Sinkonde, D., Bhalalusesa, R., Kalegele, K., Comai, S. and Matteucci, M.,
Tanzania J. Sci & Technol., Vol. 5(1), 2023 pp 31–50., DOI: https://doi.org/10.61538/tjst.v5i1.1361

Designing PostGIS Database System with Fuzzy Theory to Support Accessibility Tools for Urban Pedestrians

Daniel Sinkonde[1*], Rogers Bhalalusesa[2], Khamisi Kalegele[2], Sara Comai[3], Matteo Matteucci[3]

Abstract
The combination of fuzzy logic and crowdsourcing can be a powerful tool for generating geospatial data for pedestrians with mobility challenges in urban areas. Although potentially useful, information about the accessibility of paths that is generated through crowdsourcing is susceptible to a high degree of imprecision. Spatial data management is required for such systems, which supports the management of uncertain data. Fuzzy theory allows us to model ambiguous information. To fill this gap, an improved method based on a fuzzy relational PostGIS database (FPostGIS) is proposed. The method includes extensions to represent imprecise data within an entity-relationship (ER) data model specifically tailored for path accessibility, and a set of steps for the derivation of FPostGIS from this extended ER model. According to the case study, this methodology has been applied in the design and development of decision support application within the Maps for Easy Paths (MEP) project. This application stores and retrieves accessibility information about a particular path and allows performing spatial operations and analysis inside the database.

***Keywords**: Ambiguous information, database design, FuzzyMEP, Imprecise and uncertain information, PostGIS, rule modeling.*

[1] Department of Electronic and Telecommunication Engineering (ETE)
Mbeya University of Science and Technology (MUST), Tanzania. [*]*Corresponding author: dsinkonde2013@gmail.com*

[2] Department of Information and Communications Technology (ICT). The Open University of Tanzania, Dar es Salaam, Tanzania

[3] Dipartmento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano 20133 Milano, Italy

INTRODUCTION

Numerous real-world systems and applications, like those that use crowdsourcing, must deal with ambiguous and contradictory information, vagueness, uncertainty, and imprecision in data. The application of fuzzy theory in combination with crowdsourcing finds its application in various contexts. It aids in selecting the most suitable bank for arranging a mortgage, evaluating of client credibility, choosing an insurance company, purchasing a property, selecting a car, ajob, etc. These are the first large groups of applications for decision support. The second group of applications concerns control systems. For example, fuzzy regulators could be used for checking a valve in mechanical engineering and for releasing only the right amount of steam, which is necessary for the correct operation of the device. Fuzzy regulators are used in much smaller devices such as digital cameras, washing machines, controlling mechanism of cars, and the like for controlling many variables, ranging from the correct photographic exposure to the setting of the time needed to wash properly specific clothes in a washing machine (Bezdek, 2011). There are many examples of successful application of fuzzy theory (Bojadziev, 2007).

The applications of database technology involving fuzzy theory in filtering information and assisting in decision-making include manipulating uncertain and imprecise information to support navigation (Chen et al., 2012). Navigation systems provide spatial data which can be retrieved by users in order to make decisions regarding the best path to follow in a given moment and under given circumstances. Such databases require information that is created dynamically. For such database systems, information management components that support managing this ambiguous data are necessary. Fuzzy theory allows us to model ambiguous information. Fuzziness has garnered a lot of attention in relational database systems (RDBs), but little has been done to model fuzziness in conceptual data models for PostGIS Database Systems. To fill this gap, the researchers have proposed a design methodology for developing fuzzy RDBs (Chaudhry et al., 1999). This methodology, based on the Entity-Relationship (ER) design methodology of De Sousa et al., 2018, describes a sequence of steps to implement a fuzzy RDB. However, there seems to be no previous work in developing the ER design methodology FPostGIS for the accessibility of a path for people with mobility challenges. Then, the authors propose a generic data model that employs the ER data model to describe fuzzy rules. The syntax chosen for the fuzzy rules allows expression of explicit data against a consequent in addition to the traditional data for a consequent. This implies that the fired rules can have conflicting data. New techniques are proposed for making decisions on these rules so as to allow decision making on contradictory information.

In Section 2 of this paper, we review the state of the art in fuzzy theory and fuzzy database modeling. Section 3discusses the fuzzy association rule ER design methodology. Section 4 contains a description of the decision-making process and the fuzzy rule firing mechanism utilized by MEP in the PostGIS Database System. This study is concluded in Section 5 with a summary of the findings and an outline of future research.

State of the Art of Fuzzy Theory

Fuzzy logic is a convenient tool for handling imprecise and uncertain data in automatic decision-making systems, asreported by Edward et al., 2009. For example, Zadeh, Lotfi A. (2015) describes many applications in the areas of information sciences and control systems. According to Purian et al., 2013 using fuzzy logic is shown to be a very promising methodology for modeling traffic and path planning; mobile robots are finding a free way without encountering barriers in different environments targeting to reach to the destination. Whilst these and others, such studies have shown a greater consensus has grown around the peculiar idea of using Fuzzy theory to handle imprecise information of exemplary article by Medina et al., (1994) and hence to bring models closer to the real application. Modeling the real world using fuzzy logic is an interesting approach for the quality and condition of the accessibility of a path.

In a review work of Nguyen et al., (2012), Fuzzy theory has been used for control applications, but to our knowledge fuzzy databases have not been previously utilizedfor predicting the condition and the quality of the accessibility of a path.

Maps for Easy Paths (MEP)

The MEP (Maps for Easy Paths) (http://mep5x1000.wix.com/mepapp) is an ongoing project, proposing a set of tools and mobile apps for the enrichment of geographical maps with information about the accessibility of urban pedestrian pathways for people with mobility challenges. They range from users with manual or electric wheelchair, to the elderly with/without mechanical support, to people in temporary situations of reduced mobility by providing with information about accessible routes. It collects motion data from sensors commonly available in mobile devices and reconstructs the travelled path (Comai et al., 2017). The underlying idea is that a path that can be travelled by a person with motor disabilities can be considered accessible also for other persons having the some (or a lower) type of disability.

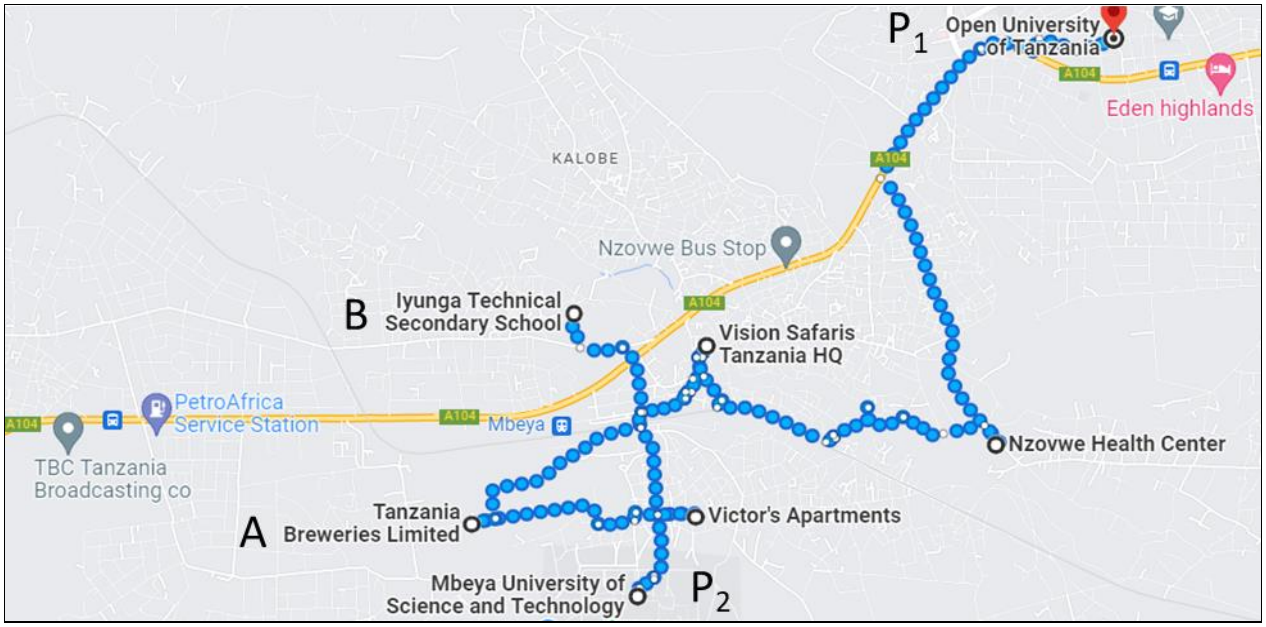


Figure 6. Describing the part of the accessibility of the path which has been travelled by a user and the other part not
Source: Own processing

Introduction to the problem

Fig. 1, shows some paths connecting some points of interest (POIs): points *A* (Tanzania Breweries Limited), *B* (Iyunga Technical Secondary School), P_1(Open University of Tanzania) and P_2(Mbeya University of Science and Technology). The paths in the figure have been travelled and mapped by users with some mobility problems. In particular, Paul follows a routine where he traverses the pathAP_1 to reach his university office every workday. Using his smartphone, he can easily map this path.

Imagine now that Peter has a smartphone with our MEP application as in Figure 1 and does not know anything about the accessibility of the streets of this AP_2 . The data collected implicitly are uploaded on the MEP server for further processing, which will end up in the construction of the path taken by the user. Once Paul reaches the endingpoint of the path AP_1, he can rate his path, for example as a “medium” accessible path. Path AP_1 is therefore associated with metadata ‘L = Level of accessibility = 2′ on the MEP server. In the same way, the other paths in the map will be associated with the profile of the users who collected the data and with their ratings. Imagine now that Peter, a second person with similar mobility problems as Paul, wants to reach the target place P_2 in Fig. 1, starting from point *A*. He can connect to MEP app and query for the path.Since Paul has partially passed that path and the information is being stored, and he rated the path as medium (or L = 2), the MEP app can retrieve the available information for all the pathsand predict the condition of the accessibility of the path AP_2 for Peter using a fuzzy logic model approach.

In this case, the path AP_1 is a route frequently taken by Paul (having similar mobility problems), the path AP_2 has been partially taken by him, while the other part of the route has never been taken before by any user or has been taken by users with different (possibly lower) mobility problems. Considering only the two paths of Paul and Peter, we have:a sequence of coordinates forAP_1 and a sequence of coordinates for AP_2. We can define the partial intersection of the two paths with the following definitions.

Subset: Path AP_2 is said to be a subset of path AP_1 if and only if $\exists x, x \in AP_2 \Rightarrow x \in AP_1$.

Proper Subset: Path AP_2 is said to be a proper subset of path AP_1 if and only if $\forall x, x \in AP_2 \Rightarrow x \in AP_1$. In this case, we write $AP_2 \subset AP_1$. The empty set that contains no members is denoted by Ø. Two paths are disjoint if they do not have any elements in common, that means, if

$$AP_1 \cap AP_2 = \emptyset \qquad (1)$$

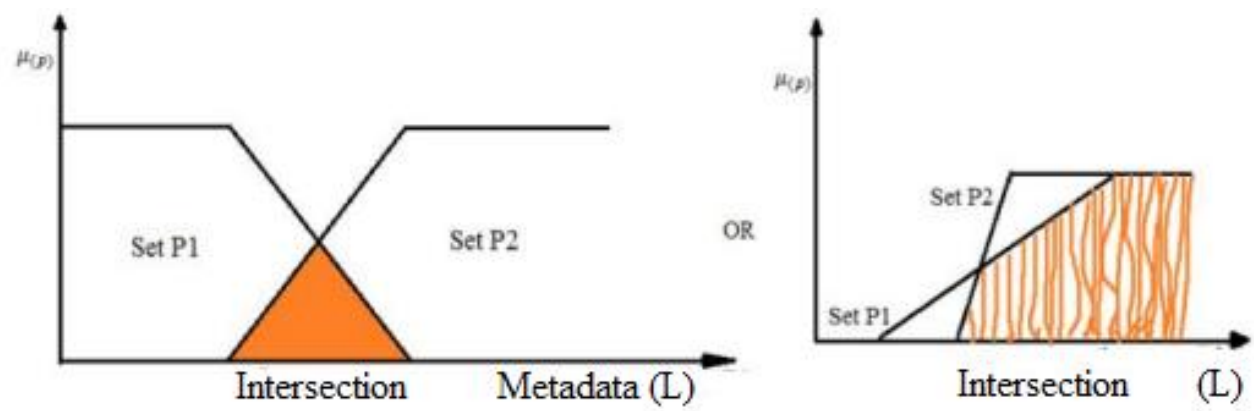
In traditional set theory, a set is defined as a collection of distinct elements. For each element, it either belongs to the set (membership degree = 1) or doesn't belong to the set (membership degree = 0). This is known as a crisp set, where membership is sharply defined.

Fuzzy set theory generalizes this concept by allowing membership degrees to be any value between 0 and 1. This means that elements can belong to a set with varying degrees of membership. In the context of fuzzy sets, an element's membership in a fuzzy set is described using a membership function. For the fuzzy set P_1, the degree to which an element A belongs to P_1 is denoted as $\mu_{P_1}(A)$. This value is a real number in the interval (0, 1). When $\mu_{p1}(A) = 0$, A does not belong to P_1 at all, and when $\mu_{p1}(A) = 1$, A fully belongs to P_1. Therefore, the fuzzy set $P1$ consists of ordered pairs [A, $\mu_{p1}(A)$] for all elements A that belong to the fuzzy set P_1

$$P_1 = [A,\ \mu_{P_1}(A)] : A \in P_1 \qquad (2)$$

One approach to define fuzzy subsets of the intersection set is shown in the following figure:

Figure 7. Fuzzy Intersection



(***Source*** *: Sinkonde, et al, p.4. 2017)*

To compute membership values with fuzzy intersection, the minimum is used:

$$\mu P_1(L) \cap \mu P_2(L) = Min\,[\mu P_1(L), \mu P_2(L)] \quad (3)$$

Table 1A. Fuzzy linguistic variables

Linguistic Variables Label	*Barriers*	*Comment*	*Pavement*
Linguistic Values	-Low -Medium -High	-Low -Medium -High	-Narrow -Dark no light -Wide
Metadata		1- 4	
Membership Function	-µLow -µMedium -µHigh	-µLow -µMedium -µHigh	-µLow -µMedium -µHigh

Table 1B. Frequency of Special Characters Membership Values for Barriers

Linguistic labels	Criticality rate (L)			
Variables	1	2	3	4
High (H)	0	0	0	1
Medium (M)	0	0	0.7	0.4
Low (L)	0	0.4	0.3	0
Very Low (VL)	0.3	0.2	0	0

Source: Own processing

Fuzzy PostGIS Relational database

The Fuzzy Relational Database Model (called FPostGIS) extends a relational model by incorporating concepts from fuzzy set theory, thus addressing the lack of precision in quantitative data. There are five examples of unreliable information generated through crowdsourcing: contradictory information regarding one issue, imprecise information, vague information, uncertain information, and ambiguous information. Then, first, there is the imprecision in the degree of membership of a tuple in a relation, and second, there is the imprecision in a data value [18]. The use of dynamic database for the control of MEP is relatively new [9] and has made it widely applicable, flexible, and portable.

Fuzzy Relation: Let Pbe the intersection of n discourses $P_1, P_2, \dots, P_n$, and its Cartesian product. Then, an $n-ary$ fuzzy relation$r$ in P is a relation which is characterized by an −variety membership function ranging over$P$, so that $\mu_r \rightarrow [0,1]$. The fuzzy relation$r$ tuple can be stated as follows:

$$t_j = \langle P_{j_1}, P_{j_2}, \dots, P_{j_n}, \mu_r(P_{j_1}, P_{j_2}, \dots . P_{j_n})\rangle$$

with $P_{j_1} \in P_1, \dots, P_{j_n} \in P_n$ (4)

Example: Consider the fuzzy relation model ending path prediction shown in Table 2. This relation has 5 attributes, such as Object ID, Metadata, Direction, Starting Point or (Starting Name) and representing the degree of certainty that the Object IDa_1 is in the State a_2, and the composite key$a_1 a_2$.

Table 2. Fuzzy Relational Model Database Ending Path Prediction

Actual Variables				Prediction state		
ObjectID	Metadata	Starting Point	Path Predicted point	State Estimation	Total Ending Path prediction	μ
415	1	75790	38152	+0.32	38152.34	0.1
419	1	75793	38156	-1.13	38155.13	0.3
422	2	75796	38159	-2.31	38157.31	0.7
432	1	75806	38169	-2.27	38167.27	0.4
435	1	75809	38172	-2.07	38170.07	1.0
436	1	75810	38173	-2.00	38171.00	0.6

MATERIALS AND METHODS

Inthissection, we propose to extend the ER data model to represent fuzziness. Then, we describe the design methodology for implementing fuzzy relational databasesfrom fuzzy conceptual data definition like Thalheim, B. (2013).

Traditional ER model to aFuzzy ER model

In this critical part of the conceptual data, the model is designed, starting with investigating the issue of designing methods which are twofold, due to separation between the construct, the common ER model and attach 'f' to the entities and relationships that are fuzzy. In addition, the design methodology for fuzzy relational databases is an extension of the design methodologies for crisp relational databases (Fahrner, C., &Vossen, G. 1995).

Common ER data model

The Entity-Relationship (ER) data model developed by Thalheim, B. (2013) is one of the paradigms that are most frequently used for the conceptual data modeling step of the database design process. The work of Teorey et al. (1986) describes a design methodology for implementing relational databases from an ER schema. The steps could be as follows:

- **Method 1: Use ER to model the application domain requirements:** The data requirements are analyzed and modeled using an ER diagram. The ER diagram in Figure 3 shows how the basic concepts of ER modeling are expressed.

- **Method 2: Transformation ER to model to relational tables:** Building relationships is a crucial step that captures the associations or interactions between entities. Relationships define how entities are connected and can be 1-to-1, 1-to-n, or n-to-n. The cardinality and participation constraints are then assigned to specify the number of occurrences and the participation requirements of entities in relationships. Cardinality determines the number of instances one entity can be associated with another, while participation indicates whether participation in a relationship is mandatory or optional.

- **Method 3: Normalization of the relations:** Normalize all relations by following three steps: the first normal form (1NF), the second normal form (2NF), and the third normal form (3NF).

- **Method 4: Validation and iteration:** The designed data model is then validated by examining its accuracy, consistency, and adherence to the requirements. Iterative refinement may be necessary based on feedback and additional requirements. Validation ensures that the data model represents scenarios around the world.

- **Method 5: Documentation of the design:** The document provides a comprehensive overview of the data model's entities, attributes, relationships, constraints, and other relevant details of the data model.

The ER Methodology with a Fuzzy Extension

This section describes the ER process design and extensions of the ER data model to accommodate fuzzy data. The literature normally expects the entities' keys to be crisp or non-fuzzy. This section describes the ER data model extensions and ER design methods for handling uncertain data.

The ER Fuzzy Extension

This section explains the extensions to the ER data model and the ER design methodology to cope with fuzzy data.

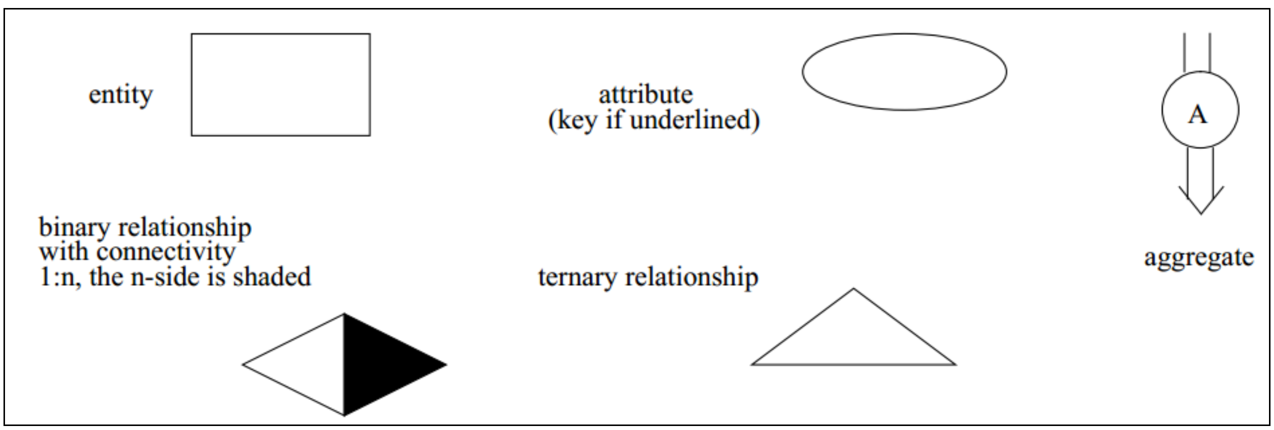


Figure 3: Basics of ER Modeling

Fuzzy Comparison Function: Fuzzy extensions of ER (Entity-Relationship) data modeling design procedures include incorporating fuzzy logic concepts into the traditional ER modeling techniques. The ER data model for the MEP project represents the most comprehensive set of tools and an innovative solution for the enrichment of geographic maps with information about the accessibility of urban areas for people with mobility challenges. As a result, a comparison function has been constructed for each entity that is involved in a fuzzy match for the relevant comparison operator. With the letters "fm_{θ}," which stand for the particular fuzzy match's comparison functions, where θ is the comparison function for the particular fuzzy match.

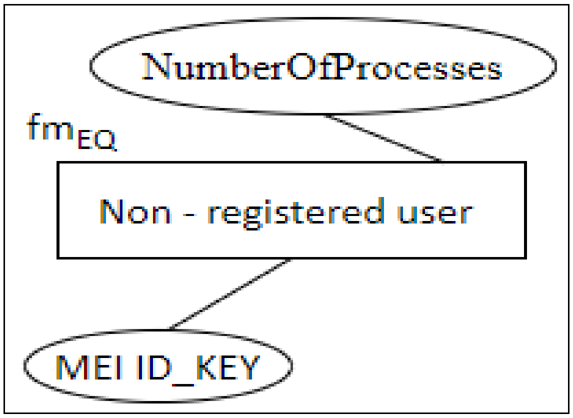


Figure 4: ER Representation of the Entity Non – Registered User

DBFuzzifier: In some circumstances, it is more useful to consider a crisp entity with one or more attributes fuzzified, rather than considering the entity in its original crisp form. In these situations, we suggest that fuzzifying the relevant EN attributes in order to define a new entity, EN^F, on top of EN.

Example: Consider a FuzzyMEP database with two entities, an entity Registered user (disability and active citizen), with 2 attributes no requirement and step_free, with the domain {low, medium, high, very-high}, and the entity Non - registered user which can visualize all information about the accessibility of the paths on their smartphones / tablet / pcwith the attributes EquipmentNum (meiid_key) and NumberOfProcesses with the domain {0,..., 1000000}. A join can be executed between the entities registered user and non-registered user, based on the attributes step_free and NumberOfProcesses, if we can define a match between the attribute values of step_free and NumberOfProcesses. One way of carrying out this translation is to partition the attribute NumberOfProcesses in the entity non-registered user into the crisp sets low, medium, high, very-high. However, a "better" translation employs fuzzy theory, and thus we define a new entity $\text{Non} - \text{registered user}^F$ on top of the entity non-registered user by mapping each instance of non-registered user into the fuzzy sets low, medium, high and very-high based on the attribute NumberOfProcesses.

We adapt the fuzzification operator defined by Lee and Chuen-Chien (1990) for this purpose. This operator has the effect of transforming crisp data into fuzzy data and is defined by:
$\mathrm{x} = \text{fuzzifier}(x_0)$ where x_0 is a crisp input value from a process; x is a fuzzy set.

Note the definition of the fuzzifier function depends upon the application requirements. In particular, this definition is determined by the universe which are fuzzifying to, which in turn determines the mapping of the elements of x_0 to various fuzzy sets.

The constrain the fuzzifier, now called the "**DBFuzzifier**", so that it can be applied to derive a fuzzy relation. This operator takes as its input parameters an entity and a fuzzifier defined on an attribute of this entity and maps this entity to a fuzzified entity.

Definition: $\mathrm{EN}^F =$ DBFuzzifier $(EN,\ fuzzifier(B_0))$, where EN is an entity with attributes $< K, \mathrm{A}_1, \dots, \mathrm{A}_\mathrm{L}, B_0 >$, with key K, non-key attributes A_i for i= 1, ..., L, and B_0 the non-key attribute to be fuzzified. For any $Z =< K, B_j, \mathrm{A}_1, .., \mathrm{A}_\mathrm{L}, \mathrm{B}_0 >\in EN$, with $k \in K$, $a_i \in A_i$ for i= 1, ..., L, and

$b_0 \in B_0, EN^F$ has the collection of tuples $Z_j^F = < K, B_j, A_1, \ldots, A_L, \mu(B_j)$, for j = 1, ..., n, with fuzzifier (B_0) = , i.e., the fuzzifier maps the attribute to a fuzzy set with finite cardinality n.

Example: Assume we need to fuzzify the entity non-registered user from the last example based on the attribute NumberOfProcesses. The domain of NumberOfProcesses in Non - registered user is {0, ..., 1000000}. In Non-register user^F, domain (- NumberOfProcesses) is a fuzzy set over the universe {low, medium, high, very high}. So with Non-register user^F = DBFuzzifier(Non-register user^F, fuzzifier (NumberOfProcesseso)), each tuple in Non - registered user is transformed to up to four fuzzy tuples in A Non-register user^F , one corresponding to each of the four sets low, medium, high, very high. To take a specific instance, the tuple <234, 230000> in non-registered user may be mapped to the tuples < 234, low, 0.1>, <234, medium, 0.9>. The membership grade of 230000 in the fuzzy sets high and very high is zero, so there are no corresponding tuples in the non-registered user relation. The ER construct we propose for the DBFuzzifier concept is shown in Figure 5.

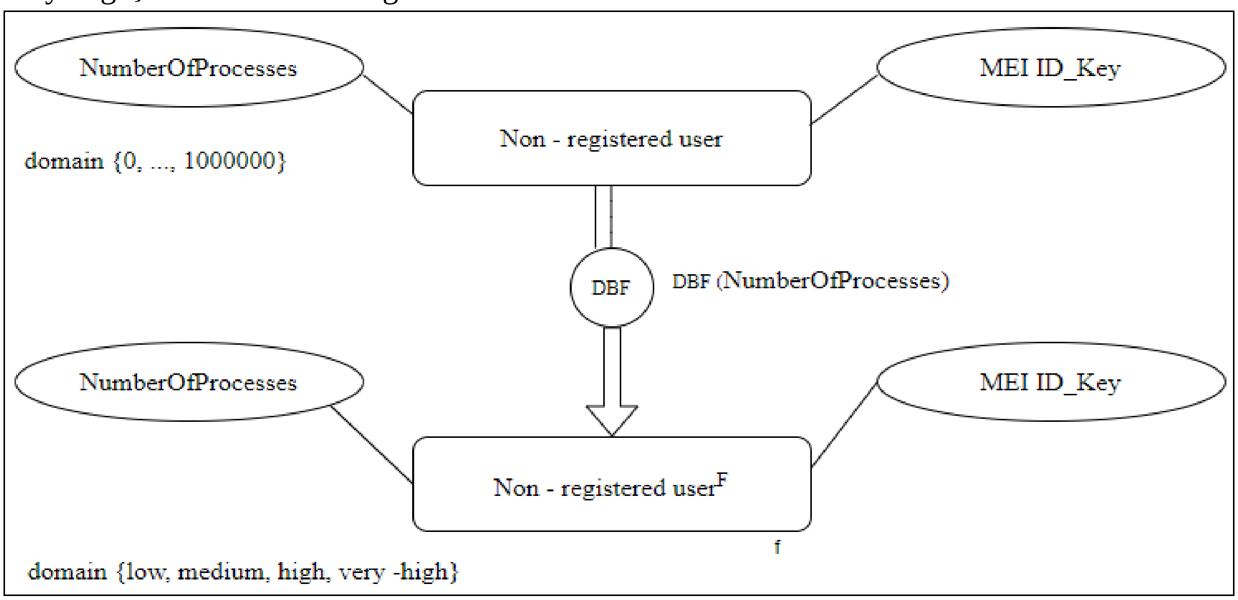


Figure 5 : DBFuzzifier Construct *(**Source :** Sinkonde, et al, 2023)*

Fuzzy ER Model Mapping to Relational Implementation

The mapping of this conceptual model to relations is required after the ER data model has been built.With the exception of adding a further membership property, fuzzy entities and fuzzy n-to-n connections can be mapped to relational databases in the same way as their crisp counterparts (showed in Section 3.1). Nevertheless, the mapping approach for converting fuzzy 1-to-1 and 1-to-n relationships to tables needs to be revised.

Converting the DBFuzzifier component and related entities to tables: Both input entity, EN and output entity, EN^F to the DBFuzzifier should be translated into separate tables. As an illustration, both non-registered user and $\text{Non} - \text{registered user}^F$ in Figure 5 ought to be mapped to different tables.

Methodology for Fuzzy Conceptual design
We now present the FPostGIS database design method that incorporates the additional components mentioned in Section 3.2 into the ER design method (Teorey, Yang, and Fry, 1986).

Method 1: Constructing an extended fuzzy ER data model

- Construct the common ER model.
- Attach 'f' to the entities and relationships that are fuzzy (see Figure 5).
- Show the DBfuzzifier construct for entities whose attributes are fuzzified at various levels.
- Attach 'fm_θ' to entities to be used in a fuzzy match, where θ is the desired comparison operator.

Method 2: Converting the ER model to relational tables.

- Convert crisp entities and crisp relationships to tables in the same way as MEP server (crisp) databases are converted to tables (as described in Section 3.1, Method 2).
- Fuzzy entities denoted by an 'f' must be converted to tables in the same way that crisp entities are, except adding an additional attribute to the fuzzy membership.
- Create fuzzy comparison functions for entities denoted with 'fm_θ'.

Method 3: Normalization of the relations.

- Normalize all relations by following method 2 by using functional dependencies, multi-valued dependencies, and restricted fuzzy functional dependencies.

Method 4: Guaranteeing correct interpretation of the fuzzy relational operators.

- This phase focuses on data operations rather than the data itself. It is needed only if the database management system (DBMS) used does not support fuzzy data. In the absence of a commercially available fuzzy DBMS, fuzzy logic can be applied to define fuzzy rules and constraints that govern the behavior of the entities and relationships in the data model in Section 2 The RDBMS might be extended to provide queries on fuzzy data, or queries embedded in host language programs could modify the results in the host language program.

Method 5: Fuzzy Querying and Reasoning

- By integrating fuzzy extensions into the ER data model design methods, it becomes possible to represent and handle uncertain or imprecise information more effectively, enabling better modeling of real-world scenarios where ambiguity and imprecision are prevalent by Hsieh et al., (2010).

FUZZIFYING THE MEP POSTGIS DATABASE

After each FuzzyMEP run, the rules in the PostGIS-database are evaluated to determine which of them should be invoked and what degree of certainty should be associated with this construction of the rule. Section 4.1 describes rule construction. Section 4.2 outlines our approach to handle conflicting information in rule consequents, while Section 4.3 describes the process for selecting the suitable algorithm based on rule building.

Determining the Grade of a Constructed Rule

As explained in Table 1A and Table 1B, the rules used in the system have the general form:

Rule One:	If the linguistic input term is (comment =metadata is 1) AND (barrier = very low), THEN (accessibility of a path = comment+µp+coordinate is Excellent)
Rule Two:	If the linguistic input term is (comment = metadata is 1) AND (barrier = low), THEN (accessibility of a path = comment+µp+coordinate is Excellent)
Rule Three:	If the linguistic input term is (comment = metadata is 1) AND (barrier =medium), THEN (accessibility of a path = comment+µp+coordinate is ok)
Rule Four:	If the linguistic input term is (comment = metadata is 1) AND (barrier = high), THEN (accessibility of a path = comment+µp+coordinate is poor)
Rule Five:	If the linguistic input term is (comment = metadata is 2) AND (barrier = very low), THEN (accessibility of a path =comment+µp+coordinate is Excellent)
Rule Six:	If the linguistic input term is (comment = metadata is 2) AND (barrier = low), THEN (accessibility of a path =comment+µp+coordinate is Excellent)

*(**Source**: Sinkonde, et al, p.4. 2017*

Therefore, If *predicate* [AND *predicate*] ..., then [NOT] *action*, α.

Consider an arbitrary rule R with $n$predicates on the LHS denoted by: If P_1AND ... AND P_n, then Action, α. To evaluate the LHS, each of these predicates will be matched to facts in the database. These facts may be fuzzy and/or the operand may be fuzzy. We note that the use of the DBFuzzifier allows us to reduce the complexity of the situation, by assuring that each value in the database corresponds to an element, and not a fuzzy set. This will have a positive impact on the performance of the matching process.

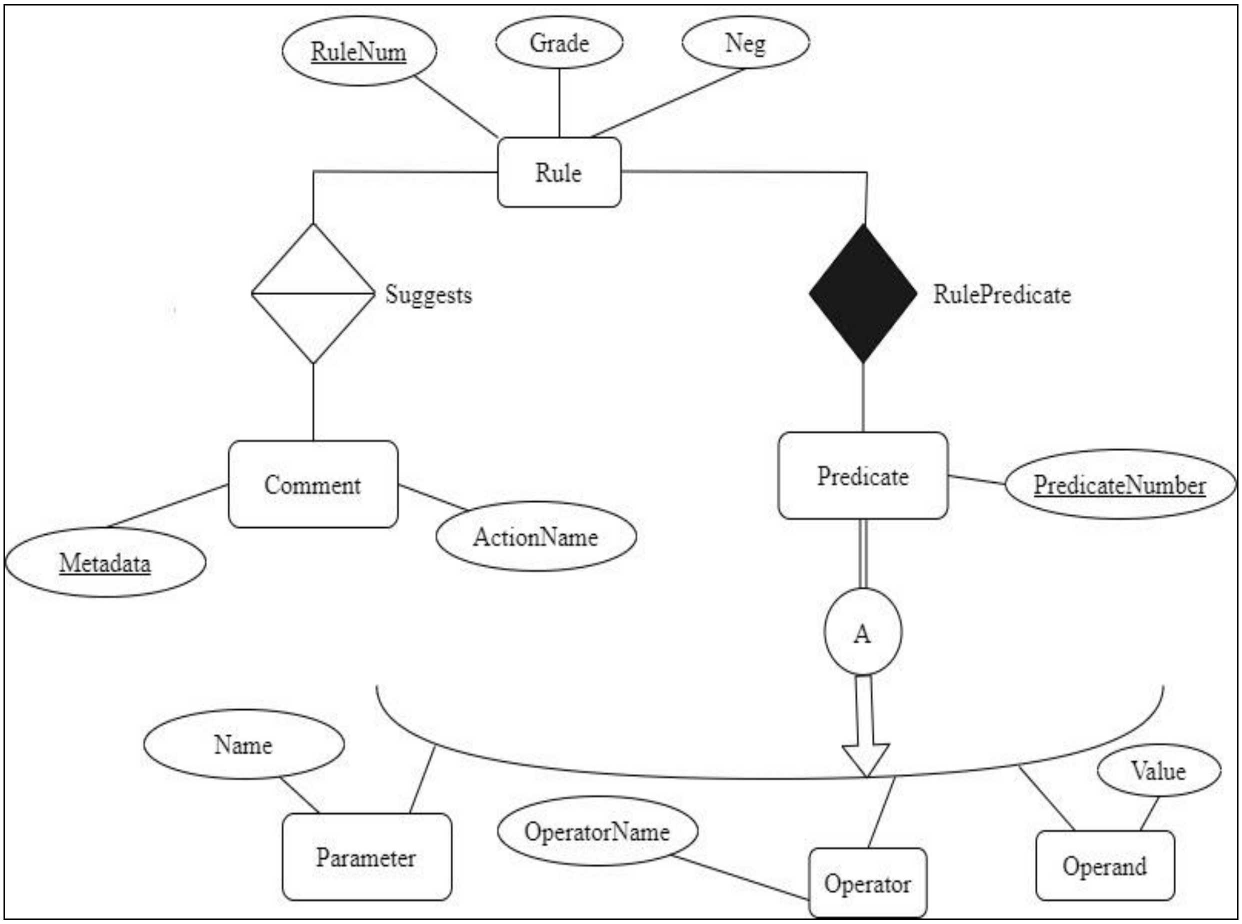


Figure 6: MEP ER Diagram of a Fuzzy Rule
*(**Source:** Sinkonde, et al, 2023)*

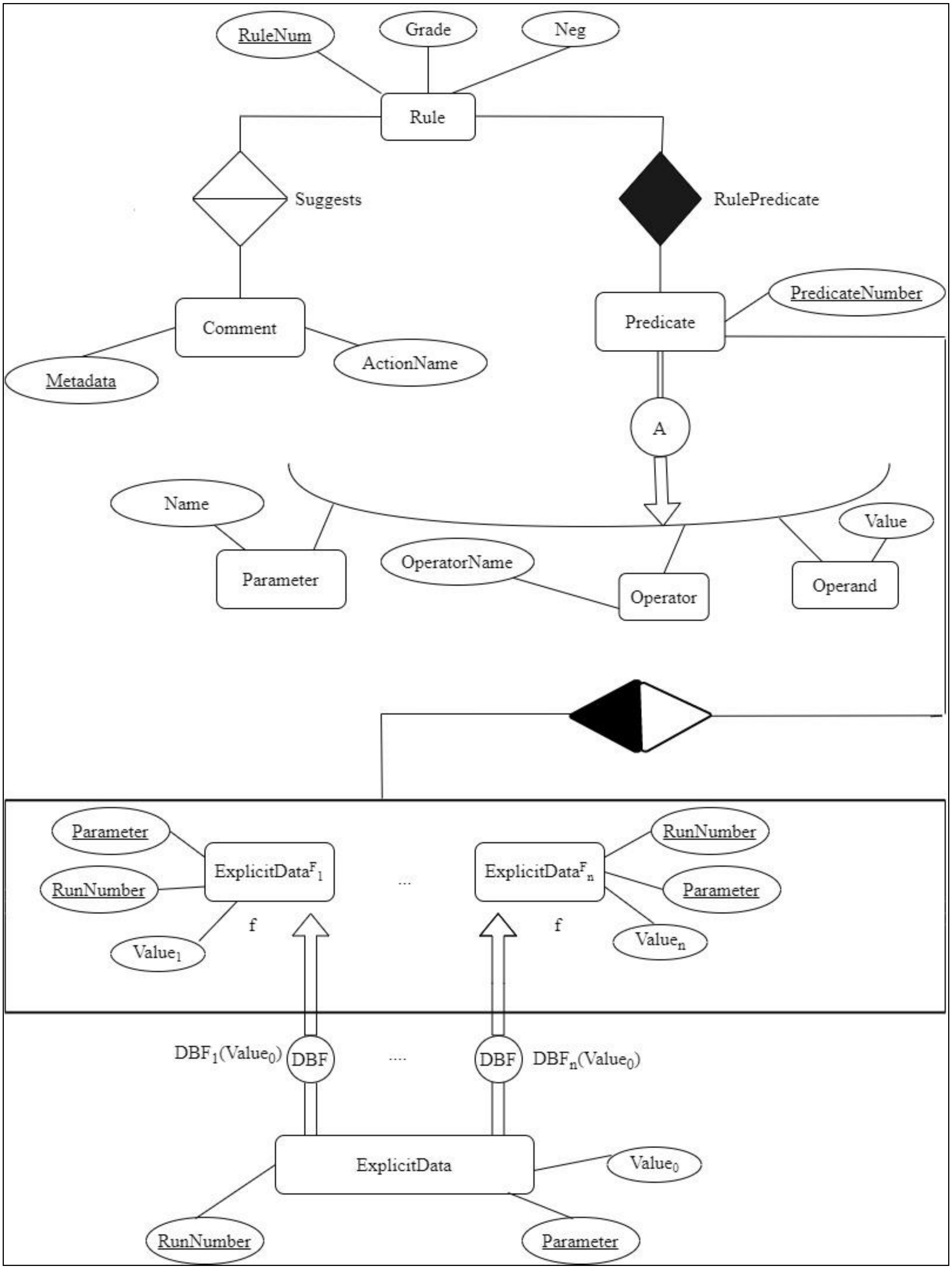


Figure 7: MEP ER Diagram of a Fuzzy Rule Base and the "facts" Data
*(**Source:** Sinkonde, et al, 2023)*

The predicate evaluation of a rule S thus results in $\delta_1, \delta_2, \dots, \delta_n$ which are the grades of match of the predicates $P_1, P_2, \dots, P_n$ of S. For the execution of the rule, an overall confidence γ needs to be calculated. γ is a function of the certainty factor α, and $\delta_1, \delta_2, \dots, \delta_n$. Since the predicates on the LHS are ANDed together, δ, the overall grade of match of the LHS, is defined as the minimum of $\delta_1, \delta_2, \dots, \delta_n$. Finally, we define$\gamma$, which is the confidence of the consequent as a result of the rule firing, as min (α, δ).

Example: Consider the following rule:
Rule 1: if barrier b_1 is reported on path h_1 the precision is high.
Rule 2: if barrier b_2 is reported on path h_2 the precision is low.
Rule 3: if barrier b_1 is reported on path h_2 the precision is high.
Rule 4: If pedestrian X using mobility aid a_1 is passing through path $P$and barrier b_1 is reported to be present across path$P$, then the alternative route h_1 is the most viable one.

Handling Imprecise information

The type of rule mentioned in the previous paragraph in the previous paragraph allows determination of rules that comment against an action. Therefore, when new rules are added to the existing rule base, some rules may be inconsistent in sense that one rule may clearly communicate the instructions of another rule.

Example: Consider that the rule base has the following rule:

Rule 1: If error $> 1\beta$ AND error $< 2\,\beta$, then precision is high, 0.8

New rules are added to the rule base:

Rule 2: If error $< 1\beta$ then the alternative route h_1 is the most viable one, 0.6

Rule 3: If error = low, then NOT precision is high, 0.6

Now Rule 3's consequent is in contradictory information regarding one issue with Rule 1's consequent.

In a review work of Sikchi et al. (2013), reports the main contributions taken from the literature for utilized the concepts used in the fuzzy expert system described how to handle contradictory information. A particular *action* may appear as a consequent in more than one rule. It may appear negated in some rules and non-negated in others. After each run, when the overall confidences for the consequents of all the rules have been evaluated, the grades of each occurrence of an action are unified. This is done by associating each action with two confidence levels, the *Upper Confidence* level and the *Lower Confidence* level.

The Upper Confidence represents the certainty that the particular action is supported by the rules constructed and is defined as the maximum of the overall grade (β) of all the matched rules in which that action appears non-negated as a consequent. The Lower Confidence is a measure of the degree to which the matched rules advise against a consequent. It is defined as 1 minus the maximum of the overall grade (β) of all the constructed rules in which that action appears negated as a consequent.

Note that if an action does not appear negated in any of the rules with a non-zero degree of match, its Lower Confidence is given a special "blank" value, indicating absence of knowledge about the Lower Confidence. Likewise, if an action does not appear non-negated in any of the rules with a non-zero degree of match, its Upper Confidence gets the value "blank".

Decision Making Process

The calculation of the Upper and Lower Confidences unifies the advice of all the rules with non-zero degree of match. At this stage, the large number of actions in our conflict set with varying degrees of support. Now a decision has to be made as to which actions have (not) been given sufficient support by the rule base for the present run, and hence should (not) be executed. To make this decision, we propose the following algorithm:

Let the Lower and Upper Confidences of each action be represented by the two-tuple (X_1, X_2) where X_1 is the Lower Confidence, and X_2 is the Upper Confidence. Both X_1 and X_2 are fuzzy grades, i.e., $X_1, X_2 \in [0, 1]$. The decision problem can then be modeled as defining decision regions on the area (0, 0), (0, 1), (1, 1), (1, 0) with the value of X_1 mapped along the x-axis and that of X_2 mapped along the y-axis [see Figure 8]. Details of the approach can be found in Zhang*et al*., (2012).

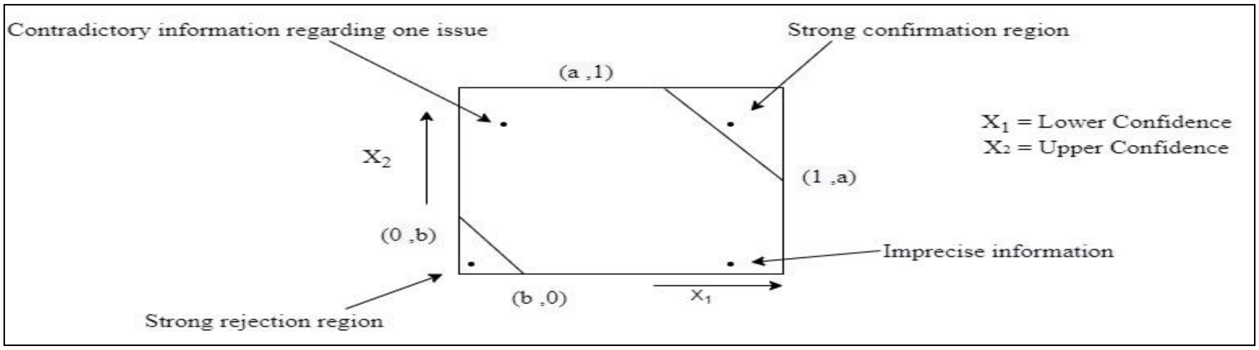


Figure 8: Two-Dimensional Representation of the Action-Related Support FuzzyMEP
Source: Own processing

A point with high X_1 and high X_2 represents an action that is strongly supported by the rules, since the Upper Confidence, X_2, which is the degree of support for taking the action, is high, and the Lower Confidence, X_1, which is 1 minus the degree of support for not taking the action, is also high. Conversely, a point with low X_1 and low X_2 represents an action which the rules strongly advise against. A point near the (1, 0) corner indicates an action for which both the support for and against is low, i.e., a region of *imprecise information*. A point near the (0, 1) corner corresponds to an action for which we have *contradictory information regarding one issue*, since the support for the action is high, and 1 minus the support against the action is low.

We utilize the above regions to employ the following rule selection strategy: Giving equal weight to the Lower and Upper Confidences, a *strong confirmation* region can be defined as the triangular region (a, 1), (1, 1), (1, a). Any action whose (X_1,X_2) representation falls in this region should be executed. Similarly, any action whose (X_1,X_2) representation falls in the *strong rejection* region (0, b), (0, 0), (b, 0) should not be executed. The constants 'a' and 'b' delineate these regions and hence determine thresholds for accepting or rejecting an action. These different regions are depicted in Figure 8.

A scenario where none of the conclusions falls in the strong confirmation region, and not all of the conclusions fall in the strong rejection region, presents us with a special case. In such a case the rule constructing has not ruled out all the conclusions but has also not explicitly suggested a particular set of conclusions. An example of this case is depicted in Figure 9. The conclusion with the maximum support is defined to be the one which is closest to the strong confirmation region. This conclusion with maximum support is taken as the decision reached by the rule invocation. In Figure 10, our algorithm would thus select the action denoted by (Y_1,Y_2).

Since it is plausible that none of the rules in the conflict set have a consequent where an action appears negated, subsequently it is possible that the Lower Confidence is assigned the value "blank". Similarly, the Upper Confidence will be assigned the value "blank" if an action appears negated in some of the rules in the conflict set, but does not appear non-negated in any of these rules. So, an action can have a "blank" for either X_1 or X_2. An action will not be associated with both (X_1,X_2).= ("blank", "blank"), rather it would not be listed at all. Since a "blank" indicates an absence of information, a "blank" as a Lower Confidence is interpreted as the value 1 for X_1, while a "blank" for an Upper Confidence is interpreted as the value 0 for X_2 [see Figure9].

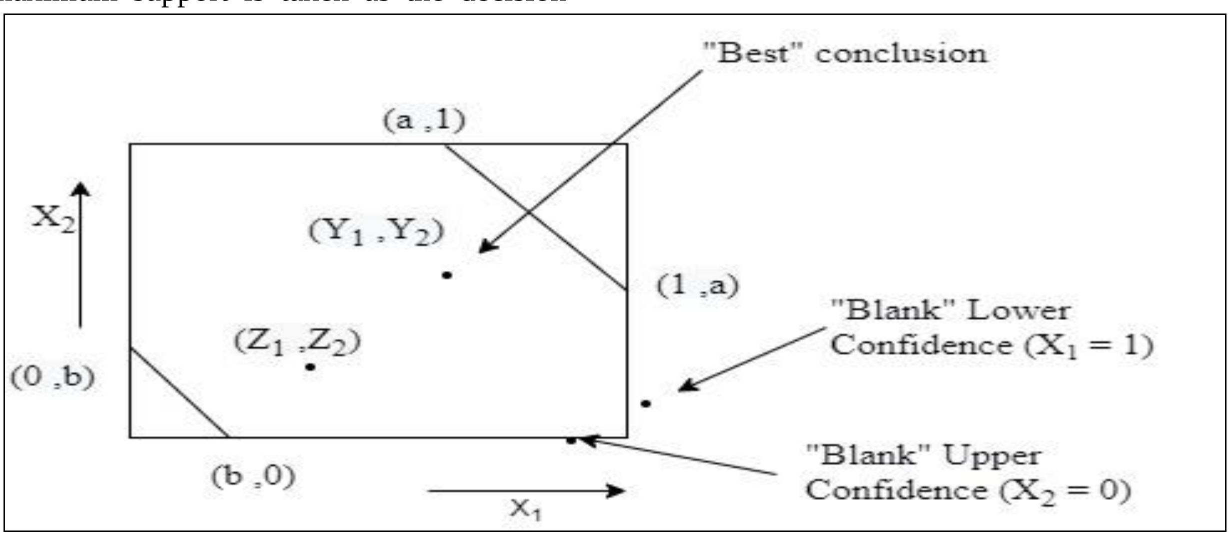


Figure 9: Selecting the suitable algorithm based on rule building.
Source*:* Own processing

CONCLUSIONS AND PERSPECTIVES

This paper makes an important contribution to the MEP project through the design methodology for fuzzy relational PostGIS databases (FPostGIS). More importantly, no findings from any design methodology for the building of fuzzy relational databases for the accessibility of a path have been published. In this paper, we propose a design methodology for FPostGIS based on accessibility information about a particular path. In this article, we propose a fuzzy extension (both graphical and formal definitions of extensions) to the ER model. Additionally, we describe a novel design methodology for mapping such fuzzy ER models for fuzzy relational databases. To that end, we are enhancing path reconstruction on a set of pedestrian paths by combining fuzzy logic and crowdsourcing methodologies. We show that the MEP server requires capabilities for managing imprecise data. Research has shown that the method has designed for FPostGIS can play a vital role in improving information about the quality-condition of pedestrian walkway accessibility and improving the quality of life for people with mobility challenges in urban areas.

RECOMMENDATIONS FOR FUTURE WORK

The research work in this paper focuses on the MEP project through the design methodology for fuzzy relational PostGIS databases (FPostGIS) for urban pedestrian accessibility. Fuzzy theory demonstrated its superior performance in modeling ambiguous information by Li et al., (2013) and Kang et al., (2020). There are still some open issues to be investigated in the future as an extension of this research. We propose research on the fuzzy information-based decision-making process where a single high confidence in a given conclusion overrides several low confidences in that conclusion. It's also important to research problems with the MEP server, such as issues with imprecise data or imprecise rules.

Disclosure statement

No potential conflict of interest was reported by the authors.

ACKNOWLEDGMENTS

The authors would like to acknowledge the contribution of the following associations: CRAiS (http://www.crais.eu/ - in Italian), Lions Club International Area Milanese, Comodalbasso (http://www.comodalbasso.org/ - in Italian), Ledha Milano (http://www.ledhamilano.it/ - in Italian). Special thanks to the Fondazione Politecnico di Milano for the generosity and support through the scholarship. We thank the Open University of Tanzania, Tanzania under the department of Information and Communications Technology (ICT)and Mbeya University of Science and Technology (MUST) for supporting this work and for allowing us to use their resources.

REFERENCES

Bezdek, Vaclav. (2011). Possible use of fuzzy logic in database. Journal of Systems Integration, 2(2), 31-46.

Bojadziev, George. (2007). Fuzzy logic for business, finance, and management (Vol. 23): World Scientific.

Buckles, Billy P, & Petry, Frederick E. (1993). A fuzzy representation of data for relational databases. In *Readings in Fuzzy Sets for Intelligent Systems* (pp. 660-666). Morgan Kaufmann.

Cao, TH. (2022). A relational database model and algebra integrating fuzzy attributes and probabilistic tuples. Fuzzy Sets and Systems, 445, 123-146.

Chaudhry, Nauman A, Moyne, James R, & Rundensteiner, Elke A. (1999). A design methodology for databases with uncertain data. Paper presented at the Scientific and Statistical Database Management, 1999. Proceedings., Seventh International Working Conference on.

Chaudhry, Nauman, Moyne, James, &Rundensteiner, Elke A. (1999). An extended database design methodology for uncertain data management. Information sciences, 121(1-2), 83-112.

Chen, Guoqing. (2012). Fuzzy logic in data modeling: semantics, constraints, and database design (Vol. 15): Springer Science & Business Media.

Comai, S., Kayange, D., Mangiarotti, R., Matteucci, M., Yavuz, S.U., Valentini, F.: Mapping City Accessibility: Review and Analysis. In: Studies in Health Technology and Informatics, vol. 217, pp. 325–331. IOS Press (2015).

Comai, S., De Bernardi, E., Matteucci, M., Salice, F. (2017). Maps for Easy Paths (MEP): Enriching Maps with Accessible Paths Using MEP Traces. Int. Conf. Smart Objects and Technologies for Social Good. GOODTECHS 2016. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 195. Springer, Cham, pp 254–263.

De Sousa, Victor Martins, & Cura, Luis Mariano delVal. (2018). Logical design of graph databases from an entity-relationship conceptual model. Paper presented at the Proceedings of the 20th International Conference on Information Integration and Web-based Applications & Services.

Edward, T, Tanya, L, & Mo, J. (2009). Introduction to fuzzy logic control with application to mobile robotics. NASA Center for Autonomous Control Engineering Department of Electrical and Computer Engineering University of New Mexico.

Fahrner, C., &Vossen, G. (1995). A survey of database design transformations based on the entity-relationship model. Data & Knowledge Engineering,15(3), 213-250.

Hsieh, H.-I., Lee, T.-P., & Lee, T.-S. (2010). Data Mining in Building Behavioral Scoring Models. 2010 International Conference on Computational Intelligence and Software Engineering, 1-4. https://doi.org/10.1109/CISE.2010.5677005

Li, S. T., & Tsai, F. C. (2013). A fuzzy conceptualization model for text mining with application in opinion polarity classification. Knowledge-Based Systems, 39, 23-33.

Kang, C., Yu, X., Wang, S. H., Guttery, D. S., Pandey, H. M., Tian, Y., & Zhang, Y. D. (2020). A heuristic neural network structure relying on fuzzy logic for images scoring. IEEE transactions on fuzzy systems, 29(1), 34-45.

Ma, Zongmin, & Yan, Li. (2022). Data modeling and querying with fuzzy sets: A systematic survey. Fuzzy Sets and Systems.

Medina, Juan Miguel, Pons, Olga, & Vila, Maria Amparo. (1994). GEFRED: a generalized model of fuzzy relational databases. Information sciences, 76(1-2), 87-109.

MEP project website: http://mep5x1000.wix.com/mepapp

Nguyen, Hung T, &Sugeno, Michio. (2012). Fuzzy systems: modeling and control (Vol. 2): Springer Science & Business Media.

Petry, Fred, &Yager, Ronald. (2022). Intuitionistic and Interval-Valued Fuzzy Set Representations for Data Mining. Algorithms, 15(7), 249.

Purian, F. K., &Sadeghian, E. (2013, December). Mobile robots path planning using ant colony optimization and Fuzzy Logic algorithms in unknown dynamic environments. In 2013 international conference on control, automation, robotics and embedded systems (CARE) (pp. 1-6). IEEE.

Sinkonde, D., Mselle, L., Shidende, N., Comai, S., & Matteucci, M., (2018). Developing an Intelligent PostGIS Database to Support Accessibility Tools for Urban Pedestrians. Urban Science, 2(3), 52. Urban Sci. 2018, 2, 52; doi:10.3390/urbansci2030052

Sinkonde, D., Mselle, L., Shidende, N., Comai, S., Matteucci, M.: Accessible urban by modeling the explicit data using fuzzy logic. In: ACM International Conference Proceeding Series, pp. 1–7. ACM (2017). https://doi.org/10.1145/3231830.3231844

T. Teorey, D. Yang and J. Fry, "A logical design methodology for relational databases using the extended entity-relationship model," ACM Computing Surveys 18, 2 (June 1986) 197-222.

Teorey, T. J., Yang, D., & Fry, J. P. (1986). A logical design methodology for relational databases using the extended entity-relationship model. *ACM Computing Surveys (CSUR)*,*18*(2), 197-222.

Thalheim, Bernhard. (2013). Entity-relationship modeling: foundations of database technology: Springer Science & Business Media.

Zadeh, Lotfi A. (2015). Fuzzy logic—a personal perspective. Fuzzy Sets and Systems, 281, 4-20.

Zhang, Fu, Yan, Li, & Ma, Zong Min. (2012). Reasoning of fuzzy relational databases with fuzzy ontologies. International Journal of Intelligent Systems, 27(6), 613-634.