ISSN: 2586-761x(Print)
ISSN: 2635-5817 (Online)
Vol. 03, No. 01, June 2020
International Journal of Advanced Social Sciences
Source: http://ictaes.org
Manuscript received: April 20, 2020; Revised: May 15, 2020 ; Accepted: May 25, 2020

Fuzzy Logic based Approach for Horizontal Data Fragmentation in Distributed Database System

Anil Verma[1], Sahaj Shakya[2]

[1]*Department of Electronics and ComputerEngineering, Institute of Engineering,Pulchowk Campus, Nepal*
[2]*Master in Computer Engineering, Neal College of Information Technology, Nepal*

anil@ioe.edu.np[1], saz.shakay@gmail.com2

Abstract

Distributed Database (DDB) has been developed to allocate a larger amount of data and information over a different sites of network. While allocating the data to present time, the data is partitioned to small fragments called fragmentation. Fragmentation is a vital part strategy to distribute the data into portions of database stored in multiple physical location and processing of distributed among multiple nodes of database. Horizontal fragmentation may improve the performance of database system but the common challenges are how the data should be fragmented. Fragmentation relying on statistical and empirical data with its precedence value is one of the effecting ways to fragment the data but implementation of precedence value along with fuzzy logic provided a scattering of load and data completely over different sites of the network.

Keywords: *Distributed Database, Fuzzy logic, Big Data Analysis*

1. Introduction

Distributed Database could be an assortment of fragment/distributed information unit that logically belongs to the identical system and spreads over an electronic network sites. The first issue of distributed information is to style the fragmentation and allocation strategy. There are several ways to distribute the data to various sites such as replication, allocation and fragmentations. Fragmentation helps to improve reliability, performance cost and security. It can be achieved by horizontal, vertical and mixed. Better Fragmentation provides better processing and distributed computing to solve computational problems with less ease of access of data. In the case of Horizontal

Corresponding Author : Anil Verma
Author's affiliation : Department of Electronics and ComputerEngineering, IoE, Pulchowk Campus, Nepal
Email: anil@ioe.edu.np
Copyright © ICT-AES

fragmentation, tuples are grouped in accordance to values of one or more field. Applications performance which is powerfully plagued by distributed database is closely impacted to the Horizontal fragmentation,

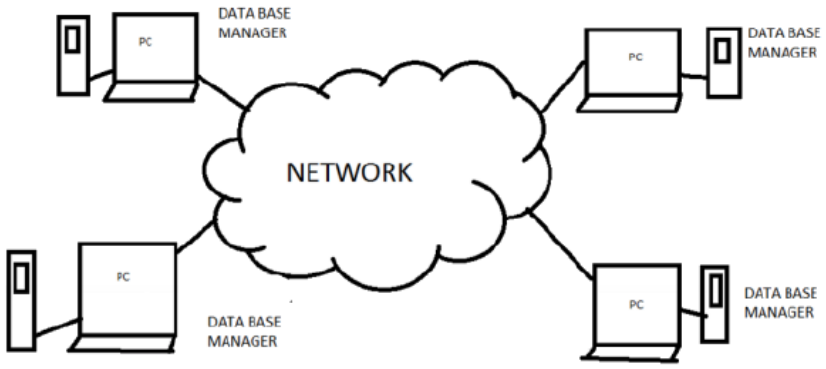


Figure 1. Distributed Processing.

Figure 1 shows how data distributed over a network helps to achieve parallel processing if it is stored on a single location maintaining transparency. But there is no proper way to determine how the data should be fragmented or how to data arrives from the site. Implementing ML algorithm: Fuzzy Logic provides a better understanding to manage the data and predict the data flow throughout the network. Fuzzy logic can be described as a method of reasoning resembling the human reasoning. Since the relation with pure mathematics are not enough to specify execution, fuzzy logic is implemented to determine the intermediate possibilities between digital values and choosing the best possible decision. There is no find tuned algorithm to determine the best predicate for the database. This resource is integrated with Professor A.B Deshmuk to develop a new kind of strategy. The main concept of this contribution are combining empirical data with strategical analysis for choosing the best predicate values.

2. Materials and Methods

2.1. MCRUDE Matrix

A MCRUDE Matrix a modification of CRUD Matrix specially designed for create, read, updated and delete of request in distributed data based on Horizontal Fragmented rows where predicates of attributes of a relation is taken as row and application of sites are taken as column.

Table 1. MCRUD Matrix.

Attribute V/ S Site	SITE 1		SITE 2		SITE 3	
	APP 1	APP 2	APP 1	APP 2	APP 1	APP 2
Attribute 1	C		R		D	
Attribute 2	R	U		U		R
Attribute 3: Site2			RUD		R	
Attribute 4: Site3		R			CRUD	R
Attribute 5: Site1	RD	CR	R		D	

2.2. ALP Table

An ALP table is a new technique of Fragmentation technique where the relation is fragmented according to the locality of precedence of attributed. The importance of attributes is provided by ALP Value with respect to the various sites.

Table 2. CRUDE Matrix with weights.

Attribute V/ S Site	SITE 1		SITE 2		SITE 3	
	APP 1	APP 2	APP 1	APP 2	APP 1	APP 2
Attribute 1	C		R		D	
Attribute 2	R	U		U		R
Attribute 3: Site2			RUD		R	
Attribute 4: Site3		R			CRUD	R
Attribute 5: Site1	CRUD	RU	R		R	

Rule:

I. Add weighted values from same sites.

II. Subtracts the weight from different sites.

Assumptions taken: C => 2; R => 1; U => 3; D => 2

Site1: (8+4)-(1)-(1) = 10

III. Repeat step II for all the attributes and construct ALP Table according t precedence weight

Table 3. ALP Table.

Attribute Name	Weights
Attribute 5: Site1	10
Attribute 4: Site3	8
Attribute 3: Site2	5

2.3. Fuzzy Logic System

In a distributed system, rel. pure mathematics isn't enough to specific execution strategy. Fuzzy technique intents on handling the probability of halfway truth, wherever reality quality might extend between fully certifiable and all false [3]. It extends the mathematically reason based on a set of rule bases to determine the intermediate possibilities of possible decisions.

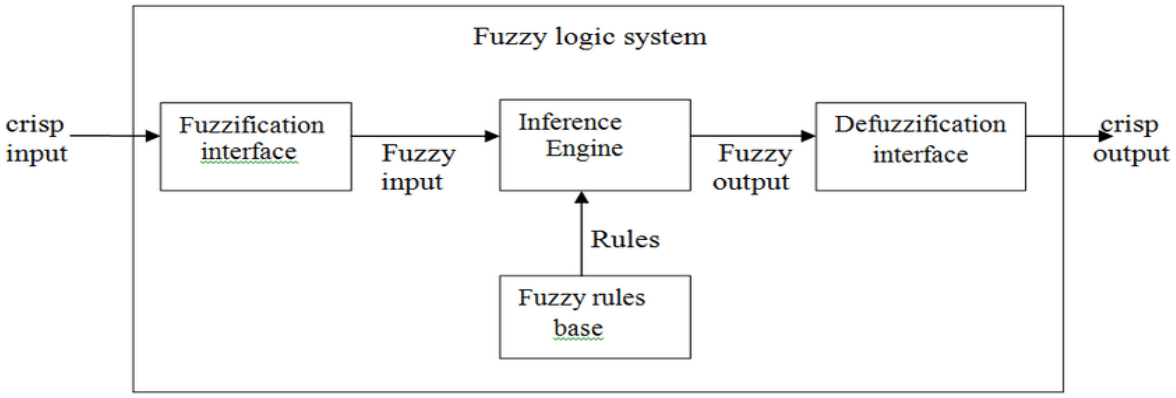


Figure 2. Fuzzy Logic Architecture.

Table 3. Fuzzy Logic System.

Application Areas	Uses	Fuzzy Logic
Automobiles	Anti-lock brakes	Use fuzzy logic to controls brakes in hazardous cases depend on car speed, acceleration, wheel speed, and acceleration
	Auto transmission	Fuzzy logic is used to control the fuel injection and ignition based on throttle setting, cooling water temperature, RPM, etc.
Electronics	Elevator Control	Use it to reduce waiting for time-based on passenger traffic
	Microwave oven	Sets lunes power and cooking strategy

2.4. Membership Functions

Each point in the Input space points are mapped to membership value between 0 and 1 shown in the graph. Input space is also referred as the universe of discourse or universal set (u), which contain all the possible elements of concern in each particular application [7]. It is denoted by "μ".

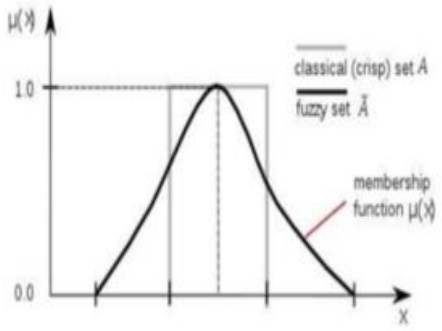


Figure 3. Membership Function.

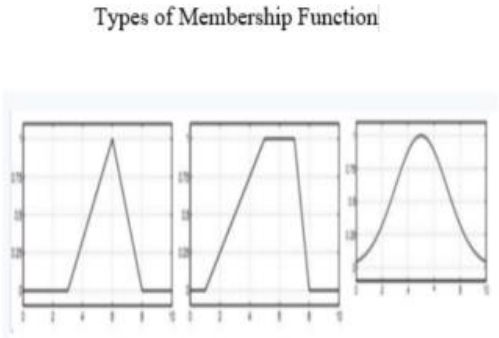


Figure 4. Triangular, Trapezoidal, Bell-shaped.

2.4. Linguistic Variable

It is a variable of values that are in words or in natural language. Each A linguistic concept is corresponded to each fuzzy set for instance: Very low, low, Average, High, Very High on which the linguistic terms and numerical values corresponds to fuzzy sets and refers these variables.

Compatibility function characterizes the mathematical meaning of a linguistic value:

c: U → [0,1], associates with each u in U its compatibility with X

For example: Height (Small, Medium, Tall)

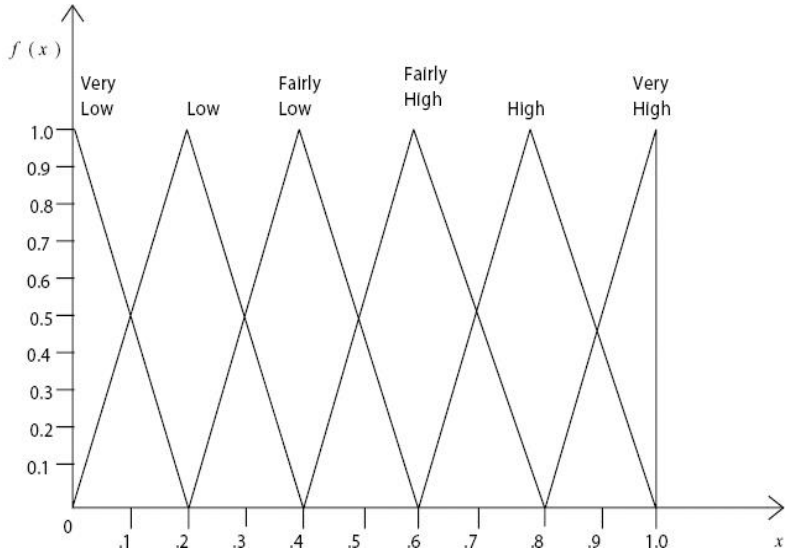


Figure 5. Linguistic Variable.

2.5. Methods

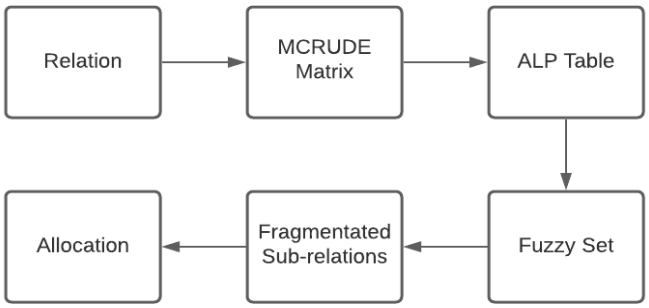


Figure 6. Block Diagram.

Relation contains various attributes describing the properties. With respect to the data distribution to various sites, attributes of the relation do not have much importance; A modified crude (MCRUD) Matrixbased on crude with the combination of ALP is assigned some weights to determine the contribution of attribute to therelations. The data sets are converted to fuzzy set to formulate the inference engine for the appropriate horizontal fragmentation. The attribute values are then fuzzified to choose the best predicate values for horizontal fragmentation for allocation of the data.

2.5.1Algorithm

Input: Total number of sites: S = {S 1 , S 2 ,… ,Sn}

Relation to be fragmented: R Modified CRUD matrix: MCRUD[R]

ALP Data: ALP[R]

Crips Data: Crips[C]

Fuzzy Data: Fuzzy [F]

Output: Fragments F = {F_1, F_2, F_3,…, F_n}

Step A: Initialize the table

Acc No	Type	CustId	Open Date	Balance	Branch
01	Saving	101	05/01/12	30000	KTM
02	Current	102	18/01/12	48000	KTM
03	Current	103	10/02/12	15900	PKR
04	Saving	104	06/03/12	37750	DRN
05	Current	105	12/03/12	50000	KTM
06	Saving	106	25/03/12	25000	PKR
07	Current	107	28/03/12	45000	DRN

Figure 7. Relation ACCOUNT.

Step B: Construction of ALP [R] from MCRUD [R] on the basis of cost functions

Cost Function

- Assign weights to CRUD
- Assumptions: Update occurs maximum so assign maximum weight to update value.
- Assumptions: Read occurs least so assign minimum weights.
- Assumptions: C=2, R=1, U=3, D=2

Site / Application / Entity,Attribute.Predicate	Site1			Site2			Site3		
	Ap 1	Ap 2	Ap 3	Ap 1	Ap 2	Ap 3	Ap 1	Ap 2	Ap 3
Accounts.AccNo<10000	C		RU						R
Accounts.AccNo>=10000		R							
Accounts.Type=Saving	CRD	RU	RUD		R				
Accounts.Type=Current		RU	R				CRUD	RU	R
. . .									
Accounts.Balance<50000	R		R			CRUD			R
Accounts.Balance>=50000		CR							
Account.Branch= KTM	CRUD	RU	CRUD			R	R		
Account.Branch=PKRur		R		CRUD	CRUD	R		R	
Account.Branch=DRNbai							CRUD	RD	CRU

Figure 8. MCRUD Matrix.

Rules:

I. Add weighted values from same sites.

II. Subtracts the weight from different sites.

- Branch => KTM (8+4+8) – (1+1)= 18.
- Branch => PKR is (8+8+1) – (1+1)= 15.
- Branch => DRN is (8+3+6) – 0 = 17.

III. Compare weights within the attributes and choose the maximum.

KTM(18) > DRN(17) > PKR(15)

IV. Calculate the total weight for the attribute (Branch).

So ALP of Branch => 18+15+17 = 50.

VI. Calculate the probability for the highest attribute.

P(KTM) = (18/50) = 0.36 => x1

VII. Do Similar for every attributes on the Table.

VIII. Compare the total attribute weight and arrange them according to their precedence value of weights.

IX.

Attribute Name	Weights
Branch	50
Type	22
Balance	10
Open Date	7
Acc No	6
CustId	6

Figure 9. ALP Table.

X. Calculate the total weight of every attributes.

Weight(Account) = 50 + 22 + 10 +7 + 6 + 6 = 101

VI. Calculate the probability of the highest attributes.

P (Branch) = 50/101 = 0.495 => x2

Step C: Fuzzy Set:

I. Rule Base

Input Fuzzy set (Antecedents):

Probability within attribute (x1)

- **Universe (crisp value): Scale 0 to 1**
- **Fuzzy set: Small, medium, large**

Probability among the whole attribute (x2)

- **Universe (crisp value): Scale 0 to 1**
- **Fuzzy set: Small, medium, large**

Output Fuzzy set (Consequents): Precedence order (x3)

- **Universe: Scale**
- **Fuzzy set: Small(A), medium(B), large(C)**

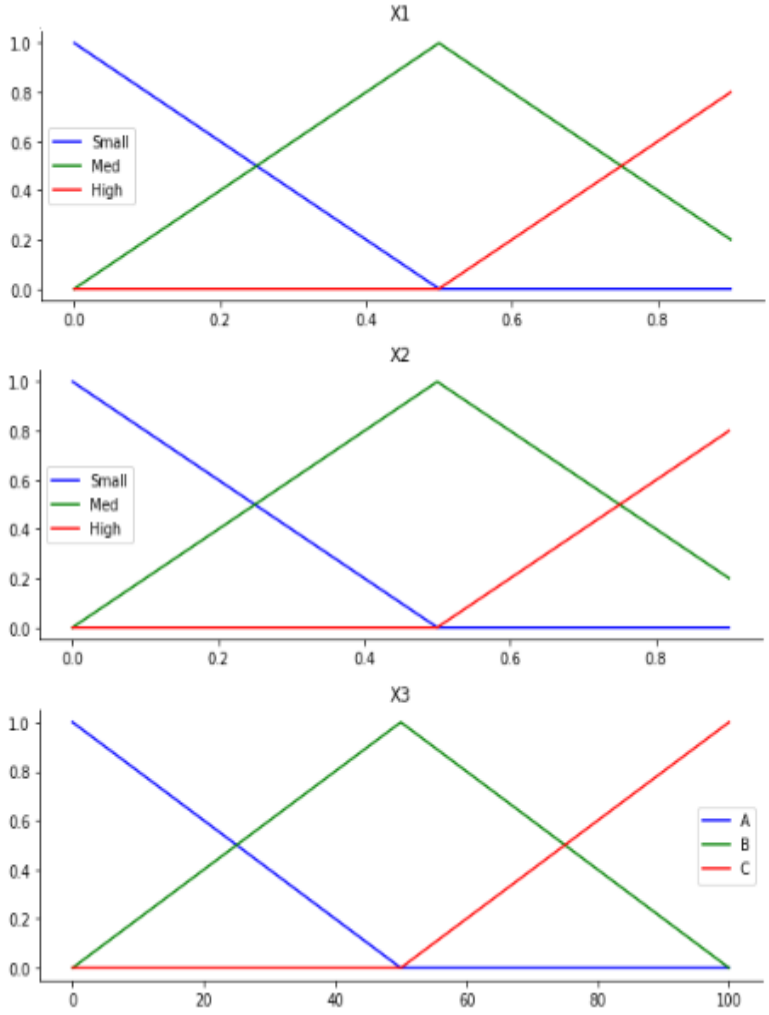


Figure 10. Rule base.

II. Rule Base

In case of common antecedence, remove the replications.

Rule R_j: if x_1 is $A_{1,j}$ and x_n is $A_{n,j}$ then Class C_j with RW_j(Rule based on Chi-FRBCS)

- If x1 is small and x2 is small then Output is Class B
- If x1 is small and x2 is medium then Output is Class B
- If x1 is small and x2 is large then Output is Class A
- If x1 is medium and x2 is small then Output is Class B
- If x1 is medium and x2 is medium then Output is Class B
- If x1 is medium and x2 is large then Output is Class A
- If x1 is large and x2 is small then Output is Class C
- If x1 is large and x2 is medium then Output is Class C
- If x1 is large and x2 is large then Output is Class C

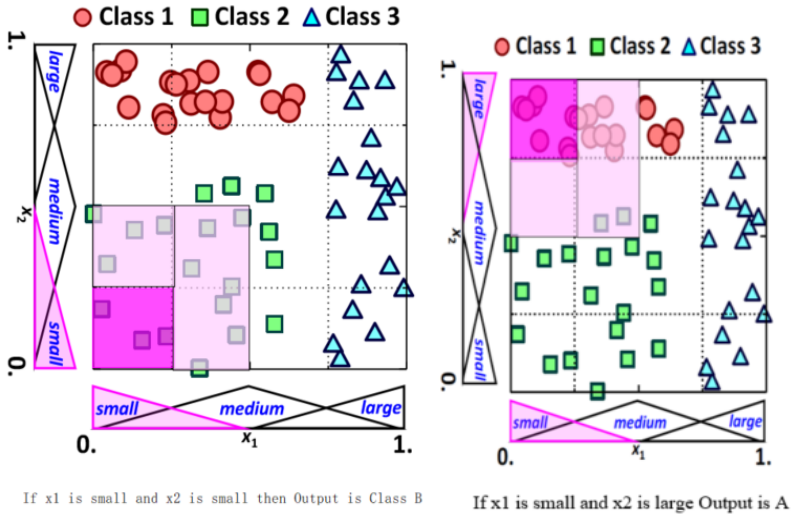


Figure 11. Rule Selection based on A, B & C Class Data Distribution.

III. Fuzzification

Fuzzy Operator is used to multiply the part of antecedents and the antecedent is resolve to a single number between 0 and 1.

Operators:

- IS: the relation x is A is quantified by the membership degree of x to the fuzzy set A.
- AND: conjunction operator, denoted ^, the most common operators are minimum and product.
- OR: disjunction operator, the most common are maximum and sum.

μ(x1) and μ(x2) = Min (μ(x1), μ(x2))

For KTM Predicate: x1(0.36) ^ x2(0.496) = Min (0.36, 0.496) = 0.36

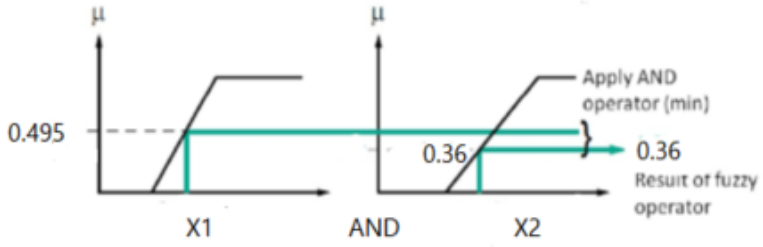


Figure 12. AND Predicate Fuzzification.

IV. Aggregate the data

The Aggregation of information is represented by membership functions where reasoning and fuzzy rule base mechanism is applied. Operators which describes mixtures of the previous ones with allowing some degree of compensation is important to calculate.With the activity of each output membership function known, all output membership functions must be combined. This is typically done using a maximum operator.

- If x1 is small and x2 is small then Output is Class B

  $\mu_{(x1)}$ = small and $\mu_{(x2)}$ = small => min(x1, x2) => Class B ➔Agg1
- Aggregation =Max(Agg1,Agg2,Agg3, …., Aggn)

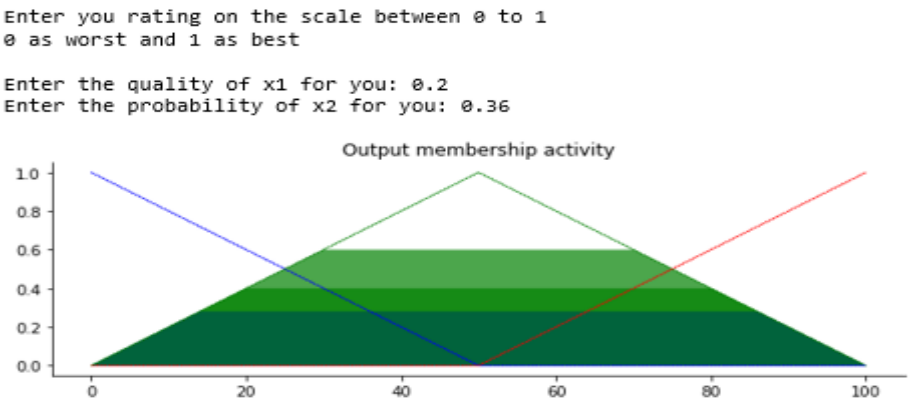


Figure 13. Aggregation.

V. Defuzzification

Defuzzification is the process of converting the fuzzy data into a crisp value. There are many process to defuuzify the data. Centroid Alogoithm is used in our case.

$$\mu_A(x)=\begin{cases} 0, & \text{if } x \le a \\ \frac{x-a}{m-a}, & \text{if } a < x \le m \\ \frac{b-x}{b-m}, & \text{if } m < x < b \\ 0, & \text{if } x \ge b \end{cases}$$

Figure 14. Centroid Algorithm .

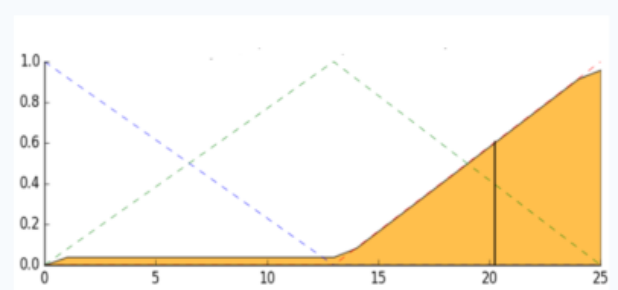


Figure 15. Defuuzified crisp value.

Step D: Rearrange P so that #P = #S based on the deffuzified data.

In the Accounts relation, attribute having highest locality precedence is chosen where and fuzzy set was created on Branch attribute.

P= {p1: KTM, p2: Branch=DRN, p3: Branch= PKR}

Step E: Fragment R using P as selection of Fuzzy set Values

- Allocate F to S

3. Results

From the experimental, probability distribution of fuzzy data is obtained as show on figure 17. The values with the highest probability is chosen as the best predicate set for horizontal fragmentation. The Fragmentation is done on the basis of higher precedence of attributes. The result shows data with no unwanted fragmentation and replication with the decrease in rate of access of data.

```
Enter 1 for Centroid, 2 for Bisector and other any number of mean:
Enter the number: 1
The Result of probability is  0.5 of your total attributes
```

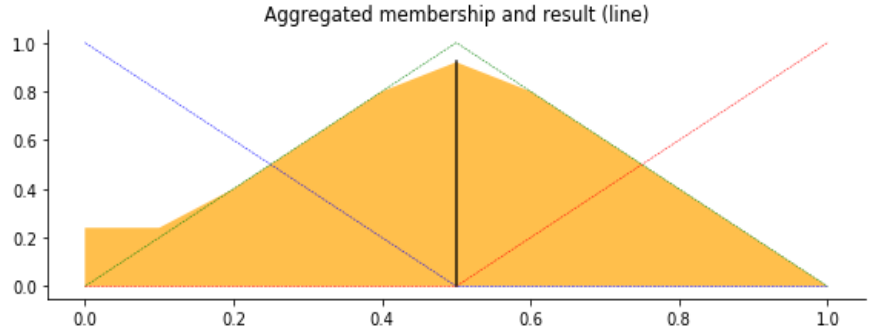


Figure 16. Aggregated Results.

X1	X2	X3
0.46	0.39	0.5
0.5	0.7	0.15
0.5	0.5	0.45
0.12	0.66	0.2

Figure 17. Order of Predicate.

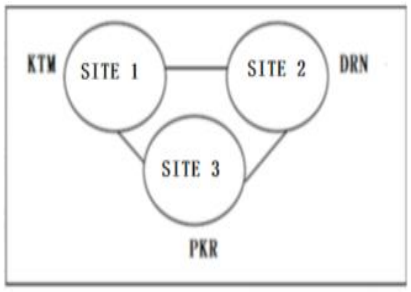


Figure 18. Order of Arrangement of attribute.

Part of Accounts Related allocated to Site 1

Acc No	Type	CustId	Open Date	Balance	Branch
01	Saving	101	05/01/12	30000	KTM
02	Current	102	18/01/12	48000	KTM
05	Current	105	12/03/12	50000	KTM

Part of Accounts Related allocated to Site 2

Acc No	Type	CustId	Open Date	Balance	Branch
04	Saving	104	06/03/12	37750	DRN
07	Current	107	28/03/12	45000	DRN

Part of Accounts Related allocated to Site 3

Acc No	Type	CustId	Open Date	Balance	Branch
03	Current	103	10/02/12	15900	PKR
06	Saving	106	25/03/12	25000	PKR

Figure 19. Order of Horizontal Fragmentation.

4. Discussions

This paper is integrated with Professor A.B Deshmuk to develop a new kind of strategy. Combining the ALP precedence and predicate set to a fuzzy set helps to predict the better rate of operation on the various sites of data. Combining the Artificial Algorithm such as fuzzy logic with the ALP strategies on development of new algorithm produces much more refine values. However this approaches has some limitation such as in case of Data with medium probabilities, this algorithm takes much processing time due to large number of distributions and is also limited only to Relational Database. This approach can be further implemented by using Decision Based Tree and also can be implemented in Distributed object Orient Database.

5. Conclusion

Horizontal fragmentation may improve the performance of database system, but the common challenges is how the data should be fragmented. The first issue of distributed information is to style the fragmentation and allocation strategy. At the initial stage of design in distributed database with having no data access statistic and query frequencies, an efficient Fragmentation and allocation at the initial stage of distributed database is extremely difficult Fragmentation relying on statistical and empirical data with its precedence value is one of the effecting way to fragment the data. Statistical Analysis on empirical data and ALP precedence with Fuzzy set is found to produce better predicate with select for horizontal fragmentation.

References

[1] Dr.A.D.Gawande, Prof. A.B.Deshmukh, Ms. P. R. Bhuyar. Horizontal Fragmentation Technique in Distributed Database, Sipna College of Engineering & Technology, Amravati, India.

[2] O. T. Timi-Johnsonand, P. O. Asagba. Query Optimization of A Distributed Database System Using Fuzzy Logic, Department of Computer ScienceUniversity of Port Harcourt, Nigeria

[3] R. Sudheer, P. Manoj Kumar, G. Srividya, M. Pavithra. A Vertical Fragmentation Strategy for Distributed Database and Fuzzy Logic, Saveetha School of Engineering, Chennai, India

[4] Bosc, P., Pivert, O.: Fuzzy queries and relational databases. In: Proceedings of the 1994 ACM symposium on applied computing, pp. 170–174. ACM (1994)

[5] Bosc, P., Pivert, O.:Sqlf:. A relational database language for fuzzy querying. IEEE Trans. Fuzzy Syst. 3(1), 1–17 (1995)

[6] Artificial Intelligence: Fuzzy Logic System. Retrived from https://www.tutorialspoint.com/artificial_ intelligence/artificial_intelligence_fuzzy_logic_systems.html

[7] Fuzzy Logic: Introduction (2019). Retrived from https://www.geeksforgeeks.org/fuzzy-logic-introduction/

[8] Asmaa Drissi, Safia Nait-Bahloul. (2018) Horizontal fragmentation for fuzzy querying databases, Distributed and Parallel Databases volume 37, pages441–468

[9] M. Elkano, M. Galar. CHI-BD (2017): A Fuzzy Rule Classification of big data, DOI: 10.1016/j.fss.2017.07.003

[10] M. Ra, “Horizontal partitioning for distributed database design,” In Advances in Database Research, World Scientific Publishing, pp. 101–120, 1993.

[11] Raipukar, A.R. And Bamnote, G.R. (2013). Query Optimization in Distributed Database, International Journal of Innovation Research in Computer And Communication Engineering, Issn (Print): 2320-9798, Vol.1, Issue 2, 422-426.

[12] D. G. Shin, and K. B. Irani, “Fragmenting relations horizontally using a knowledge based approach,” IEEE Transactions on Software Engineering (TSE), Vol. 17, No. 9, pp. 872–883, 1991.