

Data Modeling Dealing With Uncertainty in Fuzzy Logic

Angélica Urrutia¹, José Galindo², Leoncio Jimenez¹, and Mario Piattini³

1 Dpto. de Computación e Informática, Universidad Católica del Maule,
Chile,

aurrutia, ljimenez@ucm.cl

2 Dpto. Lenguajes y Ciencias de la Computación, Universidad de Málaga,
España, jgg@lcc.uma.es

3 Escuela Superior de Informática, Universidad de Castilla-La Mancha,
Mario.Piattin@uclm.es

Abstract. This paper shows models of data description that incorporate uncertainty like models of data extension EER, IFO among others. These database modeling tools are compared with the pattern FuzzyEER proposed by us, which is an extension of the EER model in order to manage uncertainty with fuzzy logic in fuzzy databases. Finally, a table shows the components of EER tool with the representation of all the revised models.

1 Introduction

On occasions the term “imprecision” embraces several meanings between which we should differentiate. For example, the information we have may be incomplete or “fuzzy” (diffuse, vague), or we may not know if it is certain or not (uncertainty), or perhaps we are totally ignorant of the information (unknown), we may know that that information cannot be applied to a specific entity (undefined), or we may not even know if the data can be applied or not to the entity in question (“total ignorance” or value “null”) [1]. Each of these terms will depend on the context in which they are applied.

The management of uncertainty in database systems is a very important problem [2] as the information is often vague. Motro states that fuzzy information is content-dependent, and he classifies it as follows:

- Uncertainty: It not possible to determine whether the information is true or false. For example, “John may be 38 years old”.
- Imprecision: The information available is not specific enough. For example, “John may be between 37 and 43 years old”, — disjunction — “John is 34

or 43 years old”, — negative — “John is not 37 years old”, or even unknown.

- **Vagueness:** The model includes elements (predicates or quantifiers) which are inherently vague, for example, “John is in his early years”, or “John is at the end of his youth”. However, once these concepts have been defined, this case would match the previous one (imprecision).
- **Inconsistency:** It contains two or more pieces of information, which cannot be true at the same time. For example, “John is 37 and 43 years old, or he is 35 years old”; this is a special case of disjunction.
- **Ambiguity:** Some elements of the model lack a complete semantics (or a complete meaning). For example, “It is not clear whether they are annual or monthly salaries”.

Zadeh as described in [3] introduces the fuzzy logic, in order to deal with this type of data. Traditional logic, because it is bi-valued, can only operate with concepts like: yes or no, black or white, true or false, 0 or 1, which allowed just for a very limited knowledge representation. Although there are other logics which take more truth values, namely multi-valued logics, fuzzy logic is one extension which takes endless truth levels (or degrees), associating the concept of membership degree or truth degree in an interval $[0,1]$ within the fuzzy logic theory.

Fuzzy databases have also been widely studied [1,4], with little attention being paid to the problem of conceptual modeling [5]. This does not mean that there are no publications, however, but that they are sparse and with no standard. Therefore, there have also been advances in modeling uncertainty in database systems [6-9] including object-oriented database models [9].

At the same time, the extension of the ER model for the treatment of fuzzy data (with vagueness) has been studied in various publications [4,7,10-15], but none of these refer to the possibility of expressing constraints by using the tools by fuzzy sets theory. In [8] a summary of some of these models can be found.

On the other hand, the main methodologies of databases design [8,16-19] have not paid attention to the modeling of data with uncertainty, although the intent of uncertainty modeling of the real world is rarely absent.

Based on these concepts, in this paper we will discuss different approaches, by various authors, related to the uncertainty conceptual modeling problem in database models. Closing the modeling stage, in sections 2, 3, 4, 5, 7 we present a Fuzzy Enhanced Entity-Relationship model, also known as FuzzyEER, a tool for fuzzy database modeling with many advantages with respect to the modeling tools presented in this section 6: fuzzy values in the attributes, degree in each value of an attribute, degree in a group of values of diverse attributes, as well as, fuzzy entities, fuzzy relationships, fuzzy aggregation, fuzzy constraints. In section 8 includes a comparison of FuzzyEER and some other fuzzy models.

2 The Zvieli and Chen Approach

In [15] is the first great approach in ER modeling. They allow fuzzy attributes in entities and relationships and introduced three levels of fuzziness in the ER model:

1. At the first level, entity sets, relationships and attribute sets may be fuzzy, namely, they have a membership degree to the model. For example, in Figure 1 the fuzzy entity “Company” has a 0.9 membership degree, the relationship “To Accept” has a 0.7 membership degree and the fuzzy attribute “Electronic mail” has a 0.8 membership degree.
2. The second level is related to the fuzzy occurrences of entities and relationships. For example, an entity Young_Employees must be fuzzy, because its instances, its employees, belong to the entity with different membership degrees.
3. The third level concerns the fuzzy values of attributes of special entities and relationships. For example, attribute Quality of a basketball player may be fuzzy (bad, good, very good...).

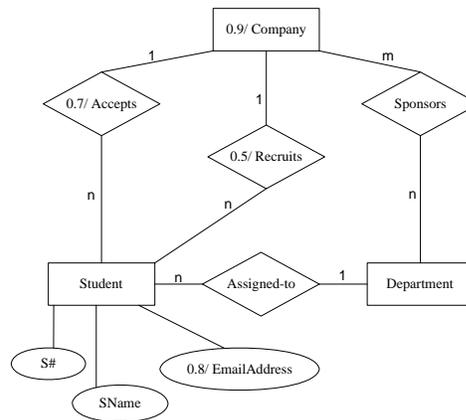


Fig. 1. Example with membership degrees to the model in some sets (entities, relationships or attributes): The first level of the Zvieli and Chen approach [3].

The first level may be useful, but at the end we must decide whether such an entity, relationship or attribute will appear or will not appear in the implementation. The second level is useful too, but it is important to consider different degree meanings (membership degree, importance degree, fulfillment degree...). A list of authors using different meanings may be found in [20]. The third level is useful, and it is similar to writing the data type of some attributes, because fuzzy values belong to fuzzy data types.

3 Proposal of Yazici et al.

In [21] propose an extension of the IFO model, shown in Figure 2 a), for the processing of imprecise data, and special treatment of data where similarity exists in a label. They call this extension ExIFO, and by means of examples they explain the implementation and validation of the representation of a fuzzy conceptual scheme by looking at a representation of uncertain attributes. In the model three new constructors are added and using these new constructors it is possible to represent explicitly attributes that have uncertain values.

The ExIFO conceptual model [9] allows imprecision and uncertainty in database models, based on the IFO conceptual model [9,21]. They use fuzzy-valued attributes, incomplete-valued attributes and null-valued attributes. In the first case, the true data may belong to a specific set or subset of values, for example the domain of this attribute may be a set of colors {red, orange, yellow, blue} or a subset {orange, yellow} where there is a similarity relation between the colors. In the second case, the true data value is not known, for example, the domain of this attribute may be a set of years between 1990 and 1992. In the third case, the true data value is available, but it is not expressed precisely, for example the domain of this attribute may be the existence or not of a telephone number. For each of these attribute types, there is a formal definition and a graphical representation. In this study, the authors introduce a high-level primitives to model fuzzy entity type whose semantics are related to each other with logic operators OR, XOR or AND. An example involving an Employee-Vehicle scheme is used in Figure 2 b) to illustrate the aggregation and composition of fuzzy entity types. The main contribution of this approach is the use of an extended NF² relation (Non First Normal Form) to transform a conceptual design into a logical design. Consequently, the strategy is to analyze the attributes that compose the conceptual model in order to establish an NF² model.

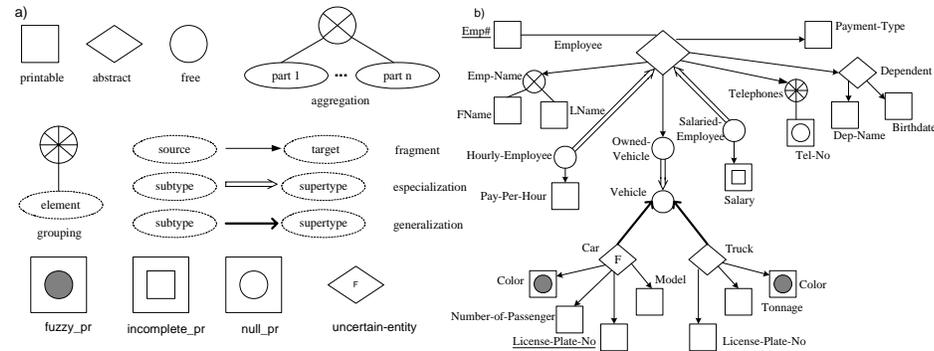


Fig. 2. Fuzzy ExIFO Model proposed by Yazici and Merdan [21]. a) Notation. b) Example Employee-Vehicle.

The study in [22] is, mainly, a conceptual modeling approach for the representation of complex-uncertain information [21] using object-oriented paradigm and an algorithm for transforming a conceptual schema specification of the model introduced here (ExIFO) into a logical schema of the fuzzy object-oriented databases model (FOOD) is proposed. ExIFO attempts to preserve the acquired strengths of

semantic approaches, while integrating concepts from the object paradigm and fuzziness by adding new constructors.

4 The Chen and Kerre Approach

In [7,8,11] these authors introduced the fuzzy extension of several major EER concepts (superclass, subclass, generalization, specialization, category and shared subclass) without including graphical representations. The basic idea is that if $E1$ is a superclass of $E2$ and $e \in E2$, then $E1(e) \leq E2(e)$, where $E1(e)$ and $E2(e)$ are the membership functions of e to $E1$ and $E2$, respectively. They discussed three kinds of constraints with respect to fuzzy relationships but they do not study fuzzy constraints: a) The inheritance constraint means that, a subclass instance inherits all relationship instances in which it has participated as a superclass entity. b) The total participation constraint for entity E is defined when for any instance in E , $\exists \alpha_i$ such that $\alpha_i > 0$, where α_i is one membership degree in the fuzzy relationship. c) The cardinality constraints 1:1, 1:N and N:M are also studied with fuzzy relationships.

The fuzzy ER model, Chen [7] proposes a model generated by $M = (\mathbf{E}, \mathbf{R}, \mathbf{A})$ expressed by \mathbf{E} as entity type, \mathbf{R} as interrelation type, and \mathbf{A} as attributes, also including label types which generate, at the first level, $L1(M) = (\mathbf{E}, \mathbf{R}, \mathbf{A}_E, \mathbf{A}_R)$, and proposes four set types, with notation shown in Figure 3 (see an Example in Figure 1), and where μ_X is the membership function to the set X (one Entity, one Relationship or one Attribute) and D_E is the domain of \mathbf{E} composed of all possible entity types concerned:

- $\mathbf{E} = \{\mu_E(\mathbf{E})/\mathbf{E} : \mathbf{E} \in D_E \text{ and } \mu_E(\mathbf{E}) \in [0,1]\}$.
- $\mathbf{R} = \{\mu_R(\mathbf{R})/\mathbf{R} : \mathbf{R} \text{ is a relationship type involving entity types in } D_E \text{ and } \mu_R(\mathbf{E}) \in [0,1]\}$.
- $\mathbf{A}_E = \{\mu_{AE}(\mathbf{A})/\mathbf{A} : \mathbf{A} \text{ is an attribute type of entity type } \mathbf{E} \text{ and } \mu_{AE}(\mathbf{A}) \in [0,1]\}$.
- $\mathbf{A}_R = \{\mu_{AR}(\mathbf{B})/\mathbf{B} : \mathbf{B} \text{ is an attribute type of relationship type } \mathbf{R} \text{ and } \mu_{AR}(\mathbf{B}) \in [0,1]\}$.

The participation constraint (Figure 3) is modeled setting that an entity E λ -participates in R if for every e of E , there exists a f in F such that $\mu_R(e,f) \geq \lambda$. The cardinality constraint is shown at the end of Figure 3, where N and M are fuzzy sets. The concept of fuzzy quantifier is not used in this approach.

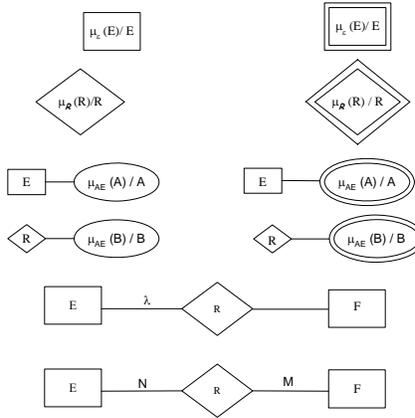


Fig. 3. ER Fuzzy notation proposed by Chen [7].

At the second level, for each entity type E and relationship type R, the sets of their values can be fuzzy sets, reflecting possible partial belonging of the corresponding values to their types. The third level of fuzzy extensions concerns with attributes and their values. For each attribute type A, any of its values can be a fuzzy set.

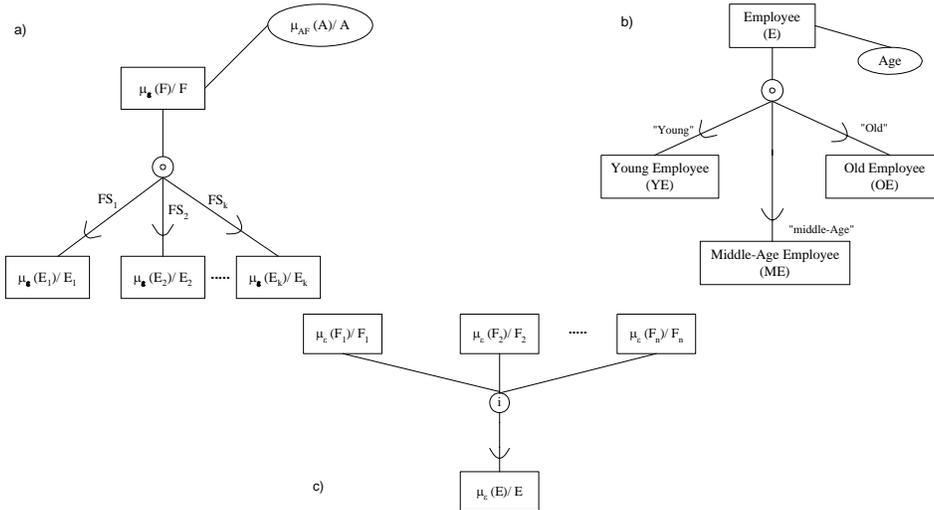


Fig. 4. Notation proposed by Chen [7]: a) An attribute-defined overlapping specialization whit $FS_i, \in F(\text{Dom}(A))$ at the first level, b) Employee in an overlapping specialization with the fuzzy attribute Age, c) Shared subclass intersection .

Later on, in another section, an attribute-defined specialization is defined with $FS_i \in F(\text{Dom}(A))$, where all the FS_i are fuzzy sets on $\text{Dom}(A)$, the domain of the attribute A . Graphically, this kind of attribute-defined specialization can be represented as shown in Figure 4 a). Figure 4 b) shows the entity Employee, and the fuzzy attribute Age with the labels “Young Employee”, “Middle-Aged Employee”, and “Old Employee”. He also includes the fuzzy definition for categories and shared subclass, i.e. union and intersection (see Figure 4 c)). This proposal, makes always reference to linguistic labels, and to the trapezoidal function over an attribute or specific entity, not to a set of different attributes or different entities. This author, just like [21], establishes his data models from the attributes, and creates the object class or entity by using generalization and specialization tools.

In [7] defines that a linguistic variable X is composed of the tuple (T, U, G, M) where: T is the set of linguistic terms of X , U is the universe of discourse, G is the set of syntactic rules that generate the element T , and M is the set of semantic rules translated from T that correspond to the fuzzy subset of U . With this, he defines a conceptual model and its mathematical representation. For example, let us $X = \text{Age}$ in Figure 5, T is generated via G by the set $\{\text{Young}, \text{Middle-Aged}, \text{Old}\}$. Each term of T is specifically handled by M by fuzzy sets. The type of correspondence between an entity and a fuzzy entity is also established, as well as the set of values that a membership degree obtains from a fuzzy set: $1:1$, $1:N$, $N:M$, incorporating fuzziness to the ER model.

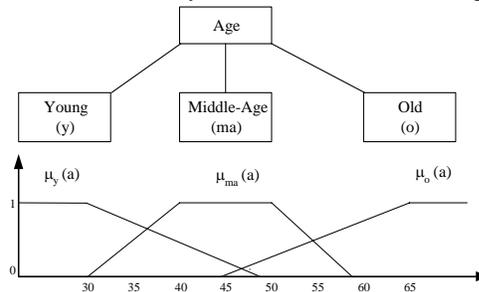


Fig 5. Linguistic variable “Age” with its corresponding values and conceptual model, according to Chen [7].

5. Proposal of Ma et al.

In [13] work with three levels of [15] incorporate in the Fuzzy Extended Entity-Relationship model (FEER model) a way of managing complex objects in the real world at a conceptual level, associating an importance degree of each of the components (attributes, entities, etc.) to the scheme. However, their definitions (of generalization, specialization, category, and aggregation) impose very restrictive conditions. They also provide an approach to mapping a FEER model to a Fuzzy Object-Oriented Database scheme (FOODB).

Figure 6 1) shows the following: a) single-valued attribute type, b) multivalued attribute type, c) disjunctive fuzzy attribute type, d) conjunctive fuzzy attribute type, e) null attribute type, f) open or null attribute type, g) disjunctive imprecise attribute type, h) conjunctive imprecise attribute type, i) entity with grade of membership, j) relationship with grade of membership, and k) attribute with grade of membership.

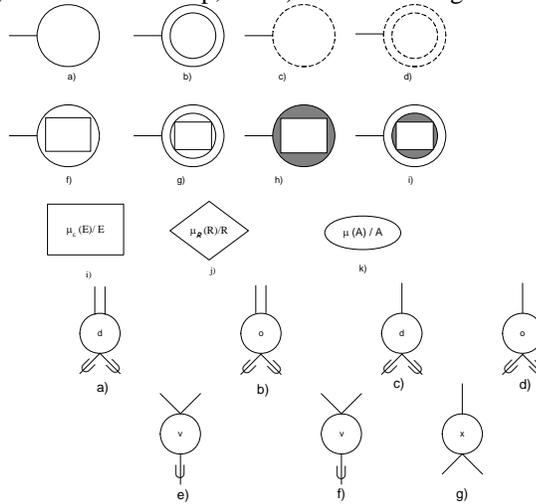


Fig. 6. FEER Notation by Ma et al. [13]: 1) Fuzzy attributes, entities, and interrelations, 2) Specialization, aggregation, and fuzzy categories

In addition, Figure 6 2) shows the following notations: a) fuzzy total and disjoint specialization, b) fuzzy total and overlapping specialization, c) fuzzy partial and disjoint specialization, d) fuzzy partial and overlapping specialization, e) fuzzy subclass with fuzzy multiple superclasses, f) fuzzy category, and g) fuzzy aggregation.

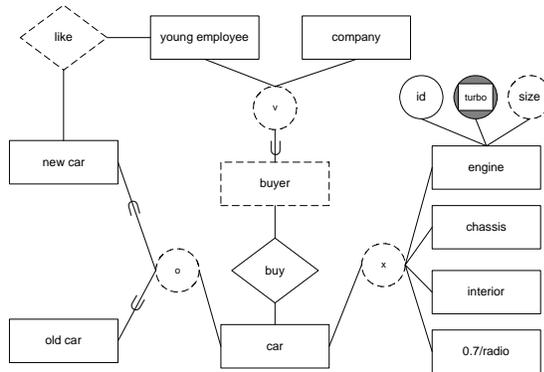


Fig. 7. Example of Ma et al. [13] notation for a car assembly company case.

Figure 7 shows an example of the EER Fuzzy model utilizing some of the notions proposed by Ma et al [13]. Thus, the “car” entity is a superclass with two fuzzy subclasses “new car” and “old car” in an overlapping specialization. Besides, the “young employee” fuzzy entity, having fuzzy instances from the “company” entity, consists of the “union” category from the fuzzy entity “buyer”. Also, “young employee” has a fuzzy relationship “like”. Finally, the “car” entity is an aggregation of some entities: “engine”, “chassis”, “interior” and “radio” (with an associated fuzzy degree of 0.7). Note that “engine” has some fuzzy attributes like *size* and *turbo*.

Ma et al. [23] introduce an extended object-oriented database model to handle imperfect as well as complex objects. They extend some major notions in object-oriented databases such as objects, classes, objects-classes relationships, subclass/superclass, and multiple inheritances.

6. Proposal of Urrutia et al.

The notation of the FuzzyEER model. It defines fuzzy attributes Type 1, Type 2 and Type 3 (Type 4 it is not shown here), and Fuzzy Relationship, Fuzzy degree associated to an attribute shown to Example 1 [24-29]. This fuzzy entity, attributes and relationship it is very common to be seeking someone with particular features for representing a specific character, which must have particular physical characteristics.

Example 1: For a real estate agency, the entity District can have the attributes (District_Id, Name, Quality). The attributes District_Id and Name are crisp. The attribute Quality of the district is defined as a fuzzy attribute Type 3, with the following labels: {Low, Regular, Good, Excellent}.

The relationship of proximity of the neighborhoods can be represented as the fuzzy relationship Close_to, which appears in Figure 8. This expresses that a proximity degree exists between any two districts.

Furthermore, the entity Landed_Property is modeled with some attributes, which can also be seen in Figure 8. Each landed property can be situated in such a place that it belongs to several districts, or that it belongs to one district but it is relatively close to another or other districts. For example, for a property, it can be indicated that its neighbourhood has the following possibility distribution (0.5/North, 1/East, 0.2/Plaza_España), indicating that it is situated in the eastern district closer to the northern district than to the España square district.

If District were an attribute of Landed_Property it would be sufficient to define it as Type 3, to define each district as a label and establish a similarity relationship (or proximity in this case) for every two districts. But this is a special case, because District is an entity with some attributes and it is related to the Landed_Property entity, so that a property may be related to several districts (3 at most). At the same time, a district for a certain landed property may have a membership degree that measures to what extent that property belongs to that district. In our model it is represented by the degree $G^{\text{Membership}}$.

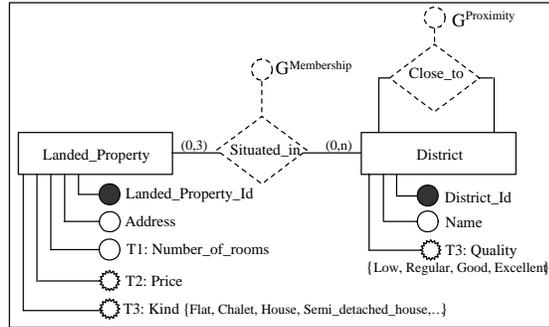


Fig. 8. Example 1, Fuzzy relationships with Fuzzy degree associated to an attribute and Fuzzy attribute Type 1, 2 y 3.

Therefore, the fuzzy relationship **Close_to** in fact generates a similar structure to a Type 3 fuzzy attribute. On the other hand, the fuzzy relationship **Situated_in** generates a similar structure in the **Landed_Property** entity, as if that entity had a Type 3 fuzzy attribute called **District**. The model reflects that the entities **Landed_Property** and **District** are related in such a way that each landed property may be situated in a maximum of 3 districts and each one of those associations gives the degree at which that landed property belongs to the district. Due to the fact that **District** has several attributes it can not be used as a fuzzy attribute Type 3 of **Landed_Property**.

A more detailed example of this case is found in [26,28]. other types of treatment of uncertainty: Fuzzy degree to the model, fuzzy degree with its own meaning, fuzzy aggregation of entities, fuzzy aggregation of attribute are treated in [25,26,29]. In the example 2 the Fuzzy entity is shown, other entities: weak entity (existence and identification) are treated in [26, 27].

Example 2: We may consider a fuzzy entity **Employee**, with an attribute which stores the total number of hours worked per week. For each employee, a membership degree to the entity can be defined, in such a way that the employees will belong to the **Employee** type of entity with a certain degree, according to the number of weekly hours. This degree will be calculated by dividing the total number of hours worked by the minimum number of hours, so that the belonging is total. Note that this is a derived fuzzy attribute in order to obtain the membership degree to the entity.

Figure 9 models this example, where $Q(h)$ is the calculus of the degree and h is the number of hours worked per week. We can see that, $Q(h) = \min\{1, h/m\}$, where m is the minimum number of hours for the total membership.

Therefore, if $m = 35$, an employee who works in the company for 15 hours, will be considered an employee with a degree of 0.43 (the result of the division $15/35$), so that this degree can be maintained in diverse calculations (selections with different aims, gratifications...).

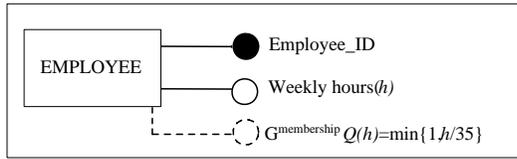


Fig. 9. Example 2, Fuzzy entity with a membership degree for each instance, which depends on the number h of hours worked per week.

Example 3: Let us consider an entity for Special Employees with its own attributes (extra payment, number of awards, motive...). A member of this shared subclass must be an engineer, a chief (boss) and a permanent employee. Figure 10 depicts this model with the following participation constraint: Almost all the chiefs and permanent employees must be special employees. It is interesting to note how this constraint means that almost all the chiefs and permanent employees must also be engineers (because all special employees belong to the engineer superclass).

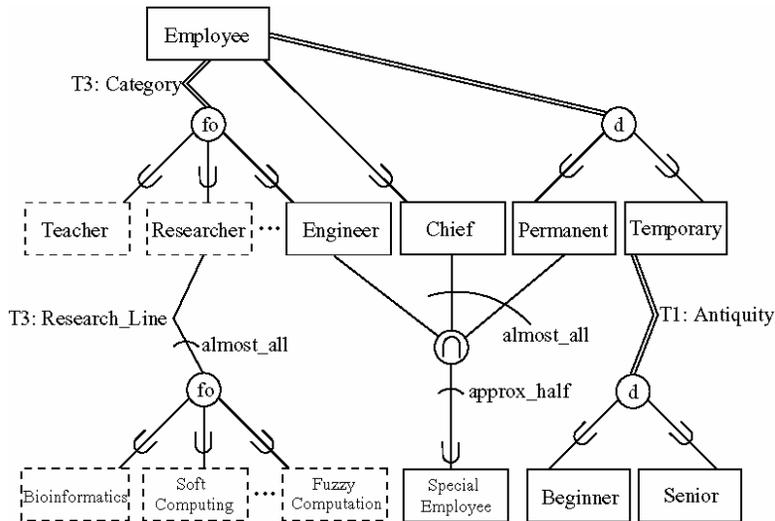


Fig. 10. Examples 3, Three Fuzzy Attribute-Defined Specializations and Fuzzy Constraints in a Shared Subclass.

On the other hand, the fuzzy completeness constraint establishes that approximately half of the employees who are engineers, chiefs and permanent employees must be special employees [24,30,31]. At the present time one has to disposition a tool CASE that it allows to model part of FuzzyEER.

7. Approaches by Other Authors

In [14] proposes an extension of the ER model with fuzzy values in the attributes, and a truth value can be associated with each relationship instance. In addition, some special relationships such as same-object, subset-of, member-of... are also introduced. In [32] applied Zadeh's extension principle to calculate the truth value of propositions. For each proposition, a possibility distribution is defined on the doubleton true, false of the classical truth values. In this way, the concepts such as entity, relationship and attribute as well as subclass, superclass, category, generalization and specialization... have been extended.

In [33] discussed of two types of imperfect information, appearing in database applications: fuzzy information representing information with inherent gradations, for which it is impossible to define sharp or precise borders, and uncertain or imprecise information, representing information which is (temporarily) incomplete due to a lack of sufficient or more precise knowledge. Dealing with this kind of imperfect information within the formal and crisp environment of a computer, is based in this paper upon the fuzzy set theory and its related possibility theory, which offers a formal framework to model imperfect information, and upon the object-oriented paradigm, which offers flexible modeling capabilities. The result is the UFO database model, a “fuzzy” extension of a full-fledged object-oriented database model.

This research discusses the UFO database model in detail in three steps. First, it is shown how fuzzy information is handled: meaningful fuzzifications of several object-oriented concepts are introduced in order to store and maintain fuzzy information, and to allow a flexible or “soft” modeling of database application. Then, it is discussed how uncertainty and imprecision in the information are handled: possible alternatives for the information are stored and maintained by introducing role object, which are tied like shadows to regular objects in the database; they allow the processing of uncertainty and imprecision in a, to the user, implicit and transparent way, and they also allow the modeling of tentative behavior and of hypothetical information in the database application. Both the static and the dynamic aspects of (imperfect) information are developed in the UFO database model, and imperfect information is considered at the data level as well as at the metalevel of a database application. The process of “extending” an object-oriented database model to the UFO database model, as discussed here, adheres, as closely as possible, to the original principles of the object-oriented paradigm, to allow a flexible and transparent, but semantically sound modeling of imprecise information. The object-oriented database model, which the extension process starts off from, adheres to the standard proposal ODMG-93, to allow for practical implementations of the UFO database model. For the same purpose, this paper also discusses an interface of the UFO database model to an extended relation database model, capable of handling some imperfect information, and for which some prototypes are already available.

In [5] propose a method for designing Fuzzy Relational Databases (FRDB) following the extension of the ER model of [15] taking special interest in converting crisp databases into fuzzy ones. The way to do so is to define n linguistic labels as n fuzzy sets over the universe of an attribute. After, each tuple in the crisp entity is transformed to up to n fuzzy tuples in a new entity (or n values in the same tuple).

Each fuzzy tuple (or value) does not store the crisp value but a linguistic label and a grade of membership giving the degree to which the corresponding crisp entity belongs to the new entity. Finally, the crisp entity and the new fuzzy entity are mapped to separate tables.

Their ER model includes fuzzy relationships as relationships with at least one attribute, namely, the membership grade. They propose FERM, a design methodology for mapping a Fuzzy ER data model to an crisp relational database in four steps (constructing a Fuzzy ER data model, transforming it to relational tables, normalization and ensuring correct interpretation of the fuzzy relational operators). They also presented the application of FERM to build a prototype of a fuzzy database for a discreet control system for a semiconductor manufacturing process.

In [5] expand the model presented in [10], focusing on their proposal for the *control processes* example. In each process imprecise values are observed, associated to linguistic labels, and every value involves a process called “DBFuzzifier construct”.

8 Fuzzy Comparison of Some Fuzzy Models

In other section we discussed some conceptual models proposed by other authors. None of these investigations uses a CASE support tool proposed to help in a system design that involves uncertainty.

Table 1. Comparison of some fuzzy models: FEER, ExIFO, Fuzzy ER and FuzzyEER

Fuzzy Models/ Components	ExIFO Yazici y Merdan (1996)	Fuzzy ER Chen (1998)	FEER Ma et al. (2001)	FuzzyEER Urrutia et al. (2003)
1. Fuzzy values in fuzzy attributes	Yes*	Yes*	Yes*	Yes
Type 1	Yes*	Yes*	Yes*	Yes
Type 2	Yes*		Yes*	Yes
Type 3				Yes
Type 4				Yes
2. Fuzzy degree associated to an attribute	Yes*	Yes*	Yes*	Yes
3. Fuzzy degree assoc. to some attributes				Yes
4. Fuzzy degree with its own meaning				Yes
5. Fuzzy degree to the model	Yes*	Yes	Yes*	Yes
6. Fuzzy entities	Yes*	Yes*	Yes*	Yes
7. Fuzzy weak entity (existence)				Yes
8. Fuzzy weak entity (identification)				Yes
9. Fuzzy relationship	Yes*	Yes*	Yes*	Yes

10. Fuzzy aggregation of entities	Yes*	Yes*	Yes*	Yes
11. Fuzzy aggregation of attributes	Yes	Yes	Yes	Yes
12. Fuzzy degree in the specialization	Yes*	Yes	Yes	Yes
13. Fuzzy degree in the subclasses		Yes	Yes	Yes
14. Fuzzy constraints		Yes*		Yes
15. Graphic and CASE Tool				Yes
16. Fuzzy oriented object	Yes		Yes	

Our proposal has a tool called FuzzyCASE, which allows us to model using EER and FuzzyEER tools. It incorporates all the notations shown in this work and in other works related to the FuzzyEER model (like the fuzzy constraints which are not detailed in Table 1, but which have not been dealt with by any author in his publications).

Some of the most important models are those proposed by [13, 21]. Table 1 shows a comparison of the FuzzyEER model with those models. Each cell shows a “Yes” if the model has that component or modeling tool (even if it has another notation). In the opposite case the cell is empty. On the other hand, if the cell has a “Yes*” this means that the component has been confined in that model but with different characteristics than those of the FuzzyEER model, or, its characteristics are limited and more reduced than those of the FuzzyEER model proposed here. In general this difference is caused by the use of different types of domains and treatment of imprecision, or, by a type of degree.

The filling of the Table 1 was based in two real experiences. First of all, in [26,28] we explore the role played by a client in a Real Estate Agency located in Malaga (Spain), where the issues was the degree measures the importance with which a client is “seeking for” or “offering” a property, without taking into account the “similarity” between the two roles. In this requirement we use the four models: FEER, ExIFO, Fuzzy ER and FuzzyEER, but the only that allow representing Type 3 and 4 was last one. Second of all, the Fuzzy Autopoietic Knowledge Management (FAKM) model develop in [15]¹, uses a FuzzyEER model to represent Knowledge attribute Type 1, Type 2, and Type 3 related at the quality control process of the

¹ The aim of the model is to integrate the system theory of living systems, the cybernetic theory of viable systems, and the autopoiesis theory of the autopoietic systems, with the hope of going beyond the knowledge management models that are based, among other things, on Cartesian dualism cognition/action (i.e., on a model of symbolic cognition as the processing of representational information in a knowledge management system). Instead, the FAKM model uses a dualism organization/structure to define an autopoietic system in a socio-technical approach [15].

paper in a manufacturing company located within the Maule Region south of Santiago in Chile, studied in [26].

9 Conclusions and Future Lines

Fuzzy databases have been widely studied with the aim of allowing the storage of imprecise or fuzzy data and the imprecise queries about the existing data [34,35].

However, the application of fuzzy logic to databases has traditionally paid little attention to the problem of the conceptual model. Few investigations study a complete and exhaustive notation of the many characteristics, which may be improved using fuzzy logic. The FuzzyEER model intends to do so and in this study we have focused on the following: types of fuzzy attributes (T1, T2, T3 and T4), fuzzy degrees associated or not with different items and with different meanings, degrees with respect to the model, fuzzy aggregations, fuzzy entities and relationships, fuzzy weak entities and degrees in a specialization.

All these concepts allow to extend the EER model to a fuzzy FuzzyEER model [26]. Therefore it may be stated that a data model which contemplates fuzzy data, allows us to represent a type of data in an information system, which the traditional systems do not deal with and so, this information is lost in such systems. This reduces the risk of obtaining empty answers from queries in the database, since fuzzy logic allows us to use a finer scale of discrimination, as it considers the interval $[0,1]$ instead of the set $\{0,1\}$. In other words, it allows us to recover instances that would not be obtained using a precise method, as they only partially meet the imposed conditions. Furthermore, the set of instances can be ordered according to the level at which it satisfies the conditions. This leads the way for creating queries and operations, which would be non-viable in a traditional system

Some of the FuzzyEER notations may be used in a FSQL (Fuzzy SQL) server, which is an extension of SQL for permitting fuzzy queries and operations [34].

In the Table 1 it has been shown that the model FuzzyEER proposed by the authors of this paper, allows to model using the components of EER a treatment of uncertainty: degree and attributes that generate outlines that model in good part the uncertainty using fuzzy logic.

Finally, we think that this paper gives an interesting overview of the subject, first step to establish a formal approach to deal with uncertainty in fuzzy relational databases. As a future line we hope to get this formal approach, nevertheless, for the moment we have in [15, 26,28] two serious works related.

References

1. M. Umamo and S. Fukami. Fuzzy Relational Algebra for Possibility-Distribution-Fuzzy-Relational Model of Fuzzy Data. Journal of Intelligent Information System, 3, pp. 7-28. (1994)

2. A. Motro. Management of Uncertainty in Database System, Modern Database System the Object Model, Interoperability and Beyond, Won Kim, editor, Addison-wesley publishing Company (1995).
3. A. Zvieli and P. Chen. ER Modeling and Fuzzy Databases. 2nd International Conference on Data Engineering, pp. 320-327. (1986).
4. N. Marín, Pons O., Vila M.A.. Fuzzy Types: "A New Concept of Type for Managing Vague Structures". International Journal of Intelligent Systems, 15, pp. 1061-1085 (2000).
5. N. Chaudhry, J. Moyne J., and E.A. Rundensteiner; An Extended Database Design Methodology for Uncertain Data Management. Information Sciences, 121, pp. 83-112. (1999).
6. B.P. Buckles and F.E. Petry. Uncertainty Models in Information and Database Systems. Information Sciences, 11, pp. 77-87. (1985).
7. G. Chen. Fuzzy Logic in Data Modelling, Semantics Constraints, and Databases Design. The Kluwer International Series on Advances in Database Systems. Series Editor A.K. Elmagarmid, U.S.A (1998).
8. E.E. Kerre and G.Q. Chen. An Overview of Fuzzy Data Models. In Studies in Fuzziness: Fuzziness in Database Management Systems, pp. 23-41. Eds. P. Bosc and J. Kacprzyk. Physica-Verlag (1995).
9. A. Yazici and R. George. Fuzzy Database Modeling. Physica-Verlag (Studies in Fuzziness and Soft Computing), New York (1999).
10. N. Chaudhry, J. Moyne, and E.A. Rundensteiner. A Design Methodology for Databases with Uncertain Data. 7th International Working Conference on Scientific and Statistical Database Management, pp. 32-41, Charlottesville, VA (1994).
11. G.Q. Chen, E.E. Kerre. Extending ER/EER Concepts Towards Fuzzy Conceptual Data Modeling. IEEE International Conference on Fuzzy Systems, 2, pp. 1320-1325 (1998).
12. E.E. Kerre and G. Chen. Fuzzy Data Modeling at a Conceptual Level: Extending ER/EER Concepts. In "Knowledge Management in Fuzzy Databases", pp. 3-11. Eds. O. Pons, M.A. Vila and J. Kacprzyk. Ed. Physica-Verlag (2000).
13. Z.M. Ma, W.J. Zhang, W.Y. Ma, and Q. Chen. Conceptual Design of Fuzzy Object-Oriented Databases Using Extended Entity-Relationship Model. International Journal of Intelligent System, 16(6). pp. 697-711 (2001).
14. E. Ruspini. Imprecision and Uncertainty in the Entity-Relationship Model, In Fuzzy Logic in Knowledge Engineering, Eds. H. Prade and C.V. Negoita. Verlag TUV Rhein-land, pp.18-22. (1986)
15. L. Jiménez. Gestion des connaissances imparfaites dans les organisations industrielles : cas d'une industrie manufacturière en Amérique Latine. Ph. Doctoral Thesis, Institut National Polytechnique de Toulouse, Toulouse, France. <http://ethesis.inp-toulouse.fr/archive/00000140/> (2005)
16. C. Batini, S. Ceri, S. Navathe. Diseño Conceptual de Bases de Datos. Addison_Wesley/Díaz de Santos. (1994).
17. T. Connolly, C. Begg. Database System, a Practical Approach to Design, Implementation and Management. Third edition, Addison Wesley (2001).
18. A. De Miguel, M. Piattini, E. Marcos. Diseño de Bases de Datos Relacionales. Rama (1999).
19. R. Elmasri and S.B. Navathe. Fundamentals of Database Systems. Addison-Wesley, Third Edition (2000).

20. J. Galindo, A. Urrutia, R. Carrasco, and M. Piattini. Fuzzy Constraints Using the Enhanced Entity-Relationship Model. Proceedings published by IEEE-CS Press of XXI International Conference of the Chilean Computer Science Society (SCCC 2001), pp. 86-94. Punta Arenas (Chile). <http://computer.org/proceedings/sccc/1396/13960086abs.htm> (2001).
21. A. Yazici and O. Merdan. Extending IFO Data Model for Uncertain Information. 4th International Conference on Information Processing and Management of Uncertainty, IPMU'96, pp. 1283-1282 (vol. III). Granada (Spain) (1996).
22. A. Yazici and A. Cinar. Conceptual Modeling for the Design of Fuzzy Object Oriented Database. In Knowledge Management in Fuzzy Databases, Eds. O. Pons, M.A.Vila and J. Kacprzyk. Ed. Physica-Verlag, pp. 12-35 (2000).
23. Z.M. Ma, W.J. Zhang, and W.Y. Ma. Extending Object-Oriented Databases for Fuzzy Information Modeling. Information Systems, 29, pp. 421-435 (2004).
24. J. Galindo, A. Urrutia, and M. Piattini. Fuzzy Databases: Modeling, Design and Implementation. Idea Group Publishing Hershey, USA (2005).
25. A. Urrutia, J. Galindo, and L. Jiménez. "Extensión del Modelo Conceptual EER para Representar Tipos de Datos Difusos". I+D Computación, Noviembre de 2002, Volumen 1, número 2, ISSN:1665-238X. <http://www.sd-cenidet.com.mx/Revista>, (México) (2002).
26. A. Urrutia. Definición de un Modelo Conceptual para Bases de Datos Difusas. Ph. Doctoral Thesis, University of Castilla-La Mancha (Spain) (2003).
27. A. Urrutia, J. Galindo, and M. Piattini. *Propuesta de un Modelo Conceptual Difuso*. Libro de Ingeniería de Software, Ediciones Cyted Ritos2, ISBN;84-9602307-9, pp 51-76 (2003).
28. A. Urrutia and J. Galindo. Algunos Aspectos del Modelo Conceptual EER Difuso: Aplicación al Caso de una Agencia Inmobiliaria, XI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF'2002), pp. 359-364. León (Spain) (2002).
29. A. Urrutia, J. Galindo, and M. Piattini. Modeling Data Using Fuzzy Attributes. IEEE Computer Society Press, XXII International Conference of the Chilean Computer Science Society (SCCC 2002), pp. 117-123. Copiapó (Chile). ISBN: 0-7695-1867-2 ISSN: 1522-4902 (2002).
30. J. Galindo, A. Urrutia, M. Piattini: "Fuzzy Aggregations and Fuzzy Specializations in Fuzzy EER Model". Chapter VI in "Advanced Topics in Database Research", Vol. 3, pp. 105-126. Ed. Keng Siau. Idea Group, Inc.. (2004) <http://www.idea-group.com>
31. Galindo J., Urrutia A., Carrasco R., Piattini M.: "Relaxing Constraints in Enhanced Entity-Relationship Models using Fuzzy Quantifiers". IEEE Trans. on Fuzzy Systems. pp. 780-796 (2004).
32. R. M. Vandenberghe. An Extended Entity-Relationship Model for Fuzzy Databases Based on Fuzzy Truth Values, Proceeding of 4th International Fuzzy Systems Association World Congress, IFSA'91, Brussels, pp. 280-283 (1991).
33. N. Van Gysegheem and R. De Caluwe. Overview of the UFO Database Model, Proceeding of EUFIT96, September 2-5, Aachen, Germany, pp.858-862 (1996).
34. J. Galindo. Tratamiento de la Imprecisión en Bases de Datos Relacionales: Extensión del Modelo y Adaptación de los SGBD Actuales. Ph. Doctoral Thesis, University of Granada (Spain). (www.lcc.uma.es) (1999).
35. F.E. Petry. Fuzzy Databases: Principles and Applications (with chapter contribution by Patrick Bosc). International Series in Intelligent Technologies. Ed. H.J. Zimmermann. Kluwer Academic Publ. (KAP) (1996).

