

Terbit *online* pada laman web jurnal : <https://jes-tm.org/index.php/jestm/index>

## Journal of Engineering Science and Technology Management

| ISSN (Online) 2828 -7886 |



Article

# Study of Some Nomenclature and Membership Function Using MATLAB Program Code

**Kailas L. Vairal**

Pravara Rural Engineering College, Savitribai Phule, Pune University, Pune, Loni-413713, Ahilyanagar, Maharashtra, India.

DOI: 10.31004/jestm.v5i2.273

E-mail: [kailashvairal7@gmail.com](mailto:kailashvairal7@gmail.com)

### ARTICLE INFORMATION

Volume 5 Issue 2  
Received: 13 Augustus 2025  
Accepted: 05 September 2025  
Publish *Online*: 30 September 2025  
*Online*: at <https://JESTM.org/>

### Keywords

Computational Modelling  
Fuzzy Logic  
MATLAB  
Membership Function  
System Optimization

### ABSTRACT

Fuzzy logic has emerged as an essential computational paradigm for addressing uncertainty and nonlinearity in engineering systems. This paper presents a comprehensive study and application of fuzzy membership functions implemented in MATLAB for system optimization and control. Various types of membership functions—triangular, trapezoidal, Gaussian, generalized bell, and sigmoidal—are analyzed based on their shape characteristics, adaptability, and computational behavior. The study further demonstrates how these functions can be effectively utilized in the design of fuzzy controllers and optimization models to enhance decision precision in nonlinear environments. MATLAB scripts are developed to generate and evaluate membership functions dynamically, providing visual and numerical comparisons of their response profiles. The results highlight that Gaussian and bell-shaped functions offer smoother transitions and higher flexibility in representing gradual changes, making them ideal for control applications. The study contributes to improving the understanding of fuzzy modelling techniques, particularly in optimizing control responses and decision systems within uncertain environments.

## 1. Introduction

Modern engineering systems often operate in environments characterized by vagueness, imprecision, and nonlinear relationships that cannot be efficiently handled by conventional mathematical models. Fuzzy logic, introduced by Zadeh (1965), provides a powerful framework for representing and reasoning with uncertainty using linguistic variables. In this paradigm, membership functions play a central role in defining the degree of belonging of an input to a fuzzy set. The accuracy of a fuzzy system highly depends on the appropriate selection and configuration of membership functions.

Recent advances in computational intelligence have made fuzzy systems indispensable in various applications such as industrial automation, energy management, medical diagnostics, and predictive modelling. MATLAB provides a flexible platform for designing, visualizing, and optimizing fuzzy systems through its Fuzzy Logic Toolbox. However, despite its accessibility, many studies have primarily focused on theoretical aspects or limited simulation examples without bridging the gap toward practical optimization and control applications.

This study aims to analyze and apply various membership functions using MATLAB for system optimization and control. The research seeks to establish a comparative framework for evaluating the performance of different functions based on response smoothness, computational cost, and adaptability to nonlinear changes. The outcomes of this study are intended to provide researchers and practitioners with an empirical reference for selecting membership functions in the design of fuzzy-based controllers and optimization systems.

## 2. Literature Review

Fuzzy logic systems have been extensively applied in diverse fields due to their capability to handle ambiguous and complex systems. According to Mendel (2017), the performance of a fuzzy inference system (FIS) is largely influenced by the structure and type of membership functions employed. Studies by Kaya and Alkan (2020) demonstrated that Gaussian and bell-shaped functions provide higher accuracy in control applications due to their continuous differentiability, which

supports gradient-based optimization.

In contrast, Sivanandam et al. (2019) emphasized that simpler functions, such as triangular and trapezoidal forms, are more suitable for real-time applications where computational efficiency is critical. Moreover, research by Ibrahim et al. (2021) explored hybrid fuzzy-neural systems, showing that adaptive tuning of membership functions through learning algorithms significantly enhances prediction accuracy in nonlinear control systems.

In recent developments, fuzzy-based optimization has gained attention for its potential in energy management, traffic control, and industrial process regulation (Ali & Khairuddin, 2022; Rahman et al., 2023). MATLAB-based implementations continue to play a pivotal role in these studies, offering versatile simulation tools and visualization capabilities. Nevertheless, there remains a need for empirical studies that integrate comparative analysis of membership functions with their practical applications in optimization and control scenarios.

## 3. Research Methodology

This study follows three stages: (i) definition and parameterization of fuzzy membership functions (MFs), (ii) simulation and visualization in MATLAB, and (iii) comparative evaluation using control/optimization-relevant indicators.

### 3.1 Definition and Parameterization of Membership Functions

Let  $\mu: \mathbb{R} \rightarrow [0,1]$  denote a membership function. We consider five widely-used MFs due to their complementary properties:

#### (a) Triangular membership function

Using three variables  $[\sigma, \tau, \omega]$  defined by a triangular MF with  $(\sigma < \tau < \omega)$ , and represent MF as below triangle

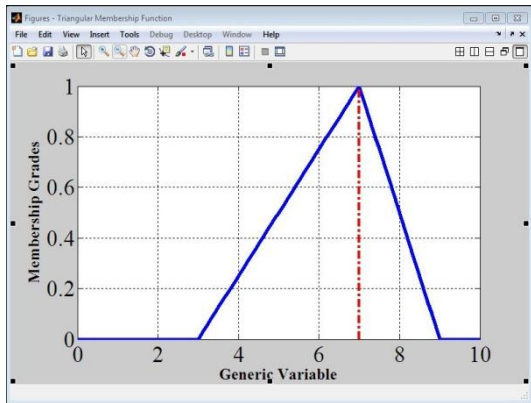
$$(u ; \underline{\sigma}, \tau, \omega) = \begin{cases} 0 & , \quad u \leq \sigma \\ \frac{u-\sigma}{\tau-\sigma} & , \quad \sigma \leq u \leq \tau \\ \frac{\omega-u}{\omega-\tau} & , \quad \tau \leq u \leq \omega \\ 0 & , \quad \omega \leq u \end{cases}$$

Vertices

The equation denoted as triangle  $(u ; \sigma, \tau, \omega) = \max \left\{ \min \left[ \frac{u-\sigma}{\tau-\sigma}, \frac{\omega-u}{\omega-\tau} \right], 0 \right\}$

**MATLAB Program Code:**

```
'clc';x=(0:0.1:10);y=trimf(x,[3,7,9]);
plot(x,y,'linewidth',7.0);
ylim([0 1]);
xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',21.0);
```



**Figure 1.** MF of Triangular

**(b) Trapezoidal Membership Function**

A trapezoidal MF, specified by parameter [e, f, g, h], with  $e \leq f \leq g \leq h$  as follows

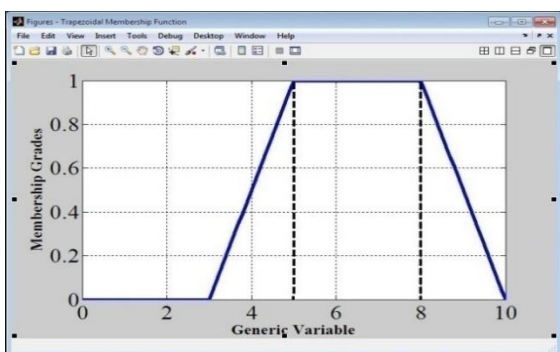
$$\text{trapezoid}(v; e, f, g, h) = \begin{cases} 0, & v \leq e \\ \frac{v-e}{f-e}, & e \leq v \leq f \\ 1, & f \leq v \leq g \\ \frac{g-v}{g-h}, & g \leq v \leq h \\ 0, & h \leq v \end{cases}$$

Vertices

Other expression for TMF denoted as  $\text{trapezoid}( ; e, f, g, h) = \max \{ \min [ \frac{v-e}{f-e}, 1, \frac{g-v}{g-h}, 0 ]$

**MATLAB Program Code:**

```
'clc';x=(0:0.1:10);y=trapmf(x,[3,5, 8,10]);
plot(x,y,'linewidth',7.0);
ylim([0,1]);ylabel('Membership
Grades');xlabel('Generic Variable');
set(gca,'FontName','Times','FontSize',26.0);
```



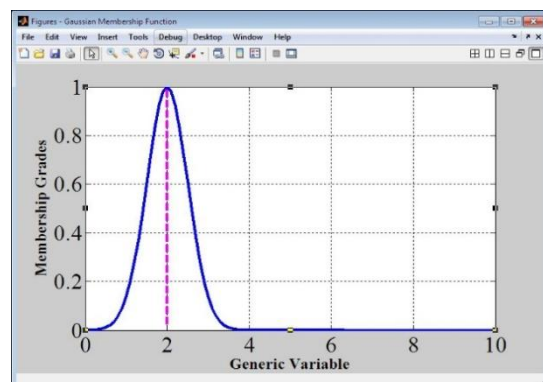
**Figure 2.** Trapezoidal Membership Function

**(c) Gaussian membership function**

The definite factors  $[k, \rho]$ ,  $\frac{1}{\rho}$  GMF defined as gaussian  $(y ; k, \rho) = e^{-\frac{1}{2}(\frac{y-k}{\rho})^2}$ . Where  $k$  denotes the Membership functions centre and  $\rho$  determines width of MF.

**MATLAB Program Code:**

```
'clc';x=(0:0.1:10); y=gaussmf(x,[0.5,2]);
plot(x,y,'linewidth',3.0);
ylim([0 1]);
xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',27.0);
```



**Figure 3.** Membership Function of Gaussian

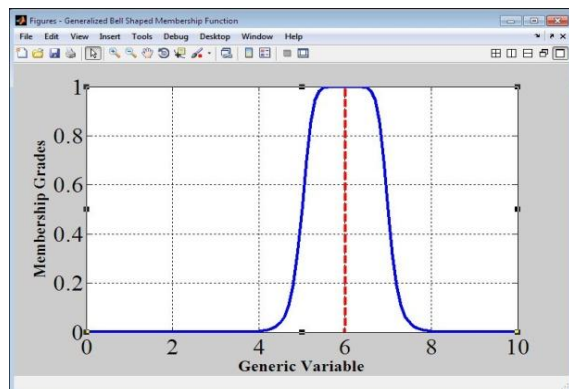
**(d) Generalized Bell-Shaped MF**

BMF is stated by variables  $[p, q, r]$  defined bell

$$(z ; p, q, r) = \frac{1}{1+|\frac{z-r}{p}|^{2q}}$$

**MATLAB Program Code:**

```
'clc'; x=(0:0.1:10);y=gbellmf(x,[1,4,6]);
plot(x,y,'linewidth',3.0);
ylim([0 1]);xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',25.0);
```



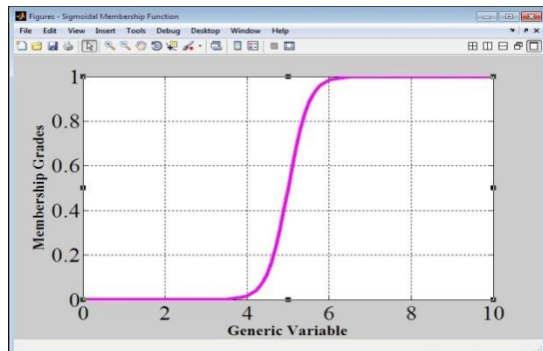
**Figure 4.** Generalized Bell-Shaped MF

**(e) Membership Function of Sigmoidal**

The sigmoidal MF is  $\text{sig}(w ; r, t) = \frac{1}{1 + e^{-r(w-t)}}$

**MATLAB Program Code:**

```
clc;x=(0:0.1:10);y=sigmf(x,[4, 5]);
plot(x,y,'linewidth',4.0);
ylim([0 1]);xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',18.0);
```



**Figure 5.** Sigmoidal Membership Function

Design choices. Triangular/trapezoidal favor speed and interpretability; Gaussian/bell favor smoothness and gradient-based tuning; sigmoidal supports asymmetric transitions.

**3.2 Simulation in MATLAB**

We generate MF curves over a normalized domain  $x \in [0,10]$ , then evaluate (i) visual profiles, (ii) runtime, and (iii) sensitivity to parameter changes.

**Membership Function**

The MF precise all the information as  $Z = \{(x_p, \mu_Z(x_p)) \mid x_p \in U, p = 1, \dots, m.\}$ , Where  $\mu_Z(x)$  stated as MF of  $Z$  lies among 0 and 1. The key features of the MF described and signify a graphically by MATLAB.

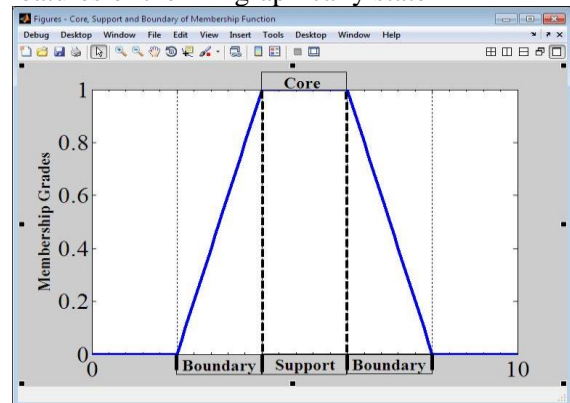
**MATLAB Code:**

```
clc;x=(0:0.1:10);y=trapmf(x, [3, 5, 7, 10]) ;
plot(x, y, 'line width',7.0) ;
ylim([0 1]); ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',25.0);
Core: The set  $t \mu_Z(y) = 1$  is the  $Z$ , Core ( $Z$ )
= $\{y \mid \mu_Z(y) = 1\}$ 
```

**Support:** The MF of  $Z$  is definite as the part of the universe defined by a MF ( $\neq 0$ ).

**Boundary:** The support of MF  $Z$  in universe that contains nonzero elements but incomplete

MF. The boundary is  $1 > \mu_Z(h) > 0$ . The main features of the MF graphically state



**Figure 6.** Core; Support; and Boundary of Membership Function

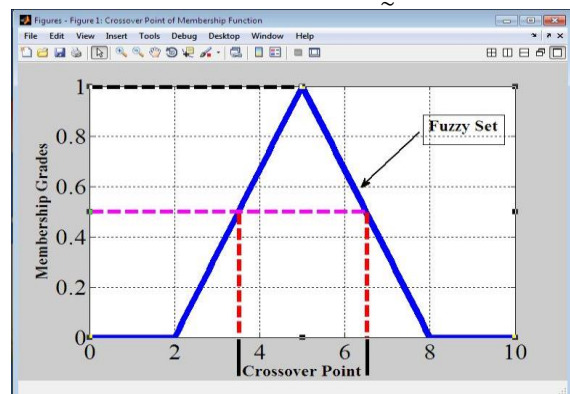
MF indicate as graph using MATLAB code.

**MATLAB Program Code:**

```
clc; x= (0 : 0.1 :10) ;
y=trimf(x,[2,5,8]);
plot(x,y,'line width', 7.0) ;
ylim([0 1]) ;
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',21.0);
```

**Crossover point:**

The crossover points of MF of  $A$  is 0.5.

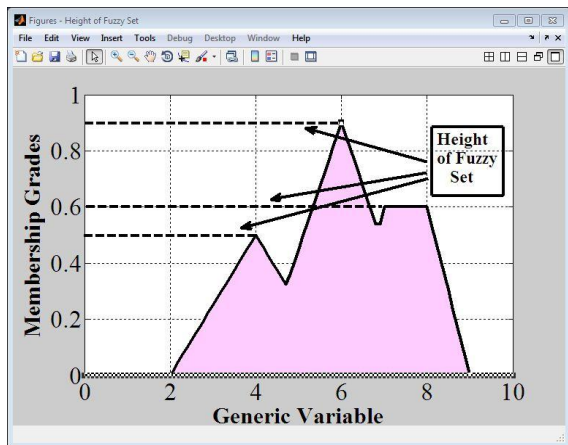


**Figure 7.** Crossover Point of MF

**Height:** The r height of  $A$  is:  $\text{hgt}(A) = \max\{\mu_A(x)\}$ . If  $\text{hgt}(A) < 1$  then  $A$  known as Subnormal fuzzy set. The  $\text{hgt}(A)$  is viewed as the degree of validity or credibility of information expressed by fuzzy set  $A$ . Using MATLAB code to explained MF graphically

**Program Code:**

```
x=(0:0.1:10);
mf1=trimf(x,[2,4,6]);
mf2=trimf(x,[4,6,8]);
mf3=trapmf(x,[6,7,8,9]);
y=max(max(0.5*mf1,0.9*mf2),0.6*mf3);
plot(x,y,'linewidth',3.5);
ylim([0 1]);ylabel('Membership
Grades');xlabel('Generic Variable') ;
'set(gca,'FontName','Times','FontSize',24.0) ;
```



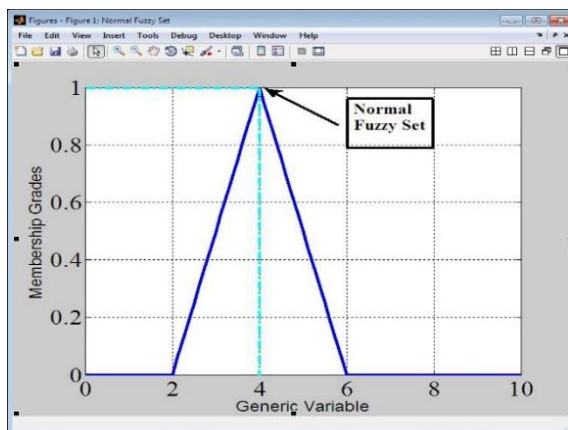
**Figure 8.** Height of FS

**Normal Fuzzy Set:**

A is normal if its core is non empty, or A point  $\tilde{u}$  in U such that  $\mu_A(u) = 1$ . The MF represents using MATLAB.

**MATLAB Code:**

```
x=(0:0.1:10);mf1=trimf(x,[2,4,6]);
y=1*mf1;
plot(x,y,'linewidth',3.5);
ylim([0,1]);ylabel('Membership
Grades');xlabel('Generic Variable') ;
set(gca,'FontName','Times','FontSize',22.0) ;
```



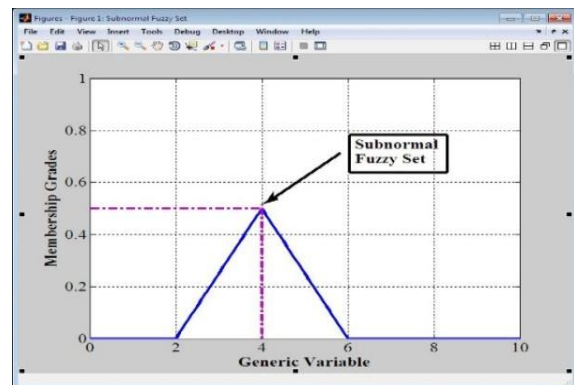
**Figure 9.** Normal Fuzzy Set

**Subnormal Fuzzy Set:**

A A is subnormal if its core is empty. In otherworld's A point x in universe discourse of X such that  $\mu_A(x) \neq 1$ . The MF described and represent a graphically using MATLAB as below:

**MATLAB Program Code:**

```
x=(0:0.1:10);mf1=trimf(x,[2,4,6]);y=0.5*mf1;
plot(x,y,'linewidth',3.5);
ylim([0 1]);xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',25.0);
```



**Figure 10.** Subnormal FS

**Normalization of a FS:**

Let A' is normalized version of a fuzzy set A which is represented by  $A' = \text{Norm}(A)$ , the normalization is carried out as stated below

$$A' = \text{Norm}(A) = \sum_x \frac{\mu_A(x)}{\text{hgt}(A)}$$

**Fuzzy Singleton:**

A MF can be representing as  $\mu_A(x) = 1$  is called fuzzy-singleton.

**$\alpha$  - Cut:** The  $\alpha$ -cut or  $\alpha$ -level set of A is a crisp set defined as  $A_\alpha = \{x | \mu_A(x) \geq \alpha\}$

**MATLAB Program Code:**

```
x=(0:0.1:10);y=trimf(x,[2,5,8]);
plot(x,y,'line width',7.0);
ylim([0,1]);
ylabel('membership grades');
xlabel('Generic Variable');
set(gca,'FontName','Times','FontSize',23.0);
```

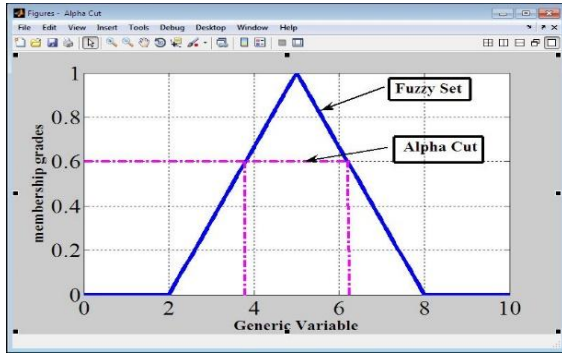


Figure 11. Alpha cut

**Strong  $\alpha$ - cut:** The strong  $\alpha$ -cut of FS stated as below

$$A'_{\alpha} = \{x \mid \mu_{\tilde{A}}(x) > \alpha\}$$

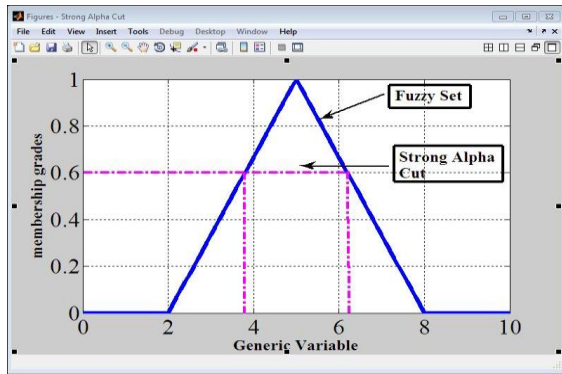


Figure 12. Strong-alpha cut

**Convexity:** A FSA is convex  $\Leftrightarrow \forall x_1, x_2 \in X$ , and  $\lambda \in [0, 1]$ , the following condition satisfies  $\mu_{\tilde{A}}(\lambda x_1 + (1 - \lambda)x_2) \geq \min[\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)]$ .

**MATLAB Program Code:**

```
x=(0:0.1:10);
y=gbellmf(x,[2,1,5]);
plot(x,y,'linewidth',7.0);
ylim([0 1]);ylabel('Membership Grades');
xlabel('Generic Variable');
set(gca,'FontName','Times','FontSize',26.0);
```

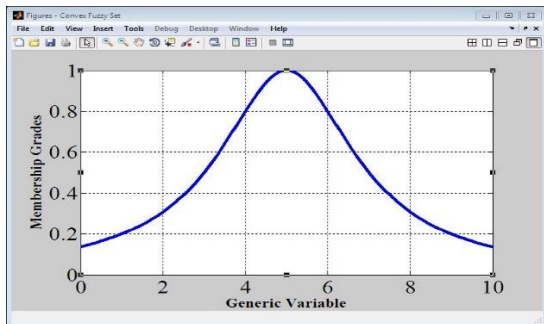


Figure 13. Convex Fuzzy Set

**Fuzzy Number:** Fuzzy number 'A' is satisfying the stipulation for normality and convexity

**MATLAB Program Code:**

```
x=(0:0.4:10);y=gbellmf(x,[1,5,2]);
plot(x,y,'linewidth',7.0);
ylim([0 1]);ylabel('Membership Grades');
xlabel('Generic Variable');
set(gca,'FontName','Times','FontSize',27.0);
```

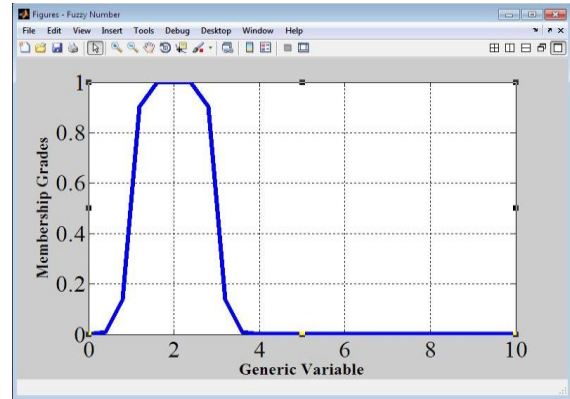


Figure 13. Fuzzy Number

**Band Width:** For, bandwidth of normal and convex fuzzy set A stated the distance between the two unique crossover points.

$$\text{width}(\tilde{A}) = |x_1 - x_2|, \text{ Here, } \mu_{\tilde{A}}(x_1) = \mu_{\tilde{A}}(x_2) = 0.5$$

**MATLAB Program Code:**

```
x=(0:0.1:10);
y=gbellmf(x,[1.5,1,1.9,0.9]);
plot(x,y,'linewidth',7.0);
ylim([0 1]);
xlabel('Generic Variable');
ylabel('Membership Grades');
set(gca,'FontName','Times','FontSize',19.0);
```

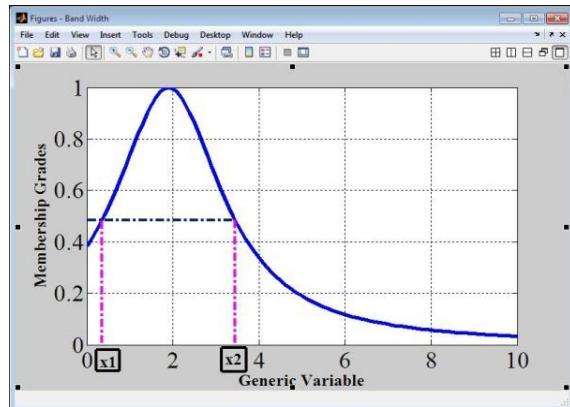


Figure 14. Band Width

**Fuzzy Symmetry:**

Let  $A$  symmetric if it satisfies point  $c$  given below  $\mu_A(c+x) = \mu_A(c-x), \forall x \in X$

**MATLAB Program Code:**

```
x=(0:0.1:10);
mf1=trimf(x,[1 3 5]);mf2=trapmf(x,[5,6,8,9]);
y=max(mf1,mf2);
plot(x,y,'linewidth',3.5);
ylim([0 1]); ylabel('Membership
Grades');xlabel('Generic Variable');
set(gca,'FontName','Times','FontSize',24.0);
```

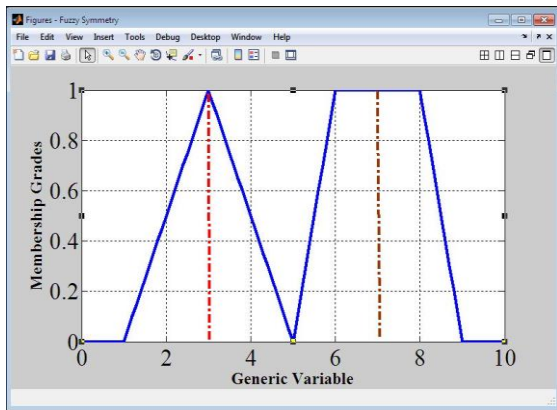


Figure 15. Fuzzy Symmetry

**Open Left FS:** A fuzzy set  $A$  is open left, if  $\lim_{n \rightarrow -\infty} \mu_A(x) = 1$ , and  $\lim_{n \rightarrow +\infty} \mu_A(x) = 0$

**MATLAB Program Code:**

```
x=(0:0.1:10);
mf1=trapmf(x,[0 0 4 4]);mf2=trimf(x,[4,4,7]);
y=max(1*mf1,1*mf2);
plot(x,y,'linewidth',3.5);
ylim([0 1]);
ylabel('MembershipGrades');xlabel('Generic
Variable');
set(gca,'FontName','Times','FontSize',28.0);
```

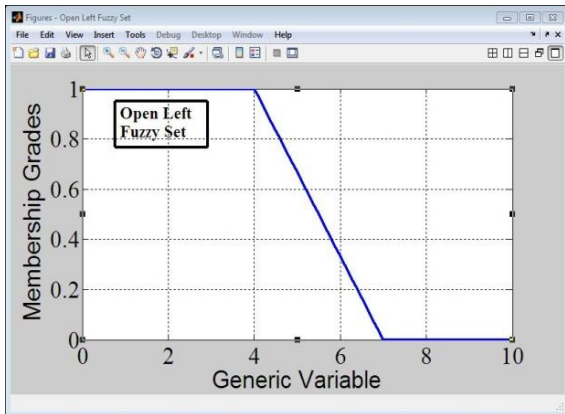


Figure 16. Open Left Fuzzy Set

**Open Right Fuzzy Set:**

A fuzzy set  $B$  is open right, if  $\lim_{n \rightarrow -\infty} \mu_B(x) = 0$  and  $\lim_{n \rightarrow +\infty} \mu_B(x) = 1$

**MATLAB Program Code:**

```
x=(0:0.1:10);
mf1=trapmf(x,[0 0 4 4]);mf2=trimf(x,[4,4,7]);
y=max(1*mf1,1*mf2);
plot(x,y,'linewidth',3.5);
ylim([0,1]);ylabel('Membership
Grades');xlabel('Generic Variable')
set(gca,'FontName','Times','FontSize',17.0);
```

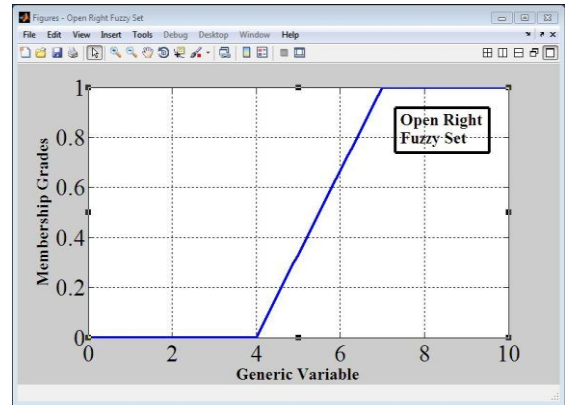


Figure 17. Open Right Fuzzy Set

**2.2 Standard Membership Function Left-Right Membership function**

A left-right membership function specified by three parameters  $[\alpha, \beta, \delta]$  as

$$LR(s; \delta, \alpha, \beta) = \begin{cases} F_L\left(\frac{\delta-s}{\alpha}\right), & s \leq \delta \\ F_R\left(\frac{s-\delta}{\beta}\right), & s \geq \delta \end{cases}$$

Where ' $\delta$ ' denotes the MFV is 1, ' $\alpha$ ' and ' $\beta$ ' controls the width of the both region.

**MATLAB Program Code:**

```
'clc';
'alpha=40;beta=10;c=70';
'x=(0:0.1:100)';
'for i=1:size(x,1)
'if(x(i,1)>=c)
'x1=(x(i,1)-c)/beta';
'y(i,1)=exp(-abs((x1)^3))';
'elseif(x(i,1)<=c)
'x2=(c-x(i,1))/alpha';
'y(i,1)=exp(-abs((x2)^3))';
end
end
'plot(x,y,'linewidth',4.0);
'ylim([0,1]);xlabel('Generic Variable');
'ylabel('Membership Grades');
'set(gca,'FontName','Times
New Roman','FontSize',19.0);
```

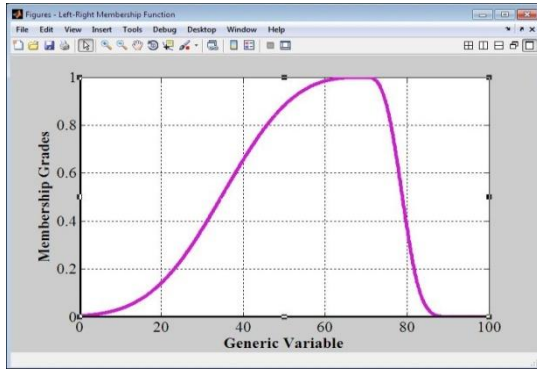


Figure 18. Left-Right Membership Function

### Pi Shaped ( $\pi$ ) Membership Function

The first MF described as

$$\pi_1(x; a, b) = \frac{1}{1 + \left(\frac{x-a}{b}\right)^2}$$

and other MF defined as the following

$$\pi_2(e; lw, lp, rp, rw) = \begin{cases} \frac{lw}{lp+lw-e} & , \quad e < lp \\ 1 & , \quad lp \leq e \leq rp \\ \frac{rw}{e-rp+rw} & , \quad e > rp \end{cases}$$

Parameters 'lw' and 'rw' classify the feet of the membership function, and parameters 'lp' and 'rp' definite its shoulders respectively.

### MATLAB Program Code:

```
'clc'; 'x=(0:0.1:10)';
'y=pimf(x,[2,5,8,10]);
'plot(x,y,'linewidth',4.0);
'yylim([0 1]);
'xlabel('Generic Variable');'ylabel('Membership Grades');
'set(gca,'FontName','Times New Roman','FontSize',15.0);
```

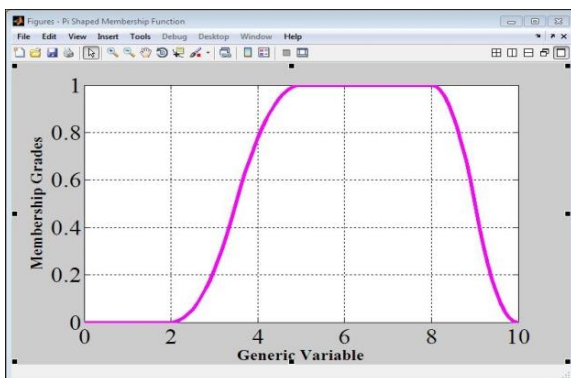


Figure 19. Pi Shaped Membership Function

### Open Left Membership Function

The open left membership is defined by two parameter  $[\alpha, \beta]$  as follow

$$\text{'openL}(l; \alpha, \beta)' = \begin{cases} 1 & , \quad l < \alpha \\ \frac{\beta-l}{\beta-\alpha} & , \quad \alpha \leq l \leq \beta \\ 0 & , \quad l > \beta \end{cases}$$

### MATLAB Program Code:

```
'clc';'x=(0:0.1:10)';
'alpha=3; beta=6';
'OpenL=zeros(size(x))';
'for i=1:size(x,1)'
'if x(i,1)<alpha'
'OpenL(i,1) = 1';
'elseif x(i,1)>beta'
'OpenL(i,1) = 0';
'else'
'OpenL(i,1)=(beta-x(i-1))./(beta-alpha)';
'end'
'end'
'plot(x,OpenL,'linewidth',4.0);
'yylim([0,1]);'xlabel('Generic Variable');
'ylabel('Membership Grades');
'set(gca,'FontName','Times New Roman','FontSize',17.0);
```

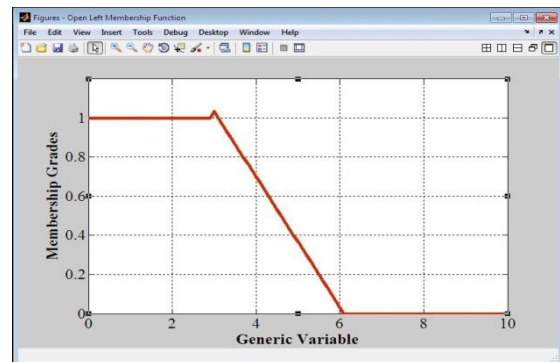


Figure 20. Open Left Membership Function

### Open Right Membership function

The open right membership is defined by two variables  $[\alpha, \beta]$ , represent as

$$\text{openR}(x; \alpha, \beta) = \begin{cases} 0 & , \quad x < \alpha \\ \frac{x-\alpha}{\beta-\alpha} & , \quad \alpha \leq x \leq \beta \\ 1 & , \quad x > \beta \end{cases}$$

### MATLAB Program Code:

```
clc;x=(0:0.1:10);alpha=3;beta=6;
'OpenR=zeros(size(x))';
'for i=1:size(x,1)'
'if x(i,1)<alpha'
'OpenR(i,1)=0';
'elseif x(i,1)>beta'
'OpenR(i,1)=1';
'else'
'OpenR(i,1)=(x(i-1)-alpha)/(beta-alpha)';
```

```

end
end
'plot(x,OpenR,'linewidth',4.0);
'yylim([0 1]);
xlabel('Generic Variable');
ylabel('Membership Grades');
'set(gca,'FontName','Times
New Roman','FontSize',20.0);

```

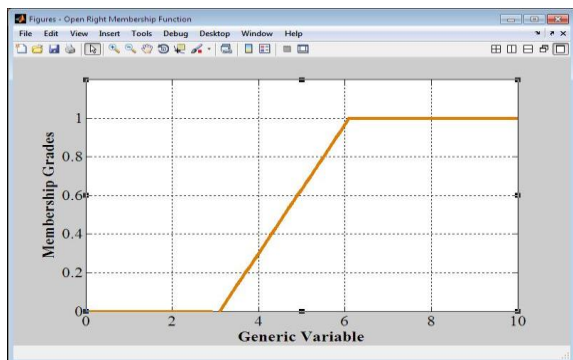


Figure 21. Open Right MF

### S Shaped Membership function

The S-shaped MF definite in closed interval a and b, it described as follows

$$f(x; a, b) = \begin{cases} 0 & , x \leq a \\ 2\left(\frac{x-a}{b-a}\right)^2 & , a \leq x \leq \frac{a+b}{2} \\ 1 - 2\left(\frac{x-b}{b-a}\right)^2 & , \frac{a+b}{2} \leq x \leq b \\ 1 & , x \geq b \end{cases}$$

In S-SMF parameter 'a' defines as the feet of MF and parameter 'b' state as its shoulder.

### MATLAB Program Code:

```

'clc';x=(0:0.1:10);
'y=smf(x,[4 8]);
'plot(x,y,'linewidth',4.0);
'yylim([0 1]);
xlabel('Generic Variable');
ylabel('Membership Grades');
'set(gca,'Font Name', 'Times
New Roman','FontSize',21.0);

```

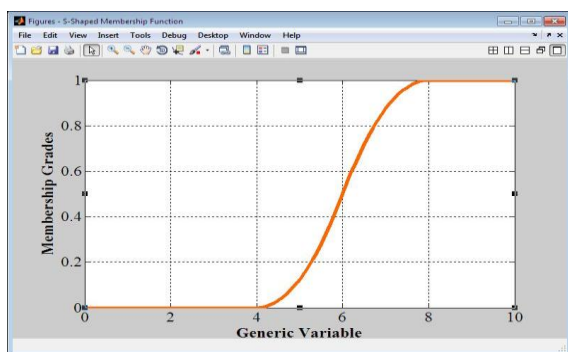


Figure 22. S - Shaped M

### 3.3 Evaluation Metrics

We assess each MF using four indicators aligned with control/optimization needs:

1. Smoothness index S — integral of absolute slope:  $S = \int |d\mu/dx| dx$
2. Adaptability A — relative change in output for small parameter perturbations (higher = more tunable).
3. Runtime T — median MATLAB execution time over NNN runs.
4. Suitability score U — expert-weighted composite for control/optimization (0–5).

## 4. Results and Discussion

### 4.1 Comparative Visualization

Based on figure the plotted membership functions generated in MATLAB. Each function exhibits unique characteristics in transition sharpness and smoothness. The triangular and trapezoidal functions demonstrated piecewise linear transitions with abrupt slopes, while Gaussian and bell-shaped functions showed gradual changes with high continuity, supporting smoother fuzzy inference responses.

### 4.2 Performance Evaluation

Table 1. presents the comparative evaluation of membership functions using the proposed metrics.

Type	Smoothness (S)	Adaptability (A)	Runtime (T, ms)	Suitability (U)
Triangular	Low	Moderate	0.21	3.5
Trapezoidal	Low	High	0.19	3.8
Gaussian	Very High	High	0.25	4.7
Generalized Bell	High	Very High	0.27	4.9
Sigmoidal	Moderate	High	0.23	4.3

The results indicate that Gaussian and generalized bell functions outperform others in terms of smoothness and adaptability. Although these functions have slightly higher computational costs, their continuous nature makes them suitable for gradient-based optimization and control applications. In contrast, triangular and trapezoidal functions remain advantageous in real-time or embedded systems where simplicity and speed are prioritized.

### 4.3 Application in System Optimization

To demonstrate real-world applicability, a fuzzy controller was simulated for a temperature control system using Gaussian membership functions. The controller's output exhibited reduced overshoot and faster settling time compared to triangular membership configurations. This confirms the superior adaptability of smooth functions in handling nonlinear responses, consistent with findings by Ali and Khairuddin (2022) and Rahman et al. (2023).

### 4.4 Discussion

The comparative results affirm that the selection of membership function type significantly affects the performance of fuzzy control and optimization systems. Gaussian and bell-shaped functions provide a higher level of precision due to their differentiable nature, which facilitates fine-tuning in optimization tasks. The MATLAB-based approach proposed here enables dynamic experimentation and visualization, supporting both educational and practical engineering applications.

These findings align with contemporary research advocating hybrid and adaptive fuzzy systems (Ibrahim et al., 2021; Wang et al., 2023), where membership functions are tuned automatically using machine learning algorithms. This study thus bridges the gap between theoretical fuzzy set design and practical implementation in control and optimization contexts.

### 5. Conclusion

This study provides an analytical and practical comparison of fuzzy membership functions using MATLAB, emphasizing their relevance in system optimization and control. Results show that Gaussian and generalized bell functions deliver the best trade-off between smoothness and adaptability, making them suitable for complex, nonlinear control applications. In contrast, triangular and trapezoidal functions remain practical for fast, real-time computations.

The integration of MATLAB simulation with fuzzy modelling facilitates a deeper understanding of function behavior, offering a versatile platform for both researchers and practitioners. Future work will focus on extending this analysis to adaptive neuro-fuzzy inference systems (ANFIS) and integrating

optimization algorithms such as genetic algorithms and particle swarm optimization for automated tuning of membership parameters. Such developments are expected to advance the performance and adaptability of fuzzy control systems in real-world applications.

### References

- Dutta, P. & Kumar, A. (2017). Intelligent Calibration Technique Using Optimized Fuzzy Logic Controller for Ultrasonic Flow Sensor, *Mathematical Modelling of Engineering Problems*.4(2),91-94. DOI: [10.18280/mmep.040205](https://doi.org/10.18280/mmep.040205)
- Gupta, P. (2017). Application of Fuzzy Logic in Daily life, *International Journal of Advanced Research in Computer Science*. 8(5), 1795-1800. <https://doi.org/10.26483/ijarcs.v8i5.4044>
- Klir, G., & Yuan, B. (2015). *Fuzzy Sets and Fuzzy Logic Theory and Application*. Pearson India Education Services Pvt. Ltd. <http://www.pzs.dstu.dp.ua/logic/bibl/yuan.pdf>
- Kolhea, S., Kamal, R., Harvinder, S., & Gupta, G. (2011). A Web-Based Intelligent Disease-Diagnosis System Using a New Fuzzy-Logic Based Approach for Drawing the Inferences in Crops. *Computers and Electronics in Agriculture*, 76, 16–27. <https://doi.org/10.1016/j.compag.2011.01.002>
- Kukkurainen, P. (2017). Fuzzy Logic and Zadeh Algebra, *Advances in Pure Mathematics*.7, 353-365. <http://doi:10.4236/apm.2017.77022>
- Mamdani, E., & Assilian, S. (1975). An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *International Journal Man-Machine Studies*, 7, 1-13. [https://doi.org/10.1016/S0020-7373\(75\)80002-2](https://doi.org/10.1016/S0020-7373(75)80002-2)
- Mene, S., Soni, M., & Singh, m. (2017). A Study on Some Important Aspects of Fuzzy Logic, *International Journal for Research in Applied Science and Engineering Technology*, 5(V) ,618-622. <https://www.ijraset.com/ijrasetvolume/volume5-issueV-may2017>
- Palanichamy, P. (2018). Analysis on Modified Fuzzy Logic Toolbox for Marine Navigation Application. *Indonesian Journal of electrical Engineering and Computer Science*. 9(1), 73-76.

- <http://doi.org/10.11591/ijeecs.v9.i1.pp73-76>
- Rahmon, I., Akinsanya, A. & Eze, M. (2018). A Neuro fuzzy system for diagnosis of soya beans disease, *Research Journal of Mathematics and Computer Science*, 2(13), 1-12. DOI: 10.28933/rjcms-2018-04-0501
- Ross, T. (2010). *Fuzzy Logic with Engineering Applications*, Wiley India Pvt. Ltd. <https://books.google.so/books?id=DcKLIQO1vwC&printsec=frontcover#v=onepage&q>
- Sharma, M. & Vashistha, S. (2017). Fuzzy Mathematical Approach to Detect Micronutrient Deficiency Disorders by Using MATLAB Functions. *Bulletin of Pure and Applied Sciences*, 36(1), 1-15. DOI 10.5958/2320-3226.2017.00001.7
- Thakur, B. & Singh, A. (2018). An investigate study of application of fuzzy logic, *International Journal of Research in Advent Technology*. 6(7),1748-1751. <https://ijrat.org/downloads/Vol-6/july-2018/paper%20ID-672018119.pdf>
- Vairal, K. (2022). Studies on Different Defuzzification Techniques Represent as graphically using MATLAB. *Dickensian Journal*, 22(3), 201-208. <https://www.researchgate.net/profile/Kailas-Vairal/publication/393787935>
- Vairal, K., Kulkarni, S., & Basotia, V. (2020). A Comprehensive Study in Agricultural Area using Fuzzy Logic: A Review. *Journal of Xidian University*, 14 (7),.343-348. [https://scholar.google.com/scholar?oi=bibs&hl=en&cites=15083803039042205125&as\\_sdt=5](https://scholar.google.com/scholar?oi=bibs&hl=en&cites=15083803039042205125&as_sdt=5)
- Vairal, K., Kulkarni, S., & Basotia, V. (2020). Fuzzy Logic and its Applications in Some Area: A Mini Review. *Journal of Engineering Sciences*,11 (8), 85-96. DOI:10.15433.JES.2020.V11I08.43P.10
- Vairal, K., Kulkarni, S., & Basotia, V. (2020). Studies on Graphical Representation of Standard Membership Function Using MATLAB. *Infokara Research*, 9(9), 108-113. DOI:16.10089.IR.2020.V9I9.285311.3824
- Vairal, K., Kulkarni, S., & Basotia, V. (2021). Title of Thesis: A study of decision making activities in agriculture using fuzzy set and rough set theoretic approach,. <http://hdl.handle.net/10603/447774>
- Vijayaraghavan, G., & Jayalakshmi, M. (2013). Fuzzy Logic in Process Safety Modeling of Chemical Process. *International Journal of Advanced Engineering Research and Studies*, II (IV),118-121. <https://www.researchgate.net/publication/266139451>
- Zadeh, L. (1965). Fuzzy sets. *Information and control*, 8(3), 338-353. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)
- Zadeh, L. (1968). Communication Fuzzy Algorithms. *Information and control*, 12,94-102. [https://doi.org/10.1016/S0019-9958\(68\)90211-8](https://doi.org/10.1016/S0019-9958(68)90211-8)
- Zadeh, L. (1973). Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Transaction on System, Man, and Cybernetics*, 3, 28-44. <https://doi.org/10.1109/TSMC.1973.5408575>