

| **Title:** Implementation of de-fuzzification methods (Center of Sum, Centre of  Gravity, Mean of Maximum). |
| --- |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Objective:** To understand de-fuzzification methods

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected Outcome of Experiment:**

**CO4 :** Apply basics of Fuzzy logic and neural networks **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

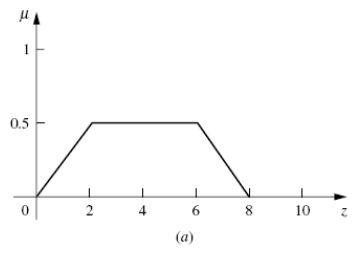
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
**Pre Lab/ Prior Concepts:**

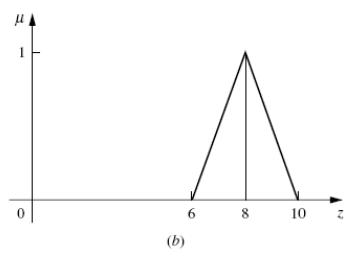
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Defuzzification :

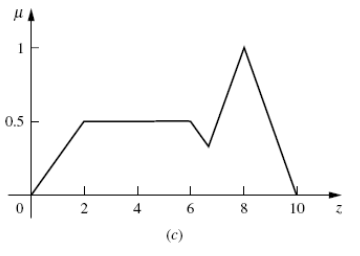
Defuzzification is the process of producing a quantifiable result in Crisp logic, given fuzzy sets and corresponding membership degrees. It is the process that maps a fuzzy set to a crisp set. It is typically needed in fuzzy control systems. These will have a number of rules that transform a number of variables into a fuzzy result, that is, the result is described in terms of membership in fuzzy sets. Defuzzification is the conversion of a fuzzy quantity to a precise quantity, just as fuzzification is the conversion of a precise quantity to a fuzzy quantity. µ

For example, **Fig (a)** shows the first part of the Fuzzy output and **Fig (b)** shows the second part of the Fuzzy output.





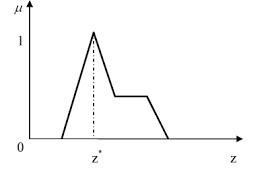
Then **Fig (c)** shows the union of the two parts (a) and (b).



Different Defuzzification methods

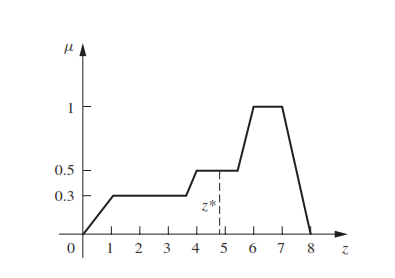
1. **Max membership method**

This method is also known as height method and is limited to peak output functions. This method is given by the algebraic expression:  
**µ**(z\*) >= **µ**(z) for all z ∊ Z.



1. **Center of gravity or centroid**

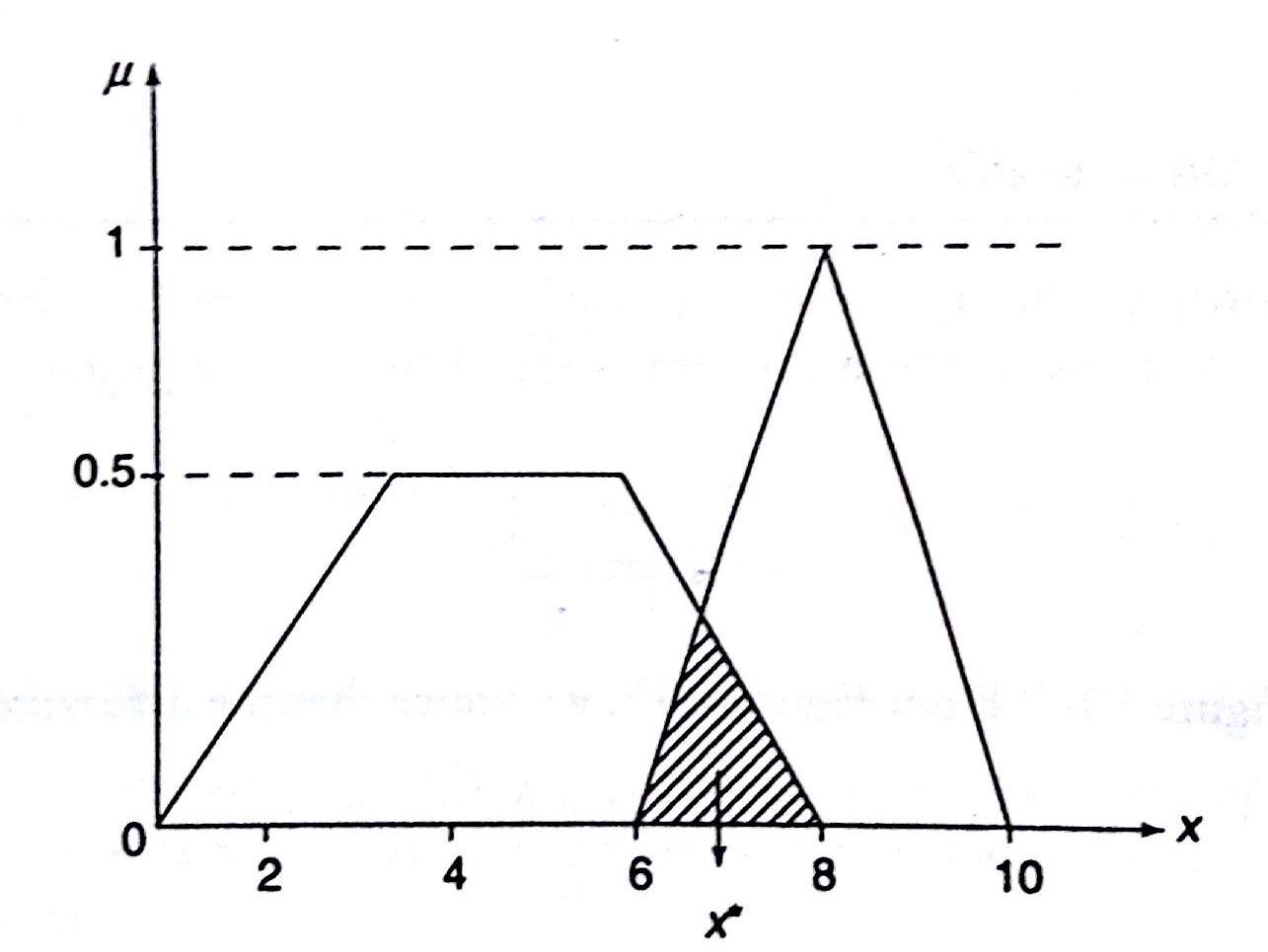
This method is also known as the centre of mass, centre of area or centre of gravity. It is the most commonly used defuzzification method. The defuzzified output z\* is given by:  
**z\* = ∫µ(z).zdz / ∫µ(z)dz**



1. **Centre of sums**

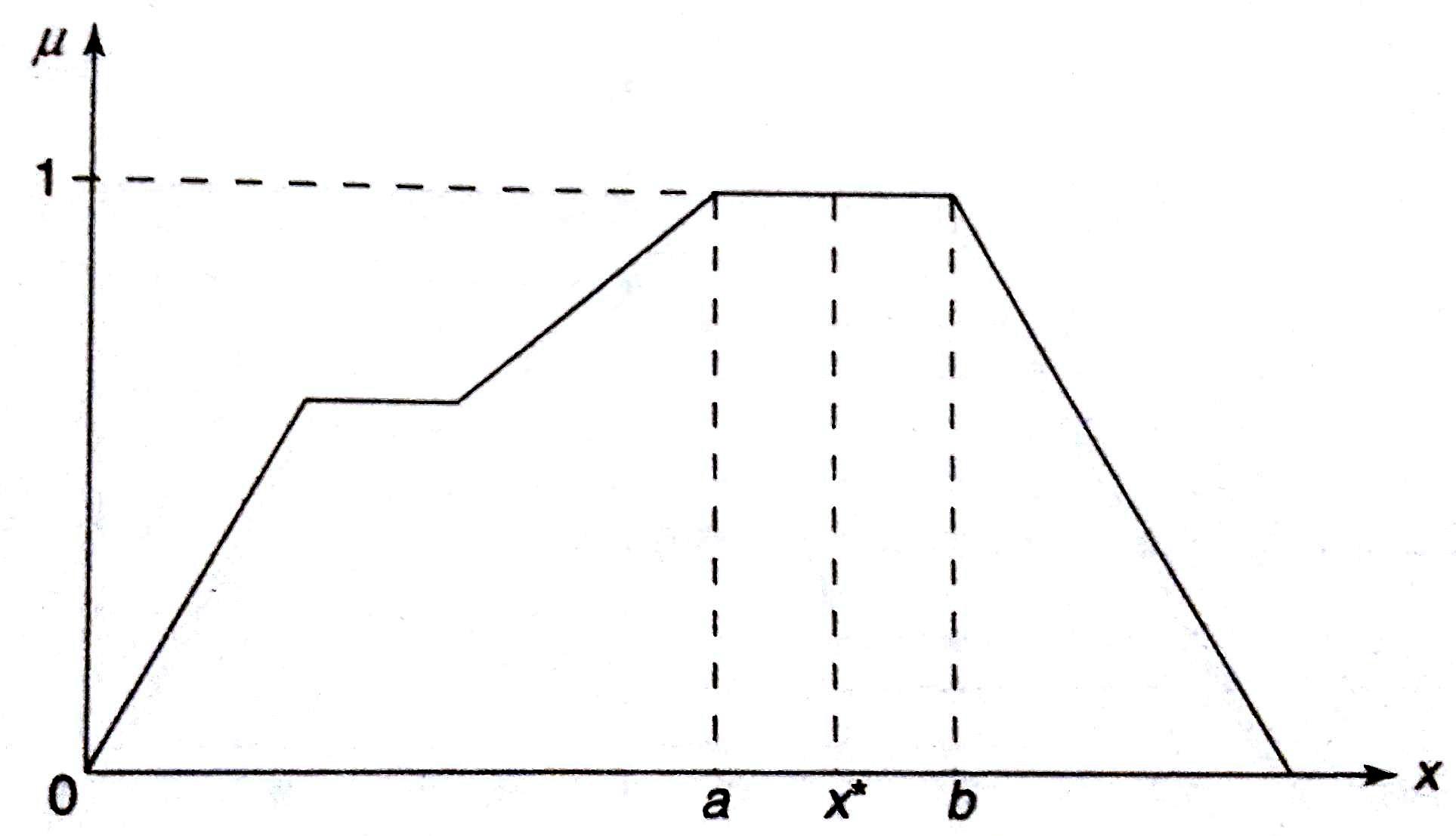
This method employs the algebraic sum of the individual fuzzy subsets instead of their union. The calculations here are very fast, but the main drawback is that the intersecting areas are added twice. The defuzzified value z\* is given by

**z\* = ∫ z\*∑µ(z).zdz / ∫ ∑µ(z)dz**



1. **Mean of maximum method**

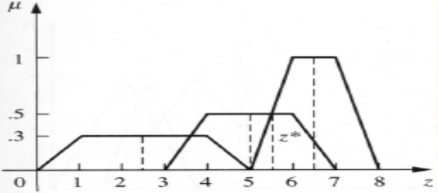
This method is also known as the middle of the maxima. This is closely related to the max-membership method, except that the locations of the maximum membership can be nonunique. The output here is given by:  
**z\* = ∑z' / n**; where z' is the maximum value of the membership function.



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Weighted average method**

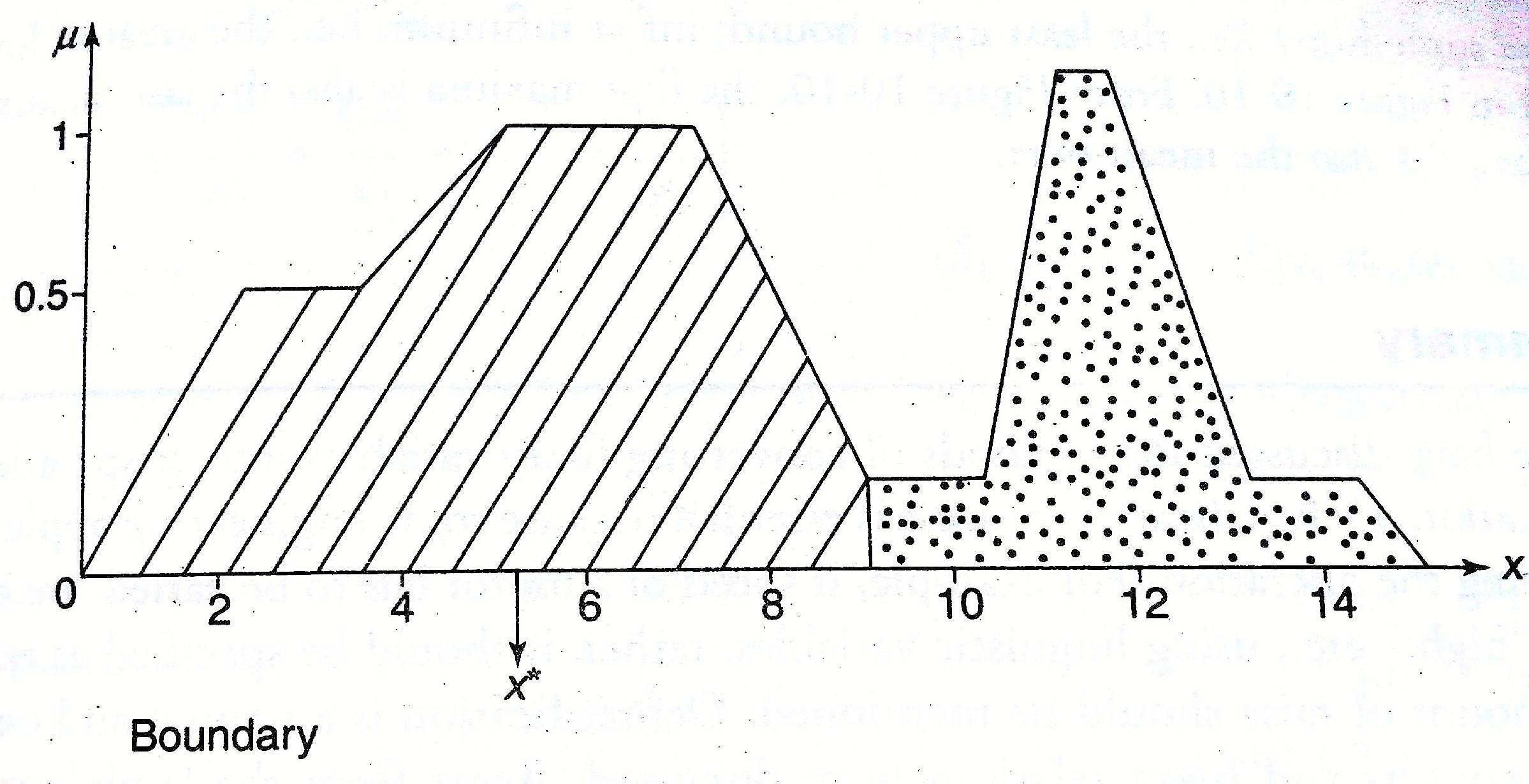
This method is valid for symmetrical output membership functions only. Each membership function is weighted by its maximum membership value. The output in the case is given by  
z\* = ∑µ(z').z' / ∑µ(z') ; where z' is the maximum value of the membership function.



1. **Centre of Largest Area**

This method can be adopted when the output of at least two convex fuzzy subsets which are not overlapping. The output, in this case, is biased towards a side of one membership function. When output fuzzy st has at least two convex regions, then the centre of gravity of the convex fuzzy subregion having the largest are is used to obtain the defuzzified value z\*. The value is given by

z\* = ∫ µc(z).zdz / ∫ ∑µc(z)dz



**Implementation Details:**

import numpy as np

import matplotlib.pyplot as plt

fuzzy\_set = {1: 0.1, 2: 0.6, 3: 0.9, 4: 0.4, 5: 0.3}

def max\_membership(fuzzy\_set):

return max(fuzzy\_set, key=fuzzy\_set.get)

def centroid(fuzzy\_set):

numerator = sum(x \* mu for x, mu in fuzzy\_set.items())

denominator = sum(mu for mu in fuzzy\_set.values())

return numerator / denominator if denominator != 0 else 0

def center\_of\_sums(fuzzy\_set):

numerator = sum(mu \* x for x, mu in fuzzy\_set.items())

denominator = sum(mu for mu in fuzzy\_set.values())

return numerator / denominator if denominator != 0 else 0

def mean\_of\_maximum(fuzzy\_set):

max\_value = max(fuzzy\_set.values())

max\_locations = [x for x, mu in fuzzy\_set.items() if mu == max\_value]

return sum(max\_locations) / len(max\_locations) if max\_locations else 0

def weighted\_average(fuzzy\_set):

numerator = sum(mu \* x for x, mu in fuzzy\_set.items())

denominator = sum(mu for mu in fuzzy\_set.values())

return numerator / denominator if denominator != 0 else 0

def center\_of\_largest\_area(fuzzy\_set):

max\_area = max(fuzzy\_set.values())

largest\_region = {x: mu for x, mu in fuzzy\_set.items() if mu == max\_area}

return centroid(largest\_region)

def plot\_fuzzy\_set(fuzzy\_set, method\_name, result):

plt.figure(figsize=(10, 5))

plt.plot(fuzzy\_set.keys(), fuzzy\_set.values(), label='Fuzzy Set', marker='o')

plt.axvline(x=result, color='r', linestyle='--', label=f'{method\_name} = {result:.2f}')

plt.title(method\_name)

plt.xlabel('Values')

plt.ylabel('Membership Degree')

plt.legend()

plt.grid()

plt.show()

methods = {

"Max Membership Method": max\_membership(fuzzy\_set),

"Centroid Method": centroid(fuzzy\_set),

"Center of Sums": center\_of\_sums(fuzzy\_set),

"Mean of Maximum": mean\_of\_maximum(fuzzy\_set),

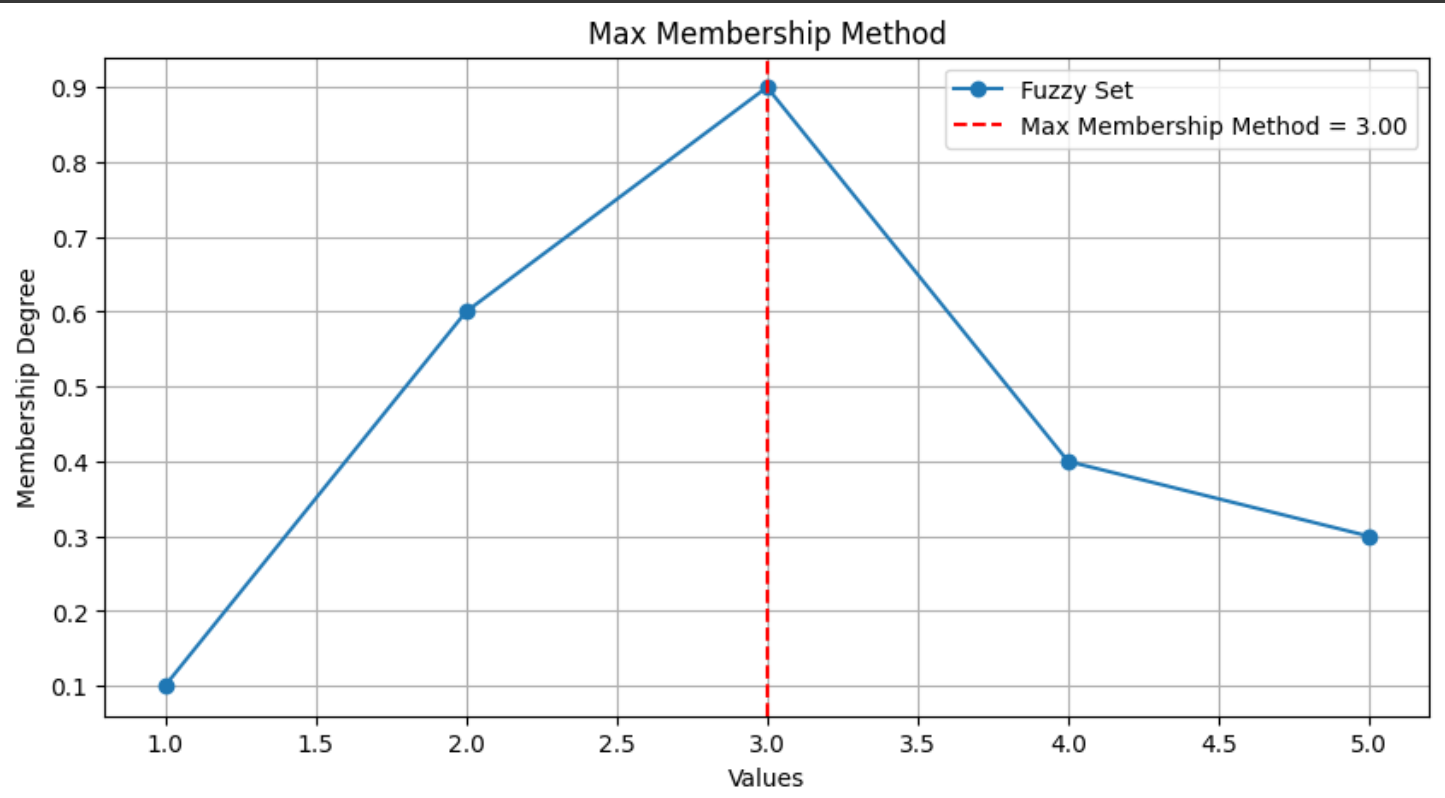
"Weighted Average": weighted\_average(fuzzy\_set),

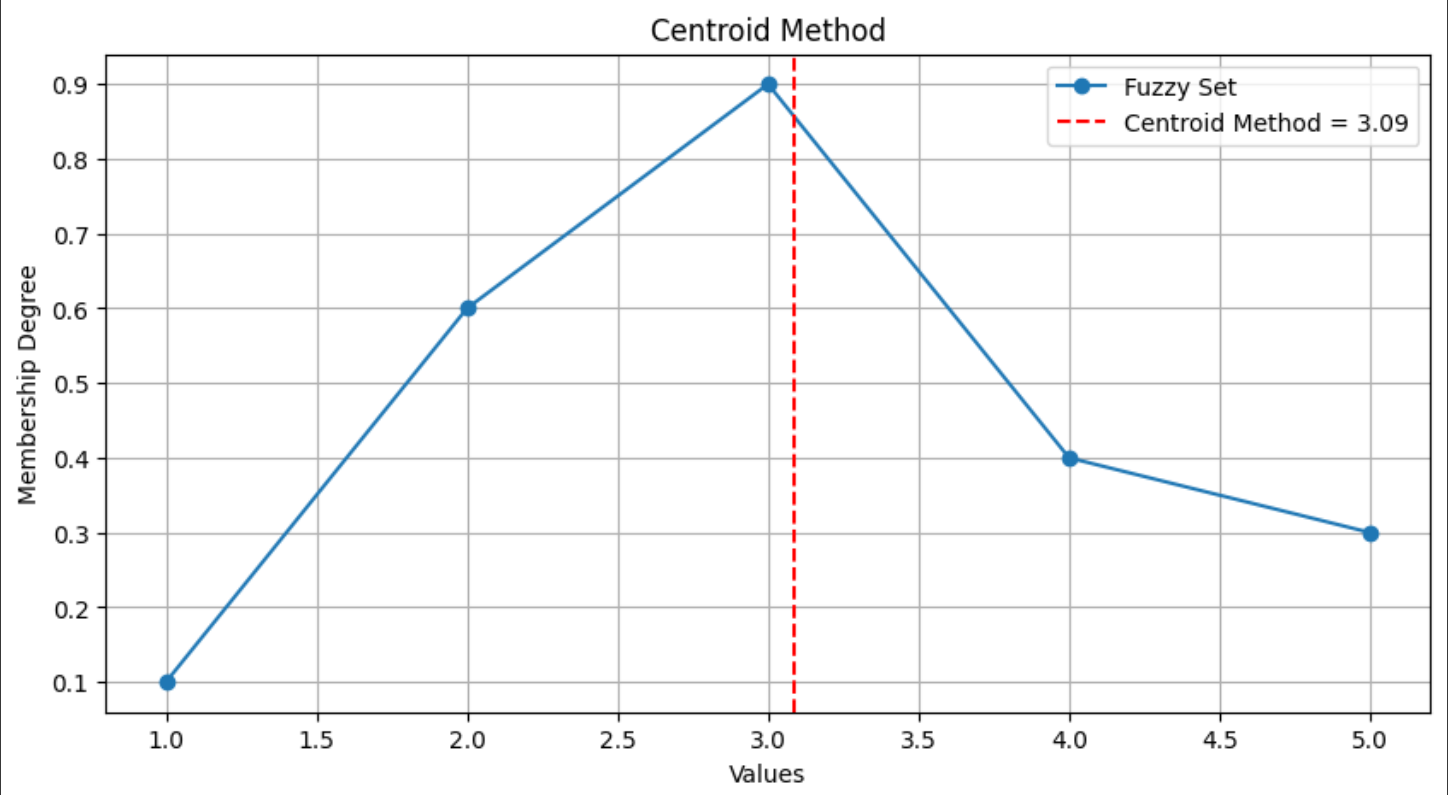
"Center of Largest Area": center\_of\_largest\_area(fuzzy\_set)

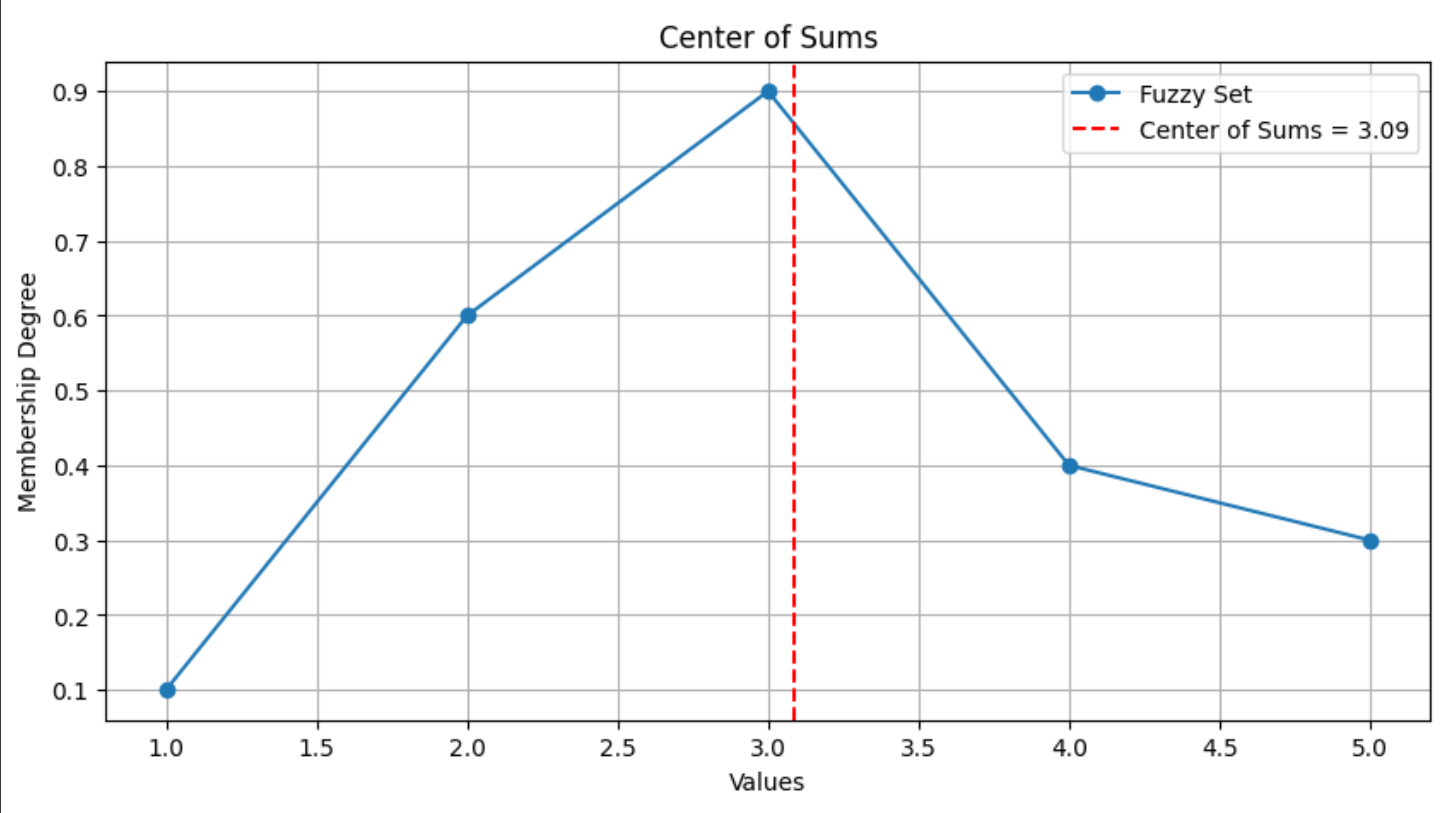
}

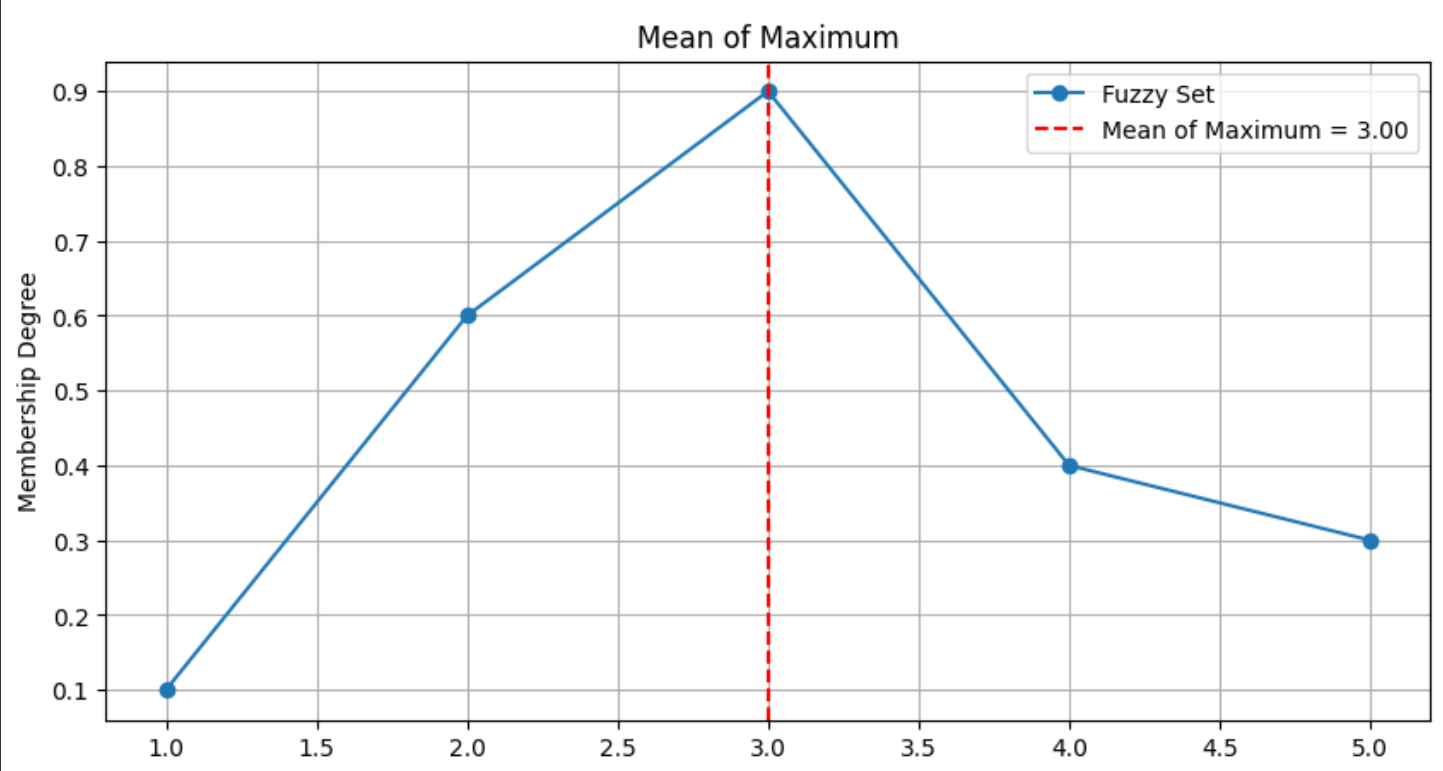
for method\_name, result in methods.items():

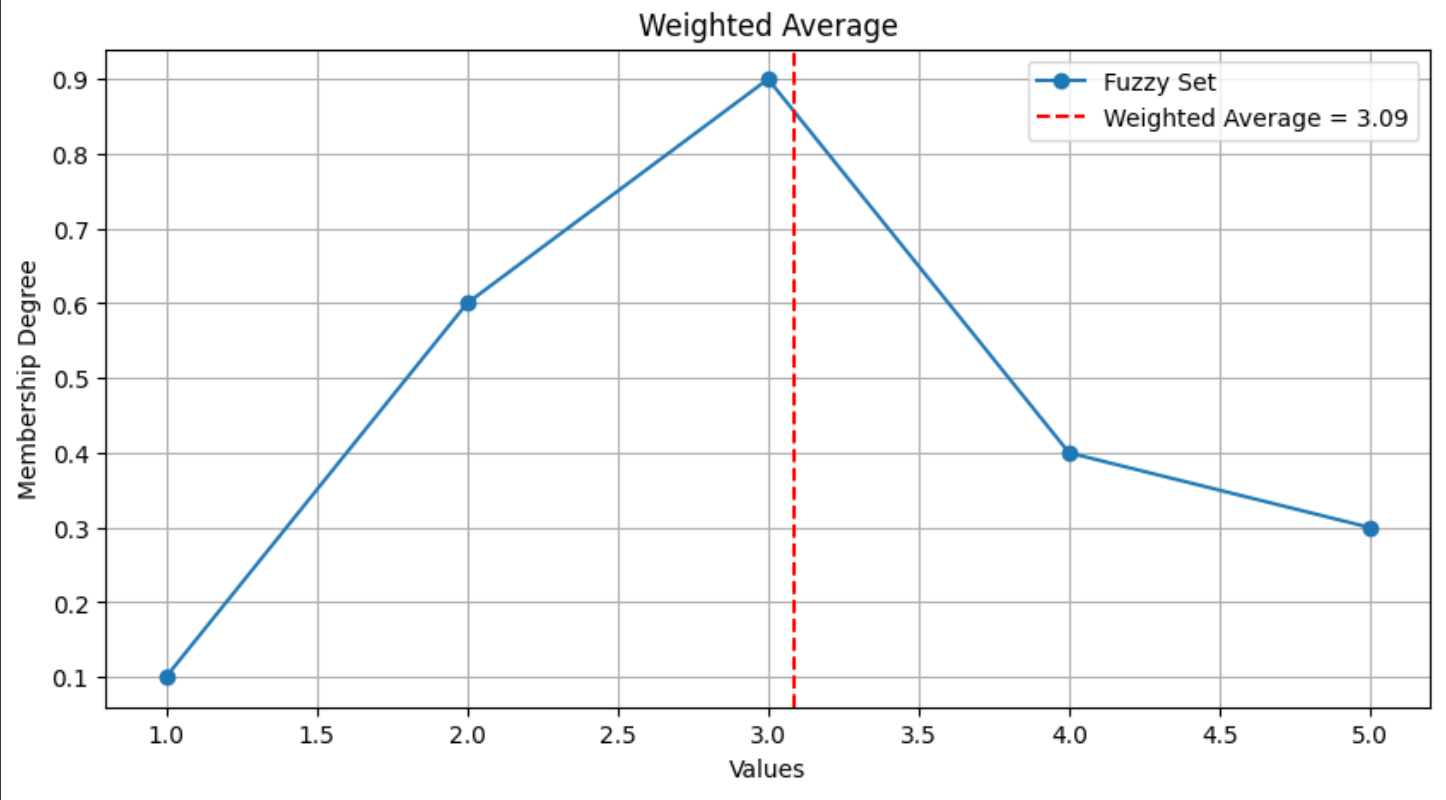
plot\_fuzzy\_set(fuzzy\_set, method\_name, result)

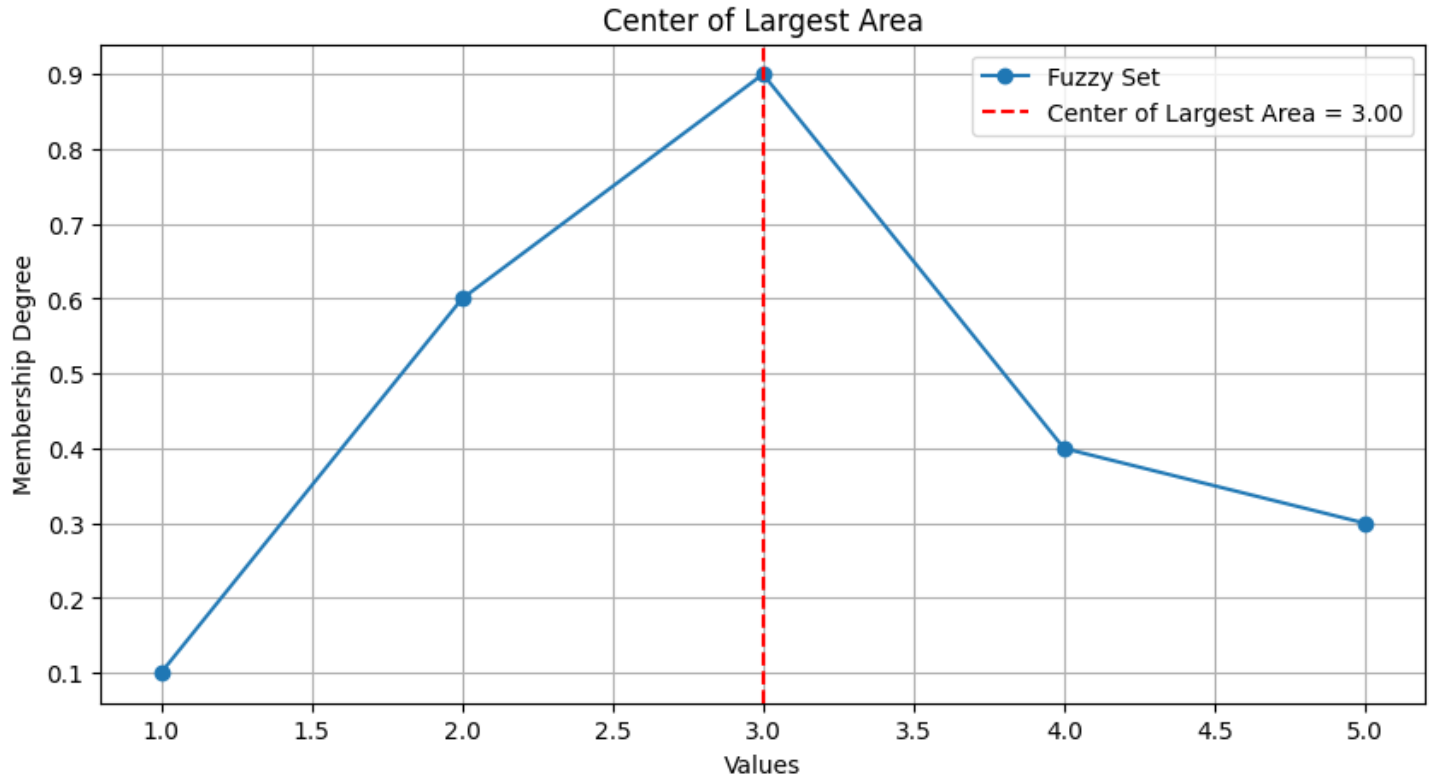












**Conclusion:** Implementation of defuzzification methods was done successfully.

**Post Lab Descriptive Questions :**

Find crisp value corresponding to the following fuzzy output sets using various de-fuzzification methods.

