

BAB III

LANDASAN TEORI

Bab ini akan memaparkan berbagai teori yang melandasi penulis dalam membangun sistem yang nantinya akan dibuat.

3.1. Pengertian Optimalisasi

Secara umum pengertian optimalisasi menurut Kamus Besar Bahasa Indonesia (Depdikbud, 1995:628) adalah berasal dari kata optimal yang berarti terbaik atau pencarian nilai terbaik (minimum atau maksimum) dari beberapa fungsi yang diberikan pada suatu konteks. Adapun pengertian dari optimalisasi adalah sebuah proses, cara dan perbuatan (aktivitas/kegiatan) untuk mencari solusi terbaik dalam beberapa masalah, dimana yang terbaik sesuai dengan kriteria tertentu.

3.2. Lampu Lalu Lintas

3.2.1. Pengertian

Satu metode yang paling penting dan efektif untuk mengatur lalu lintas di persimpangan adalah dengan menggunakan lampu lalu lintas (Kandaga & Tjahjadi, 2011). Lampu lalu lintas adalah sebuah alat elektrik (dengan sistem pengaturan waktu) yang memberikan hak jalan (pada saat lampu menyala warna hijau) pada satu arus lalu lintas atau lebih sehingga aliran lalu lintas ini bisa melewati persimpangan dengan aman dan efisien atau lampu yang mengendalikan arus lalu lintas yang terpasang di persimpangan jalan, tempat penyeberangan pejalan kaki (*zebra cross*) dan tempat arus lalu lintas lainnya, namun yang menjadi ruang lingkup penelitian ini adalah lampu lalu lintas di persimpangan jalan. Banyaknya kendaraan yang lalu lalang di kota besar menyebabkan kemacetan yang sangat mungkin terjadi (Yudanto, et al., 2013). Lampu lalu lintas dibangun sangat sesuai untuk :

- a. Penundaan berlebihan pada rambu berhenti dan rambu pengendali kecepatan.

- b. Masalah yang timbul akibat tikungan jalan.
- c. Tabrakan sudut sisi.
- d. Kecelakaan pejalan kaki.

Keuntungan dengan merancang lampu lalu lintas yang benar pada persimpangan:

- a. Mengurangi frekuensi tipe kecelakaan tertentu, khususnya kecelakaan tipe sudut kanan.
- b. Menghasilkan pergerakan lalu lintas yang teratur.
- c. Menyediakan arus yang kontinyu bagi iring-iringan kendaraan melalui koordinasi yang memadai pada kecepatan tertentu di rute tertentu.
- d. Memungkinkan kendaraan dan pejalan kaki untuk melintasi lalu lintas yang sangat ramai.
- e. Pengendalian lalu lintas menjadi lebih ekonomis dibandingkan metode manual.

Kerugian memasang lampu lalu lintas yang buruk pada persimpangan :

1. Meningkatkan frekuensi kecelakaan.
2. Penundaan yang terlalu lama.
3. Pelanggaran lampu lalu lintas.
4. Perjalanan memutar melalui rute yang lebih jauh.

Secara umum, lampu lalu lintas dipasang pada suatu persimpangan berdasarkan alasan spesifik berikut :

- a. Untuk meningkatkan keamanan sistem secara keseluruhan.
- b. Untuk mengurangi waktu tempuh rata-rata di sebuah persimpangan, sehingga meningkatkan kapasitas.
- c. Untuk menyeimbangkan kualitas pelayanan di seluruh aliran lalu lintas.

3.2.2. Warna Lampu Lalu Lintas

Warna yang digunakan pada lampu lalu lintas :

- a. Lampu menyala merah: waktu yang diperuntukkan bagi pergerakan kendaraan untuk berhenti dan diperuntukkan bagi pejalan kaki untuk menyeberang jalan.
- b. Lampu menyala kuning: waktu yang diperlukan untuk memberikan peringatan kepada pengemudi bahwa nyala lampu akan berganti.
- c. Lampu menyala hijau: waktu yang diperuntukkan bagi pengendara untuk melewati persimpangan.

3.3. Persimpangan Jalan

Persimpangan adalah simpul pada jaringan jalan yang merupakan pertemuan antar jalan dan perpotongan lintasan kendaraan. Persimpangan merupakan factor yang penting dalam menentukan kapasitas dan waktu perjalanan pada suatu jaringan jalan, khususnya daerah perkotaan. Terdapat 2 jenis persimpangan yaitu persimpangan bersinyal dan persimpangan tanpa sinyal.

3.4. Software Matlab

Matlab merupakan bahasa pemrograman yang digunakan pada teknik komputasi yang digunakan untuk menyelesaikan matematik dan metode numerik. Matlab menyediakan fasilitas untuk visualisasi, komputasi, pemrograman, dan pengolahan basis data. Selain itu pada Matlab juga atersedia fitur yang dikelompokkan berdasarkan aplikasi tertentu yaitu *toolbox*. Matlab ini sendiri merupakan bahasa pemrograman yang termasuk tingkat tinggi dikembangkan oleh MathWorks, tujuan adanya matlab ini adalah pengguna dapat melakukan analisis data, mengembangkan algoritma, membuat model, dan dengan adanya fungsi *built-in* dan *tools* akan memudahkan untuk mengeksplorasi berbagai pendekatan dan memperoleh sebuah solusi yang lebih cepat dibandingkan menggunakan *spreadsheets* atau bahasa pemrograman seperti *C/C++* atau *Java*. Mengapa demikian ? Karena pada Matlab memiliki konsep *array/matrik* sebagai standar variabel elemennya tanpa perlu pendeklarasian *array* seperti pada bahasa

pemrograman lainnya. Matlab dapat juga diintegrasikan dengan bahasa pemrograman *C*, *Java*, *.Net*, dan *Microsoft Excel* (Kodong, 2015). Beberapa bagian penting yang terdapat pada interface Matlab, yaitu:

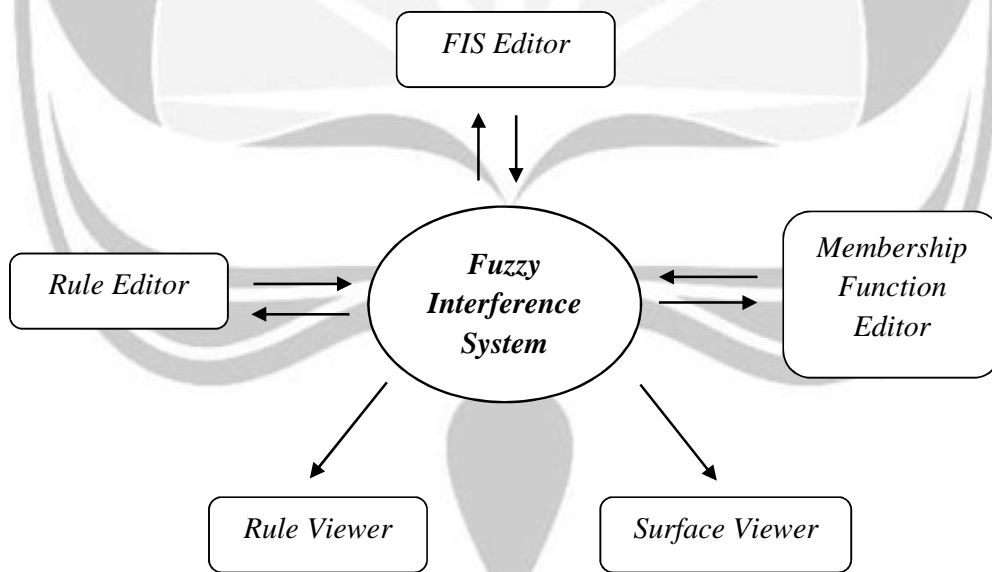
1. *Command Windows* adalah jendela untuk memberikan perintah secara manual dan berfungsi untuk menjalankan seluruh fungsi yang disediakan oleh Matlab.
2. *Workspace* untuk menginformasikan kepada pengguna tentang variabel yang dipakai selama menggunakan Matlab.
3. *Current Directory* untuk memilih direktori yang aktif.
4. *Command History* untuk menyimpan perintah yang sudah pernah ditulis pada *command window*.

Dalam Matlab juga memiliki User Interface atau yang sering didengar dengan nama GUI (*Graphical User Interface*) yang dibangun dengan obyek grafis. Dalam GUI Matlab sudah disediakan *uicontrol* yang tersedia pada editor *figure* dengan beberapa *uicontrol* pengguna dapat dimudahkan, beberapa *uicontrol* yang dapat digunakan sebagai berikut:

1. *Pushbutton* berfungsi untuk menghasilkan sebuah tindakan jika diklik.
2. *Toggle Button* berfungsi sama seperti *pushbutton* hanya saat *pushbutton* ditekan maka tombol akan kembali seperti semula jika mouse dilepas.
3. *Radio Button* mirip dengan tombol *check box* hanya pada *radio button* hanya dapat menandai atau memilih satu pilihan.
4. *Check Box* berfungsi jika terdapat beberapa pilihan mandiri atau tidak tergantung dengan pilihan lainnya.
5. *Edit Text* berfungsi untuk memasukkan atau menginputkan suatu data dari keyboard.
6. *Static Text* berfungsi untuk menghasilkan teks bersifat statis atau tetap.
7. *Slider* berfungsi untuk menginputkan nilai yang diinginkan tidak menggunakan keyboard hanya menggeser slider secara vertikal atau horisontal ke nilai yang diinginkan sehingga pemasukan nilai data dapat dilakukan secara fleksibel.
8. *Panel* berfungsi untuk mengelompokkan kontrol yang berhubungan.

9. *List Box* berfungsi untuk menampilkan semua daftar item yang terdapat pada *string property* dan item yang ada dapat dipilih satu atau lebih.
10. *Pop Menu* berfungsi menampilkan daftar pilihan yang didefinisikan pada *string property* ketika mengklik tanda panah pada aplikasi yang memiliki kesamaan seperti *radio button*.
11. *Axes* berfungsi untuk menampilkan grafik atau gambar.

Fuzzy Logic Toolbox merupakan fasilitas *Graphical User Interface (GUI)* yang diberikan Matlab untuk mempermudah membangun suatu sistem *fuzzy* atau sekumpulan *tool* yang akan membantu dalam perancangan sistem *fuzzy*. Ada 5 GUI *tools* yang digunakan untuk membangun, mengedit, dan mengobservasi sistem penalaran pada *fuzzy*, yaitu *fuzzy inference system (fis) editor*, *membership function editor* dapat membaca dan memodifikasi *fis*, *rule editor*, *rule viewer*, dan *surface viewer* hanya dapat membaca tidak dapat mengedit atau memodifikasi.

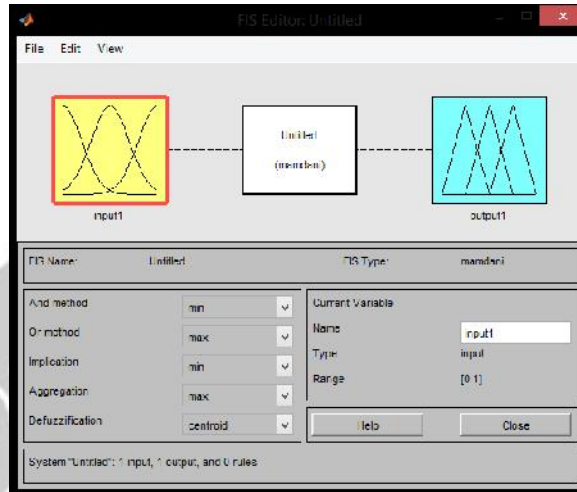


Gambar 3.1 Fuzzy Logic Toolbox Matlab

Keterangan :

- a. *Fuzzy Inference System (FIS) Editor*

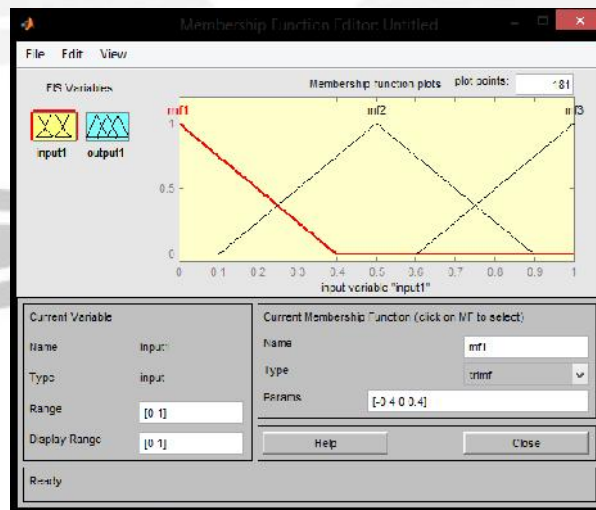
Digunakan untuk langkah awal membuat suatu penalaran *fuzzy* yang baru, dengan menuliskan “*fuzzy*” pada *command line* pada matlab.



Gambar 3.2 Tampilan FIS Editor

b. *Membership Fuction Editor*

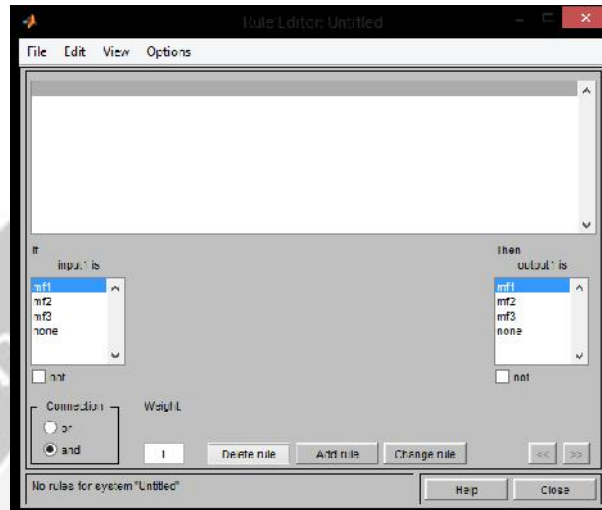
Digunakan untuk mengedit fungsi keanggotaan himpunan *fuzzy* untuk tiap-tiap variabel *input* dan *output*.



Gambar 3.3 Tampilan Membership Fuction Editor

c. *Rule Editor*

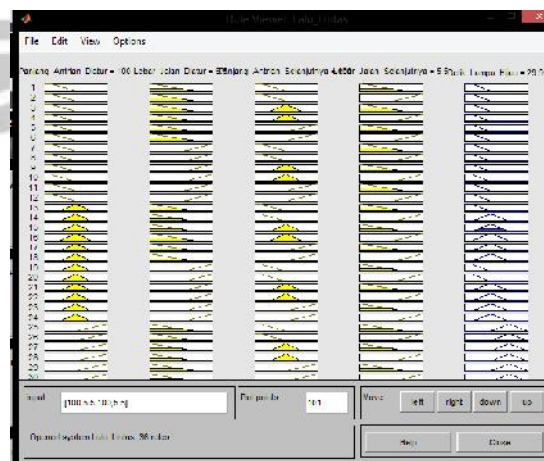
Digunakan untuk mengedit dan menampilkan aturan yang akan atau telah dibuat sebelumnya.



Gambar 3.4 Tampilan Rule Editor

d. *Rule Viewer*

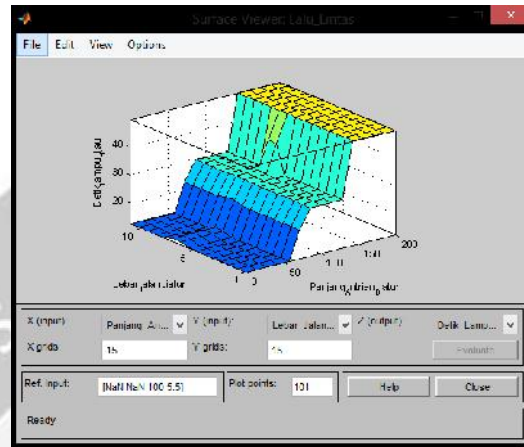
Digunakan untuk melihat alur penalaran *fuzzy* pada sistem meliputi pemetaan *input* yang diberikan ke tiap-tiap variabel *input*, aplikasi operator dan fungsi implikasi. Komposisi aturan sampai pada penentuan *output* tegas pada defuzzifikasi.



Gambar 3.5 Tampilan Rule Viewer

e. *Surface Viewer*

Digunakan untuk melihat gambar pemetaan antara variabel-variabel *input* dan *output*.



Gambar 3.6 Tampilan Surface Viewer

3.5. Metode Logika *Fuzzy*

3.5.1. Pengertian

Logika *fuzzy* adalah cabang dari sistem kecerdasan buatan (*Artificial Intelligent*) yang mengemulasi kemampuan manusia dalam berfikir ke dalam bentuk algoritma yang kemudian dijalankan oleh mesin. Algoritma ini digunakan dalam berbagai aplikasi pemrosesan data yang tidak dapat direpresentasikan dalam bentuk biner. Logika *fuzzy* diperkenalkan pada tahun 1965 oleh Lotfi Zaeh yang merupakan perlengkapan matematika untuk melakukan komputasi variabel yang bersifat tidak pasti. Menurut buku yang dibuat oleh Kusumadewi, logika *fuzzy* menginterpretasikan pernyataan yang samar menjadi sebuah pengertian yang logis (Kusumadewi, 2010) atau *fuzzy* secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai benar atau salah secara bersamaan. Logika *fuzzy* digunakan untuk mengestimasi sesuatu, mengambil keputusan, dan sebagai kontrol mekanik. Dalam himpunan *fuzzy* dikenal derajat keanggotaan yang memiliki rentang nilai 0 sampai 1. Berbeda dengan himpunan tegas (*crisp*) dengan nilai

keanggotaan hanya terdapat dua kemungkinan, yaitu 1 dan 0. Teori *fuzzy* menyediakan mekanisme untuk mempresentasikan ukuran variabel linguistik seperti “padat”, “sedang”, “tidak padat”, dan sebagainya (Sivanandam, Deepa, & Sumanthi, 2007).

Beberapa alasan mengapa orang-orang menggunakan logika *fuzzy* untuk menyelesaikan kasus ini, karena konsep logika *fuzzy* lebih mudah dimengerti, sederhana, fleksibel, memiliki toleransi terhadap data-data yang tidak tepat, dan didasarkan pada bahasa alami.

3.5.2. Inferensi *Fuzzy*

Metode Mamdani merupakan salah satu metode *inferensi fuzzy* yang digunakan untuk menyelesaikan masalah yang berkaitan dengan optimalisasi lalu lintas. Metode Mamdani juga sering dikenal sebagai metode *Max-Min* yang telah diperkenalkan oleh Embrahin Mamdani tahun 1975.

Proses untuk mendapatkan *outputnya* dilakukan dalam 4 tahap, yaitu :

a. Pembentukan Himpunan *Fuzzy*

Pada metode mamdani, baik variabel input maupun variabel *output* dibagi menjadi satu atau lebih himpunan *fuzzy*.

b. Fuzzifikasi

Proses penentuan derajat keanggotaan berdasarkan data masukkan pengguna dari panjang antrian kendaraan dan lebar jalur.

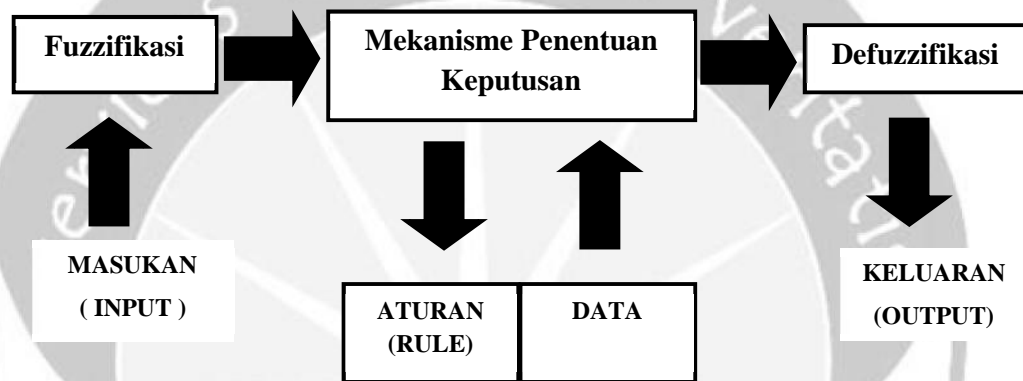
c. Inferensi Berdasarkan *Rule*

Dalam inferensi menggunakan metode mamdani adalah khususnya metode *minimum* dengan menggunakan operator *AND*. Nilai input akan dimasukkan dan telah dihitung akan menempati derajat keanggotaannya masing-masing dan untuk keluaran masing-masing *rule* akan diambil nilai terkecil atau *minimum* dari semua nilai yang telah dihitung.

d. Defuzzifikasi

Proses untuk menentukan keputusan akhir yang dihasilkan oleh penalaran dalam bentuk *fuzzy* berdasarkan pengeluaran *rule* berupa derajat keanggotaan menjadi variabel numerik *nonfuzzy*. Pada proses ini diperoleh

dari suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam *range* tertentu maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output*. Pada penulisan tugas akhir ini metode yang digunakan untuk defuzzifikasi komposisi aturan Mamdani adalah *Mean of Maximum* (MOM) dengan mengambil nilai rata-rata domain *fuzzy* yang memiliki nilai keanggotaan pada daerah *fuzzy* maksimum. (Koten, 2012).



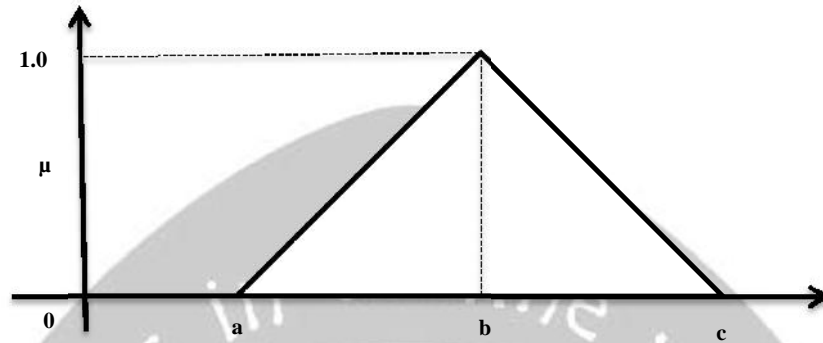
Gambar 3.7 Mekanisme Penentuan Keputusan Logika Fuzzy

3.5.3. Fungsi Keanggotaan

Fungsi Keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotannya yang memiliki interval antara 1 dan 0. Melalui pendekatan fungsi maka akan mendapatkan nilai keanggotaan. Contoh gambar kurva atau grafik fungsi keanggotaan sebagai berikut:

a. Kurva Segitiga

Kurva segitiga pada dasarnya adalah 2 garis linear yang digabungkan.

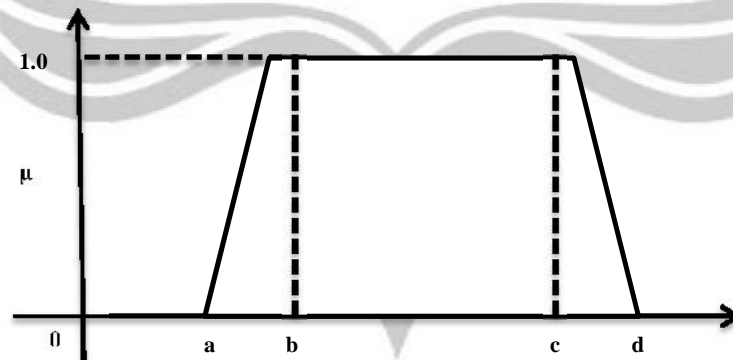


Gambar 3.8 Kurva Segitiga

Dengan fungsi keanggotaan :

$$\mu_{[x]} = \begin{cases} 0; x \leq a \text{ atau } x \geq c \\ (x - a)/(b - a); a < x \leq b \\ (c - x)/(c - b); b < x < c \end{cases} \dots\dots (3.1)$$

b. Kurva Trapesium

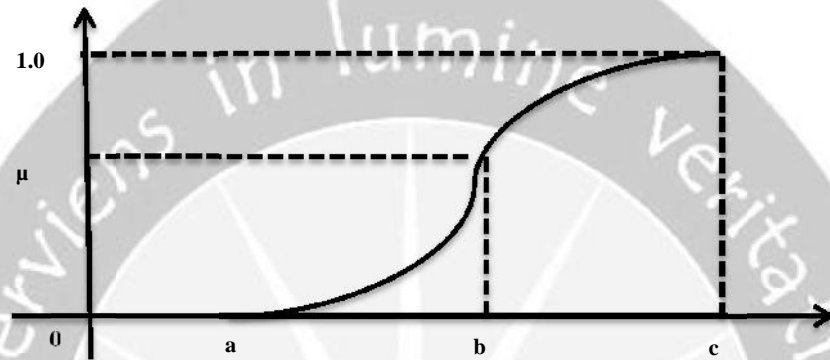


Gambar 3.9 Kurva Trapesium

Dengan fungsi keanggotaan :

$$\mu_{[x]} = \begin{cases} 0; x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); a < x \leq b \\ 1; b < x \leq c \\ (d-x)/(d-c); c < x < d \end{cases} \dots\dots (3.2)$$

c. Kurva Sigmoid

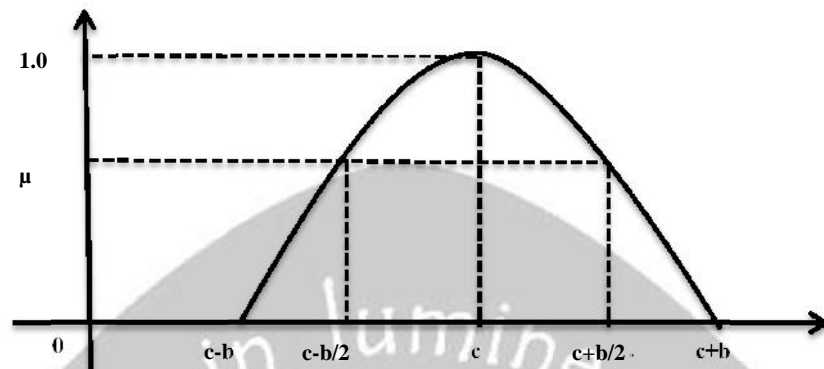


Gambar 3.10 Kurva Sigmoid

Dengan fungsi keanggotaan :

$$\mu_{[x; a, b, c]} = \begin{cases} 0; x \leq a \\ 2((x-a)/(c-a))^2; a < x \leq b \\ 1 - 2((c-x)/(c-a))^2; b < x < c \\ 1; x \geq c \end{cases} \dots\dots (3.3)$$

d. Kurva Phi



Gambar 3.11 Kurva Phi

Dengan fungsi keanggotaan :

$$\sim[x; a, b, c] = \begin{cases} \sim[x; c - b, c - b / 2, c]_{sigmoid} ; x \leq c \\ \sim[x; c, c + b / 2, c + b]_{sigmoid} ; x > c \end{cases} \dots\dots (3.4)$$