Representation of weakly structured imprecise data for fuzzy querying

Rallou Thomopoulos, Patrice Buche, Ollivier Haemmerlé

Representation of weakly structured imprecise data for fuzzy querying

Rallou Thomopoulos[a], Patrice Buche[a] and Ollivier Haemmerlé[a]

[a]INA P-G, UER d'informatique/INRA BIA, 16 rue Claude Bernard, 75231 Paris Cedex 5, France
Tel : +33 1 44 08 16 79, +33 1 44 08 16 75, +33 1 44 08 72 29. Fax : +33 1 44 08 16 66
E-mail : {Rallou.Thomopoulos,Patrice.Buche,Ollivier.Haemmerle}@inapg.inra.fr

In the present paper we introduce an extension of the conceptual graph model suitable to the representation of data which are modelized using fuzzy sets. We extend the specialization relation of the conceptual graph model to fuzzy conceptual graphs. Lastly we introduce a new way of comparing conceptual graphs, using the idea that a graph may be compatible with another graph with a given degree d, which allows to make more flexible comparisons of fuzzy conceptual graphs. This work takes place within a project that aims at building a tool for the analysis of microbial risks in food products.

Keywords: Fuzzy databases, Conceptual graphs, Imprecise data, Soft querying, Microbiology.

1. INTRODUCTION

Our research project is part of a national programme which aims at building a tool for the analysis of microbial risks in food products. We are concerned with the storage and the querying of data that come from the bibliography of microbiology. These data have several particularities: (i) they are polymorphic information in a field that is continuously growing; we call them "weakly structured data"; (ii) they are often imprecise because of the complexity of the biological processes involved; (iii) they are not exhaustive, as the bibliography does not cover all possible experimental factors and conditions. These particularities have the following respective consequences: (i) it is difficult to determine a classic database schema to store all the useful information; (ii) it is necessary to represent imprecise information; (iii) it is necessary to enlarge the querying in order to provide close answers when the exact information is missing.

The approach we chose consists in designing a unified querying system (called UQS) that simultaneously scans two separate bases : a relational database containing the structured information, and a conceptual graph knowledge base containing the data that do not fit in the structure of the relational database. The justification and the structure of the unified querying system have already been presented in [2]. To retrieve information from the conceptual graph knowledge base, the user's query is translated into a conceptual graph which is used to scan the knowledge base. In this paper, our objective is to extend the conceptual graph model in order to be able to represent imprecise data - including numerical values - and enlarged queries.

Classically the conceptual graph model allows one to represent symbolic data [16]. A numerical value cannot be represented otherwise than symbolic data. We propose a way of introducing a numerical domain of values within the framework of the basic conceptual graph model.

Concerning enlarged querying and imprecise information management, the bibliography in the database framework covers two kinds of problems. In a first category of papers, the fuzzy set framework has been shown to be a sound scientific way of modelling flexible queries [1]. In the second category of papers, the fuzzy set framework has also been proposed to represent imprecise values by means of possibility distributions [14].

Besides, the introduction of the fuzzy set theory into the conceptual graph model has been studied by Morton [10] and extended by several works such as [17,3]. Compared to the previous approaches, we propose a more homogeneous and integrated way of combining conceptual graphs and fuzzy sets: (i) we propose a homogeneous representation of fuzzy types[1] and fuzzy markers[1]; (ii) the domain of these fuzzy sets is built in accordance with the support[1].

Combining a knowledge representation model and a way of introducing imprecision has been proposed in other previous works. In particular, we can cite formalisms that describe ontologies like the object model [7], or information retrieval using terminological logics [15]. The latter are part of the "knowledge representation" subfield of artificial intelligence and more specifically semantic networks, just as the conceptual graph model.

The original contribution of this paper is thus mainly to provide an extension of the conceptual graph model suitable to the representation of imprecise data and enlarged queries, by using the fuzzy set framework and by proposing a mechanism allowing a flexible comparison of conceptual graphs; and secondly to propose a natural way of representing numerical values within the basic conceptual graph model.

Section 2 briefly presents the representation models that we use, i.e. what we use fuzzy sets for, and what the conceptual graph model is. Section 3 describes our choice for the representation of numerical values in the conceptual graph model, and the extension that we propose for the representation of fuzzy values. In section 4 we extend the specialization relation in order to allow comparisons of conceptual graphs that contain fuzzy concepts.

2. PRELIMINARY NOTIONS

2.1. Fuzzy sets

In our application we need firstly to be able to represent imprecise data, secondly to use enlarged querying. To perform this we use the fuzzy set theory [18].

Definition 1 *A* **fuzzy set** *A on a domain X is defined by a membership function μ_A from X to [0, 1] that associates with each element x of X the degree to which x belongs to A.*

[1] These notions are explained in Section 2

The domain X may be continuous or discrete. These two cases are illustrated by the examples given in Figure 1. The fuzzy set *MyMilkProductPreferences* is also noted :
1/Full milk + 0.5/Half-skimmed milk.

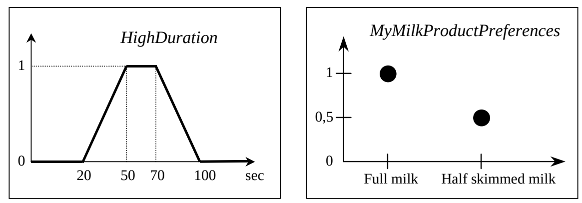

Figure 1. Fuzzy sets *HighDuration* and *MyMilkProductPreferences*

A fuzzy set may be interpreted in two ways:

1. as the expression of preferences on the domain of a selection criterion. For example the fuzzy set *HighDuration* in Figure 1 may be interpreted as a preference concerning the required value of the criterion *Duration*: a duration between 50 and 70 seconds is fully satisfactory, values outside this interval may also be acceptable, but with smaller preference degrees;

2. as an imprecise datum represented by a possibility distribution. For example the fuzzy set *MyMilkProductPreferences* may be interpreted as an imprecise datum if the kind of milk that was used in the experiment is not clearly known: it is very likely to be full milk, but half-skimmed milk is not excluded.

Of course either a continuous or a discrete domain can be used to express a preference as well as an imprecise datum.

In our application, "imprecise data" refer to:

- data known with a given variability, e.g. a concentration measure can take different values if we make the same experiment several times, because of the complexity of the underlying biological processes. This measure is not to be represented by a precise value, but by a minimum-maximum interval of values, e.g. [49.8 U/ml, 51.1 U/ml], corresponding to the extrema of the obtained results;

- data whose precision is limited by the measuring techniques. For example by using a method able to detect bacteria beyond a given concentration threshold (e.g. 10^2 cells per gramme), not detecting any bacterium means that their concentration is below this threshold. This imprecise value is noted "$< 10^2$ cells/g";
- vague data, like "in products having a weak water activity (a_w), microorganisms with spores can appear". In this example [20] the piece of information "weak water activity" may be represented by a fuzzy set.

The fuzzy set framework allows one to represent a precise value, an interval or a fuzzy value using the same formalism.

2.2. The conceptual graph model

The weakly structured data of the application are represented using the conceptual graph model, which is a knowledge representation model based on labelled graphs, introduced by Sowa [16]. We use the formalization presented in [13]. In the conceptual graph model, knowledge is divided into two parts: the terminological part (the support) and the assertional part (the conceptual graphs). In this section, we briefly and intuitively present the conceptual graph model through the example of our application.

The support

The support provides the ground vocabulary used to build the knowledge base: the types of concepts used, the instances of these types, and the types of relations linking the concepts. It describes the hierarchical organization of these elements.

The set of concept types is partially ordered by a *kind of* relation. *Universal* and *Absurd* are respectively its greatest and lowest elements. Figure 2 presents a part of the set of concept types used in the application.

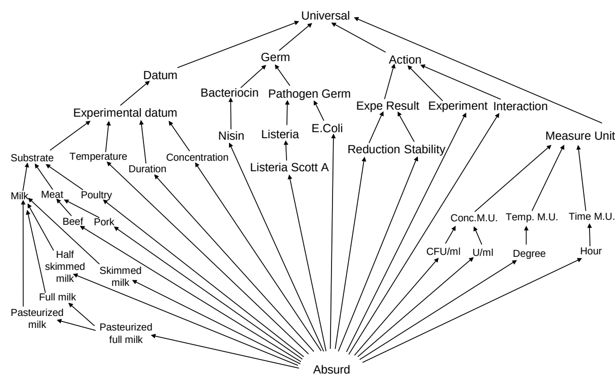

Figure 2. A part of the concept type set for the microbial application

The concepts can be linked by means of relations. The set of relation types is partially ordered by a *kind of* relation. Each relation type is characterized by an arity, and a signature which specifies the maximal concept types that a given relation can link together. The set of relation types we use contains relation types such as *Agt*, which is a binary relation having *(Action, Germ)* as a signature. It means that "an Action has for agent a Germ" (for example an interaction can have a bacterium as an agent).

The third set of the support is the set of individual markers. Each individual marker represents an instance of a concept. For example, *Celsius degree* can be an instance of *Degree*. The generic marker (noted $*$) is a particular marker referring to an unspecified instance of a concept.

The conceptual graphs

The conceptual graphs, built upon the support, express the factual knowledge. They are composed of two kinds of vertices: (i) the *concept vertices* (noted in rectangles or in brackets) which represent the entities, attributes, states, events; (ii) the *relation vertices* (noted in ovals or in parentheses) which express the nature of the relations between concepts.

The *label* of a concept vertex is a pair defined by the type of the concept and a marker (individual or generic) of this type. The label of a relation vertex is its relation type.

The information contained in the conceptual graph knowledge base describes the behaviour of pathogen germs (increase, reduction or stability of their concentration) in food products during different processes. For example, the conceptual graph given in Figure 3 is a representation of the information: "the experiment E1 carries out an interaction I1 between Nisin and Listeria Scott A in full milk and the result is reduction".

Definition 2 *The knowledge base* $KB = \{G_1, \ldots, G_p\}$ *containing the weakly structured knowledge of our system is a set of connected, possibly cyclic conceptual graphs.*

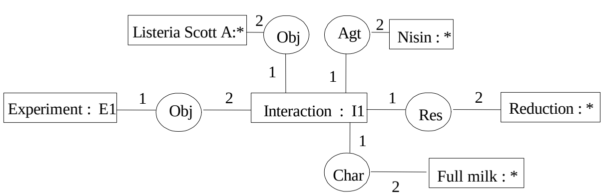

Figure 3. An example of a conceptual graph

Specialization relation, projection operation

The set of conceptual graphs is partially ordered by the specialization relation (noted $\leq$), which can be computed by the projection operation (a kind of graph morphism allowing a restriction of the vertex labels authorized in the support): $G' \leq G$ if and only if there is a projection of G into G'. An example is given in Figure 4.

Since it allows the search for conceptual graphs which are specializations of (which contain more precise information than) another conceptual graph, the projection operation is widely used for the querying of conceptual graph knowledge bases. We then call a "query graph" a conceptual graph that we try to project into each graph of the knowledge base, called "factual graphs".

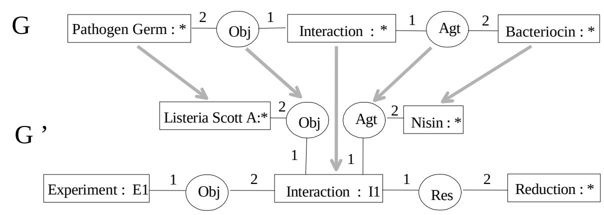

Figure 4. There is a projection from G into G', $G' \leq G$ (G' is a specialization of G)

The question of the existence of a projection of a graph into another graph is NP-complete [11]. However there are polynomial cases, for instance the question of the existence of a projection of an acyclic graph into a graph. We use the polynomial algorithm of [12], which means that we have to use necessarily acyclic query graphs.

3. REPRESENTING NUMERICAL VALUES AND FUZZY VALUES IN THE CONCEPTUAL GRAPH MODEL

3.1. Representing numerical values

The microbiological data stored, as well as the user's queries, include numerical values, like temperatures, concentrations, durations. In the conceptual graph model that we use [13], individual markers are identifiers for instances: an individual marker is a symbolic datum that identifies a given instance in a unique way. Two different instances are necessarily noted by two different individual markers so there is no ambiguity.

As implied by the definition of the model, two incompatible concept types[1] cannot have a com-

[1] With the term "incompatible" we mean two types whose greatest common subtype is *Absurd*

mon instance and therefore cannot share a common individual marker. For instance, let us suppose that the type *Full milk* and the type *Pasteurized milk* have a non-absurd greatest common subtype *Pasteurized full milk*. If 'sample1' is an individual marker of the concept type *Full milk* and also of the concept type *Pasteurized milk*, then it is necessarily a marker of *Pasteurized full milk*. Now let us consider the types *Duration* and *Temperature*. As they have no greatest common subtype different from *Absurd*, they cannot share a common marker. Thus '30' cannot be a marker of both *Duration* and *Temperature*, neither can any numerical value be a marker of several concept types if these types do not have a non-absurd greatest common subtype.

We propose to adopt another representation of numerical values, based on a different support. This representation is in conformity with the basic conceptual graph model.

Here are two different examples proposed by Sowa [16] to represent numerical values. Sowa deals with the representation of measures, where he distinguishes the object on which the measure is made, the parameter that is measured, the measure itself and its name. For instance the measure of the length of a bar of 25.4 cm is represented by:

[BAR]→(CHRC)→[LENGTH]→(MEAS)→[MEASURE]→(NAME)→["25.4 cm"]

contracted to:

[BAR]→(CHRC)→[LENGTH : @25.4 cm].

The drawback of this representation is that the measure appears as a string in which the value and the unit are not distinguished. Besides, Sowa [16] deals with the representation of numbers in a different way. He proposes to distinguish the number itself and the names assigned to it. For example the following graph presents two possible names for the number *four*:

["IV"]←(NAME)←[NUMBER : #27018]→(NAME)→["4"].

The use of a distinct representation for numbers and measures does not highlight the link between a number and a measure, although a measure can contain a number, as in the previous example. Moreover it does not allow one to handle typed data (strings, numerical values, ...), which we wish to introduce in our application so as to be able to perform numerical processing, in particular numerical comparisons and calculations.

Therefore in order to represent numerical values, we propose to introduce the concept type *NumericalValue* into the support. It is a subtype of the more general type *Value*. We introduce the relation type *NumVal(Datum, NumericalValue)*, subtype of the more general relation type *Val(Datum, Value)*.

Definition 3 *A numerical value is a marker of a specific concept type. The set of markers associated with this type may be uncountable*[2]*.*

This concept type is called *NumericalValue* in our application. Such a marker is represented by an integer or a real number in a conceptual graph. In the following, the set of markers associated with the type *NumericalValue* is assumed to be $\mathbb{R}$.

The designation of these types, as well as the signatures of the relation types introduced, are given as an example and can be modified and adapted to other applications. Other subtypes of the concept type *Value* and the relation type *Val* may also be considered and organized into a hierarchy, such as strings, real numbers, integers and so on.

The conceptual graph of Figure 5 extends Figure 3 with additional information, including numerical values represented on the basis of the new support. It can be interpreted as "the experiment E1 carries out an interaction I1 between Nisin at a concentration of 50 U/ml and Listeria Scott A in skimmed milk during 2 hours at a temperature of 37 degrees and the result is reduction" [9].

Let us note that the specialization relation remains unchanged by the introduction of numerical markers: * is more general than all the individual markers - including numerical ones - which are not comparable.

3.2. Representing fuzzy values

We propose to introduce the representation of fuzzy values concerning both concept types and markers.

[2] This is an exception to the definition of the support as established by the definition of [13]

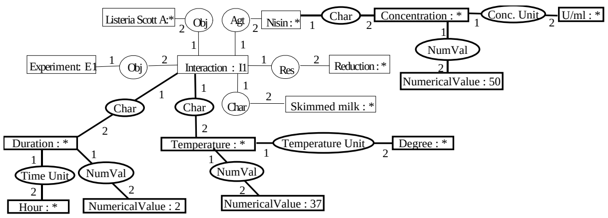

Figure 5. An example of a conceptual graph representing numerical values

Information of the application stored in conceptual graphs (factual graphs or query graphs) may be represented in two ways: (i) as individual markers; for instance this is the case for numerical values (30, 50, etc.); (ii) as concept types; for instance this is the case for substrates (Milk, Beef, etc.). In both cases, we must be able to represent them as fuzzy information, as explained in Section 2. It is thus necessary to define both fuzzy types and fuzzy markers.

Morton [10] firstly introduced fuzziness in the conceptual graph model. He distinguished perceptual, propositional and linguistic fuzziness, respectively concerning entity, attribute, and information concepts. Perceptual fuzziness represents the compatibility between an individual marker and its type within an entity concept vertex. It is materialized by a compatibility degree, for instance [GIRL : Sue | 0.6] expresses a doubt about Sue being a girl. Propositional fuzziness is represented by a truth degree or a fuzzy truth value associated with one or several conceptual graphs defining a statement. Linguistic fuzziness concerns metric attributes, which can have either a precise measure or a label that stands for a crisp or fuzzy subset of what is called the "universe of discourse".

In [17], linguistic fuzziness is proposed for non-metric attributes, and fuzzy relation concepts are introduced, by associating a certainty degree to relations. For example: [GIRL:Sue | 0.6]←(AGNT | 0.5)←[EAT:#80]→(OBJ)→(PIE) means, according to the authors, that it is not certain whether it is a girl (probably called Sue) who performs the eating. The interpretation of such fuzzy propositions seams unclear and different cases are hard to distinguish, for instance "it is not certain that Sue is a girl" should be different from "it is not certain that the girl in question is Sue", from "it is not certain that it is a girl", from "it is not certain that she is eating", from it is not certain that she is doing something" and so on.

In our work, the semantics of fuzzy markers is that of Morton's linguistic fuzziness. Metric and non-metric concepts are not distinguished as they are treated homogeneously, and the "universe of discourse" is clearly defined as part of the set of individual markers defined in the support of the conceptual graph model. We do not handle fuzzy relations, as in our context fuzziness concerns the data and not the way they are linked. We focus on a homogeneous approach of both concept types and markers. In both cases, fuzziness is represented in the same way, by means of a normalized fuzzy set.

In [3], the notion of conjonctive fuzzy type is proposed, which is a conjonction of types associated with the same individual marker with different fuzzy truth values), e.g. {(Tall_man, *true*), (Young_man, *very false*)}.

In our approach, using fuzzy types, we do not question the unicity of an individual marker's type: in our representation a fuzzy type represents a disjunction of possible types (with different possibility degrees), e.g. (1/Full milk + 0.5/Half skimmed milk), associated with the generic marker, which is different from a conjonctive fuzzy type as proposed in [3].

Definition 4 *The* **reference domain** *Ref(t) associated with the concept type t is the set of individual markers that conform to t.*

$$\forall t \in T_C, Ref(t) = \{m \in I \mid \tau(m) \leq t\}$$

where T_C is the set of concept types defined in the support, I is the set of individual markers and τ an application from I to T_C that associates a minimum concept type with each individual marker.

The reference domain associated with a concept type is thus a subset of I. It may be finite or infinite, continuous or discrete. For example, if the markers that conform to the concept type *NumericalValue* are the real numbers, then *Ref(NumericalValue)*=$\mathbb{R}$ is continuous and infinite. If there are two individual markers *T1* and *T2* that conform to the concept type *Temperature*, then *Ref(Temperature)* = {*T1, T2*} is discontinuous and discrete.

Definition 5 *A* **fuzzy marker** m_f *of concept type* t *is a fuzzy set defined on Ref(t).*

It represents a disjunction of individual markers of type t modified by a coefficient between 0 and 1.

Remark 1 *A "classic" individual marker* m *of type* t *can be considered as a particular fuzzy marker: its membership function associates the value 1 with* m*, and the value 0 with the rest of the domain Ref(t). The generic marker * can be considered as a particular fuzzy marker of type* t*: its membership function associates the value 1 with any element of Ref(t).*

Definition 6 *A* **concept with a fuzzy marker** *is a concept vertex whose label is a pair* (t, m_f)*, where* t *is an element of* T_C *and* m_f *is a fuzzy marker of the concept type* t*.*

The conceptual graph represented in Figure 6 includes a concept with a fuzzy marker, of type *NumericalValue*.

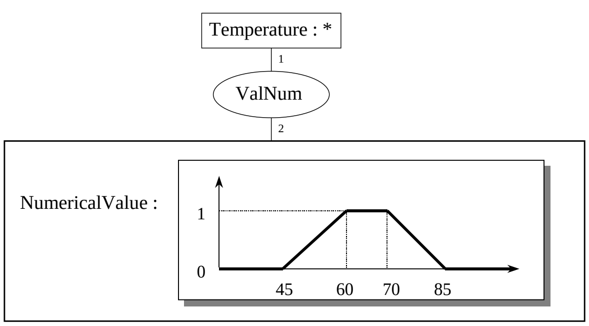

Figure 6. An example of a concept with a fuzzy marker

Definition 7 *A* **fuzzy type** t_f *is a fuzzy set defined on a subset* D_{t_f} *of incomparable*[3] *concept types of* T_C*.*

For example the fuzzy set *MyMilkProductPreferences* represented in Figure 1 is a fuzzy type defined on a subset of the concept types given in Figure 2.

Remark 2 *A "classic" concept type* t *can be considered as a particular fuzzy type. Its membership function is defined on one element* $\{t\}$ *and takes the value 1 for this element.*

Definition 8 *Let* t_f *be a fuzzy type defined on* D_{t_f}*. The* **reference domain** *Ref(*t_f*) associated with the fuzzy type* t_f *is the union of the reference domains of the elements of* D_{t_f}*:*

$$\mathit{Ref}(t_f) = \bigcup_{t \in D_{t_f}} \mathit{Ref}(t)$$

For example the reference domain of the fuzzy type *MyMilkProductPreferences* is the set of markers that conform to the type *Full milk* or to the type *Half-skimmed milk*.

Definition 9 *A* **concept with a fuzzy type** *is a concept vertex whose label is a pair* (t_f, m)*, where* t_f *is a fuzzy type and* m *is the generic marker *.*

Remark 3 *The generic marker * can once again be considered as the fuzzy marker defined on Ref(*t_f*) whose membership function associates the value 1 with any element of Ref(*t_f*).*

For instance, let us suppose that the user's preferences concerning the substrate are *MyMilkProductPreferences* represented in Figure 1. In conceptual graph terms, this substrate is the concept [Full milk : *] with the degree 1, or the concept [Half-skimmed milk : *] with the degree 0.5, which is represented by the concept with a fuzzy type of Figure 7.

The use of fuzzy types does not question the unicity of an individual marker's type: in our representation a fuzzy type represents a weighted

[3] within the meaning of the specialization relation

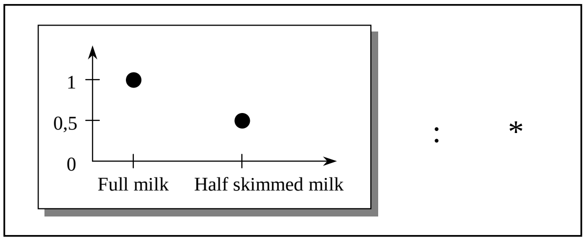

Figure 7. An example of a concept with a fuzzy type

disjunction of possible types, associated with the generic marker, e.g. [(1/Full milk + 0.5/Half-skimmed milk) : *]. This is different from a conjunctive fuzzy type as proposed in [3], which is a conjunction of types (with different fuzzy truth values) associated with the same individual marker, e.g. {(Tall_man, *true*), (Young_man, *very false*)}.

4. COMPARISON OF FUZZY CONCEPTS, THE SPECIALIZATION RELATION

The specialization relation of the conceptual graph model, presented in Section 2, allows one to perform comparisons of conceptual graphs. After having extended the model to represent fuzzy concepts (concepts with a fuzzy marker or with a fuzzy type), the next step is to be able to order conceptual graphs that include fuzzy concepts (called "fuzzy graphs"), and in particular to be able to compare a fuzzy query graph with fuzzy factual graphs. To perform this comparison, we extend the specialization relation to fuzzy concepts, then we propose to relax this comparison, which is an all-or-nothing process, by introducing a more flexible comparison that effects fuzzy querying.

4.1. The notion of specialization for fuzzy sets

The notion of specialization for fuzzy sets is based on the inclusion relation: A is a specialization of B if and only if A is included in B. An example is given in Figure 8 on a continuous domain. This definition applies to both discrete and continuous domains.

Definition 10 *Let A and B be two fuzzy sets defined on a domain X. A is included in B (noted $A \subseteq B$) if and only if their membership functions μ_A and μ_B satisfy the condition:*

$$\forall x \in X, \mu_A(x) \leq \mu_B(x).$$

Let F(X) be the set of all possible fuzzy sets on the domain X. Inclusion is a partial order relation in F(X).

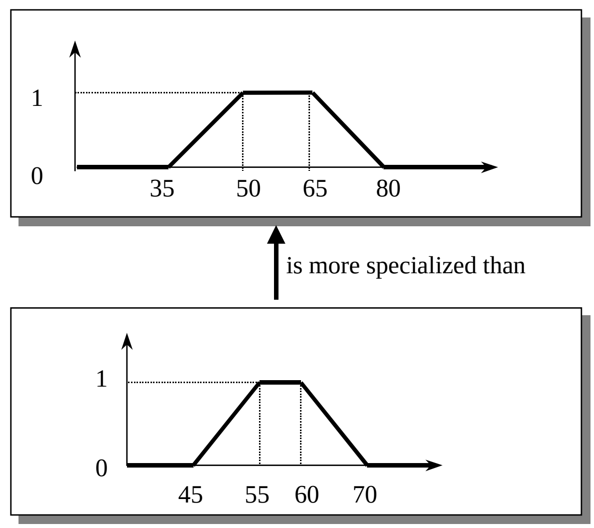

Figure 8. Example of specialization for fuzzy sets

4.2. Extension of the specialization relation to fuzzy concepts

Definition 11 *Let t and t' be two fuzzy types on the domains D_t and $D_{t'}$ respectively. Their characteristic functions are noted χ_t and $\chi_{t'}$. t' is a specialization of t if and only if:*
$\forall x' \in D_{t'}$ $(\chi_{t'}(x') \neq 0)$, $\exists x \in D_t$, $x' \leq x$ *and* $\chi_{t'}(x') \leq \chi_t(x)$.

An example of a projection involving fuzzy types is given in Figure 9.

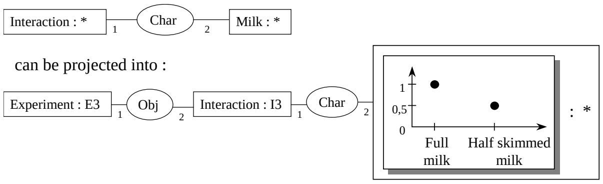

Figure 9. An example of a projection involving fuzzy types

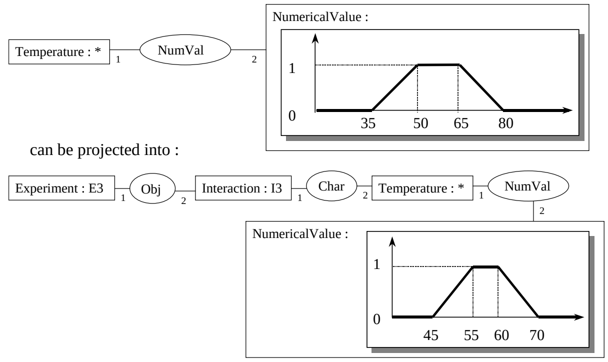

Figure 10. An example of a projection involving fuzzy markers

Remark 4 *If t and t' are "classic" types, this definition is in agreement with the classic specialization relation: t (resp. t') is represented by the fuzzy set defined on $\{t\}$ (resp. $\{t'\}$) that associates the value 1 with t (resp. t'). We still have: t' is a specialization of t if and only if $t' \leq t$.*

Definition 12 *Let m and m' be two markers of types t and t', defined on Ref(t) and Ref(t') respectively. m' is a specialization of m if and only if Ref(t) is included in Ref(t') and $m' \subseteq m$, where $\subseteq$ is the classic inclusion relation defined for fuzzy sets.*

An example of a projection involving fuzzy markers is given in Figure 10.

Note that in Definition 12 there are four possible cases for m (resp. m'). m (resp. m') can be: an individual marker of a simple type; a fuzzy marker of a simple type; a generic marker of a simple type; a generic marker of a fuzzy type.

If m and m' are two individual markers (of the simple types t and t', $t' \leq t$), this definition is in agreement with the classic specialization relation: m (resp. m') is represented by the fuzzy set that associates the value 1 with m (resp. m') and 0 with the rest of Ref(t) (resp. Ref(t')). Then m' is a specialization of m iff $m' \subseteq m$, that is iff $m' = m$.

If m is the generic marker (of a simple or a fuzzy type t) and m' an individual marker (of a simple type t', $t' \leq t$), we also have the classic specialization relation: m is represented by the fuzzy set that associates the value 1 with any element of Ref(t), m' is represented by the fuzzy set that associates the value 1 with m' and 0 with the rest of Ref(t'). Then m' is a specialization of m because $m' \subseteq m$ is always true.

Let us consider two fuzzy types, t defined on a set of n simple types, and t' defined on a set of n' simple types. The checking of the inclusion of a concept with the fuzzy type t' in a concept with the fuzzy type t, has a complexity in $O(n' \times n)$. Similarly, if we consider two fuzzy markers, m defined on a discrete domain composed of n individual markers, and m' defined on a discrete domain composed of n' individual markers, the checking of the inclusion of a concept with the fuzzy marker m' in a concept with the fuzzy marker m also has a complexity in $O(n' \times n)$. In the case where m and m' are defined on a continuous domain, in order to avoid a significant increase of the complexity, we have chosen to limit the fuzzy sets used to "trapezoidal" ones: such a trapezoidal membership function has five phases, limited by four abscissa values (a, b, c, d). It takes the value 0 until a, then increases to 1 from a to b, keeps the value 1 from b to c, decreases to 0 from c to d, and keeps the value 0 from d. Checking the inclusion can then be done in constant time.

Definition 13 *Let $l = (t, m)$ and $l' = (t', m')$ be the labels of two concepts, where t and t' can be fuzzy types, m and m' can be fuzzy markers (we recall that a type and its marker cannot be fuzzy simultaneously). Then l' is a specialization of l if and only if t' is a specialization of t and m' is a specialization of m.*

Property 1 *This extended projection operation remains a partial preorder on the set of conceptual graphs (with possibly fuzzy concepts).*

Proof 1 *As mentioned in Definition 10, the inclusion relation of fuzzy sets is a partial order in the set of fuzzy sets defined on a same domain X. For this reason the specialization relation, extended to conceptual graphs that include fuzzy concepts, preserves its reflexivity and transitivity properties. As all the comparisons of "classic" (non fuzzy) conceptual graphs remain unchanged, we still do not have the antisymmetry property (it is a preorder) and incomparable graphs still cannot be compared (it is a partial preorder).*

As we intuitively presented above, comparisons of fuzzy concept vertices can be done in constant or polynomial time depending on the cases. Searching a projection from an acyclic graph into a graph, using the algorithm of [12] extended to fuzzy concepts, thus remains a problem with polynomial complexity.

Using this extended projection operation, the comparison of two conceptual graphs leads to a binary result: a graph G can be projected into a graph G' or cannot, there is no intermediate solution. However a more flexible comparison of fuzzy sets should allow one to evaluate the compatibility between a fuzzy query graph and a fuzzy factual graph. Therefore we propose to introduce a relation of compatibility with a degree d between two conceptual graphs.

4.3. A more flexible comparison of fuzzy concepts

Two scalar measures are classically used to evaluate the compatibility between a fuzzy selection criterium and a correspondent imprecise datum: (i) a degree of possibility of matching [19]; (ii) a degree of necessity of matching [5]. Within the framework of this paper, we only deal with the former.

Definition 14 *Let m and m' be two markers of types t and t', respectively defined on Ref(t) and Ref(t'), with characteristic functions χ_m and $\chi_{m'}$. Then m' is compatible with m with the possibility degree d (noted $m' comp_d m$), where d has the following value:*

- *$d = 0$ if $Ref(t) \cap Ref(t') = \emptyset$;*
- *otherwise $d = \Pi(m\ ; m')$.*

$\Pi(m\ ; m')$, degree of possibility of matching between m and m', measures the maximum compatibility between m and m' and is defined by:
$\Pi(m\ ; m') = sup_{x \in Ref(t) \cap Ref(t')} min(\chi_m(x), \chi_{m'}(x))$.

Note that this measure of the degree of possibility with which m' is compatible with m is symmetrical.

An example is given in Figure 11.

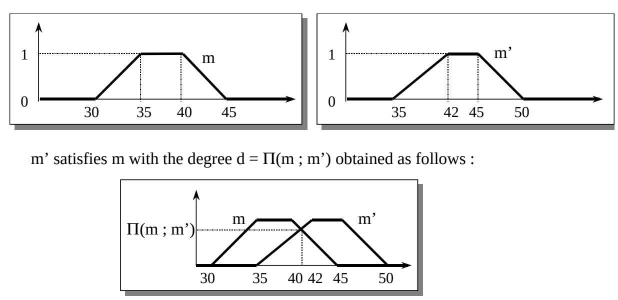

Figure 11. Flexible comparison of two markers m and m' of type *NumericalValue*

Remark 5 *For two "classic" individual markers m and m', $\Pi(m\ ; m')$ takes the value 1 if $m = m'$, 0 if not. If m or m' is the generic marker, $\Pi(m\ ; m') = 1$.*

Definition 15 *Let t and t' be two fuzzy types, respectively defined on the domains D_t and $D_{t'}$. Their characteristic functions are noted χ_t and $\chi_{t'}$. Then t' is compatible with t with the possibility degree d (noted $t' comp_d t$), where d is determined as follows:*

Let A be the set of all pairs (x, x') from $D_t \times D_{t'}$ satisfying $x' \leq x$.

- *if $A = \emptyset$, $d = 0$;*
- *otherwise $d = sup_{(x,x') \in A} min(\chi_t(x), \chi_{t'}(x'))$.*

For example, the fuzzy type:
t' = 1/Full milk + 0.5/Half-skimmed milk
is compatible with the fuzzy type:
t = 0.6/Milk + 1/Beef + 0.3/Poultry
with the degree:

$$d = sup(min(\chi_t(Milk), \chi_{t'}(Full\ milk)), min(\chi_t(Milk), \chi_{t'}(Half\text{-}skimmed\ milk)))$$
$$= sup(min(0.6, 1), min(0.6, 0.5))$$
$$= sup(0.6, 0.5) = 0.6.$$

Note that this measure of the degree of possibility with which t' is compatible with t is not symmetrical, because it involves the specialization relation. For instance, in the previous example, t is compatible with t' with the degree 0.

Remark 6 *For two "classic" types t and t', $\Pi(t\ ; t')$ takes the value 1 if $t \leq t'$, 0 if not.*

Definition 16 *Let $l = (t, m)$ and $l' = (t', m')$ be the labels of two concepts c and c', where t and t' can be fuzzy types, m and m' can be fuzzy markers (we recall that the type and its marker cannot be fuzzy simultaneously). Then c' is compatible with c with the degree of possibility d (noted $c' comp_d c$), where d is defined as follows:*

Let $d1$ be the degree with which t' is compatible with t ($t' comp_{d1}$ t). Let $d2$ be the degree with which m' is compatible with m ($m' comp_{d1}$ m). Then $d = min(d1, d2)$.

The *min* operator is used for the conjunction of the compatibility degrees, as presented in [6].

For instance, for:
c = [Full milk : 1/sample32 + 1/sample35] and
c' = [0.5/Full milk + 1/Half-skimmed milk : *],
we have:
$d1 = 0.5$ (Full milk has the degree 1 in c and 0.5 in c', Half-skimmed milk is not comparable with Full milk),
$d2 = 1$ (both sample32 and sample35 have the degree 1 in c and also in c', where the generic marker * stands for the fuzzy sets that associates the degree one with every marker of Full milk and Half-skimmed milk)
$d = min(0.5, 1) = 0.5$, thus $c' comp_{0.5}\ c$.

Definition 17 *Let G and G' be two conceptual graphs that can possibly include fuzzy concepts. Then the graph G' is compatible with the graph G with the degree d (noted $G' comp_d\ G$) if there is an ordered pair (f, g) of mappings, f (resp. g) from the set of relation types (resp. concept types) of G to the set of relation types (resp. concept types) of G', such that:*

- *edges and their numbering are preserved;*
- *relation vertex labels may be restricted.*

d is then determined as follows:

Let C_G be the set of concept vertices of G. For each concept vertex $c \in C_G$, let d_c be the degree of possibility with which $g(c)$ is compatible with c. Then $d = min_{c \in C_G} d_c$.

Remark 7 *If G can be projected into G' (G' is a specialization of G), then G' is compatible with G with the degree 1.*

For example let us consider the graph G given in Figure 12 and the graph G' given in Figure 13. G' is compatible with G with the degree of possibility $d = 0.5$, which corresponds to the degree of possibility with which the concept vertex [Half-skimmed milk : *] of the graph G' is compatible with the concept vertex [1/Skimmed milk + 0.5/Half-skimmed milk : *] of the graph G, all the other concept vertices of G being satisfied with the degree of possibility 1 by their image in G'.

As explained in Section 4.2, searching a projection from an acyclic graph into a graph, both possibly including fuzzy concepts, is a problem with polynomial complexity. Calculating the degree of possibility of matching is done in constant time. The algorithm of [12] adapted to compute if an acyclic graph is compatible with a graph

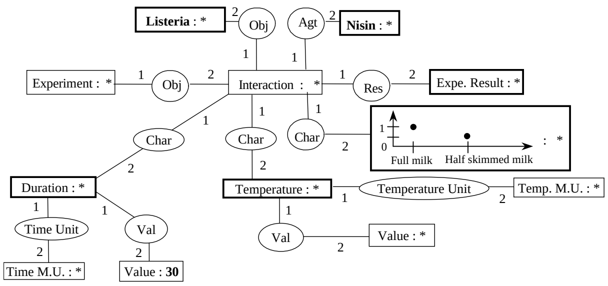

Figure 12. An example of a query graph G

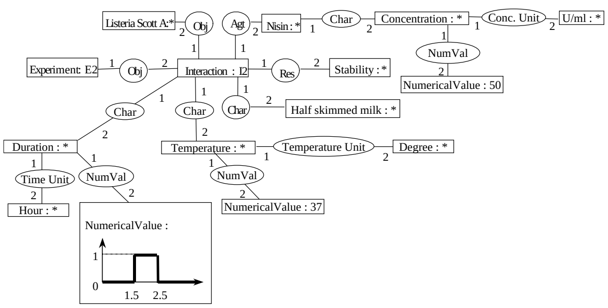

Figure 13. An example of a factual graph G'

(both possibly including fuzzy concepts) with a given possibility degree , thus remains a problem with polynomial complexity, but it supplies more solutions.

5. CONCLUSION AND PERSPECTIVES

Within the context of the creation of a tool for decision-making aid in the domain of food risk control, the specificities of the data led us to follow the steps presented in this paper: in the conceptual graph model, we have presented a choice for the representation of numerical values and a way of representing fuzzy data. In order to allow comparisons in this extended model, we have proposed an extension of the specialization relation. Lastly we have softened this comparison by introducing a relation of compatibility with a degree d between two graphs, allowing enlarged querying.

The originality of our approach is the combination of two models that complement each other to satisfy the purposes of the application. Indeed the data and the queries of the project require a flexible data structure and fit to a hierachical classification, which is brought by the conceptual graph model. On the other hand they include numerical data and fuzzy data, which the conceptual graph model is not designed for [13], but which are handled by the fuzzy set theory [19]. This combined approach is also original because it integrates fuzzy sets in the conceptual graph model tightly; fuzzy sets are built upon the support of the conceptual graph model and provide a homogeneous extension of the model.

A prototype of this work has been implemented using the CoGITo platform [8] and a microbiological knowledge base is under construction, in cooperation with the group of microbiologist experts working on the project. It includes information from three kinds of publications:

- documents that synthesize experimental results of different previous articles on a given subject. These publications cannot be stored as recordings in the relational database which is dedicated to the description of complete and detailed experiments;
- documents that give qualitative information only. Qualitative data are not exploitable by querying the relational database, where they can only be stored as plain text; the keywords and the semantics of the connections between them are not highlighted.
- documents whose content is not directly related to the relational database theme. There are no attributes that fit to these data in the relational database, but they can be stored as concepts in the conceptual graph model.

About one hundred graphs, each composed of around fifty vertices, have been registered in the knowledge base up to now. Nested conceptual graphs [4] - i.e. conceptual graphs that include concept vertices whose description itself is represented by a conceptual graph - could be used in order to represent information at various levels of detail.

Our very next work will be to study other comparison degrees (in particular the degree of necessity of matching [5]) in order to refine the comparison of fuzzy sets. In a more distant future, we will have to adapt our system to enable non-specialists of the conceptual graph model to use it. An important work on the interfacing of our system has to be done. In particular, during the knowledge acquisition stage, by providing conceptual graph patterns, that biologists could complete in order to enter data in the knowledge base.

REFERENCES

1. P. Bosc, L. Lietard, and O. Pivert. Soft querying, a new feature for database management system. In *Proceedings DEXA'94 (Database and EXpert system Application), Lecture Notes in Computer Science #856*, pages 631–640. Springer-Verlag, 1994.
2. P. Buche and O. Haemmerlé. Towards a unified querying system of both structured and semi-structured imprecise data using fuzzy views. In *Proceedings of the 8th International Conference on Conceptual Structures, Lecture Notes in Artificial Intelligence #1867*, pages 207–220, Darmstadt, Germany, August 2000. Springer-Verlag.
3. T.H. Cao. *Foundations of Order-Sorted Fuzzy Set Logic Programming in Predicate Logic and Conceptual Graphs.* PhD thesis, University of Queensland, Australia, 1999.
4. M. Chein, M.L. Mugnier, and G. Simonet. Nested graphs: A graph-based knowledge representation model with fol semantics. In *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 524–524, Trento, Italy, June 1998. Morgan Kaufmann Publishers.
5. D. Dubois and H. Prade. *Possibility Theory - An Approach to Computerized Processing of Uncertainty.* Plenum Press, New York, 1988.
6. D. Dubois and H. Prade. *Fuzziness in Database Management Systems, P. Bosc and J. Kacprzyk eds.*, chapter Tolerant fuzzy pattern matching : an introduction, pages 42–58. Heidelberg:Physica Verlag, 1995.
7. D. Dubois, H. Prade, and J.P. Rossazza. Vagueness, typicality and uncertainty in class hierarchies. *International Journal of Intelligent Systems*, 6:167–183, 1991.
8. O. Haemmerlé and O. Guinaldo. CoGITo v3.3 : plate-forme de développement d'applications sur les graphes conceptuels. *Technique et Science Informatiques*, 18(9):933–965, Novembre 1999.
9. D.S. Jung, F.W. Bodyfelt, and M.A. Daeschel. Influence of fat and emulsifiers on the efficacy of nisin in inhibiting listeria monocytogenes in fluid milk. *Journal of Dairy Science*, (75):387–393, 1992.
10. S.K. Morton. *Conceptual graphs and fuzziness in artificial intelligence.* PhD thesis, University of Bristol, 1987.
11. M.L. Mugnier and M. Chein. Polynomial algorithms for projection and matching. In *Proceedings of the Seventh Annual Workshop on Conceptual Structures, Lecture Notes in Artificial Intelligence #754*, pages 239–251, Las Cruces, New Mexico, 1992.
12. M.L. Mugnier and M. Chein. Characterization and algorithmic recognition of canonical conceptual graphs. In *Proceedings of the First International Conference on Conceptual Structures, Lecture Notes in Artificial Intelligence #699*, pages 294–311, Quebec City, Canada, 1993. Springer-Verlag.
13. M.L. Mugnier and M. Chein. Représenter des connaissances et raisonner avec des graphes. *Revue d'Intelligence Artificielle*, 10(1):7–56, 1996.
14. H. Prade. Lipski's approach to incomplete information data bases restated and generalized in the setting of zadeh's possibility theory. *Information Systems*, 9(1):27–42, 1984.
15. F. Sebastiani. A probabilistic terminological logic for modelling information retrieval. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval,*

pages 122–130, Dublin, Ireland, July 1994. ACM/Springer.
16. J.F. Sowa. *Conceptual structures - Information processing in Mind and Machine*. Addison-Welsey, 1984.
17. V. Wuwongse and M. Manzano. Fuzzy conceptual graphs. In *Proceedings of the First International Conference on Conceptual Structures, Lecture Notes in Artificial Intelligence #699*, pages 430–449, Quebec City, Canada, August 1993. Springer-Verlag.
18. L.A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
19. L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1:3–28, 1978.
20. M.H. Zwietering, T. Wijtzes, J.C. de Wit, and K. Van't Riet. A decision support system for prediction of the microbial spoilage in foods. *Journal of Food Protection*, 55:973–979, 1992.