
plt.plot(y_durasi, lama, 'r', label='Lama')
plt.title('Fungsi Keanggotaan Durasi Hijau')
plt.xlabel('Durasi (detik)')
plt.ylabel('Derajat Keanggotaan')
plt.legend()
plt.grid(True)

plt.tight_layout()
plt.show()
```

3. Tambahkan garis vertikal pada $x = 70$ dan $y = 49.45$ untuk menunjukkan input dan output pada grafik yang dihasilkan.

SECARA MANUAL

1. Gambar sumbu: X (0-150 untuk input, 10-90 untuk output), Y (0-1).
2. Plot fungsi keanggotaan sebagai segitiga:
 - sepi : (0,0) ke (0,1) ke (50,0).
 - sedang : (30,0) ke (75,1) ke (120,0).
 - padat : (100,0) ke (150,1).
 - pendek : (10,0) ke (10,1) ke (35,0).
 - sedang : (25,0) ke (50,1) ke (75,0).
 - lama : (65,0) ke (90,1).
3. Tandai $x = 70$ dengan $\mu_{\text{sedang}} = 0.8889$.
4. Gambar kurva agregat (segitiga 25-75 dipotong 0.8889) dan tandai centroid pada 49.45.

KESIMPULAN

Perhitungan manual menghasilkan durasi hijau ≈ 49.45 detik untuk 70 kendaraan, yang konsisten dengan output `skfuzzy`. Grafik dapat digunakan untuk memvisualisasikan proses fuzzy logic secara intuitif.

CARA MENGGUNAKAN

1. Salin teks di atas ke editor teks (misalnya Notepad, VS Code).

2. Simpan sebagai file dengan ekstensi `.md` (contoh: `fuzzy_calculation.md`).
3. Buka dengan aplikasi seperti Visual Studio Code, Typora, atau konversi ke Word/PDF menggunakan alat seperti Pandoc (`pandoc fuzzy_calculation.md -o fuzzy_calculation.docx`).

Jika Anda memerlukan bantuan tambahan untuk mengonversi atau menyesuaikan, beri tahu saya!