Semantically-driven flexible division in fuzzy object oriented models

Nicolás Marín[1] Carlos Molina[2] Olga Pons[1] M. Amparo Vila[1]

1. Department of Computer Science and AI, University of Granada, 18071, Granada, Spain
2.Department of Informatics, University of Jaén, Campus Las Lagunillas, A3, 23071, Jaén, Spain
Email: nicm@decsai.ugr.es, carlosmo@ujaen.es, {opc,vila}@decsai.ugr.es

Abstract— *Fuzzy Databases have been a fertile research area that has produced a wide variety of remarkable solutions for the storage and manipulation of imperfect information. Proposals can be found in the three most relevant data models: relational, object-oriented, and object-relational. Query capabilities have been especially studied in the context of the relational data model, while in object models many problems continue being a matter of research. In this paper, we focus on the resolution of "division queries" (in its relational sense) in an object oriented data model. As we will see, the presence of fuzzily described objects in the database make necessary to use suitable operators that take into account the resemblance that governs the comparison in the underlying reference universe. We also analyze the role of cardinality in this kind of queries.*

Keywords— Division, Quotient, Fuzzy, Object-Oriented, Object-Relational, Database.

1 Introduction

Fuzzy Set Theory[1] has proved to be an adequate tool in handling real world data when they are affected by imperfections of very different kinds. The application of this theory in order to extend conventional databases has led to the development of fuzzy database systems.

Object-oriented and object-relational database management systems allow the representation of schema when complex relationships make the use of Codd's relational model difficult. The object-oriented data model is more powerful from a modeling point of view, because it incorporates important features such as inheritance and encapsulation. As it was the case with the relational data model in the past, many researchers have recently tried to improve object orientation with the help of fuzzy concepts. As a result, fuzzy object-oriented database models (FOODBM) have appeared [2, 3, 4].

In [5, 6] a framework is presented which allows programmers to handle imprecision in the description of objects. An object-relational implementation of this model can be found in [7].

In this paper, we focus on the study of operators that can be used to solve a special kind of queries in fuzzy object-oriented databases: those queries that, in the relational context, are known as fuzzy divisions. Fuzzy division queries in the relational data model has been deeply studied during the last decades: from the initial works of Dubois et al.[8] till the last works of Bosc et al.[9], the interested reader can find a wide survey of remarkable proposals(e.g. [10, 11, 12]).

This work analyzes how to solve this kind of queries over a fuzzy object model, where the equality is substituted by resemblance as the basis of comparisons. Due to this fact, we have to suitably soften the division operator, so that resemblance can be appropriately taken into account.

The paper is organized as follows: section 2 briefly introduces the fuzzy object-oriented data model that is the basis of our proposal; section 3 describes the type of queries we want to solve in an object-oriented context; section 4 analyzes the problem of considering the resemblance when computing the inclusion associated to division queries; section 5 analyzes the role that the cardinality of the involved sets plays in the resolution of the query; finally, some conclusions are outlined in section 6.

2 A fuzzy object-oriented model

As we have previously mentioned in the introduction, this paper is devoted to the problem of how to solve division queries in a fuzzy object model like the one introduced in [13]. We briefly present in this section a summary of its main characteristics.

As in any object model, the state of an object is equated with a set of attribute values according to its class description. The approach described in [13] considers different types of attribute values in order to give support to the representation of fuzzily described objects. Together with the *precise* values, objects, and *crisp* collections of any conventional object-oriented data model, the fuzzy model permits to use more powerful values, as imprecise labels and fuzzy collections, in order to deal with imprecision in the state of the object.

Fig. 1 shows a description of different types of attribute values that we consider in our approach. That is, the object state can be composed by:

- Precise values: This category of values involves all the classical basic classes that usually appear in an object-oriented data model (e.g. numerical classes, string classes, etc.). Values in these domains are easily represented and compared using conventional built-in data types and the classical set of relational operators.
- Imprecise values: The case of imprecise (atomic) values is a bit more complex. In many cases, linguistic labels[14, 15, 16] are associated to this kind of values, but different types of imprecise values must be considered according to their semantics.
- Objects: the attribute value may be a reference to another object (constituting what is called a complex object).
- Collections: the attribute may be conformed by a set of values or, even, by a set of objects. Imprecision in this kind of attributes may appear at two levels:

ISBN: 978-989-95079-6-8

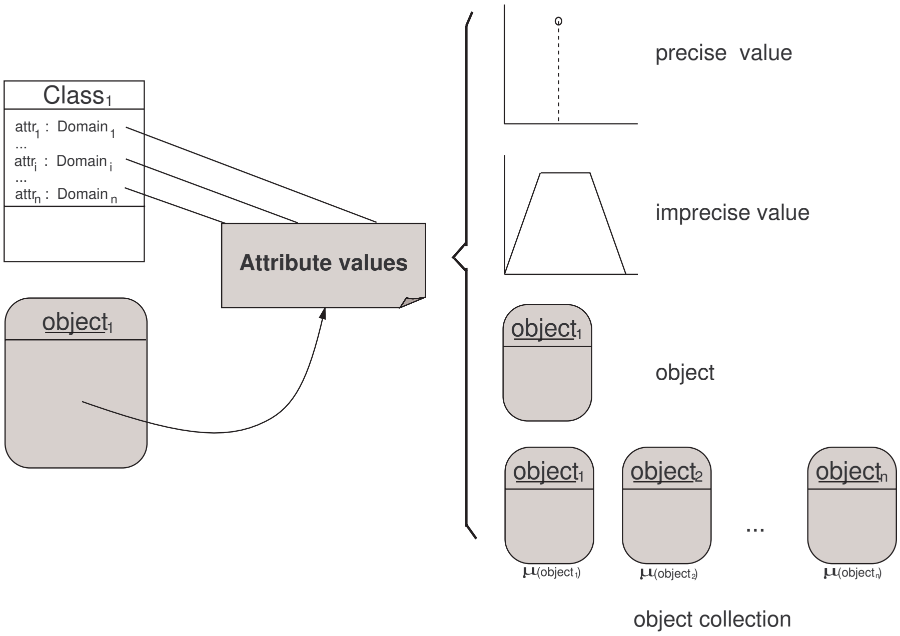

Figure 1: Different kinds of attribute values

- The set may be fuzzy. The semantics of membership degrees depends on the problem, but, in collections, the fuzzy set is considered to be conjunctive. For example, the set of languages a person can speak may be fuzzy if we take into account the degree of command for every language.
- The elements of the set may be fuzzy values or, in general, fuzzily described objects. For example, we can have a set of fuzzily described students.

An example of a fuzzy set of fuzzily described objects may be the collection of friends of a given person, if we use a degree to measure the friendship relation and the friends are described by means of fuzzy attribute values.

2.1 Generalization of equality

We do not only need a representation of fuzzily described objects in our model but also a way to manage this kind of objects. This includes a basic capability to compare the state of two objects belonging to a given class.

With fuzzily described objects like the ones presented in the previous paragraphs, this comparison has to be performed by a process which consists of two steps:

- computing degrees of resemblance for pairs of attribute values, and
- aggregating these resemblance degrees to obtain a general degree of object resemblance.

That is, if o_1 and o_2 are two objects belonging to class C which is characterized by type T_C whose structural component Str_C is an attribute set $\{x_1, x_2, \ldots, x_n\}$, then our goal is to find a resemblance degree between o_1 and o_2 by the aggregation of resemblance degrees of the pairs ($o_1.x_i$, $o_2.x_i$).

$S_{x_i}(o_1, o_2)$ will stand for the resemblance degree observed between the x_i attribute values for objects o_1 and o_2, and $S(o_1, o_2)$ is the aggregated resemblance degree we want to calculate.

In this process, not all attributes need to have the same importance: each attribute x_i has an associated weight $p_{x_i} \in [0, 1]$ that represents its relative importance in the final decision.

Figure 2 summarizes the process of calculating resemblance degrees. In order to compare objects o_1 and o_2, we first compare their attribute values and obtain partial compatibility degrees $S_{x_i}(o_1, o_2)$. Then, we aggregate them to obtain a global resemblance opinion $S(o_1, o_2)$ according to the attribute importance established in the class.

As can be observed, the comparison of two objects involves a recursive procedure. For the sake of space, we omit here the operators for base cases and the analysis of the recursive aggregation procedure used in our model. Interested readers can found a detailed description in [17].

3 Division queries in fuzzy object-oriented databases

Codd defined a set of eight basic operators for his relational model. Some of them are implemented directly in SQL, but some other require particular implementation to fit into the SQL language. Relational division is one of the eight basic operations in Codd's relational algebra[18] that has these requirements.

Suppose that we have two relations t_a and t_b whose schema are, respectively, $T_A(X, Y)$ and $T_B(Y')$, where Y and Y' are compatible attributes. The division $t_a \div t_b$ obtains a new relation t_d with schema $T_D(X)$ and with the following set of tuples:

$$\{a|a \in domain(X) \wedge (\forall b, (b \in t_b) \rightarrow (< a, b > \in t_a)\} \quad (1)$$

Assume now that T_A and T_B are fuzzy relations, i.e., that their tuples are weighted by a number between 0 and 1. That

ISBN: 978-989-95079-6-8

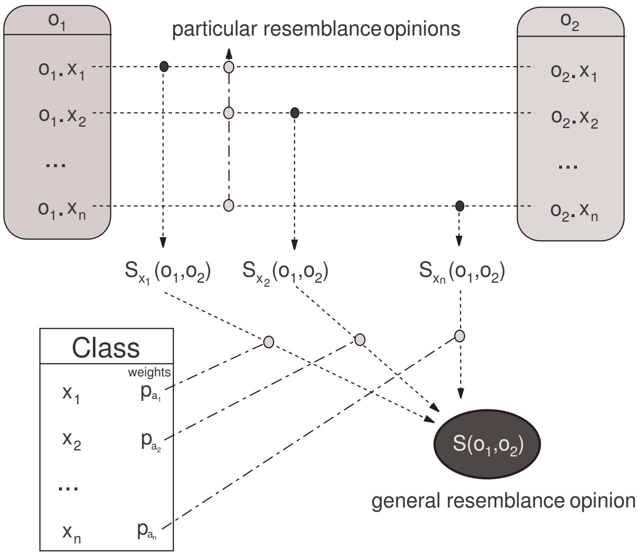

Figure 2: Obtaining the compatibility between two objects.

is, each tuple of t_a is affected by a membership degree, $\mu_{t_a}(< a, b >) \in [0, 1]$ and, similarly, each tuple of t_b has its corresponding $\mu_{t_b}(< b >) \in [0, 1]$.

The membership degree of the tuples of the division result can be obtained as follows [8]:

$$\mu_{t_d}(a) = inf_b\{\mu_{t_b}(b) \rightarrow \mu_{t_a}(a, b)\} \quad (2)$$

In the case of an object-oriented model, we have to consider (in the more general case) that attribute values are objects. Figure 3 depicts the tables with the new situation in an object-oriented context.

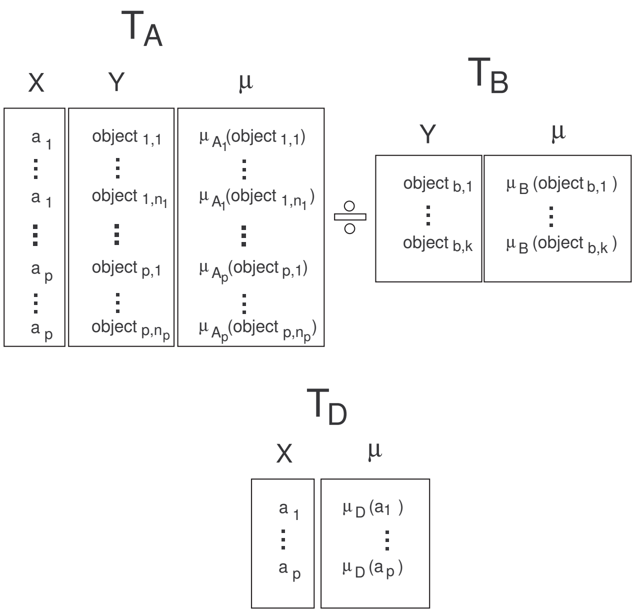

Figure 3: Tables involved in the division

Moreover, as in an object-oriented model attribute values do not necessarily have to be atomic, table T_A (and, similarly T_B) could have the schema showed in Fig. 4.

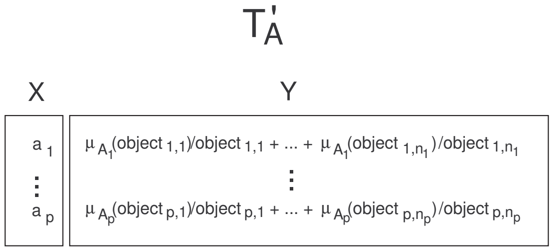

Figure 4: Object-oriented version of T_A

In any case, whatever the schema version of dividend and divisor tables are, the solution for division queries entails the resolution of the inclusion problem described in Fig. 5. Next sections are devoted to analyze this inclusion problem.

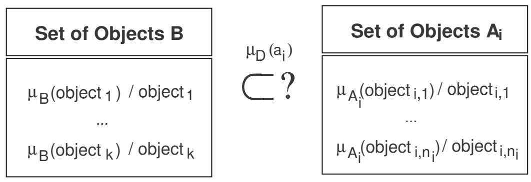

Figure 5: An inclusion problem

4 Two resemblance-based approaches

According to the previous paragraphs, in order to solve division queries in object-oriented models, we need to perform suitable computations of the above mentioned inclusion operator.

To compute the inclusion between any couple of fuzzy sets of objects (like in Fig. 5), we need to generalize the fuzzy inclusion operators, taking into account that the objects of the set may be fuzzily described.

Several proposals for the calculus of this inclusion degree can be found in the literature. In [19] the inclusion degree between two fuzzy sets A and B is calculated as follows:

$$N(B|A) = \min_{u \in U}\{I(\mu_A(u), \mu_B(u))\}, \quad (3)$$

where I stands for an implication operator, and μ_X is the membership function that describes the fuzzy set X. This degree coincides with the fuzzy division operator described in (2). The implication operator can be chosen in accordance to the properties we want the inclusion degree to fulfil [8, 10].

Nevertheless, independently of the chosen implication operator, this formulation supposes that both A and B are defined over a reference universe $\mathcal{U}$ made up of precise elements, where the classical equality is the basis of the comparisons. That is, the implication operator compares to what extent the presence of an element of the universe $\mathcal{U}$ in A forces its presence in B (i.e. we compare the membership degrees of the *same* object to both sets).

However, in the context of fuzzy databases, it frequently happens that the elements of the universe $\mathcal{U}$ are imprecise objects among which classical equality cannot be applied (as we

ISBN: 978-989-95079-6-8

commented in section 2. Instead, a similarity or a (more relaxed) resemblance relationship must be used. That is, for a given element in the set A, it is not clear which element of B has to be taken in order to compare the membership degrees.

In our example of object databases, the objects of the two set of Fig. 5 may be defined with imprecision and two *apparently* different objects (from the identity point of view) may be the *same* one (from the value equality point of view).

That is, if the reference universe $\mathcal{U}$ is formed by fuzzily described elements, a generalized version of (3) that takes into account resemblance among the involved objects must be used. We can consider two distinct ways of doing that, namely, making resemblance acts as a limit of the implication or making resemblance acts as a limit of membership to A_i.

4.1 Resemblance as a limit of the implication

If the reference universe $\mathcal{U}$ is formed by fuzzily described elements, (3) can be generalized as follows [20]:

Definition 1 (Resemblance driven inclusion degree) *Let A and B be two fuzzy sets defined over a finite reference universe $\mathcal{U}$, S be a resemblance relation defined over the elements of $\mathcal{U}$, and $\otimes$ be a t-norm. The inclusion degree of B in A driven by the resemblance relation S is calculated as follows:*

$$\Theta_S(A|B) = \min_{x \in \mathcal{U}} \max_{y \in \mathcal{U}} \theta_{B,A,S}(x,y) \quad (4)$$

where

$$\theta_{B,A,S}(x,y) = \otimes(I(\mu_B(x), \mu_A(y)), \mu_S(x,y)) \quad (5)$$

In (3), the inclusion degree is calculated taking into account the element of $\mathcal{U}$ that fulfils in a lower degree the implication condition between the membership degrees of this element to both sets. In (4), since the membership degrees of similar (but not necessarily equal from the identity point of view) elements can be compared, we restrict the implication using the resemblance degree of the two elements. In summary, for each element that belongs with a certain degree to the set A, we look for a *quite similar* object in $\mathcal{U}$ that belongs to the set B with an equal or higher degree.

Consider the example of Fig. 6.

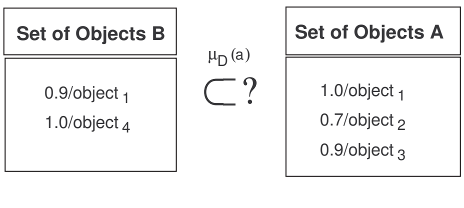

Figure 6: Example

In this example, $\{o_1, o_2, o_3, o_4\}$ is the reference universe $\mathcal{U}$ over which the resemblance relation S of table 1 is defined.

In this situation:

$\Theta_S(A|B)$= min {max {⊗(I($\mu_B(o_1)$, $\mu_A(o_1)$), $\mu_S(o_1,o_1)$), ..., ⊗ (I($\mu_B(o_1)$, $\mu_A(o_4)$), $\mu_S(o_1,o_4)$)}, ..., max{⊗(I($\mu_B(o_4)$, $\mu_A(o_1)$), $\mu_S(o_4,o_1)$), ..., ⊗(I($\mu_B(o_4)$, $\mu_A(o_4)$), $\mu_S(o_4,o_4)$)}}.

If we use the minimum as t-norm and (6) as implication operator, then:

$\Theta_S(A|B)$= min {max {1,0,0,0}, max {0,1,0,0}, max {0,0,1,0.7}, max {0,0,0.7,0}}= min{1,1,1,0.7}=0.7.

Table 1: Resemblance Relation

	o_1	o_2	o_3	o_4
o_1	1.0	0.0	0.0	0.0
o_2		1.0	0.0	0.0
o_3			1.0	0.7
o_4				1.0

$$I(x,y) = \begin{cases} 1, & \text{if } x \leq y \\ y/x, & \text{otherwise} \end{cases} \quad (6)$$

4.2 Resemblance as a limit of relationship

As an alternative to the strategy used in the previous subsection, we can consider that the resemblance only restricts the membership degree of a certain object to the set A.

The assumption of this second version of flexible inclusion is to enlarge the set A into a superset A' obtained by composing A with the resemblance relation as suggested in [9]. The idea is to expand A in the sense that the objects of the reference class $\mathcal{U}$ *similar* to an object initially present in A, are added to A. This resemblance based tolerant inclusion can be defined as follows.

Definition 2 (Resemblance based tolerant inclusion degree) *Let A and B be two fuzzy sets defined over a finite reference universe $\mathcal{U}$, S be a resemblance relation defined over the elements of $\mathcal{U}$, and $\otimes$ be a t-norm. The tolerant inclusion degree of B in A according to the resemblance relation S is calculated as follows:*

$$\Psi_S(A|B) = \min_{x \in \mathcal{U}} \max_{y \in \mathcal{U}} \psi_{B,A,S}(x,y) \quad (7)$$

where

$$\psi_{B,A,S}(x,y) = I(\mu_B(x), \otimes(\mu_A(y), \mu_S(x,y)) \quad (8)$$

According to the example of Fig. 6 and the resemblance relation defined in Table 1:

$\Psi_S(A|B)$= min {max {I($\mu_B(o_1)$, ⊗($\mu_A(o_1)$,$\mu_S(o_1,o_1)$)), , ..., I($\mu_B(o_1)$, ⊗($\mu_A(o_4)$,$\mu_S(o_1,o_4)$))}, ..., max {I($\mu_B(o_4)$, ⊗($\mu_A(o_4)$,$\mu_S(o_4,o_1)$)), ...,I($\mu_B(o_4)$, ⊗($\mu_A(o_4)$,$\mu_S(o_4,o_4)$))}}.

If we use the minimum as t-norm and (6) as implication operator, then:

$\Psi_S(A|B)$= min {max {1,0,0,0}, max {1,1,1,1}, max {1,1,1,1}, max {0,0,0.7,0}}= min{1,1,1,0.7}=0.7.

4.3 Some notes on the two approaches

Although with our example, both flexible implementations of inclusion deliver the same value, normally the behaviour of these alternatives will differ.

An example where the second approach is more optimistic than the first one is described below.

Consider the naive example of Fig. 7.

In this example, $\{o_1, o_2\}$ is the reference universe $\mathcal{U}$ over which the resemblance relation S of table 2 is defined.

If we use the minimum as t-norm and (6) as implication operator, then:

$\Theta_S(A|B)$=0.6

$\Psi_S(A|B)$=1.0

ISBN: 978-989-95079-6-8

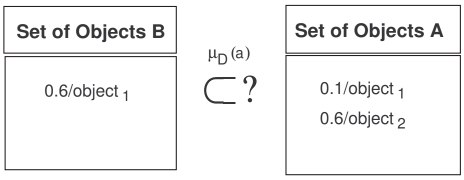

Figure 7: Example

Table 2: Resemblance Relation

	o_1	o_2
o_1	1.0	0.6
o_2		1.0

If we change the resemblance relation as in Table 3 and maintain the rest of data, then
$\Theta_S(A|B)$=0.9
$\Psi_S(A|B)$=1.0

Table 3: Resemblance Relation

	o_1	o_2
o_1	1.0	0.9
o_2		1.0

From a semantic point of view, with this configuration of the *Tolerant Inclusion*, in the computation of $\psi_{B,A,S}(x, y)$, the membership degree to A of a given object y bounds the needed resemblance between x and y. This effect can diminish if we use t-norms more restrictive than the minimum.

5 The role of cardinality

In order to finish our analysis of inclusion operators between fuzzy sets of fuzzily described objects, we now pay attention to cardinality, which may play a relevant role when computing the inclusion for two main reasons: On the first hand, if we soften the inclusion so that *similar* objects can fulfil the implication constraint, we have to be careful with the number of involved objects (the same object of A can be *matched* to more than one object of B); on the other hand, another point where the inclusion operators are able to be softened is to be tolerant with the number of objects of set B that we force to be included in A.

These two reasons lead us to new approaches to the inclusion operator which is supporting the division we want to solve.

5.1 Cardinality as a constraint of similarity

The use of resemblance relations instead of equality when computing inclusion degrees may make the obtained value be high, even if there is a great difference in terms of cardinality between the set of elements of B and the subset of elements of A matched to them during the computation of inclusion.

In some situations, cardinality may not be important. For example, imagine that A and B are two sets of *tools* and that we want to solve the question *Can a person do with tools of A all the tasks he can do with the tools of B?*. In this case, for each tool of B, we look for a tool of A with similar capabilities. In this context, it does not matter if the same tool of A is matched to more than one tool of B. That is, the number of *selected tools of A* is not relevant. However, if we need to assign this set of tools to a group of people that have to work independently, then, the number of *selected tools of A* is relevant and cardinality is important.

If we wish to distinguish between these situations, we need to weight the inclusion degree with a factor that takes into account the distance between the cardinalities of the fuzzy sets that are being compared.

Definition 3 (Cardinality Factor) *Let $\mathfrak{U}$ be a reference universe. Let $|X|$ stands for the crisp cardinality of X. CF : $P(\mathfrak{U}) \times P(\mathfrak{U}) \to [0, 1]$ is a cardinality factor if:*

1. *If* $|X| = |Y|$, $CF(X, Y) = 1$
2. *If* $|X| >= |Y|$, $CF(X, C) >= CF(Y, C)$
3. *If* $|X| >= |Y|$, $CF(C, Y) >= CF(C, X)$

A simple example of CF is the following one:

$$CF(X, Y) = \begin{cases} 1, \text{ if } |X| >= |Y| \\ |X|/|Y|, \text{ otherwise} \end{cases} \tag{9}$$

This CF can be composed with a *relative fuzzy quantifier*[21] to adjust the behavior to the user needs.

Using (9) in the example of Fig. 8 and the resemblance relation of Table 4, we have:
$\Theta_S(A|B)$=1.0
$\Psi_S(A|B)$=1.0
but CF(matched elements of A,B)=0.5.

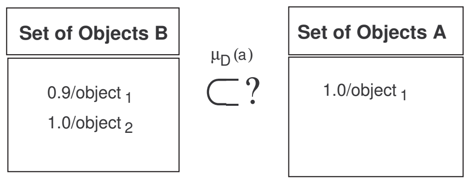

Figure 8: Example

Table 4: Resemblance Relation

	o_1	o_2
o_1	1.0	1.0
o_2		1.0

5.2 Cardinality as a relaxation of the inclusion

The other role cardinality can play in an inclusion operator is to soften the amount of elements of B that have to fulfil the inclusion relation. That is, with this kind of relaxation, the idea is to compute to what extent *almost all* elements of B are included in A. For example, in the case of tools, we want to solve the question *Can a person do with tools of A almost all the tasks he can do with the tools of B?*

There are many possible implementations of this idea in relation to what is called approximate division[8, 12]. One of the possible implementations of the corresponding approximate inclusion is to substitute the infimum in (3) by an OWA operator based on the desired *almost all* quantifier[22]. This way, we can adapt Θ and Ψ operators as follows.

ISBN: 978-989-95079-6-8

Definition 4 (Approximate Θ) *Let A and B be two fuzzy sets defined over a finite reference universe $\mathcal{U}$, S be a resemblance relation defined over the elements of $\mathcal{U}$, $\otimes$ be a t-norm, and $OWA_\mathcal{Q}$ an OWA operator induced by a regular non-decreasing quantifier $\mathcal{Q}$. The approximate inclusion degree of B in A driven by the resemblance relation S according to $OWA_\mathcal{Q}$ is calculated as follows:*

$$\Theta_{\mathcal{Q},S}(A|B) = OWA_\mathcal{Q}(\{\max_{y\in\mathcal{U}} \theta_{B,A,S}(x,y)\}_{x\in\mathcal{U}}) \quad (10)$$

Definition 5 (Approximate Ψ) *Let A and B be two fuzzy sets defined over a finite reference universe $\mathcal{U}$, S be a resemblance relation defined over the elements of $\mathcal{U}$, $\otimes$ be a t-norm, and $\mathcal{Q}$ a relative fuzzy quantifier. The approximate tolerant inclusion degree of B in A based on the resemblance relation S according to $\mathcal{Q}$ is calculated as follows:*

$$\Psi_{\mathcal{Q},S}(A|B) = OWA_\mathcal{Q}(\{\max_{y\in\mathcal{U}} \psi_{B,A,S}(x,y)\}_{x\in\mathcal{U}}) \quad (11)$$

6 Conclusions

In this paper, we have presented an approach to solve queries that are similar to the division queries of the relational data model but, now, in an object-oriented context. The fact that attribute values can be fuzzily described objects make us to extend the division operator by means of resemblance measures. We have considered two ways of incorporating such resemblance measures in the computation of inclusion, namely, restricting membership and restricting inclusion. Additionally, we have analyzed the role that cardinality can play in the division operator. Thus, we have considered to weight the inclusion operators with a cardinality factor that takes into account the number of elements of set A matched during the inclusion analysis. We have also propose two approximate inclusion operators based on the use of a quantifier for relaxing the division condition. With the described survey of operators, the user can semantically adapt the division operators to her/his needs in an object-oriented context.

Acknowledgment

Work partially supported by the projects *Representación y Manipulación de Objetos Imperfectos en Problemas de Integración de Datos: Una Aplicación a los Almacenes de Objetos de Aprendizaje*, Junta de Andalucía (TIC03175), and *Tratamiento de Información Difusa Temporal en un SGBD Relacional: Formulación Teórica, Implementación y Aplicaciones*, Estado Español (TIN2008-02066).

References

[1] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[2] Rita De Caluwe. *Fuzzy and Uncertain Object-Oriented Databases: Concepts and Models*, volume 13 of *Advances in Fuzzy Systems-Applications and Theory*. World Scientific, 1997.

[3] J. Lee, J-Y. Kuo, and N-L. Xue. A note on current approaches to extent fuzzy logic to object oriented modeling. *International Journal of Intelligent Systems*, 16:807–820, 2001.

[4] Zongmin Ma. *Advances In Fuzzy Object-oriented Databases: Modeling And Applications*. IGI Publishing, Hershey, PA, USA, 2004.

[5] N. Marín, O. Pons, and M. A. Vila. Fuzzy types: A new concept of type for managing vague structures. *International Journal of Intelligent Systems*, 15:1061–1085, 2000.

[6] F. Berzal, N. Marín, O. Pons, and M. A. Vila. Development of applications with fuzzy objects in modern programming platforms: Research articles. *Int. J. Intell. Syst.*, 20(11):1117–1136, 2005.

[7] L. Cuevas, N. Marín, O. Pons, and M. A. Vila. pg4db: A fuzzy object-relational system. *Fuzzy Sets Syst.*, 159(12):1500–1514, 2008.

[8] Didier Dubois and Henri Prade. Semantics of quotient operators in fuzzy relational databases. *Fuzzy Sets and Systems*, 78:89–93, 1996.

[9] P. Bosc, A. Hadjali, and O. Pivert. Exceptions and resemblance: Two keys for tolerant division operators. *Fuzzy Information Processing Society, 2008. NAFIPS 2008. Annual Meeting of the North American*, pages 1–6, May 2008.

[10] Patrick Bosc, Didier Dubois, Olivier Pivert, and Henri Prade. Flexible queries in relational databases—the example of the division operator. *Theor. Comput. Sci.*, 171(1-2):281–302, 1997.

[11] Xiaohui Tang and Guoqing Chen. A complete set of fuzzy relational algebraic operators in fuzzy relational databases. *Fuzzy Systems, 2004. Proceedings. 2004 IEEE International Conference on*, 1:565–569 vol.1, July 2004.

[12] P. Bosc, O. Pivert, and D. Rocacher. About quotient and division of crisp and fuzzy relations. *J. Intell. Inf. Syst.*, 29(2):185–210, 2007.

[13] F. Berzal, N. Marín, O. Pons, and M. A. Vila. Managing fuzziness on conventional object-oriented platforms: Research articles. *Int. J. Intell. Syst.*, 22(7):781–803, 2007.

[14] L. A. Zadeh. The concept of linguistic variable and its application to aproximate reasoning I. *Information Sciences*, 8:199–251, 1975.

[15] L. A. Zadeh. The concept of linguistic variable and its application to aproximate reasoning II. *Information Sciences*, 8:301–357, 1975.

[16] L. A. Zadeh. The concept of linguistic variable and its application to aproximate reasoning III. *Information Sciences*, 9:43–80, 1975.

[17] Fernando Berzal, Juan C. Cubero, Nicolás Marín, María Amparo Vila Miranda, Janusz Kacprzyk, and Slawomir Zadrozny. A general framework for computing with words in object-oriented programming. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 15(Supplement-1):111–131, 2007.

[18] E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387, 1970.

[19] J-P. Rossazza, D. Dubois, and H. Prade. A hierarchical model of fuzzy classes. In *Fuzzy and Uncertain Object-Oriented Databases. Concepts and Models.*, volume 13, pages 21–61. Advances in Fuzzy Systems- Applications and Theory, 1998.

[20] N. Marín, J. M. Medina, O. Pons, D. Sánchez, and M. A. Vila. Complex object comparison in a fuzzy context. *Information and Software Technology*, 45(7):431 – 444, 2003. Software Engineering with Computational Intelligence S.I.

[21] Lotfi A. Zadeh. A computational approach to fuzzy quantifiers in natural languages. *Computers and Mathematics with Applications*, 9(1):149 – 184, 1983.

[22] R. R. Yager. Families of owa operators. *Fuzzy Sets and Systems*, (59):125–148, 1993.