BAB II

LANDASAN TEORI

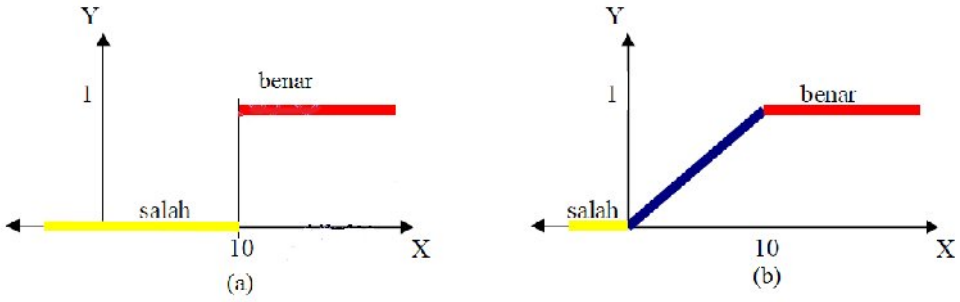
2.1 Logika Fuzzy

Logika Fuzzy pertama kali dikembangkan oleh Lotfi A. Zadeh pada tahun1965. Teori ini banyak diterapkan di berbagai bidang, antara lain representasipikiran manusia kedalam suatu sistem. Banyak alasan mengapa penggunaanlogika fuzzy ini sering dipergunakan antara lain, konsep logika fuzzy yang miripdengan konsep berpikir manusia. Sistem fuzzy dapat merepresentasikanpengetahuan manusia ke dalam bentuk matematis dengan lebih menyerupai caraberpikir manusia.Pengontrol dengan logika fuzzy mempunyai kelebihan yaitudapat mengontrol sistem yang kompleks, non-linier, atau sistem yang sulitdirepresentasikan kedalam bentuk matematis.Selain itu, informasi berupapengetahuan dan pengalaman mempunyai peranan penting dalam mengenaliperilaku sistem di dunia nyata.

Logika fuzzy juga memiliki himpunan fuzzy yang mana pada dasarnya, teorihimpunan fuzzy merupakan perluasan dari teori himpunan klasik. Dimana dengan logika fuzzy, hasil yang keluar tidak akan selalu konstan dengan input yang ada.Cara kerja logika fuzzy secara garis besar terdiri dari input, proses dan output.Logika fuzzy merupakan suatu teori himpunan logika yang dikembangkan untukmengatasi konsep nilai yang terdapat diantara kebenaran (*truth*) dan kesalahan(*false*). Dengan menggunakan fuzzy logic nilai yang dihasilkan bukan hanya ya(1) atau tidak (0) tetapi seluruh kemungkinan diantara 0 dan 1.

2.2 Perbedaan Logika Fuzzy dan Logika Tegas

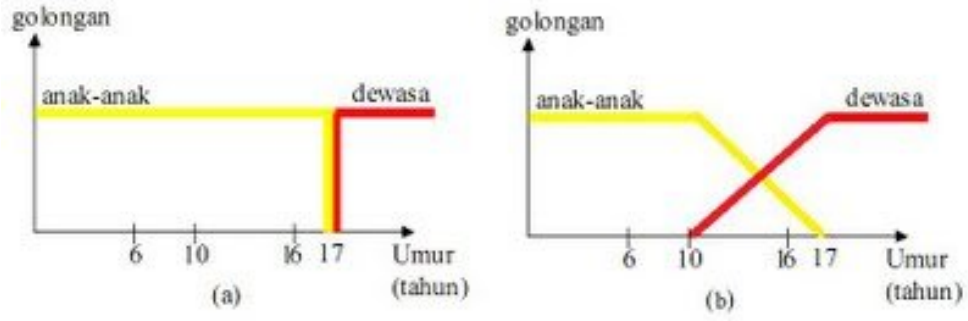
Perbedaan antara kedua jenis logika ini adalah logika fuzzy memiliki nilai 0 hingga 1, sedangkan logika tegas 0 dan 1.Secara grafik perbedaan logika fuzzy dan logika tegas ditunjukan pada gambar 2.1.



Gambar 2.1 Perbedaan logika fuzzy (a) dan logika tegas (b)

Pada gambar 2.1 (a) apabila x lebih dari atau sama dengan 10 baru dikatakan benar yaitu y=1, sebaliknya nilai x kurang dari 10 adalah salah yaitu y=0. Maka angka 9, 8, dan 7 dan seterusnya adalah dikatakan salah. Pada gambar 2.1 (b) nilai x= 9, 8, atau 7 atau nilai antara 0 – 10 dapat dikatakan benar dan dapat dikatakan salah.

Dalam contoh kehidupan kita seseorang dikatakan sudah dewasa apabila berumur 17 tahun, maka siapapun yang kurang umur dari 17 tahun didalam logika tegas akan dikatakan sebagai belum dewasa atau anak-anak. Sedangkan dalam hal ini pada logika fuzzy umur 17 tahun dapat dikategorikan dewasa tapi belum dewasa, misal untuk umur 16 tahun dan 15 tahun atau 14 tahun dan 13 tahun. Secara grafik dapat digambarkan sebagai berikut.



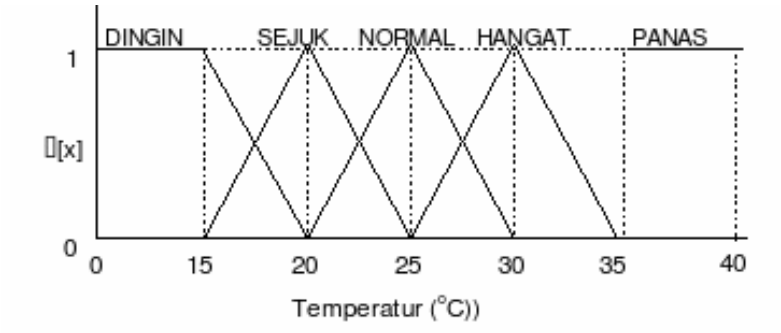
Gambar 2.2 Perbandingan contoh logika tegas (a) dan logika fuzzy (b)

2.3 Himpunan Fuzzy

Dalam teori logika fuzzy dikenal himpunan fuzzy (*fuzzy sets*) yang merupakan pengelompokan sesuatu berdasarkan variabel bahasa (*linguistik variabel*), yang dinyatakan dalam fungsi keanggotaan.Didalam semesta pembicaraan (*universe of* dicourse), Fungsi keanggotaan dari suatu himpunan fuzzy tersebut bernilai 0 sampai dengan 1.

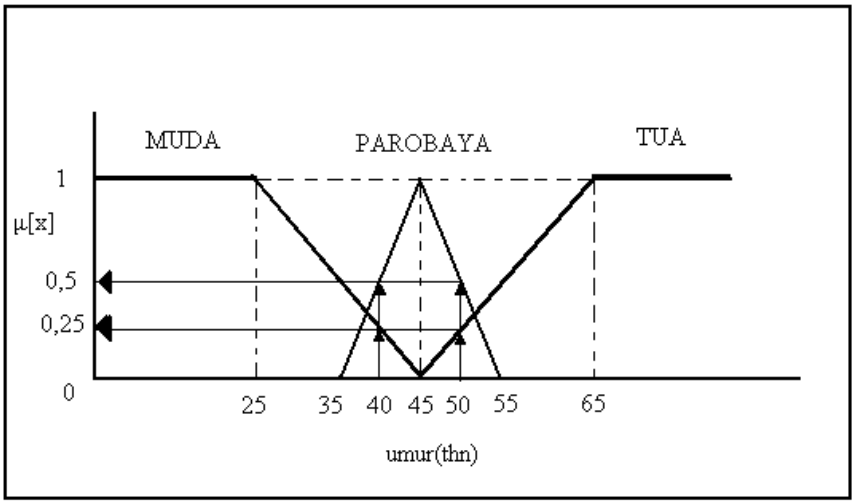
Contoh dari himpunan variabel bahasa antara lain:

Himpunan suhu atau temperatur dapat dinyatakan dengan dingin, sejuk, normal, hangat, dan panas.Grafik dari himpunan suhu ini ditunjukan pada gambar 2.3 ini.



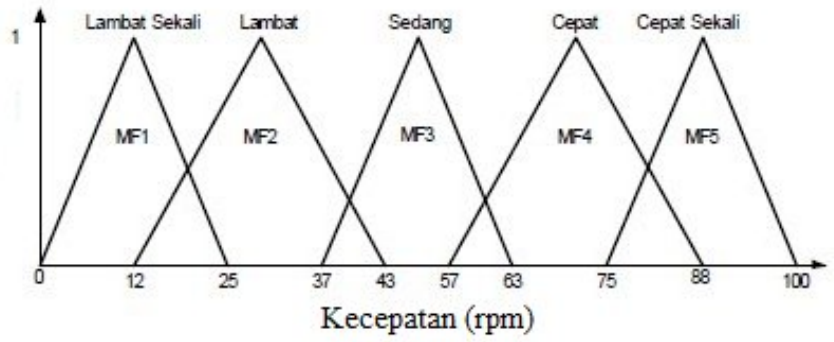
Gambar 2.3 Contoh keanggotaan himpunan temperatur atau suhu

Himpunan umur dapat dinyatakan dengan muda, parobaya, tua, dan sangat tua.Grafik dari himpunan umur ini ditunjukan pada gambar 2.4 ini.



Gambar 2.4 Contoh keanggotaan himpunan umur

Himpunan dari kecepatan dapat dinyatakan dengan lambat, normal, cepat, dan sangat cepat.Grafik dari himpunan umur ini ditunjukan pada gambar 2.5 ini.



Gambar 2.5 Contoh keanggotaan himpunan kecepatan

2.4 Dasar Logika Fuzzy

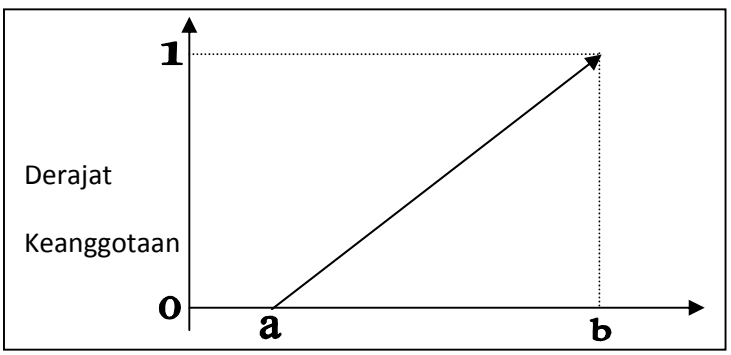
2.4.1 Fungsi Keanggotaan (*Membership Function*)

Fungsi keanggotaan merupakan suatu kurva yang menunjukkan pemetaantitik-titik input data kedalam nilai keanggotaanya (*disebut juga dengan derajat keanggotaan*) yang memiliki interval antara 0 sampai 1. Untuk mendapatkannilai keanggotaan dapat menggunakan cara pendekatan fungsi.

Ada beberapa fungsi keanggotaan yang digunakan dalam teori himpunanfuzzy adalah:

- Representasi Linier

  Pada representasi linear, pemetaan input ke derajat keanggotaan nyadigambarkan sebagai suatu gari lurus. Bentuk ini paling sederhana danmenjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas.Ada 2 keadaan himpunan fuzzy yang linear. Pertama, kenaikan himpunandimulai pada nilai domain yang memiliki derajat keanggotaan nol [0]bergerak ke kanan menuju kenilai domain yang memiliki derajat keanggotaan lebih tinggi seperti pada Gambar 2.6.

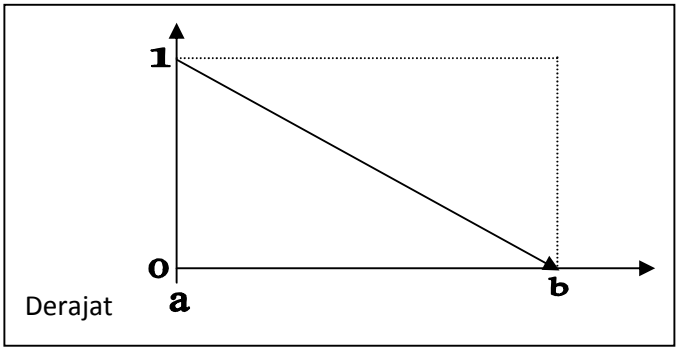


Gambar 2.6 Representasi Linear Naik

Persamaan fungsi keanggotaan linear naik:

$$\mu(x) = \begin{cases} 0; & x \le a \\ (x-a)/(b-a); & a \le x \le b \\ 1; & x \ge b \end{cases} \quad \text{.....(2.1)}$$

Kedua, merupakan kebalikan yang pertama.Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah seperti pada Gambar 2.7.



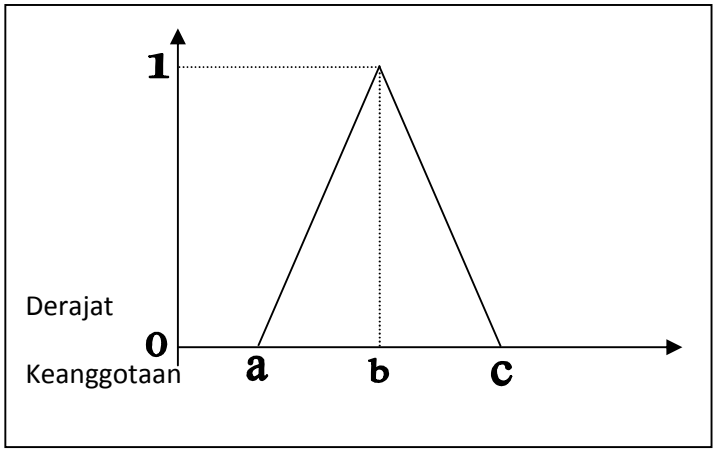
Gambar 2.7 Representasi Linear Turun

Fungsi keanggotaan linear turun :

$$\mu(x) = \begin{cases} (b-x)/(b-a); & a \le x \le b \\ 0; & x \ge b \end{cases} \quad \text{.....(2.2)}$$

- Representasi Kurva Segitiga

  Pada dasarnya merupakan gabungan antara 2 garis (*linear*) seperti terlihat pada Gambar 2.8.



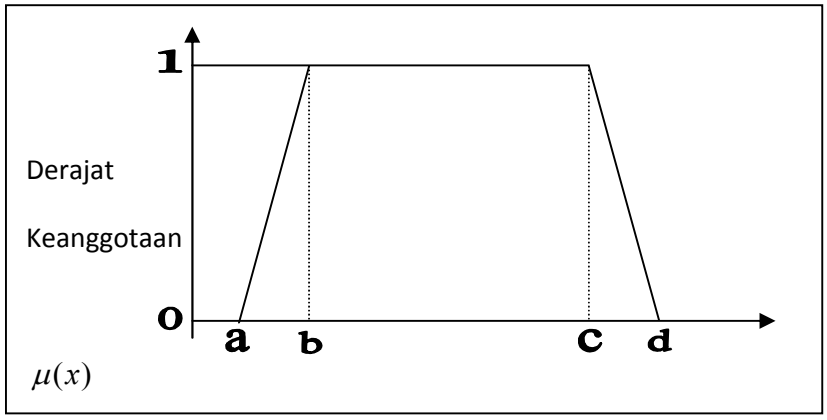
Gambar 2.8 Representasi Kurva Segitiga

Persamaan fungsi keanggotaan kurva segitiga:

$$\mu(x) = \begin{cases} 0; & x \le a \ atau \ x \ge c \\ (x-a)/(b-a); & a \le x \le b \\ (b-x)/(c-b); & b \le x \le c \end{cases} \quad \text{.....(2.3)}$$

- Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1 seperti pada Gambar 2.9.



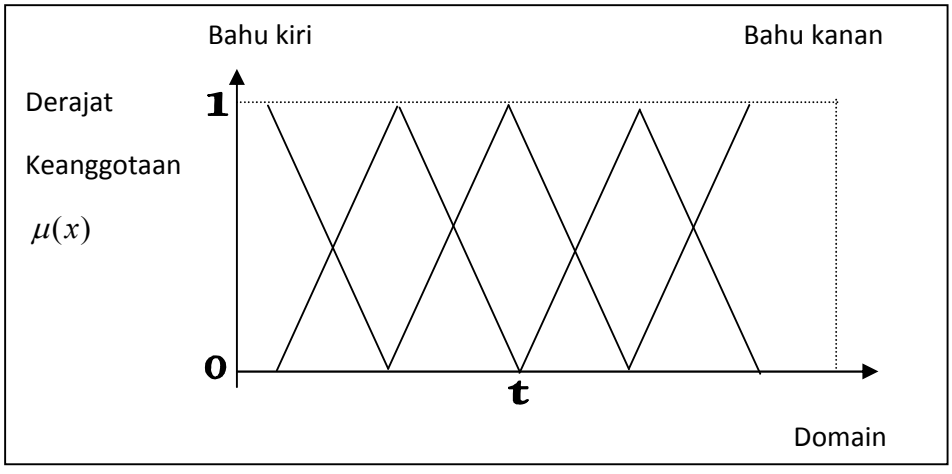
Gambar 2.9 Representasi Kurva Trapesium

Persamaan fungsi keanggotaan Kurva Trapesium :

$$\mu(x) = \begin{cases} 0; & x \le a \text{ atau } x \ge d \\ (x-a)/(b-a); & a \le x \le b \\ 1; & b \le x \le c \\ (d-x)/(d-c); & c \le x \le d \end{cases} \quad \text{.....(2.4)}$$

- Representasi Kurva Bahu

Daerah yang terletak ditengah-tengah suatu variable yang dipresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun (*misalkan: dingin bergerak ke sejuk bergerak ke hangat dan bergerak ke panas*). Tetapi terkadang salah satu sisi dari variable tersebut tidak mengalami perubahan. Sebagai contoh, apabila telah mencapai kondisi panas, kenaikan temperature akan tetap berada pada kondisi panas. Himpunan fuzzy 'bahu', bukan segitiga, digunakan untuk mengakhiri variable suatu daerah fuzzy. Bahu kiri bergerak dari benar ke salah, demikianjuga bahu kanan bergerak dari salah ke benar. Gambar menunjukkan variable temperature dengan daerah bahunya.



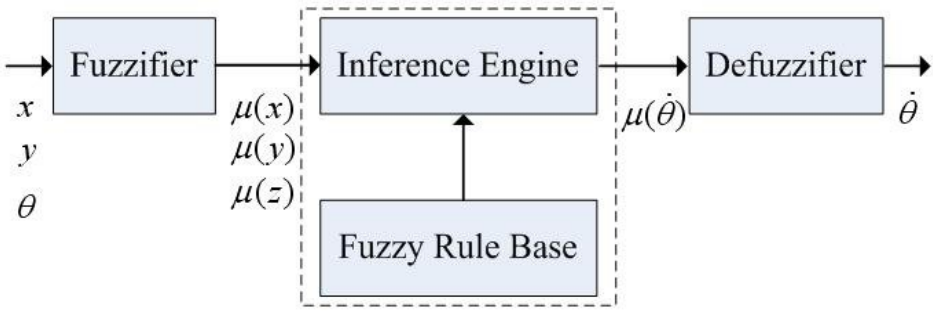
Gambar 2.10 Representasi Kurva Bahu

2.5 Cara Kerja Kontrol Logika Fuzzy

Dalam sistem kontrol logika fuzzy terdapat beberapa tahapan operasional meliputi:

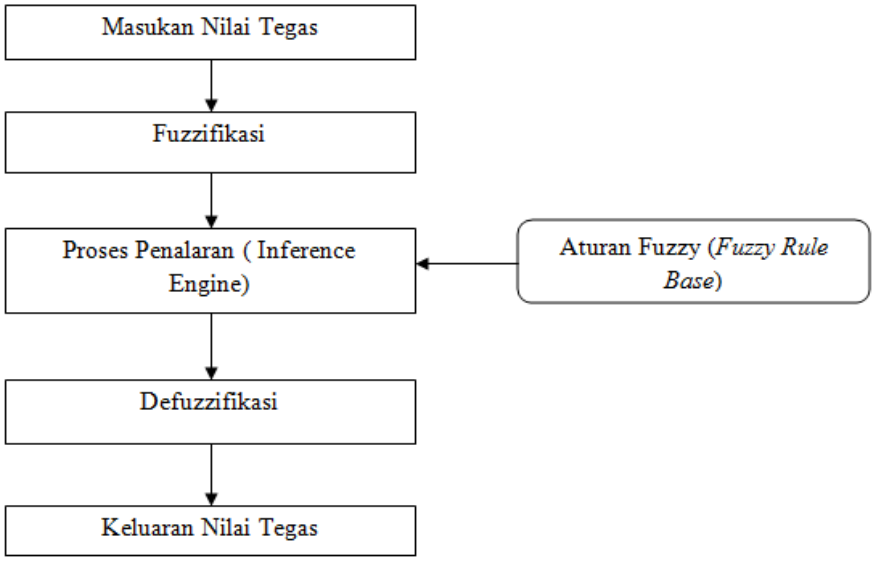
1. Fuzzifikasi.
2. mesin penalaran atau inference engine.
3. aturan dasar (*fuzzy rule*).
4. defuzzifikasi.

Blok diagram kontrol logika fuzzy ditunjukkan pada gambar 2.11.



Gambar 2.11 Blok diagram kontrol logika fuzzy

Kerangka operasional kontrol logika fuzzy ditunjukkan pada gambar 2.12.



Gambar 2.12 Kerangka kerja kontrol logika fuzzy

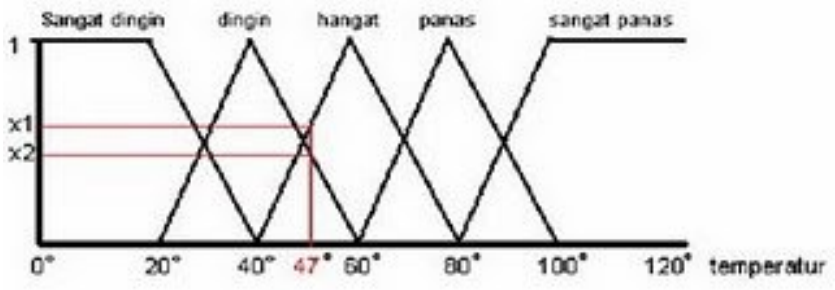
Dari gambar 2.12 sinyal masukan dari kontrol logika fuzzy dapat berupa nilai tegas. Sinyal masukan kontrol logika fuzzy dapat diambilkan dari.

1. Selisih antara nilai rujukan (*reference*) dengan nilai keluaran nyata dari kontrol logika fuzzy yang berupa nilai kesalahan (*error* = *E*).
2. Turunan pertama dari nilai error yang dikenal dengan delta error = dE

2.5.1 Fuzzifikasi

Fuzzifikasi yaitu suatu proses untuk mengubah suatu masukan dari bentuk tegas (crisp) menjadi fuzzy (variabel linguistik) yang biasanya disajikan dalam bentuk himpunan-himpunan fuzzy dengan suatu fungsi kenggotaannya masing-masing. Contoh dari proses Fuzzification adalah seperti yang ditunjukkan di gambar 2.13. Sebuah sistem fuzzy untuk mengukur suhu mempunyai 5 buah membership function yang mempunyai label sangat dingin, dingin, hangat, panas,

sangat panas. Kemudian input yang diperoleh dari crisp input adalah 47° maka pengambilan fuzzy input-nya adalah seperti pada gambar 2.13.



Gambar 2.13 Proses perubahan dari crisp/nilai tegas menjadi fuzzy input

Sehingga didapat 2 fuzzy input yang masing-masing adalah: dingin (x2) dan hangat (x1). Nilai x1 dan x2 dapat dicari dengan rumus persamaan garis.Yang menentukan sistem anda sensitif atau tidak adalah membership function ini.Jika membership function-nya banyak maka sistem anda menjadi sensitif. Yang dimaksud dengan sensitif dalam hal ini adalah jika input-nya berubah sedikit saja maka sistem akan cepat merespon dan menghasilkan suatu output lain. Output dari proses fuzzification ini adalah sebuah nilai input fuzzy atau yang biasanya dinamakan fuzzy input.

2.5.2 Aturan Dasar Logika Fuzzy

Aturan dasar atau rule base pada kontrol logika fuzzy merupakan suatu bentuk aturan relasi/implikasi “Jika-Maka” atau “If-Then” seperti pada pernyataan berikut:

“JIKA” X=A dan “JIKA” Y=B “MAKA” Z=C

Contoh dari aturan jika-maka ini pada pengendalian suhu ruangan dengan pengaturan kecepatan kipas angin melalui frekuensi variabel adalah sebagai berikut.

1. “JIKA” suhu panas dan
2. “JIKA” kecepatan kipas sangat lambat
3. “MAKA” sumber frekuensi dinaikkan sangat tinggi agar kecepatan kipas tinggi

Jadi aturan dasar kontrol logika fuzzy ditentukan dengan bantuan seorang pakar yang mengetahui karakteristik objek yang akan dikendalikan. Aturan dasar tersebut dapat dinyatakan dalam bentuk matrik aturan dasar kontrol logika fuzzy. Contoh aturan dasar pengaturan suhu ruangan dapat dilihat pada tabel berikut.

Tabel 2.1 Contoh matrik aturan dasar perancangan kontrol logika fuzzy

x/y	**B**	**S**	**K**
B	K	K	B
S	K	S	K
K	B	K	B

Dimana,

X= Suhu, Y= Kecepatan Kipas.

2.5.3 Mesin Penalaran Kontrol Logika Fuzzy (*Inference Engine*)

Mesin penalaran (*Inference Engine*) adalah proses implikasi dalam menalar nilai masukan guna penentuan nilai keluar sebagai bentuk pengambil keputusan. Salah satu model penalaran max-min. Dalam penalaran max-min proses pertama yang dilakukan adalah melakukan operasi operasi min sinyal keluaran lapisan fuzzifikasi, yang diteruskan dengan operasi max untuk mencari nilai keluaran yang selanjutnya akan difuzzifikasikan sebagai bentuk keluaran pengontrol. Operasional max-min tersebut dinyatakan sebagai berikut.

1. Operasi min (*irisan)*

$a \cap b = \min(a,b) = a \text{ if } \leq b$

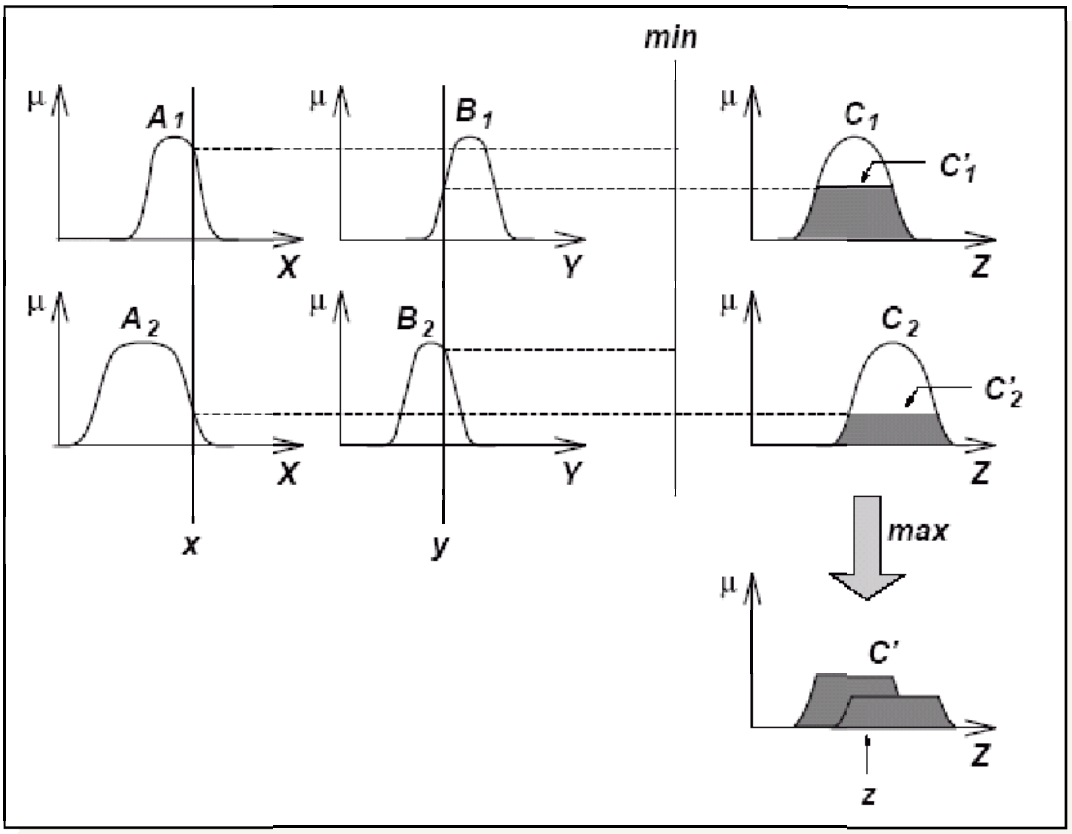
$= a \text{ if } a > b$……(2.5)

2. Operasi max (*union*)

$a \cup b = \max(a,b) = a \text{ if } \geq b$

$= a \text{ if } a \geq b$……...(2.6)

Proses penalaran max-min dapat dijelaskan dengan gambar 2.14.



Gambar 2.14 Operasi max-min secara grafis

2.5.4 Defuzzifikasi

Defuzzifikasi merupakan proses pemetaan himpunan fuzzy kemampuan tegas (crisp) proses ini merupakan kebalikan ari proses fuzzifikasi.

Metode dalam defuzzifikasi antara lain:

a. Metode Max

   Solusi himpunan fuzzy diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah fuzzy, dan mengaplikasikannya ke output dengan menggunakan operator OR (*union*). Secara umum dapat dituliskan :

   $$\mu_{sf}[x_i] = \max(\mu_{sf}[x_i], \mu_{kf}[x_i]) \quad \text{.....(2.7)}$$

b. Metode Additive (*Sum*)

   Solusi himpunan fuzzy diperoleh dengan cara melakukan bounded sum terhadap semua output daerah fuzzy. Secara umum dapat dituliskan :

   $$\mu_{sf}[x_i] = \min(1, \mu_{sf}[x_i] + \mu_{kf}[x_i]) \quad \text{.....(2.8)}$$

c. Metode Probabilistik OR (*probor*)

   Solusi himpunan fuzzy diperoleh dengan cara melakukan product terhadap semua output daerah fuzzy. Secara umum dapat dituliskan :

   $$\mu_{sf}[x_i] = (\mu_{sf}[x_i] + \mu_{kf}[x_i]) - (\mu_{sf}[x_i] * \mu_{kf}[x_i]) \quad \text{.....(2.9)}$$

   Keterangan :

   $\mu_{sf}[x_i]$: nilai keanggotaan solusi fuzzy sampai aturan ke-i

   $\mu_{kf}[x_i]$: nilai keanggotaan konsekuen aturan ke-i

Metode defuzzyfikasi yang akan digunakan adalah Metode Center of Area (*COA*). Dimana pada metode ini, solusi crisp diperoleh dengan cara mengambil titik pusat (z*) daerah fuzzy. Secara umum dirumuskan :

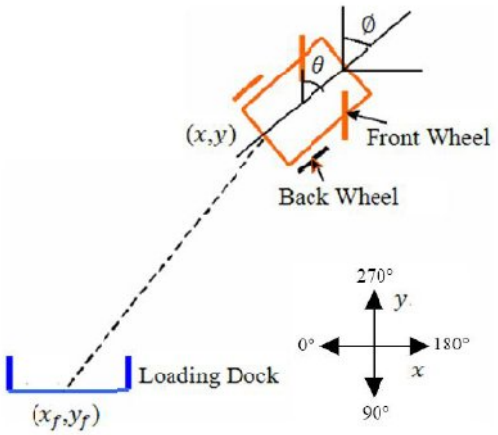
$$z^* = \frac{\int \quad ( )}{\int \quad ( )} \quad \text{.....(2.10)}$$

$$z^* = \frac{\Sigma \quad (\quad)}{\Sigma \quad (\quad)} \quad \text{.....(2.11)}$$

2.6 Permasalahan Pada Sistem Parkir Seri Otomatis

Pada simulasi ini posisi truk ditentukan oleh 3 variable ɵ, x dan y. Dimana variable θ merupakan sudut orientasi, dan untuk pasangan koordinat (x,y) merupakan sebagai posisi dimana mobil akan memulai untuk melakukan proses parkir. Sedangkan output dari simulasi ini yaitu θ, sudut kemudi (*steering angle*)yang digunakan untuk mengontrol mobil tersebut agar bisa mencapai posisi parkir yang ditentukan secara otomatis.

Mobil akan melakukan gerak maju atau mundur dari rules dan membership function yang telah ditentukan**.** Sehingga akan menghasilkan kontrol mobil dengan sudut kemudi yang tepat, agar mobil bisa mencapai posisi awal parkir yang telahditentukan.



Gambar 2.14 Perancangan mobildanlahanparkir

Untuk mempermudah pada simulasi sistem control ini diasumsikan untuk masukan Ø berada pada jarak [-90^0, -270^0] sedangkan untuk masukan x dan y berada pada jarak [0,10] dan untuk keluaran berada pada jarak [-30, 30].

Karena sistem ini masih menggunakan simulasi, jadi harus mengetahui terlebih dahulu dinamika prosedur untuk pergerakan maju-mundurnya mobil. Dan pada sistem ini menggunakan persamaan kinematika sebagai berikut.

$$x(t+1) = x(t) - \cos(\emptyset(t) + \theta(t)) - \sin(\theta(t) + \emptyset(t)) \quad (2.12)$$

$$y(t+1) = y(t) - \sin(\emptyset(t) + \theta(t)) - \cos(\theta(t) + \emptyset(t)) \quad (2.13)$$

$$\emptyset(t+1) = \emptyset(t) - \sin^{-1} \frac{(\ )}{(\ )} \quad (2.14)$$

Dimana, (x , y) = koordinat bagian belakang mobil sebagai poros titik tengah

b = panjang dari mobil tersebut

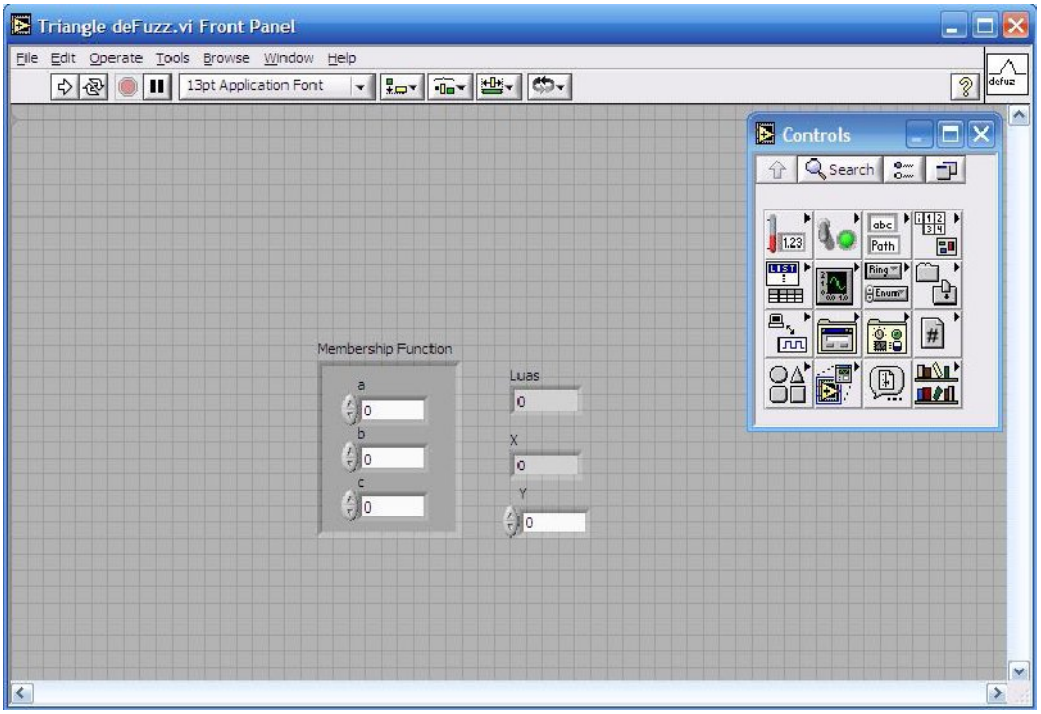
2.7 Perangkat Lunak Labview

LabVIEW adalah sebuah software pemograman yang diproduksi oleh National Instruments. Seperti bahasa pemograman lainnya yaitu C++, matlab atau visual basic, LabVIEW juga mempunyai fungsi dan peranan yang sama, perbedaannya adalah LabVIEW menggunakan bahasa pemrograman berbasis grafis atau blok diagram sedangkan bahasa pemrograman lainnya menggunakan text. Program LabVIEW dikenal dengan sebutan VI atau virtual instruments karena penampilan dan operasinya dapat meniru sebuah instrument. Pada LabVIEW, user pertama-tama membuat user interface atau front panel dengan menggunakan kontrol dan indikator, yang dimaksud dengan kontrol adalah knobs, push buttons, dials dan peralatan input lainnya sedangkan yang dimaksud dengan indikator adalah graphs, LEDs dan peralatan display lainnya. Setelah menyusun user interface, lalu user menyusun blok diagram yang berisi kode-kode VIs untuk

mengontrol front panel. Software LabVIEW terdiri dari tiga komponen utama,yaitu:

2.7.1 *Front Panel*

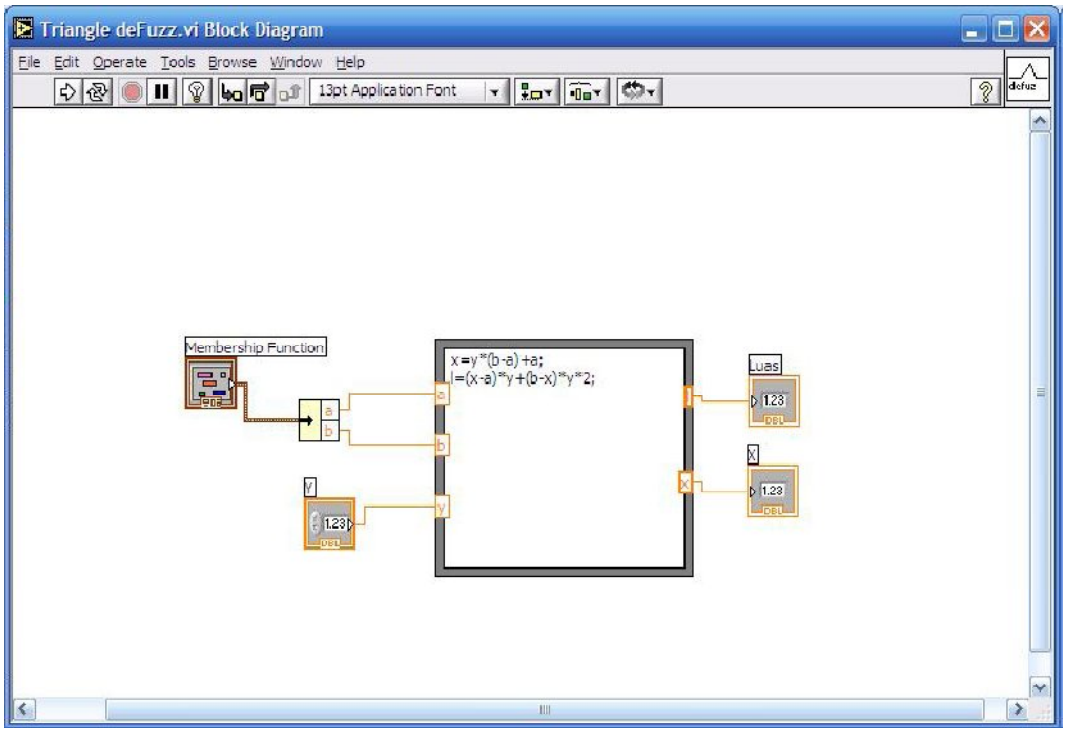
Front Panel adalah bagian window yang mempunyai background abu-abu serta mengandung kontrol dan indikator. Frontpanel digunakan untuk membangun sebuah VI, menjalankan program dan mendebug program. Tampilan dari front panel dapat di lihat pada gambar 2.10.



Gambar 2.10 *Front Panel*

2.7.2 Blok Diagram

Blok diagram adalah bagian window yang berlatar belakang putih berisi source code yang dibuat dan berfungsi sebagai instruksi untuk front panel. Tampilan dari blok diagram dapat lihat pada gambar 2.11.



Gambar 2.11 Blok diagram

2.7.3 *Control dan Function Pallete*

Control dan function pallete digunakan untuk membangun sebuah VI.

a. *Control Pallete*

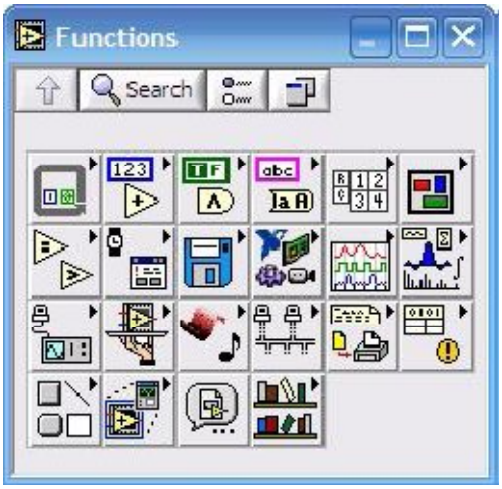
Control pallete merupakan tempat beberapa control dan indikator padafront panel, *control pallete* hanya tersedia di front panel, untukmenampilkan *control pallete* dapat dilakukan dengan mengklik windows>> show *control pallete* atau klik kanan pada front panel.Contoh *controlpallete* ditunjukkan pada gambar 2.12.



Gambar 2.12 *Control Pallete*

b. *Function Pallete*

Function pallete di gunakan untuk membangun sebuah blok diagram,*function pallete* hanya tersedia pada blok diagram, untukmenampilkannya dapat dilakukan dengan mengklik windows >> show*control pallete* atau klik kanan pada lembar kerja blok diagram. Contoh dari *function pallete* ditunjukkan pada gambar 2.13.



Gambar 2.13 *Function Pallete*