Rieger, B. (Ed.): Empirical Semantics, Bochum: Brockmeyer 1981

104

APPLICATIONS OF ALPHA EXPRESSIONS TO FUZZY RELATIONS

*Motohide Umano**

*Masaharu Mizumoto***

*Kokichi Tanaka****

* *Department of Applied Mathematics*
Faculty of Science
Okayama University of Science
Ridai-cho, Okayama
Japan

** *Department of Management Engineering*
Faculty of Engineering
Osaka Electro-Communication University
Neyagawa, Osaka
Japan

*** *Department of Information and Computer Sciences*
Faculty of Engineering Science
Osaka University
Toyonaka, Osaka
Japan

The data and information encountered in the real world do not have precisely defined criteria of membership in a certain class. In order to deal mathematically with such ambiguity, L.A. ZADEH (1965) has proposed the concept of fuzzy sets and fuzzy relations, and formulated many concepts such as fuzzy program (ZADEH 1973), fuzzy logic and approximate reasoning (ZADEH 1975) and possibility distribution (ZADEH 1978).

For easy and convenient applications of fuzzy sets and fuzzy relations, we have implemented a system for fuzzy-set manipulation based on FSTDS (Fuzzy-Set-Theoretic Data Structure) (UMANO, MIZUMOTO and TANAKA 1978a) and a system for fuzzy reasoning (UMANO, MIZUMOTO and TANAKA 1978b, 1979) using the FSTDS System. In the FSTDS System, we have to write statements using fuzzy-set operators such as UNION, INTERSECTION and COMPOSITION to describe a procedure in FSTDSL/FORTRAN.

E.F. CODD (1971, 1972), however, proposed an alpha expression which provides the relation required using predicate calculus in a relational model of databases (CODD 1970). Since this provides non procedurally and intuitively the relation required, it is very useful for manipulating ordinary relations.

In this paper, we present a definition of a fuzzy relational database, which is an extended version of Codd's relational database, and a fuzzy alpha expression whose predicate expression contains fuzzy sets as constants and fuzzy relations as predicate operators. And we describe interpretation methods for the application of such fuzzy alpha expressions to fuzzy relational databases. The interpretation methods are concerned with the processing of the grades of tuples in fuzzy relations and the compatibility of the predicate in a fuzzy alpha expression.

1. RELATIONAL DATABASES AND ALPHA EXPRESSIONS

E.F. CODD (1970) proposed a relational model of database and the method, called alpha expressions (CODD 1971, 1972), for selecting the relations required from such a database using predicate calculus.

[Definition 1] A <u>relational database</u> D is a collection of relations R_1, R_2, ..., R_r, in which each domain can be distinguished from each other by attribute names (A_{i1}, A_{i2}, ..., A_{in_i} for a relation R_i).

As Figure 1 illustrates (DATE 1975), it is convenient to represent a relation as a table, with each row representing one tuple. In Figure 1, PART is a relation name and P#, PNAME, COLOR and WEIGHT are attribute names. Since we have several relations in a database and the same attribute names may be used in different relations, each domain is specified by the attribute name modified by a relation name, that is, $R_i.A_{ij}$, where R_i is a relation name and A_{ij} an attribute name in R_i.

PART

P#	PNAME	COLOR	WEIGHT
P1	Nut	Red	12
P2	Bolt	Green	17
P3	Screw	Blue	17
P4	Screw	Red	14
P5	Cam	Blue	12
P6	Cog	Red	19

Fig. 1. Relation PART

[Definition 2] An alpha expression is denoted as

$$\{<T_1, T_2, \ldots, T_n> : P\}, \tag{1}$$

where T_i, $i = 1,2,\ldots,n$, are attribute names qualified by the appropriate relation name and P stands for a predicate and (1) defines a relation of n-tuples $<t_1,t_2,\ldots,t_n>$, $t_i \in T_i$, $i = 1,2,\ldots,n$, which satisfy the predicat P. The list $<T_1,T_2,\ldots, T_n>$ is called the target list and P the qualification expression.

In general, a predicate is formulated according to the usual rules but including attribute names qualified by a relation name which may be or not be contained in the target list. The permitted operators are the comparison operator θ (=, ≠, <, ≤, > and ≥), the Boolean operator and, or and not, and of course parentheses ( ) to enforce a desired order of evaluation.

[Example 1] For relations S, P and SP in Figure 2 which represent supplier, part and their relationship, respectively, consider the following alpha expressions (DATE 1977).

(a) W_1 = {SP.P# : SP.S# = S2}

S

S#	SNAME	STATUS	CITY
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens

P

P#	PNAME	COLOR	WEIGHT
P1	Nut	Red	12
P2	Bolt	Green	17
P3	Screw	Blue	17
P4	Screw	Red	14
P5	Cam	Blue	12
P6	Cog	Red	19

SP

S#	P#	QTY
S1	P1	3
S1	P2	2
S1	P3	4
S1	P4	2
S1	P5	1
S1	P6	1
S2	P1	3
S2	P2	4
S3	P3	4
S3	P5	2
S4	P2	2
S4	P4	3
S4	P5	4
S5	P5	5

Fig. 2. Relational representation of supplier-part-model.

This selects a set of P# components of 3-tuples in SP whose S# component is identical to a constant S2. The result is a set {P1, P2} and it is assigned to W_1. When the number of the components in the target list is one , the angles < > may be omitted for simplicity. Note that in a predicate constants are denoted without quotation marks, i.e., '...'. There is no confusion because all attribute names are qualified by relation names in this paper.

Thus, we have

$$W_1 = \{P_1, P_2\}. \tag{2}$$

(b) W_2 = {S.S# : S.CITY = Paris <u>and</u> S.STATUS > 20}

This results in a set of S# components of 4-tuples in S whose CITY component is identical to Paris and STATUS component is greater than 20, that is, it selects supplier numbers for suppliers in Paris with status > 20. The result is

$$W_2 = \{S3\}. \tag{3}$$

(c) $W_3 = \{\langle S.SNAME, S.CITY\rangle : SP.S\# = S.S\# \ \underline{\text{and}}\ SP.P\# = P2\}$

This gets a set of pairs of SNAME and CITY components in tuples in S whose S# component is identical to S# component of tuples in SP whose P# component is equal to a constant P2. We can get

$$W_3 = \{\langle Smith, London\rangle, \langle Jones, Paris\rangle, \langle Clark, London\rangle\}. \tag{4}$$

As for evaluation of alpha expressions, the predicate is evaluated ranging over all relations in the target list and if the predicate is true, then the tuple constructed from the components corresponding to the target list is added to the result. Note that we must also range over all tuples in the relations in the predicate which are not contained in the target list. For (c) in Example 1, since SP occurs in the predicate but not in the target list, we must range over all tuples in SP with ranging over S.

For linking SP's in SP.S# and SP.P#, a range variable and an existential quantifier ∃ are introduced and we might have an equivalent alpha expression to (c) in Example 1:

$$W_3 = \{\langle S.SNAME, S.CITY\rangle : \exists Z(Z.S\# = S.S\# \ \underline{\text{and}}\ Z.P\# = P2)\}, \tag{5}$$

where Z ranges over the relation SP.

In general, we can use a universal quantifier ∀. However, when we extend a relation and predicate to fuzzy ones, it will be difficult to give good interpretations to them. So we will not use existential and universal quantifiers in this paper.

2. FUZZY DATABASES AND FUZZY ALPHA EXPRESSIONS

The definition of a relational database lead to that of a fuzzy database as a collection of fuzzy relations (KUNII 1976). We can define more general and complex fuzzy databases. The most general fuzzy databases may be defined as a fuzzy set of generalized fuzzy relations (UMANO, MIZUMOTO and TANAKA 1978a). This definition is, however, so complex to manipulate that we shall have a definition which is simple but may be enough to represent fuzzy data in the real world.

[Definition 3] A <u>fuzzy database</u> D_F is a collection of fuzzy relations of fuzzy sets. A fuzzy relation R_F of fuzzy sets in $U_1, U_2, \ldots, U_n$ is defined by a membership function:

$$\mu_{R_F} : [0,1]^{U_1} \times [0,1]^{U_2} \times \ldots \times [0,1]^{U_n} \longrightarrow [0,1], \qquad (6)$$

where B^A means all functions from A to B and × is the Cartesian product.

Note that a fuzzy relation of fuzzy sets in $U_1, U_2, \ldots, U_n$ can be considered as a level-2 fuzzy relation in $U_1 \times U_2 \times \ldots \times U_n$.

To illustrate a fuzzy relation in a table form, we add a special attribute name μ. It should be, however, noted that a user need not pay attention to an attribute name μ by use of fuzzy alpha expressions although he can specify it to manipulate a fuzzy database.

[Example 2] If U_1, U_2 and U_3 are the sets of names, numerical ages and numerical height of individuals, respectively, and fuzzy sets *young*, *middle-aged* and *old* in U_2 and *short*, *middle* and *tall* in U_3 are defined, we might have

$$R_F = \{0.8/\langle\{John\}, young, tall\rangle,\ 0.6/\langle\{Jack\}, old, middle\rangle\} \qquad (7)$$

as a fuzzy relation of fuzzy sets in U_1, U_2 and U_3. When a fuzzy set has only one element with a grade value being 1 such as {John} and {Jack}, we may omit the braces { } unless there is confusion.

By a fuzzy alpha expression, we mean an alpha expression whose predicate is fuzzy, that is, it contains fuzzy sets as constant values and fuzzy relations as predicate operators.

We shall consider in the following the way of interpretation in the cases where a fuzzy database and a fuzzy alpha expression are combined.

3. APPLICATIONS OF FUZZY ALPHA EXPRESSIONS TO A FUZZY DATABASE

The interpretation methods are concerned with the grade associated with tuples in fuzzy relations and the compatibility of the predicate expression in fuzzy alpha expressions.

(1) Processing of grades associated with tuples

In order to separate the processing of the grades associated with tuples in fuzzy relations, we shall use ordinary fuzzy relations as a fuzzy database. In this case, relations in a database include only ordinary elements and the interpretation of a predicate is the same as that for ordinary databases.

We must, however, determine a grade value of a tuple in the target list which satisfies the predicate since a tuple which includes components in a result tuple belongs to a relation with some grade value. We shall have a simple example.

[Example 3] Assume that R and S are fuzzy relations shown in Figure 3. Consider the following alpha expressions.

(a) W_1 = {R.A2 : R.A1 = a}

R

A1	A2	μ
a	x	0.1
a	y	0.2
b	z	0.3
c	z	0.4

S

A1	A2	μ
x	e	0.6
x	f	0.7
y	g	0.9
z	g	0.1
z	h	0.5

Fig. 3. Fuzzy relations R and S.

We range over only a fuzzy relation R. For the first row 0.1/<a,x>, although the predicate is satisfied, we may not add a component x to the result with the compatibility 1. The tuple <a,x> belongs to the relation R with a grade 0.1, so it is reasonable to add the component x with the compatibility 0.1. For the second row, we have y with a grade value 0.2. For the third and fourth rows, the predicate is not satisfied and nothing is added to the result. Thus, we have a fuzzy set:

$$W_1 = \{0.1/x,\ 0.2/y\}. \tag{8}$$

(b) $W_2 = \{<R.A1,\ S.A2> : R.A2 = S.A1\}$

First, we range over the relation R. For the first row in R, we have the first and second rows in the relation S which satisfy the predicate. In this case, since the target list involves two relations R and S, we may have the minimum value of the two grade values, i.e., for the first row 0.1/<a,x> in R and the first row 0.6/<x,e> in S, so we have the compatibility:

$$0.1 \wedge 0.6 = 0.1 \tag{9}$$

for the tuple <a,e> for the result. Similarly, 0.1/<a,f> is added to the result for the first row in R and the second row in S.

Thus we have a fuzzy relation:

$$W_2 = \{0.1/\langle a,e\rangle,\ 0.1/\langle a,f\rangle,\ 0.2/\langle a,g\rangle,\\ 0.1/\langle b,g\rangle,\ 0.3/\langle b,h\rangle,\ 0.1/\langle c,g\rangle,\ 0.4/\langle c,h\rangle\}. \qquad (10)$$

The evaluated value of the predicate can be considered as the compatibility 1 if the predicate is satisfied and 0 if not.

As a summary of the above consideration, we have the following method.

[Method 1] Let R_1, R_2, ..., R_r be fuzzy relations and A_{i1}, A_{i2}, ..., A_{in_i} be attribute names of R_i and P be a predicate. If we have an alpha expression:

$$\{\langle R_{i_1}.A_{j_1},\ R_{i_2}.A_{j_2},\ \ldots,\ R_{i_n}.A_{j_n}\rangle : P\}, \qquad (11)$$

the compatibility of an n-tuple $\langle u_1, u_2, \ldots, u_n\rangle \in R_{i_1}.A_{j_1} \times R_{i_2}.A_{j_2} \times \ldots \times R_{i_n}.A_{j_n}$ is given by

$$p \wedge g_1 \wedge g_2 \wedge \ldots \wedge g_n, \qquad (12)$$

where p is the evaluated value of the predicate P and g_k, k = = 1,2,...,n, denotes the grade value of the tuple, in which the u_k is a component of the attribute A_{j_k} in the fuzzy relation R_{i_k}, in the fuzzy relation R_{i_k}. Thus fuzzy relation induced by the alpha expression (11) from fuzzy relations R_1, R_2, ..., R_r is the following:

$$\{p \wedge g_1 \wedge g_2 \wedge \ldots \wedge g_n / \langle u_1, u_2, \ldots, u_n\rangle\\ : \langle u_1, u_2, \ldots, u_n\rangle \in R_{i_1}.A_{j_1} \times R_{i_2}.A_{j_2} \times \ldots \times R_{i_n}.A_{j_n}\}. \qquad (13)$$

If we apply the alpha expression (a) in Example 3 to an ordinary relation, we have the image of the set {a} under R in a sense of ordinary set theory. The result omitted to a fuzzy relation is the image {a} under R in a sense of fuzzy set theory.

As for the alpha expression (b) in Example 3, we have the composition of R and S for both ordinary relations and fuzzy relations in respective senses.

So Method 1 is very reasonable for the processing of grades associated with tuples in fuzzy relations.

(2) <u>Compatibility of the predicate in fuzzy alpha expressions</u>

We shall consider a fuzzy relation whose elements are fuzzy sets as was defined in (6). Since we can apply Method 1 for the processing of grades associated with tuples, it is sufficient to consider an ordinary relation of fuzzy sets.

A problem is how to evaluate the predicate which includes fuzzy sets and get the compatibility of the predicate with relations.

Since a predicate often includes a term $r_1 = r_2$ and it seems natural to deal with it by computing a degree of a fuzzy set consistency between r_1 and r_2, we will differentiate an operator = from the other operators. We first present, therefore, an interpretation method of a predicate which contains only term $r_1 =$ $= r_2$. The terms involving the other operators are discussed later.

[Example 4] Suppose that U_1, U_2 and U_3 are a set of names, the interval [0,100] which represents ages and the interval [0,200] which does height of individuals. We have an ordinary relation PERSON of fuzzy sets in U_1, U_2 and U_3 shown in Figure 4. In Figure 4, *young*, *middle-aged* (we may denote *ma* in an abbreviation form) and *old* are fuzzy sets in U_2 and *tall*, *middle* and *short* in U_3.

Consider an alpha expression:

$$W = \{PERSON.NAME : PERSON.AGE = 25\} \tag{14}$$

For John and Betty, the compatibilities of the predicate are 0. For Mike, Taro and Jack, they will be $\mu_{young}(25)$, $\mu_{middele\text{-}aged}(25)$

and $\mu_{old}(25)$, respectively, using membership functions. Thus we have

$$W = \{\mu_{young}(25)/\text{Mike}, \mu_{middle\text{-}aged}(25)/\text{Taro}, \mu_{old}(25)/\text{Jack}\}. \quad (15)$$

If the fuzzy sets *young*, *middle-aged* and *old* are defined using S, π and Z functions (ZADEH 1975, 1978 and UMANO, MIZUMOTO and TANAKA 1978a) as

$$\mu_{young}(u) = Z(u;\ 30,25,20), \quad (16)$$

$$\mu_{middle\text{-}aged}(u) = \pi(u;\ 20,40), \quad (17)$$

$$\mu_{old}(u) = S(u;\ 40,45,50), \quad (18)$$

and illustrated in Figure 5, then we have the values of membership functions as follows:

$$\mu_{young}(25) = 0.5, \quad (19)$$

$$\mu_{middle\text{-}aged}(25) = 0.125, \quad (20)$$

$$\mu_{old}(25) = 0. \quad (21)$$

PERSON

NAME	AGE	HEIGHT
John	15	tall
Betty	22	middle
Mike	young	short
Taro	middle-aged	160
Jack	old	170

Fig. 4. Ordinary relation PERSON of fuzzy sets.

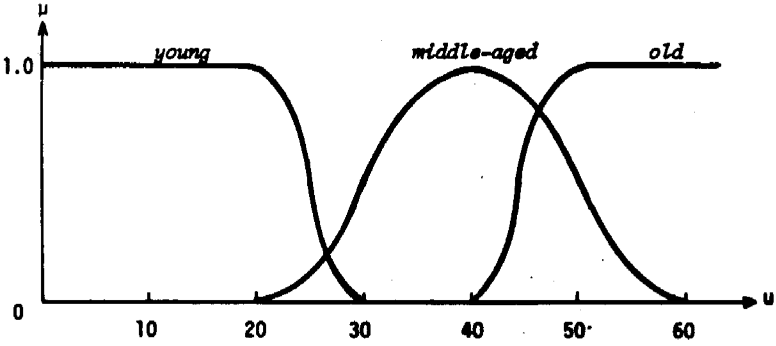

Fig. 5. Fuzzy sets *young, middle-aged* and *old*.

Thus we have a fuzzy set:

$$W = \{0.5/\text{Mike},\ 0.125/\text{Taro}\}. \tag{22}$$

This result (22) seems to agree with our intuition. This interpretation will be reasonable when we recall the meaning of the membership function.

The evaluated value of a term $r_1 = r_2$ took a number in the interval [0,1]. The Boolean operators and, or and not may be defined as

$$t_1 \ \underline{\text{and}}\ t_2 = t_1 \wedge t_2, \tag{23}$$

$$t_1 \ \underline{\text{or}}\ t_2 = t_1 \vee t_2, \tag{24}$$

$$\underline{\text{not}}\ t = 1 - t, \tag{25}$$

where t_1, t_2 and t are numbers in [0,1] and mean the evaluated values of the term $r_1 = r_2$.

An alpha expression in Example 4 includes an element 25 in U_2 as a constant. We can extend an element to a fuzzy set in U_2 as a constant, and it is called a fuzzy alpha expression. We can use an extension principle by ZADEH (1975). We shall state it as Method 2 in terms of the compatibility.

[Method 2] If we can compute a function γ of compatibility for all u in U, then a compatibility for a fuzzy set F such as

$$F = \{\mu_F(u)/u : u \in U\} \quad (26)$$

is given by

$$\gamma(F) = \{\mu_F(u)/\gamma(u) : u \in U\}. \quad (27)$$

[Example 5] For the relation PERSON of fuzzy sets in Figure 4, consider a fuzzy alpha expression:

$$W = \{\text{PERSON.NAME} : \text{PERSON.AGE} = \textit{middle-aged}\}. \quad (28)$$

The compatibilities for John and Betty will be given by $\mu_{middle\text{-}aged}(15)$ and $\mu_{middle\text{-}aged}(22)$, respectively. For Mike, Taro and Jack, it will be given using Method 2 by

$$\mu_{ma}(young) = \{\mu_{young}(u)/\mu_{ma}(u) : u \in U\} = \gamma_1, \quad (29)$$

$$\mu_{ma}(ma) = \{\mu_{ma}(u)/\mu_{ma}(u) : u \in U\} = \{t/t : t \in [0,1]\} = \gamma_2, \quad (30)$$

$$\mu_{ma}(old) = \{\mu_{old}(u)/\mu_{ma}(u) : u \in U\} = \gamma_3, \quad (31)$$

where U is the universe of discourse U_2 in Example 4. Thus the result of the fuzzy alpha expression (28) is as follows.

$$W = \{\mu_{ma}(15)/\text{John}, \mu_{ma}(22)/\text{Betty}, \gamma_1/\text{Mike}, \gamma_2/\text{Taro}, \gamma_3/\text{Jack}\}. \quad (32)$$

Note that a fuzzy set W is a type-2 fuzzy set. If the fuzzy sets *young*, *middle-aged* and *old* are given as, say, in Figure 5, we have $\mu_{ma}(15) = 0$ and $\mu_{middle\text{-}aged}(22) = 0.02$ and γ_1, γ_2 and γ_3 as illustrated in Figure 6 which are fuzzy sets in the interval [0,1].

Thus, the result is

$$W = \{0.02/\text{Betty}, \gamma_1/\text{Mike}, \gamma_2/\text{Taro}, \gamma_3/\text{Jack}\}. \quad (33)$$

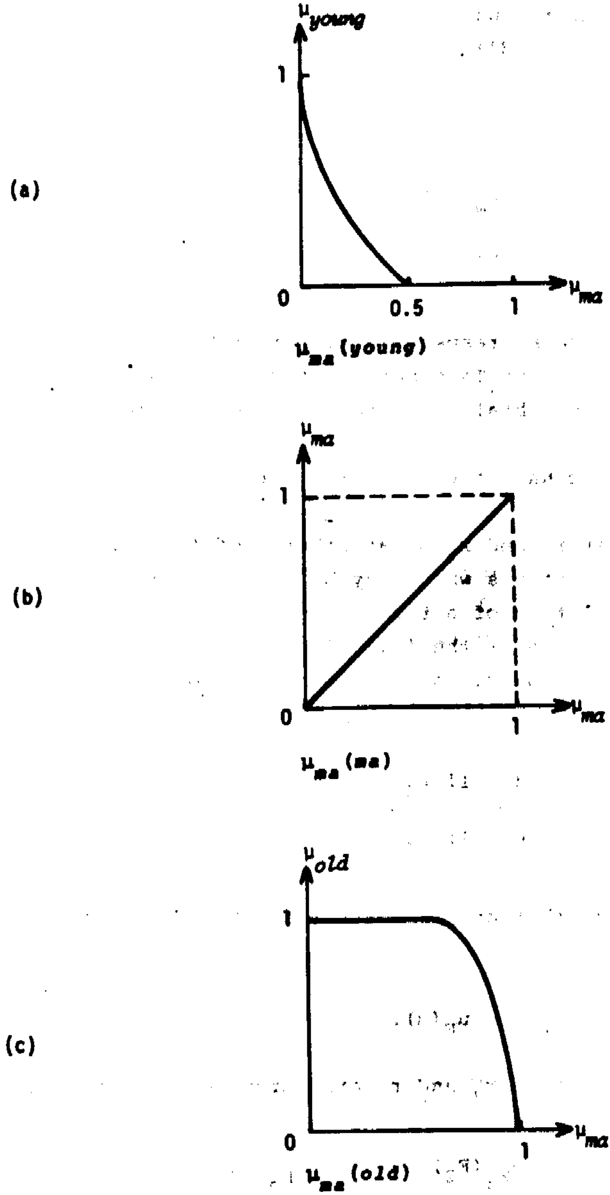

Fig. 6. Fuzzy sets $\mu_{ma}(young)$, $\mu_{ma}(ma)$ and $\mu_{ma}(old)$.

[Note] In Example 5, the compatibilities γ_1, γ_2 and γ_3 were obtained for Mike, Taro and Jack, respectively. However, we might have instead of (29) - (31) the following:

$$\mu_{young}(ma) = \{\mu_{ma}(u)/\mu_{young}(u) : u \in U\}, \tag{34}$$

$$\mu_{ma}(ma) = \{\mu_{ma}(u)/\mu_{ma}(u) : u \in U\}, \tag{35}$$

$$\mu_{old}(ma) = \{\mu_{ma}(u)/\mu_{old}(u) : u \in U\}, \tag{36}$$

for Mike, Taro and Jack, respectively. This result is the same as the previous one only for Taro but not for the others. These compatibilities could be obtained by the inverse functions of Figure 6.

As a summary, we have the following Method 3.

[Method 3] Let r_1 and r_2 be attributes of fuzzy relations of fuzzy sets or constants which may be elements or fuzzy sets. Then the compatibility γ of a term $r_1 = r_2$ in a predicate of an alpha expression is one of the following.

(1) In the case that r_1 is an element u_1 and r_2 is also an element u_2,

$$\gamma = 1 \quad \text{if } u_1 = u_2, \qquad \gamma = 0 \quad \text{if } u_1 \neq u_2. \tag{37}$$

(2) In the case that one is a fuzzy set F and the other is an element u,

$$\gamma = \mu_F(u). \tag{38}$$

(3) In the case that r_1 and r_2 are fuzzy sets F_1 and F_2,

$$\gamma = \mu_{F_1}(F_2) \quad \text{or } \gamma = \mu_{F_2}(F_1). \tag{39}$$

We cannot apply Method 3 to the other comparison operators. We shall consider the other methods for interpretation which is not dependent on the special comparison operators.

We can consider a comparison operator as a fuzzy relation and the compatibility is given by a grade value of its membership function. This method is applicable to the comparison operators such as $\approx$(approximately equal) and $\gg$ (much greater than).

[Method 4] Let r_1 and r_2 be attributes of ordinary fuzzy relations or constants of elements, and θ be a comparison operator which denotes an ordinary relation or a fuzzy relation. Then, the compatibility for a term $r_1 \theta r_2$ is given by

$$\gamma = \mu_\theta(r_1, r_2) \tag{40}$$

using the membership function of comparison operator.

We cannot get the compatibility if r_1 or r_2 is a fuzzy set. In this case, we can use the following Method 5.

[Method 5] If a membership function μ_θ of a comparison operator θ is defined for all u in U and v in V, the compatibility for a fuzzy set A in U and a fuzzy set B in V such as

$$A = \{\mu_A(u)/u : u \in U\}, \tag{41}$$

$$B = \{\mu_B(v)/v : v \in V\}, \tag{42}$$

is given by

$$\gamma = \mu_\theta(A, B) = \{\mu_A(u) \wedge \mu_B(v)/\mu_\theta(u,v) : \mu_\theta(u,v) > 0,$$
$$u \in U \text{ and } v \in V\}. \tag{43}$$

And if there exist no u and v such that $\mu_\theta(u,v) > 0$, then

$$\gamma = 0. \tag{44}$$

If either r_1 or r_2 is an element, we can use Method 5 with considering $\{1/r_1\}$.

[Example 6] For a relation in Figure 4, we have a fuzzy alpha expression:

$$W = \{\text{PERSON.NAME} : \text{PERSON.AGE} \approx \textit{middle-aged}\}. \tag{45}$$

The compatibilities are

John: $\mu_\approx(15,ma)=\{\mu_{ma}(v)/\mu_\approx(15,v) : \mu_\approx(15,v)>0,\ v \in V\}$, (46)

Betty: $\mu_\approx(22,ma)=\{\mu_{ma}(v)/\mu_\approx(22,v) : \mu_\approx(22,v)>0,\ v \in V\}$, (47)

Mike: $\mu_\approx(young,ma)=\{\mu_{young}(u)\wedge\mu_{ma}(v)/\mu_\approx(u,v) : \mu_\approx(u,v)>0,$
$u \in U$ and $v \in V\}$, (48)

Taro: $\mu_\approx(ma,ma)=\{\mu_{ma}(u)\wedge\mu_{ma}(v)/\mu_\approx(u,v) : \mu_\approx(u,v)>0,$
$u \in U$ and $v \in V$, (49)

Jack: $\mu_\approx(old,ma)=\{\mu_{old}(u)\wedge\mu_{ma}(v)/\mu_\approx(u,v) : \mu_\approx(u,v)>0,$
$u \in U$ and $v \in V$. (50)

If *young*, *middle-aged* and *old* are again shown in Figure 5, and $\approx$ in Figure 7, then the result is illustrated in Figure 8 for $\alpha = 5$ and in Figure 9 for $\alpha = 10$.

For the comparison operator >>, which might be defined by

$$\mu_{>>}(u,v) = S(u-v;\ \alpha,\beta,\gamma), \tag{51}$$

the same processing is applicable and we could have a result. But the result is omitted.

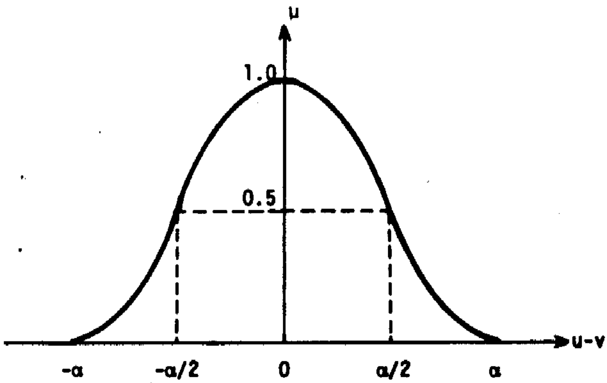

Fig. 7. Membership function of comparison operator ≈.

By several methods, we can compute the compatibility for a term $r_1 \theta r_2$. We must compute Boolean operators <u>and</u>, <u>or</u> and <u>not</u> in order to obtain a compatibility of a predicate P. If the compatibility is in the interval [0,1], we can use (23) - (25). But the compatibility of a term is in general a fuzzy set in the interval [0,1]. The operator <u>not</u> is unary, so we can apply Method 2 to it. For binary operators <u>and</u> and <u>or</u>, we can get the following.

[Method 6] If a binary operation * is defined for all u in U and v in V, the binary operation * can be extended to a fuzzy set A in U:

$$A = \{\mu_A(u)/u \ : \ u \in U\} \tag{52}$$

and a fuzzy set B in V:

$$B = \{\mu_B(v)/v \ : \ v \in V\} \tag{53}$$

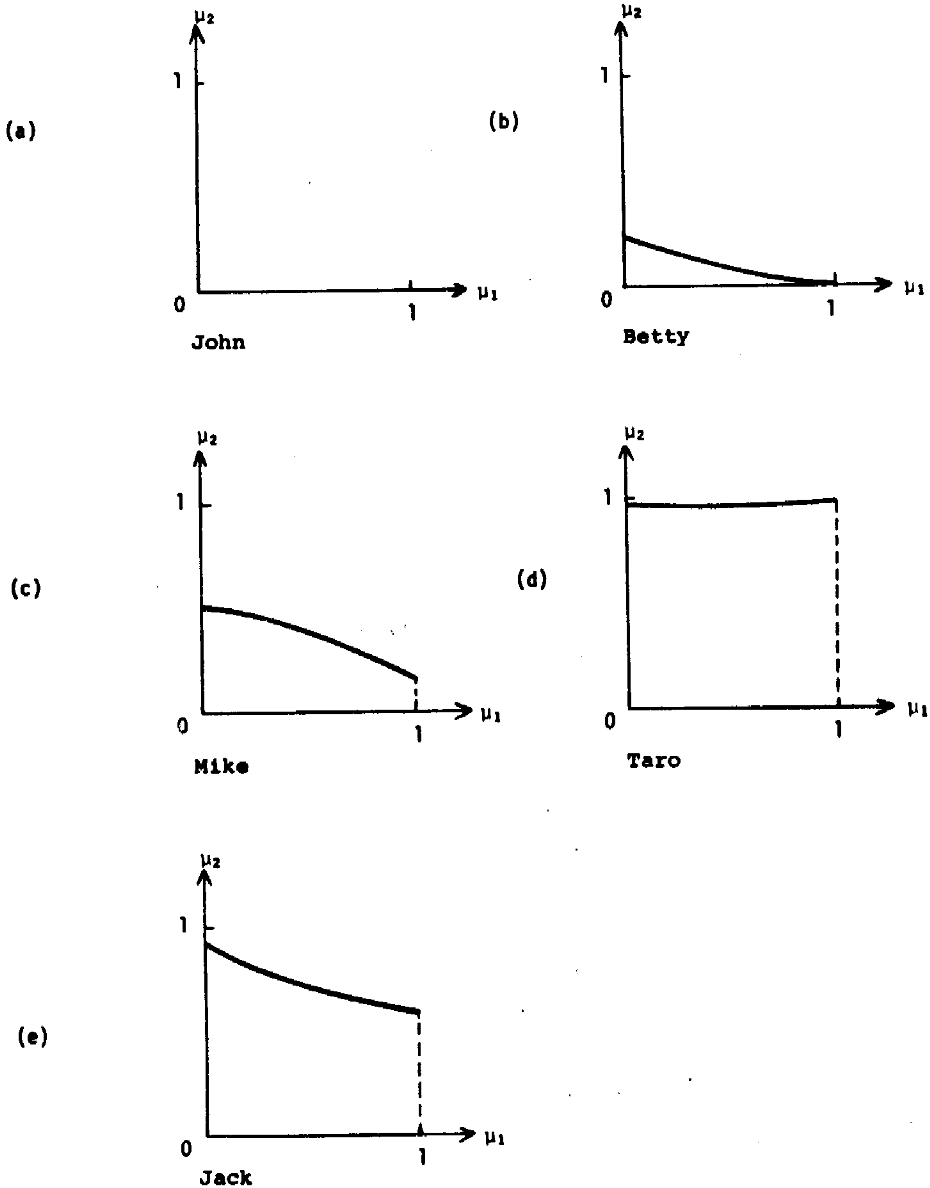

Fig. 8. Compatibilities in the case where $\alpha = 5$ in the comparison operator $\approx$.

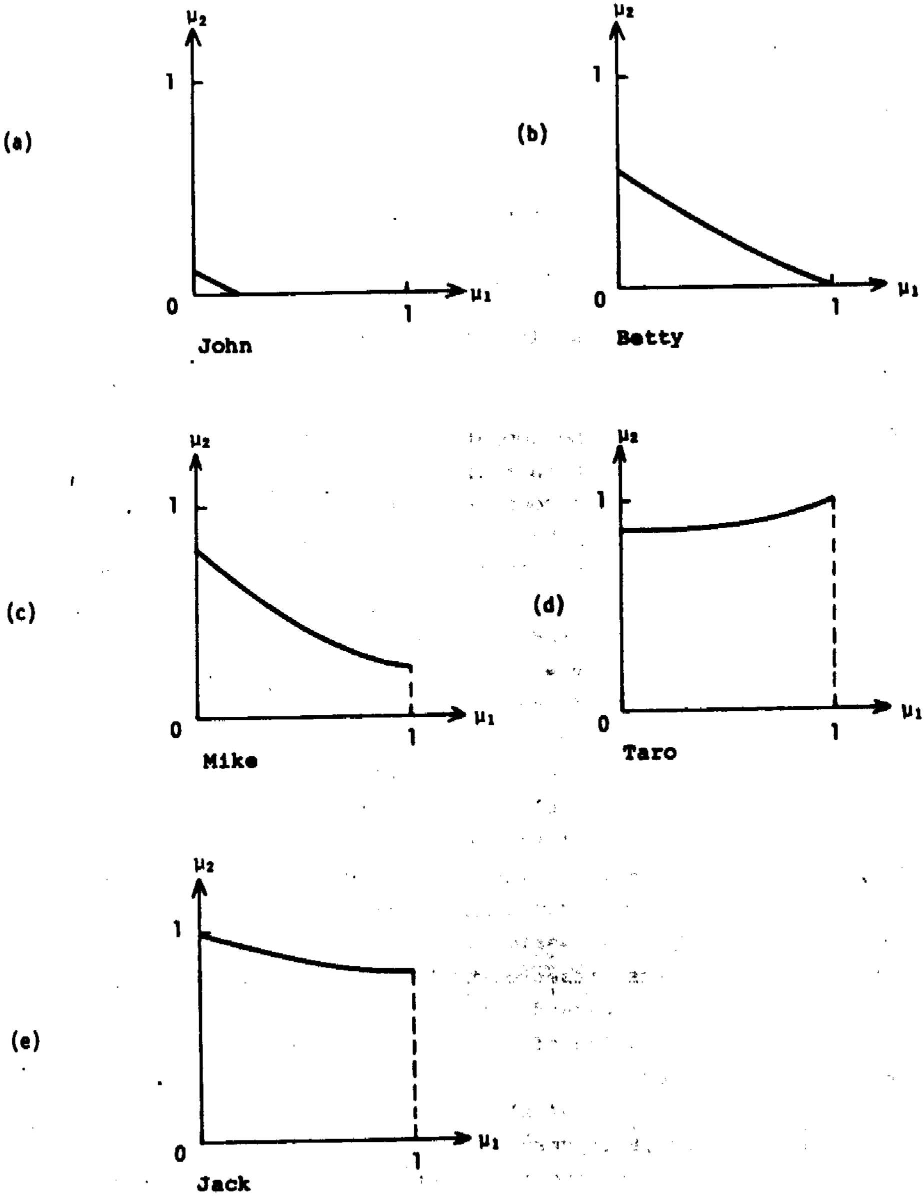

Fig. 9. Compatibilities in the case where $\alpha = 10$ in the comparison operator $\approx$.

and we have

$$A * B = \{\mu_A(u) \wedge \mu_B(v)/u * v : u \in U \text{ and } v \in V\}. \quad (54)$$

We can obtain using Methods 2 and 6

$$\underline{\text{not}}\ T = \{\mu_T(t)/1 - t : t \in [0,1]\}, \quad (55)$$

$$T_1\ \underline{\text{and}}\ T_2 = \{\mu_{T_1}(t_1) \wedge \mu_{T_2}(t_2)/t_1 \wedge t_2 : t_1, t_2 \in [0,1]\}, \quad (56)$$

$$T_1\ \underline{\text{or}}\ T_2 = \{\mu_{T_1}(t_1) \wedge \mu_{T_2}(t_2)/t_1 \vee t_2 : t_1, t_2 \in [0,1]\}, \quad (57)$$

where T, T_1 and T_2 are the compatibilities of fuzzy sets in the interval [0,1] for a term in a predicate P.

We have discussed the computation of the compatibility by several examples using Methods 2 - 6. Methods 2, 5 and 6 are given for extending a compatibility for an element in a universe of discourse to a fuzzy set in it, while Methods 3 and 4 involve the interpretation of a predicate. We can only apply Method 3 for a comparison operator = and its result may agree with our intuitions. But for two fuzzy sets, there are two interpretations and we must determine which is better. This can be overcome by a facility for choosing one of interpretations by a user specification. Method 4 can be applied to arbitrary comparison operators.

It will be very useful to introduce linguistic hedges (ZADEH 1972) such as *very, more or less, much* and *slightly,* since we can more conveniently and directly denote fuzzy sets in a fuzzy database and fuzzy alpha expressions. The interpretation method of linguistic hedges was presented by ZADEH (1972) as operators which operate on the operand fuzzy sets. So we need no more new method for interpretation of fuzzy alpha expression which contains linguistic hedges.

The result fuzzy set of alpha expression is a type-2 fuzzy set in general and it is very difficult to understand the meaning of the grade values. So the fuzzy sets of grade values in the result would be better presented in linguistic form. Approximating a fuzzy set in linguistic form by some appropriate hedges and fuzzy sets already defined will lead to a good man-machine interface for communicating naturally.

CONCLUSIONS

The fuzzy database defined in this paper has very wide flexibility and applicability because we need not get the well-defined data in the description of the real world.

We have described several interpretation methods for the applications of alpha expressions to fuzzy relations, especially, fuzzy relations of fuzzy sets in this paper.

We must investigate the processing of existential and universal quantifiers, the introduction of fuzzy quantifiers *many*, *few* and so on. The theory of normal form of fuzzy relational database and a relational completeness of fuzzy alpha expression are very interesting.

REFERENCES

CODD, E.F. (1970): A Relational Model of Data for Large Shared Data Banks. Communications of the ACM, 13, 377-387.

CODD, E.F. (1971): A Data Sublanguage Founded on the Relational Calculus. 1971 ACM SIGFIDET Workshop on Data Description, Access and Control, 35-68.

CODD, E.F. (1972): Relational Completeness of Data Base Sublanguages. Data Base Systems, 65-98 (Ed. Rustin, R.), Englewood Cliffs, N.J.: Prentice-Hall.

DATE, C.J. (1977): An Introduction to Database Systems, Second Edition. Reading, Mass.: Addison-Wesley.

KUNII, T.L. (1976): DATAPLAN: An Interface Generator for Database Semantics. Information Sciences, 10, 279-298.

UMANO, M., MIZUMOTO, M. and TANAKA, K. (1978a): FSTDS System: A Fuzzy-Set Manipulation System. Information Sciences, 14, 115-159.

UMANO, M., MIZUMOTO, M., TANAKA, K. (1978b): Implementation of Approximate Reasoning System Using FSTDS System. International Colloquium on Fuzzy Set Theory and Applications, Marseille, France, September 20-22, 1978.

UMANO, M., MIZUMOTO, M., TANAKA, K. (1979): A System for Fuzzy Reasoning. Sixth International Joint Conference on Artificial Intelligence, 917-919, Tokyo, Japan, August 20-23, 1979.

ZADEH, L.A. (1965): Fuzzy Sets. Information and Control, 8, 338-353.

ZADEH, L.A. (1972): A Fuzzy-Set-Theoretic Interpretation of Linguistic Hedges, Journal of Cybernetics, 2, 4-34.

ZADEH, L.A. (1973): Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. IEEE Transactions on System, Man and Cybernetics, SMS-3, 28-44.

ZADEH, L.A. (1975): Fuzzy Logic and Approximate Reasoning. Synthese, 30, 407-428.

ZADEH, L.A. (1978): PRUF — A Meaning Representation Language for Natural Languages. International Journal of Man-Machine Studies, 10, 395-460.