

Nama : Ferdinand S. Pakpahan

NPM : G1A021040

Mata Kuliah : Logika Fuzzy

A . Fungsi Keanggotaan

Fungsi keanggotaan merupakan komponen utama dalam sistem fuzzy yang menentukan tingkat kecocokan nilai input dengan kategori fuzzy tertentu. Dalam sistem ini, fungsi keanggotaan didefinisikan untuk setiap variabel input dan output.

1. Variabel Input dan Output

Suhu (°C):

1. dingin: [22, 22, 23, 25]
2. normal: [23, 26, 29, 30]
3. panas: [29, 32, 32, 32]

Kebisingan (dB):

1. rendah: [55, 55, 60, 65]
2. sedang: [60, 70, 80, 90]
3. tinggi: [85, 90, 90, 90]

Pencahayaan (lux):

1. redup: [150, 150, 200, 250]
2. normal: [200, 300, 400, 500]
3. terang: [450, 500, 500, 500]

Produksi (unit):

1. rendah: [120, 130, 135, 140]
2. sedang: [135, 145, 150, 155]
3. tinggi: [150, 155, 160, 160]

2. Kode untuk Definisi Fungsi Keanggotaan

```
# 1. Definisikan variabel input dan output
suhu = ctrl.Antecedent(np.arange(22, 33, 1), 'suhu')    # Suhu dalam
derajat Celsius
kebisingan = ctrl.Antecedent(np.arange(55, 91, 1), 'kebisingan')    #
Kebisingan dalam dB
pencahayaan = ctrl.Antecedent(np.arange(150, 501, 1), 'pencahayaan')
# Pencahayaan dalam lux
```

```

produksi = ctrl.Consequent(np.arange(120, 161, 1), 'produksi') #
Produksi dalam unit
# 2. Definisikan fungsi keanggotaan trapezoid untuk suhu
suhu['dingin'] = fuzz.trapmf(suhu.universe, [22, 22, 23, 25])
suhu['normal'] = fuzz.trapmf(suhu.universe, [23, 26, 29, 30])
suhu['panas'] = fuzz.trapmf(suhu.universe, [29, 32, 32, 32])
# 3. Definisikan fungsi keanggotaan trapezoid untuk kebisingan
kebisingan['rendah'] = fuzz.trapmf(kebisingan.universe, [55, 55, 60,
65])
kebisingan['sedang'] = fuzz.trapmf(kebisingan.universe, [60, 70, 80,
90])
kebisingan['tinggi'] = fuzz.trapmf(kebisingan.universe, [85, 90, 90,
90])
# 4. Definisikan fungsi keanggotaan trapezoid untuk pencahayaan
pencahayaan['redup'] = fuzz.trapmf(pencahayaan.universe, [150, 150,
200, 250])
pencahayaan['normal'] = fuzz.trapmf(pencahayaan.universe, [200, 300,
400, 500])
pencahayaan['terang'] = fuzz.trapmf(pencahayaan.universe, [450, 500,
500, 500])
# 5. Definisikan fungsi keanggotaan trapezoid untuk produksi
produksi['rendah'] = fuzz.trapmf(produksi.universe, [120, 130, 135,
140])
produksi['sedang'] = fuzz.trapmf(produksi.universe, [135, 145, 150,
155])
produksi['tinggi'] = fuzz.trapmf(produksi.universe, [150, 155, 160,
160])
# 6. Gambar fungsi keanggotaan
suhu.view(title="Fungsi Keanggotaan Suhu")
kebisingan.view(title="Fungsi Keanggotaan Kebisingan")
pencahayaan.view(title="Fungsi Keanggotaan Pencahayaan")
produksi.view(title="Fungsi Keanggotaan Produksi")
plt.show()

```

3. Visualisasi Fungsi Keanggotaan

Fungsi keanggotaan untuk masing-masing variabel akan digambarkan menggunakan fungsi `.view()`, memberikan gambaran visual yang jelas tentang bagaimana setiap nilai terdistribusi dalam himpunan fuzzy.

B. 27 Aturan Fuzzy

A. Kode untuk Aturan Fuzzy

```

# 7. Definisikan aturan fuzzy
rules = []
# Kombinasi untuk suhu 'dingin'
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] &
pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] &
pencahayaan['normal'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] &
pencahayaan['terang'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] &
pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] &
pencahayaan['normal'], produksi['rendah']))

```

```

rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] &
pencahayaannya['terang'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] &
pencahayaannya['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] &
pencahayaannya['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] &
pencahayaannya['terang'], produksi['sedang']))
# Kombinasi untuk suhu 'normal'
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] &
pencahayaannya['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] &
pencahayaannya['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] &
pencahayaannya['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] &
pencahayaannya['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] &
pencahayaannya['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] &
pencahayaannya['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] &
pencahayaannya['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] &
pencahayaannya['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] &
pencahayaannya['terang'], produksi['tinggi']))
# Kombinasi untuk suhu 'panas'
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] &
pencahayaannya['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] &
pencahayaannya['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] &
pencahayaannya['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] &
pencahayaannya['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] &
pencahayaannya['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] &
pencahayaannya['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] &
pencahayaannya['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] &
pencahayaannya['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] &
pencahayaannya['terang'], produksi['tinggi']))

```

C. Derajat Keanggotaan

Derajat keanggotaan menghitung seberapa baik input yang diberikan sesuai dengan fungsi keanggotaan yang telah didefinisikan.

A. Kode untuk Menghitung Derajat Keanggotaan

```

# 8. Setup sistem kontrol fuzzy
produksi_ctrl = ctrl.ControlSystem(rules)
produksi_simulasi = ctrl.ControlSystemSimulation(produksi_ctrl)
# 9. Masukkan nilai input (contoh)
suhu_value = 28 # Suhu dalam derajat Celsius

```

```

kebisingan_value = 90 # Kebisingan dalam dB
pencahayaannya_value = 300 # Pencahayaannya dalam lux
# Masukkan nilai input ke dalam sistem
produksi_simulasi.input['suhu'] = suhu_value
produksi_simulasi.input['kebisingan'] = kebisingan_value
produksi_simulasi.input['pencahayaannya'] = pencahayaannya_value
# Hitung derajat keanggotaan
suhu.view(sim=produksi_simulasi)
kebisingan.view(sim=produksi_simulasi)
pencahayaannya.view(sim=produksi_simulasi)
produksi.view(sim=produksi_simulasi)

```

D. α -Predikat untuk Setiap Aturan

α -predikat digunakan untuk menghitung output berdasarkan aturan fuzzy yang telah ditetapkan.

Kode untuk Menghitung Output

```

# 10. Hitung hasil output
produksi_simulasi.compute()
# 11. Lihat hasil output produksi
output_produk = produksi_simulasi.output['produksi']
print(f"Output produksi: {output_produk:.2f}")

```

E. Rata-rata Jumlah Produk

Metode defuzzy yang digunakan adalah Weighted Average, yang memberikan hasil akhir berdasarkan kontribusi dari setiap aturan.

1. Kode untuk Visualisasi Hasil Output

```

# 12. Gambar hasil output
produksi.view(sim=produksi_simulasi)
plt.show()

```

NOTE : link ke Notebook (https://colab.research.google.com/drive/1v9v6-JTQkmi5uv_b_QxEBYlr_ycLpCdt?usp=sharing)