A Fuzzification of the Relational Data Model

Doheon Lee, Myoung Ho Kim, Hyung Lee-Kwang and Yoon-Joon Lee

Dept. of Computer Science,
Korea Advanced Institute of Science and Technology,
373-1, Kuseong-dong, Yuseong-gu,
Taejon, 305-701, South Korea
Tel: +82-42-869-3563
Fax: +82-42-869-3510
E-mail : doheony@dbserver.kaist.ac.kr

Abstract

Expression and processing of vagueness, which has many real world applications, is not handled effectively in the conventional relational model. In this paper we investigate a fuzzy extension to the relational data model and propose three fuzzy relational query languages. Two of them are the *Level-1 Fuzzy Relational Algebra* and *Level-1 Fuzzy Relational Calculus*. They are fundamental query languages and serve as a theoretical framework for the fuzzy relational database. Based on them, the *Fuzzy Selective Relational Algebra* is presented to express fuzzy constants and fuzzy comparators, which are more effective to represent vagueness in user queries. We show that the three proposed query languages have the same expressive powers.

Keywords: fuzzy database, relational algebra, relational calculus, relational data model

1. Introduction

The relational data model proposed by E.F.Codd has been widely used due to its effective data independency and its simple mathematical structure[COD70]. However, the relational data model has several limitations. One of them is lack of dealing with subjective vagueness in user's data retrieval requests. Many efforts to introduce vagueness into the theory of the relational data model have been made in the past. They can be classified into two major categories, i.e. Crisp Data and Fuzzy Query(CDFQ) and Fuzzy Data and Fuzzy Query(FDFQ) categories. In the CDFQ category, queries with fuzzy concepts can be processed for database storing only crisp values[LEE92, ICH86, KAC86, MOT88, WON90]. In the FDFQ category, queries with fuzzy concepts can be processed for database which can store fuzzy values[BUC82A, BUC82B, PRA84, UMA91, ZEM85, ZEM89].

To be accepted by most database users, fuzzy database systems need to have sufficient compatibilities with conventional database systems[MOT90]. Even though the FDFQ approaches can greatly enhance database functionalities, they are too far from conventional database systems yet. On the other hand, the CDFQ approaches are more compatible with conventional database systems. They can also enhance database functionalities significantly.

Among the CDFQ approaches, ARES[ICH86] and VAGUE [MOT88] are worthy of notice. ARES and VAGUE introduced the 'similar-to' comparator into the relational algebra. They can deal with vagueness to some extent but cannot deal with fuzzy concepts such as "big", "about-5", etc. In this paper we propose a fuzzified relational data model to accomodate vagueness using the CDFQ approach.

The remainder of this paper is organized as follows. In section 2, we present the Level-1 Fuzzy Relational Algebra(FRA-1) and Level-1 Fuzzy Relational Calculus(FRC-1). They are fundamental query languages of the proposed model. In section 3, we define the Fuzzy Selective Relational Algebra(FSRA) to express fuzzy constants and fuzzy comparators. It is a query language which is more effective to represent vagueness in user queries. We also show that FRA-1, FRC-1 and FSRA have the same expressve powers. Section 4 analyses various aspects of the proposed model. Finally we give concluding remarks in section 5.

2. The Level-1 Fuzzy Relational Data Model

In this section we define two fundamental query languages in the proposed level-1 fuzzy relational data model. They are Level-1 Fuzzy Relational Algebra(FRA-1) which is a fuzzy extension of the relational algebra, and Level-1 Fuzzy Relational Calculus(FRC-1) which is a fuzzy extension of the relational calculus. As the relational algebra and the relational calculus are defined on relations, FRA-1 and FRC-1 are defined on level-1 fuzzy relations. A level-1 fuzzy relation is a subset of cartesian product of level-1 fuzzy sets[KLI88].

DEFINITION 1

Suppose that $D_1, D_2, ..., D_k$ are domains. Then a *level-1 fuzzy relation* R is defined as

$R = \{ (t, \mu_R(t)) \mid t = < x_1, x_2, ..., x_k >, 0 < \mu_R(t) \le 1, x_1 \in D_1, x_2 \in D_2, ..., x_k \in D_k \}$,

where $\mu_R(t)$ denotes the degree to which tuple t belongs to the level-1 fuzzy relation R.

For simplicity, we will use the term "fuzzy relation" to refer to *level-1 fuzzy relation* .

2.1. The Level-1 Fuzzy Relational Algebra

The Level-1 Fuzzy Relational Algebra(FRA-1) is a collection of formal operators acting on fuzzy relations and producing fuzzy relations as results. It is an extension of the conventional relational algebra by using the extension principle[KLI88].

DEFINITION 2

The *Level-1 fuzzy relational algebra* has six basic operators i.e. σ, Π, $\cup$, $\cap$, $\times$, $-$. Suppose that there are two fuzzy relations R_1 and R_2.

(1) Selection : σ

$\sigma_{X \Theta Y}(R_1) = \{ (t, \mu_{\sigma_{X \Theta Y}(R_1)}(t)) \mid X \Theta Y$ with respect to t is true, $\mu_{\sigma_{X \Theta Y}(R_1)}(t) = \mu_{R_1}(t) \}$,

where X is an attribute name, Y is either an attribute name or a crisp constant and Θ is a comparator among $=, \ne, \le, \ge, >$ and $<$.

(2) Projection : Π

$\Pi_S(R_1) = \{ (t, \mu_{\Pi_S(R_1)}(t)) \mid \mu_{R_1}(t') > 0,\ t = \text{sub-list}(t' \mid S),$
$\mu_{\Pi_S(R_1)}(t) = \text{MAX}[\mu_{R_1}(t_i)],\ t_i = \text{same-project-set}(t' \mid S) \}$,

where sub-list(t' | S) is a projected tuple of t' with respect to the attribute list S and same-project-set(t' | S) denotes $\{ t^* \mid \text{sub-list}(t^* \mid S) = \text{sub-list}(t' \mid S) \}$

(3) Union : $\cup$

$R_1 \cup R_2 = \{ (t, \mu_{R_1 \cup R_2}(t)) \mid \mu_{R_1 \cup R_2}(t) = \text{MAX}(\mu_{R_1}(t), \mu_{R_2}(t)) \}$

(4) Intersection : $\cap$

$R_1 \cap R_2 = \{ (t, \mu_{R_1 \cap R_2}(t)) \mid \mu_{R_1 \cap R_2}(t) = \text{MIN}(\mu_{R_1}(t), \mu_{R_2}(t)) \}$

(5) Cartesian Product : $\times$

$R_1 \times R_2 = \{ (t, \mu_{R_1 \times R_2}(t)) \mid \mu_{R_1}(t_1) > 0, \mu_{R_2}(t_2) > 0,\ t = \text{concatenate}(t_1, t_2).\ \mu_{R_1 \times R_2}(t) = \text{MIN}(\mu_{R_1}(t_1), \mu_{R_2}(t_2)) \}$

Here, if $t_1 = < x_1, x_2, ..., x_p >$ and $t_2 = < y_1, y_2, ..., y_q >$, then concatenate$(t_1, t_2) = < x_1, x_2, ..., x_p, y_1, y_2, ..., y_q >$.

(6) Difference : $-$

$R_1 - R_2 = \{ (t, \mu_{R_1 - R_2}(t)) \mid \mu_{R_1 - R_2}(t) = \text{MIN}(\mu_{R_1}(t), 1 - \mu_{R_2}(t)) \}$

When we restrict the value in μ attribute to be either 0 or 1, it is easy to see that FRA-1 is reduced to the relational algebra. Though we use MIN and MAX operators to represent AND and OR semantics, respectively, they were just for illustrations. Depending on the circumstances, we may use any t-norm, t-conorm operators[KLI88] instead of MIN and MAX. Note that unlikely to the relational algebra, the intersection operator cannot be represented by the combination of the other five operators.

2.2. The Level-1 Fuzzy Relational Calculus

We define Level-1 Fuzzy Relational Calculus(FRC-1) by applying the extension principle to the relational calculus. In fact, formulas in FRC-1 are syntactically equivalent to those in the domain relational calculus. Their interpretations are extended to handle fuzzy truth values.

DEFINITION 3

A *Formula in FRC-1* is either an atomic formula or compound formula.

(1) Atomic formula

- Every literal $p(X_1, X_2, ..., X_k)$ is an atomic formula, where p is a fuzzy predicate symbol[KLI88] and $X_1, X_2, ..., X_k$ are attribute names.
- Every arithmetic comparison $X \Theta Y$ is an atomic formula, where X is an attribure name, Y is either a constant or an attribute name and Θ is a comparator among $=, \ne, \le, \ge, >$ and $<$.

(2) Compound formula

Compound formulas are defined recursively. If f_1 and f_2 are formulas and X is an attribute name, the followings are also formulas.

$f_1 \wedge f_2$, $f_1 \vee f_2$, $\sim f_1$, $(\exists X) f_1(X)$, $(\forall X) f_1(X)$.

As in the relational calculus, each formula in FRC-1 represents a fuzzy relation i.e. *interpretation*. To make domain independent interpretations, we adopt the same safety criteria[ULL88] in FRC-1. From now on, we will use the term "FRC-1 formula" to refer to the safe FRC-1 formula because unsafe formula is beyond the scope of this paper.

DEFINITION 4

The followings are *interpretations of FRC-1 formulas*. We use lower-case letters to denote the formulas and upper-case letters to denote the corresponding fuzzy relations. v(f) denotes the (fuzzy) truth value of the formula f.

(1) When the formula f is $p(X_1, X_2, \dots, X_k)$,

$F = \{ (t, \mu_F(t)) \mid t = \langle X_1, X_2, \dots, X_k \rangle,\ \mu_F(t) = v(p(X_1, X_2, \dots, X_k)) > 0 \}$.

(2) When the formula f is $X \Theta Y \wedge f_2 (X_1, \dots, X_k)$,

$F = \{ (t, \mu_F(t)) \mid t =$ a list of free variables of $f,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v(X \Theta Y \wedge f_2) = MIN (v(X \Theta Y), v(f_2))$.

(3) When the formula f is $f_1 \wedge f_2$,

$F = \{ (t, \mu_F(t)) \mid t =$ a list of free variables of $f,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v(f_1 \wedge f_2) = MIN (v(f_1), v(f_2))$.

(4) When the formula f is $f_1 \vee f_2$,

$F = \{ (t, \mu_F(t)) \mid t =$ a list of free variables of $f,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v(f_1 \vee f_2) = MAX (v(f_1), v(f_2))$.

(5) When the formula f is $\sim f_1 \wedge f_2$,

$F = \{ (t, \mu_F(t)) \mid t =$ a list of free variables of $f,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v(\sim f_1 \wedge f_2) = MIN (1 - v(f_1), v(f_2))$.

(6) When the formula f is $(\exists X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)$,

$F = \{ (t, \mu_F(t)) \mid t = \langle Y_1, \dots, Y_p, Z_1, \dots, Z_q \rangle,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v((\exists X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q))$
$= MAX_X (v(f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)))$.

(7) When the formula f is $(\forall X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)$,

$F = \{ (t, \mu_F(t)) \mid t = \langle Y_1, \dots, Y_p, Z_1, \dots, Z_q \rangle,\ \mu_F(t) = v(f) > 0 \}$,

where $v(f) = v((\forall X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q))$
$= MIN_X (v(f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)))$.

In (2) and (5) of Defintion 4, we ANDed secondary formula f_2 with the original single formula in Definition 3. In any safe formula, such a single formula cannot exist alone. It has to be ANDed with another safe formula. As in FRA-1, when we restrict the truth value of each formula to be either 0 or 1, FRC-1 can be easily shown to be reduced to the relational calculus.

2.3. The Relationship between FRA-1 and FRC-1

We show that the expressive power of FRA-1 and FRC-1 is equivalent to each other. First, we show that the expressive power of FRC-1 is greater than or equal to that of FRA-1.

THEOREM 1

Every query expressible in FRA-1 is expressible in FRC-1.

Proof:

We show by induction that for every expression e of FRA-1 defining a k-ary fuzzy relation, there is a formula $f(X_1, X_2, \dots, X_k)$ of FRC-1 defining the same relation. We use E, E_1 and E_2 to denote the fuzzy relations of FRA-1 expressions e, e_1 and e_2, respectively. We also use F, F_1 and F_2 to denote the fuzzy relations corresponding to the FRC-1 formulas f, f_1 and f_2, respectively.

The basis covers the case where e is a single fuzzy relation R. If we use a fuzzy predicate r to represent a fuzzy relation R in FRC-1 formula, the corresponding formula to e is trivially $r(X_1, X_2, \dots, X_k)$.

For the induction, we consider six cases corresponding to the six basic operators of FRA-1.

Case 1: $e = e_1 \cup e_2$

We show that the FRC-1 formula, $f = f_1 \vee f_2$ represents the same fuzzy relation as E. By defintions,

$E = \{ (t , \mu_E(t)) \mid \mu_E(t) = MAX(\mu_{E_1}(t), \mu_{E_2}(t)) \}$ .... < 1.1 >

$F = \{ (t , \mu_F(t)) \mid \mu_F(t) = v(f) = v(f_1 \vee f_2) \}$
$= \{ (t , \mu_F(t)) \mid \mu_F(t) = MAX(v(f_1(t)), v(f_2(t))) \}$. ... < 1.2 >

By the induction hypothesis, we already know that E_1 and E_2 are the same fuzzy relations as F_1 and F_2, respectively. So,

$\mu_{F_1}(t) = \mu_{E_1}(t)$ and $\mu_{F_2}(t) = \mu_{E_2}(t)$ .... < 1.3 >

Now, by the interpretation of FRC-1,

$\mu_{F_1}(t) = v(f_1(t))$ and $\mu_{F_2}(t) = v(f_2(t))$ .... < 1.4>

From equations <1.1>, <1.2>, <1.3> and <1.4>

$\mu_E(t) = MAX(\mu_{E_1}(t), \mu_{E_2}(t)) = MAX(\mu_{F_1}(t), \mu_{F_2}(t)) = MAX(v(f_1(t)), v(f_2(t))) = \mu_F(t)$.

Thus, E is equal to F.

Case 2: $e = e_1 \cap e_2$

By the similar procedure to Case 1 except substituting $\cup$ and MAX with $\cap$ and MIN, respectively, we can easily show that the formula, $f_1 \wedge f_2$ represents the same fuzzy relation as E.

Case 3: $e = e_1 - e_2$

In the same manner as Case 1, we can easily show that the formula, $f_1 \wedge \sim f_2$ represents the same fuzzy relation as E.

Case 4: $e = e_1 \times e_2$

In the same manner as Case 1, we can easily show that the formula, $f_1(X_1, \dots, X_p) \wedge f_2(Y_1, \dots, Y_p)$ represents the same fuzzy relation as E when $E_1 = \{ (t_1 , \mu_{E_1}(t_1)) \mid t_1 = \langle x_1, \dots, x_p \rangle \}$ and $E_2 = \{ (t_2 , \mu_{E_1}(t_2)) \mid t_2 = \langle y_1, \dots, y_q \rangle \}$.

Case 5: $e = \sigma_{X\Theta Y}(e_1)$

In the same manner as Case 1, we can easily show that the formula, $f_1 \wedge X \Theta Y$ represents the same fuzzy relation as E.

Case 6: $e = \Pi_{X_1, X_2, \ldots, X_p}(e_1)$

Suppose that attribute list of E_1 is $(X_1, X_2, \ldots, X_p, Z_1, Z_2, \ldots, Z_q)$. In the same manner as Case 1, we can easily show that the formula, $f(X_1, X_2, \ldots, X_p) = (\exists Z_1)(\exists Z_2)\ldots(\exists Z_p) f_1(X_1, X_2, \ldots, X_p, Z_1, Z_2, \ldots, Z_q)$ represents the same fuzzy relation as E.

Q. E. D.

The next theorem shows that the expressive power of FRA-1 is greater than or equal to that of FRC-1. Then by Theorem 1, we can conclude that FRA-1 and FRC-1 have equivalent expressive powers.

THEOREM 2

Every query expressible in safe FRC-1 is expressible in FRA- 1.

Proof:

We show that for every query f of FRC-1 representing a k-ary fuzzy relation, there is an expression e of FRA-1 defining the same relation by using induction. In addition to the same notational conventions as those in Theorem 1, we use $Z'_1, Z'_2, \ldots,$ and Z'_r to denote the same attribute names $Z_1, Z_2, \ldots,$ and Z_r, respectively.

The basis covers the case where f is an atomic formula $r(X_1, X_2, \ldots, X_k)$. If we use fuzzy predicate r to represent the fuzzy relation R in FRC-1 formula, the corresponding FRA-1 expression is trivially $R(X_1, X_2, \ldots, X_k)$.

For the induction, we consider six cases corresponding to the six basic connectives and quantifiers of FRC-1.

Case 1 : $f = X \Theta Y \wedge f_1(X_1, \ldots, X_p)$ **and** FV **is a list of free variables.**

According to the FV, we can consider three sub-cases. The other possibilities are excluded due to the safety criteria.

If $FV = (X_1, \ldots, X_p)$,

We've already prove that the above formula and the expression $e = \sigma_{X\Theta Y}(e_1)$ of FRA-1 represent the same fuzzy relations in Case 5 of Theorem 1.

If $FV = (X_1, \ldots, X_p, X)$, Y is a constant α and $\Theta = "="$,

We show that FRA-1 expression, $e = e_1 \times \{ (\alpha, 1.0) \}$ generates the same fuzzy relation as F. By definitions,

$$F = \{ (t, \mu_F(t)) \mid \mu_F(t) = v(f) = v(X = \alpha \wedge f_1) \}$$
$$= \{ (t, \mu_F(t)) \mid \mu_F(t) = MIN(v (X = \alpha), v(f_1)) \}$$
$$= \{ (t, \mu_F(t)) \mid \mu_F(t) = v(f_1) \} \quad \ldots\ldots <2.1>$$
$$E = \{ (t, \mu_E(t)) \mid \mu_E(t) = \mu_{E_1}(t_1) \} \quad \ldots\ldots <2.2>$$

By the induction hypothesis, we already know that E_1 is the same fuzzy relations as F_1. So,

$$\mu_{F_1}(t_1) = \mu_{E_1}(t_1) \quad \ldots\ldots <2.3>$$

Now, by the interpretation of FRC-1,

$$\mu_{F_1}(t_1) = v(f_1(t_1)) \quad \ldots\ldots <2.4>$$

From equations <2.1>, <2.2>, <2.3>, and <2.4>

$$\mu_E(t) = \mu_{E_1}(t_1) = \mu_{F_1}(t_1) = v(f_1(t_1)) = \mu_F(t).$$

Thus F is equal to E.

If $FV = (X_1, \ldots, X_p, X)$, $Y \in \{ X_1, \ldots, X_p \}$ and $\Theta = "="$,

By th similar manner, we can easily show that FRA-1 expression, $e = \sigma_{Y'=Y}(e_1 \times \Pi_Y(e_1))$ generates the same fuzzy relation as F.

Case 2 : $f = f_1(X_1, X_2, \ldots, X_p, Z_1, Z_2, \ldots, Z_r) \wedge f_2 (Y_1, Y_2, \ldots, Y_q, Z'_1, Z'_2, \ldots, Z'_r)$,

We show that the FRA-1 expression $e = \Pi_{X_1, \ldots, X_p, Y_1, \ldots, Y_q, Z_1, \ldots, Z_r}(\sigma_{Z_1 = Z_1'}(\ldots (\sigma_{Z_r = Z_r'}(e_1 \times e_2)) \ldots))$, represents the same fuzzy relation as F. By defintions,

$$F = \{ (t, \mu_F(t)) \mid \mu_F(t) = v(f) = v(f_1 \wedge f_2) \}$$
$$= \{ (t, \mu_F(t)) \mid \mu_F(t) = MIN(v(f_1(t)), v(f_2(t))) \}. \ldots <2.5>$$
$$E = \{ (t, \mu_E(t)) \mid \mu_E(t) = MIN(\mu_{E_1}(t_1), \mu_{E_2}(t_2)) \} \ldots\ldots <2.6>$$

By the induction hypothesis, we already know that E_1 and E_2 are the same fuzzy relations to F_1 and F_2, respectively. So,

$$\mu_{F_1}(t) = \mu_{E_1}(t) \text{ and } \mu_{F_2}(t) = \mu_{E_2}(t) \quad \ldots\ldots <2.7>$$

And by the interpretation of FRC-1,

$$\mu_{F_1}(t) = v(f_1(t)) \text{ and } \mu_{F_2}(t) = v(f_2(t)) \quad \ldots\ldots <2.8>$$

From the equations <2.5>, <2.6>, <2.7>, and <2.8>

$$\mu_E(t) = MAX(\mu_{E_1}(t), \mu_{E_2}(t)) = MAX(\mu_{F_1}(t), \mu_{F_2}(t))$$
$$= MAX(v(f_1(t)), v(f_2(t))) = \mu_F(t).$$

Thus F is equal to E.

Case 3 : $f = f_1(X_1, X_2, \ldots, X_k) \vee f_2(X_1, X_2, \ldots, X_k)$,

We've already prove the above formula and the expression $e_1 \cup e_2$ of FRA-1 represent the same fuzzy relations in the Case 1 of Theorem 1.

Case 4 : $f = \sim f_1(Z_1, Z_2, \ldots, Z_r) \wedge f_2 (Y_1, Y_2, \ldots, Y_q, Z'_1, Z'_2, \ldots, Z'_r)$,

By the induction hypothesis, we already know that E_1 and E_2 are the same fuzzy relations to F_1 and F_2, respectively. We denote $DOM(f_1)$ to refer to the set of all values that appear in the formula f_1 itself and the corresponding fuzzy relation F_1, except the rational numbers for the degrees of memberships[Ull88]. Under the domain independency, the corresponding fuzzy relation to $\sim f_1$ is $DOM(f_1)^r - E_1$. So, this is a special case of the Case 1.

Case 5 : $f = (\exists X) f_1(Y_1, \ldots, Y_p, X, Z_1, \ldots, Z_q)$,

We have already proved the above formula and the expression $\Pi_{Y_1, \dots, Y_p, Z_1, \dots, Z_q}(e_1)$ of FRA-1 represent the same fuzzy relation at the Case 3 of Theorem 1.

Case 6 : $f = (\forall X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)$,

It is easy to see that $(\forall X) f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q) = \sim(\exists X) \sim f_1(Y_1, \dots, Y_p, X, Z_1, \dots, Z_q)$. So, this case is already covered in Case 4 and Case 5.

Q. E. D.

COROLLARY 1

FRA-1 and FRC-1 have the same expressive powers

Proof:

From Theorem 1 and Theorem 2, the collorary immediately follows.

Q. E. D.

In the conventional relational model, if a query language can express all of the basic operators of the relational algebra, we call them relationally complete. For the case of the fuzzy relational model, we extend the notion of the relational completeness. We can regard a query language as a *level-1 fuzzy relationally complete* language if it can express all operators of FRA-1.

3. Fuzzy Constructs in Fuzzy Relational Query Languages

FRA-1 and FRC-1 are fundamental query languages of the proposed level-1 fuzzy relational data model. Since they are straightforward extensions to the counterparts in the conventional relational data model, we do not introduce any new syntactic constructs into them.

In order to introduce constructs expressing vagueness, the Fuzzy Selective Relational Algebra(FSRA) has been proposed in [LEE92]. In this section, we briefly describe FSRA and show that its expressive power is equivalent to those of FRA-1 and FRC-1.

3.1. The Fuzzy Seletive Relational Algebra

Because the vagueness in user's data requests are mostly expressed through selection predicates, the selection operator, i.e. σ, plays a major role in fuzzy query formulations. We introduce fuzzy constants and fuzzy comparators into the selection operator to facilitate expression of vagueness.

In contrast to the conventional (crisp) constant, a fuzzy constant is defined as a fuzzy set. Examples of the fuzzy constants are "tall", "small" and "about-5". While conventional comparators are used to represent crisp comparisons, the fuzzy comparators, $\approx$, $!\approx$, are used to represent similarity-based comparisons.

DEFINITION 5

As in FRA-1, the *Fuzzy Selective Relational Algebra(FSRA)* has six basic operators i.e. σ^*, Π, $\cup$, $\cap$, $\times$, $-$. Only selection operator is further extended to express fuzzy constants and fuzzy comparators. The other operators are all the same as those of FRA-1.

Selection operator of FSRA is defined as

$$\sigma^*_{X \Theta Y}(R),$$

where X is an attribute name , Θ is a comparator among $=$, $\neq$, $\leq$, $\geq$, $>$, $<$, $\approx$, $!\approx$ and Y is either an attribute name or a constant. Here the constant is either a crisp constant or a fuzzy constant.

By using FSRA, we can express a vague query such as "Find part whose weight is heavy and color is similar to red". "Heavy" and "similar to" are fuzzy terms. Fuzzy constants and fuzzy comparators are used to express those fuzzy terms as follows.

$$\sigma^*_{\text{weight} = \text{heavy}}(\sigma^*_{\text{color} \approx \text{"red"}}(\text{PART})),$$

where PART is a fuzzy relation whose tuples are descriptions of parts. Since conventional relations can be considered as fuzzy relations whose tuples belong to the relations at degree 1, FSRA can also be used on conventional databases.

To store semantics of the fuzzy constants and fuzzy comparators, we need special kinds of relations. We call them *semantic relations*, which can be classified into three categories.

	fuzzy constant	fuzzy comparator
continuous domain	(1)	
scattered domain	(2)	(3)

Figure 1. Classification of semantic relations

The classification is based on the domains and structures of the stored informations. Because a relation can hold only discrete data, we need an approximation to store information on continuous domain. But in the case of scattered domain, we do not need such an approximation.

A fuzzy constant is expressed as a fuzzy set(unary fuzzy relation), but a fuzzy comparator is expressed as a binary fuzzy relation. A fuzzy comparison on continuous domain can be substituted by a fuzzy predicate using a fuzzy constant on continuous domain. Let's see an example. The fuzzy comparison such that " X is similar to 5.0" can be substituted by the fuzzy predicate such that " X is about-5.0". So we exclude the case of fuzzy comparators on continuous domains.

We present the schema of each semantic relation.

(1) Fuzzy Constant on Continuous domain

{ (t, μ(t)) , t = < lower-value , upper-value >, 0 < μ(t) $\leq$ 1 }

where μ(t) denotes the degree to which the values in [lower-value, upper-value] conforms to the fuzzy concept of this semantic relation. An example semantic relation of this type is shown in Figure 2.

Domain of Height = [0, 200]

High	LowerValue	UpperValue	μ
	0	15	0.0
	15	50	0.0
	50	85	0.1
	85	125	0.3
	125	160	0.5
	160	185	0.7
	185	200	1.0

graphical representation

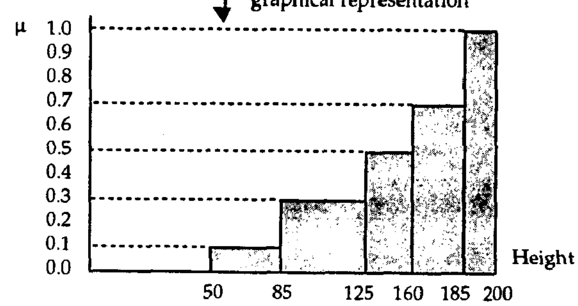

Figure 2. An example of the semantic relation.

(2) Fuzzy Constant on Scattered domain

{ (t, μ(t)) , t = < value >, 0 < μ(t) $\leq$ 1 }

where μ(t) denotes the degree to which "value" conforms to the fuzzy concept of this semantic relation.

(3) Fuzzy Comparator on Scattered domain

{ (t, μ(t)) , t = < A-value , B-value >, 0 < μ(t) $\leq$ 1 }

where μ(t) denotes the degree to which "A-value" is similar to "B-value".

3.2. Relationships among FSRA, FRA-1 and FRC-1

We have defined FRA-1, FRC-1 and FSRA. The relationships among those three query languages are shown in Figure 3.

Since we have shown that FRA-1 and FRC-1 have the same expressive powers in Corollary 1, the proof that the expressive power of FSRA is equal to FRA-1 implies that all three query languages have the same expressive powers. Clearly, th expressive power of FSRA is greater than or equal to that of FRA-1 because FSRA is a strict extension of FRA-1. If we show that the expressive power of FRA-1 is greater than or equal to that of FSRA, we can conclude that the two languages have the same expressive powers. Note that FSRA ia a level-1 fuzzy relationally complete language.

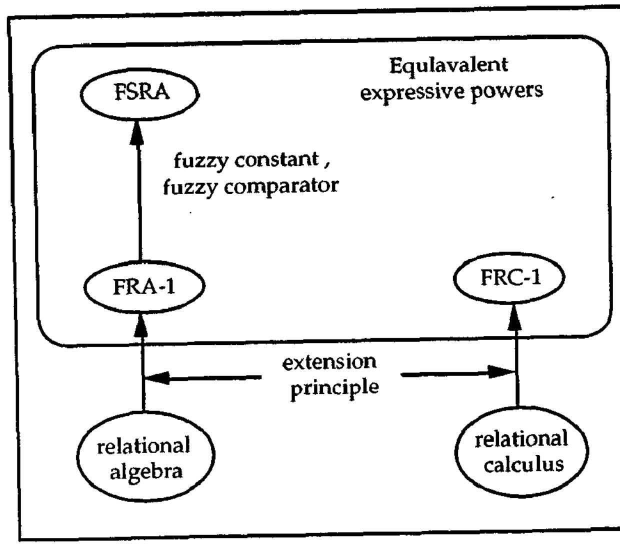

Figure 3. Relationships among FRA-1, FRC-1 and FSRA

<u>THEOREM 3</u>

Every query expressible in FSRA is expressible in FRA-1.

Proof:

We only extend the selection condition of FRA-1 to define FSRA. If we can transform any query in FSRA to that of FRA-1, the proof is completed. The following investigate each case of the extended features.

Case 1: Fuzzy Constant on Continuous Domain

$\sigma_{X_i = fuzzy\_term}(R)$
$= \Pi_{X_1, \ldots, X_k}(\sigma_{X_i \geq LowerValue}(\sigma_{X_i < UpperValue}(R \times SR_{fuzzy\_term})))$

$\sigma_{X_i \neq fuzzy\_term}(R)$
$= \Pi_{X_1, \ldots, X_k}(\sigma_{X_i \geq LowerValue}(\sigma_{X_i < UpperValue}(R \times \overline{SR}_{fuzzy\_term})))$

where SR_{fuzzy_term} is a semantic relation containing the information of fuzzy term and $\overline{SR}_{fuzzy\_term}$ is the complement of it.

Case 2: Fuzzy Constant on Scattered Domain

$\sigma_{X_i = fuzzy\_term}(R) = \Pi_{X_1, \ldots, X_k}(\sigma_{X_i = Value}(R \times SR_{fuzzy\_term}))$

$\sigma_{X_i \neq fuzzy\_term}(R) = \Pi_{X_1, \ldots, X_k}(\sigma_{X_i = Value}(R \times \overline{SR}_{fuzzy\_term}))$

where SR_{fuzzy_term} is a semantic relation containing the information of fuzzy term and $\overline{SR}_{fuzzy\_term}$ is the complement of it.

Case 3: Fuzzy Comparator on Continuous Domain

$\sigma_{X \approx Y}(R) = \Pi_{A_1, \ldots, A_k}(\sigma_{A\text{-}Value = X}(\sigma_{B\text{-}Value = Y}(R \times SR_{SIM})))$

$\sigma_{X \,!\approx\, Y}(R) = \Pi_{A_1, \ldots, A_k}(\sigma_{A\text{-}Value = X}(\sigma_{B\text{-}Value = Y}(R \times \overline{SR}_{SIM})))$

where SR_{SIM} is a semantic relation containing the similarity between two values.

<u>COROLLARY 2</u>

FRA-1, FRC-1 and FSRA have the same expressive powers.

Proof:

From Corollary 1 and Theorem 3, the collorary immediately follows. Q.E.D.

4. Analysis of the Level-1 Fuzzy Relational Model

In section 2 and section 3, we have proposed a fuzzy relational model by defining fuzzy relations and fuzzy query languages to accomodate vagueness. This section analyses various aspects of the proposed model and also describes important functional advantages over the conventional relational model.

Effective retrieval for a vague query

Requests for data can be classified into specific requests and vague requests. Vague requests include fuzzy qualifications. While specific requests can be processed effectively in the conventional query systems, vague requests are not. To process vague requests in conventional query systems, users must retry specific queries repeatedly with minor modifications until they match satisfactory data.

As an example, suppose that a user issue a data request such as " Find heavy and long parts" on the database in Figure 5.

PART

No.	Name	Col	Wgt	Hgt	μ
001	nut	red	12.8	160.7	1.0
002	bolt	green	17.2	200.8	1.0
003	screw	blue	17.2	1000.9	1.0
004	screw	red	14.1	1100.9	1.0

Figure 5. An example database

First, the user may formulate a query using the relational algebra such as,

$$\sigma_{Wgt>500}(\sigma_{Len>1000}(\text{PART})).$$

Because there is no tuple satisfying the above qualification, the result relation is null. Then he may modify the query to relax some constraints as follows.

$$\sigma_{\underline{Wgt>50}}(\sigma_{Len>1000}(\text{PART}))$$

Still, the result relation is null. So he may modify the query again such as,

$$\sigma_{\underline{Wgt>15}}(\sigma_{Len>1000}(\text{PART}))$$

Now, a tuple < 003, screw, blue, 17.2, 1000.9 > is retreived. If he becomes tired due to repeated trials of similar queries, he may be satisfied with this result, which we think is not satisfactory.

On the other hand, the proposed FSRA comes up with a solution effectively. To process queries in FSRA, we need semantic relations. Suppose we have semantic relations having semantics of "heavy" and "long" as in Figure 6.

We can express the afore-mentioned data request by using the FSRA as follows

$$\sigma^*_{Wgt=heavy}(\sigma^*_{Len=long}(\text{PART}))$$

It is transformed to a FRA-1 query as follows. For simplicity, we divide the transformed query into two subqueries.

HEAVY

LowerValue	UpperValue	μ
10	11.5	0.0
11.5	14	0.1
14	16	0.5
16	18.5	0.8
18.5	20	1.0

LONG

LowerValue	UpperValue	μ
100	500	0.0
500	800	0.1
800	1000	0.5
1000	1200	0.8
1200	1300	1.0

Figure 6. Sematic relations "HEAVY" and "LONG"

TEMP

$$= \Pi_{No,\,Name,\,Col,\,Wgt,\,Len}(\sigma_{Len \geq LowVal}(\sigma_{Len < UpVal}(\text{PART} \times \text{LONG})))$$

RESULT

$$= \Pi_{No,\,Name,\,Col,\,Wgt,\,Len}(\sigma_{Wgt \geq LowVal}(\sigma_{Wgt < UpVal}(\text{TEMP} \times \text{HEAVY})))$$

After processing the above queries, we have the result as in Figure 7.

No.	Name	Col	Wgt	Hgt	μ
001	nut	red	12.8	160.7	0.0
002	bolt	green	17.2	200.8	0.0
003	screw	blue	17.2	1000.9	0.8
004	screw	red	14.1	1100.9	0.5

Figure 7. Result of a FSRA query

As shown in Figure 7, we get the degree of query conformity for each tuple at one time. Clearly, FSRA is much more effective and flexible in processing vagueness than the conventional relational queries.

Ranking capability

In the case of a specific query, every tuple in the database either conforms to the query completely or does not conform to the query at all, i.e, query conformity is either 1 or 0. However, in the case of a vague query, the query conformity is a matter of degree. Let's suppose that an employer wants to find young employees in his company. Is an employee of 28 years, Tom fit for the employer's demand? How about the other employee of 30 years, John? Though Tom is better fit for the employer's demand, John is also fit for it to some extent. So we must rank the tuples in the answer according to their query conformities. Because the values in the μ attribute represent the query conformities in the proposed model, ranking the tuples according to the conformities to the given query can be easily achieved.

Compatibility with the conventional relational database

To be accepted by most database users, fuzzy database systems should have sufficient compatibilities with conventional database systems[MOT90]. FRA-1 and FRC-1 are strict extensions to the relational algebra and the relational calculus, respectively. The fuzzy relation is also a strict extension to the conventional relation. Thus, the proposed model is a strict extension to the relational data model and well compatible with the conventional database systems. When we restrict values in μ attributes to be either 0 or 1, the proposed model is reduced to the conventional relational model.

Individual qualifications

As a vague linguistic term may have different meanings to different users, fuzzy query systems must interprete the vague queries with individual qualifications. While one think that 1.000 dollars are big money, the other may think that they are not. Fuzzy query systems must support such kinds of individual differences. In FSRA, the meanings of vague terms, i.e. fuzzy constants and fuzzy comparators, are stored in the forms of semantic relations. As semantic relations are handled in the same manner as data relations, by using normal database operations, users can easiliy adjust semantics by modifying the contents of semantic relations. Because mappings from vague terms to the corresponding semantic relations reflect the individual qualifications, users can easily make the query system interprete the vague terms by using their own semantics.

Measure of expressive powers

There have been many efforts to accomodate vagueness in the relational data model. Some of them can handle a large variety of vagueness while the others concentrated on the specific vagueness. The proposed model provides a notion of *level-1 fuzzy relational completeness*. As in the relational data model, we can regard a query language as a level-1 fuzzy relationally complete language if it can express all operations in FRA-1.

5. Concluding Remarks

In this paper we have proposed a fuzzy relational data model. We have presented two fundamental query languages FRA-1 and FRC-1, and have presented a derived query language FSRA to express fuzzy constants and fuzzy comparators. Fuzzy constants and fuzzy comparators in FSRA are effective constructs to deal with vagueness in data retrieval requests. We have shown that FRA-1, FRC-1 and FSRA have the same expressive powers.

Unlikely to query systems accepting specific qualifications, the proposed fuzzy query systems have efficient ranking capabilities. Because the proposed fundamental query languages are strict extensions to the counterparts of the relational data model, they can be directly applicable to conventional relational databases with only additional semantic relations. In other words, the proposed model is well compatible with the conventional relational database systems. As semantic relations are handled in the same manner as data relations, the proposed model effectively support individual qualifications in query interpretations. We have mentioned the notion of level-1 fuzzy relational completeness, which is a fuzzified version of the conventional one. The notion of level-1 fuzzy relational completeness can be used as a theoretical measure of expressive powers of various fuzzy query languages.

We can extend the level-1 fuzzy relation to level-N fuzzy relation, which is a subset of cartesian product of level-N fuzzy sets[KLI88]. Such extensions need to be investigated to manage more fuzziness in the database.

References

[BUC82A] B. Buckles, E. Petry, "Fuzzy Databases and their Applications", *Fuzzy Information and Decision Processes*, M. Gupta, E. Sanchez, Eds., North-Holland., pp. 361-371, 1982

[BUC82B] B. Buckles, E. Petry, "A Fuzzy Representation of Data for Relational Databases", *Fuzzy Sets and Systems*, Vol. 7, pp.213-226, 1982

[COD70] E.F. Codd, "A Relational Model of Data for Large Shared Data Banks", *Comm. ACM*, Vol. 13, No. 6, pp. 377-387, 1970

[LEE92] D.H. Lee, H. Lee-Kwang, M.H. Kim, "A Study on the Fuzzy Selective Relational Algebra", *Proc. Int'l Conf. on Fuzzy Logic and Neural Network*, , pp. 353-356, 1992

[ICH86] T. Ichikawa, "ARES: A Relational Database with the Capability of Performing Flexible Interpretation of Queries", *IEEE Trans. Software Engineering*, Vol. 12, No. 5, 1986

[KAC86] J.Kacprzyk, A. Ziolkowski, "Database Queries with Fuzzy Linguistic Quantifiers", *IEEE Trans. SMC*, Vol. 16, No. 3, pp. 474-478, 1986

[KLI88] G.J. Klir, T.A. Folger, *Fuzzy Sets, Uncertainty, and Information*, Prentice-Hall International, Inc., pp.260-265, 1988

[MOT90] A. Motro, "Accomodating Imprecision in Database Systems: Issues and Solutions", *Data Engineering*, Vol. 13, No. 4, pp. 29-34, 1990

[MOT88] A. Motro, "VAGUE: A User Interface to Relational Databases that permit vague queries", *ACM Trans. on Office Information Systems*, Vol. 6, No. 3, pp. 187-214, 1988

[PRA84] H.Prade, C. Testemale, "Generalization Database Relational Algebra for the Treatment of Incomplete or Uncertain Information and Vague Queries", *Information Science*, Vol 34, pp. 115-143, 1984

[UMA91] M. Umano, Y. Ezawa, "Implementation of SQL-type Data Manipulation Language for Fuzzy Relational Databases", *IFSA Brussels.*, 1991

[WON90] M.H.Wong, K.S.Leung, "A Fuzzy Database-Query Language", *Information Systems*, Vol. 15, No. 5, pp. 583-590, 1990

[ZEM85] M. Zemankova, A. Kandel, "Implementing Imprecision in Information Systems", *Information Sciences*, Vol. 37, pp. 107-141, 1985

[ZEM89] M. ZemanKova, "FIIS: A Fuzzy Intelligent Information System", *Data Engineering*, Vol. 12, No. 2, pp. 11-20, 1989