An Equivalence Relation to Reduce Data Redundancy Based on Fuzzy Object Oriented Database System

Sujoy Dutta, Laxman Sahoo, Debasis Dwibedy

Abstract— In this paper, a fuzzy equivalence relation is defined, generally, superseding most of the established results. The technique of employing sets of values for tuple components to express imprecision and redundancy in relational databases was proposed by Buckles and Petry in their classic works on fuzzy relational databases [1], [2]. By employing finite scalar domains with similarity relations and special fuzzy number domains, Buckles and Petry have demonstrated that the classical properties of uniqueness of tuple interpretations and well-definedness of the relational algebra can be retained in the fuzzy relational database model. The key to the preservation of these properties is the fact that scalar domains with similarity relations and the fuzzy number domains can be partitioned into equivalence classes. However, since equivalence classes can be constructed by assuming the existence of similarity relations, it is desirable to generalize the fuzzy relational database model to one based only on equivalence classes. In this work, we show that the important properties of classical relational databases (and of fuzzy relational databases) are preserved in a generalized model built on equivalence relations on finite database domains.

Index Terms— Domain partitions, Equivalence classes, Equivalence relations, Fuzzy relational databases, Relational algebra.

I. INTRODUCTION

Discovery of Object oriented databases are considered better than the relational and other databases due to increasing demand of new approaches to deal with complex data, complex relationship existing among such data and large data intensive applications. A major goal for database research has been the corporation of additional semantics into the data model. In real-world applications, information is often vague or ambiguous or inexact. Therefore, different kinds of incomplete information or data redundancy have extensively been introduced into relational databases. However, many studies have been carried out on the development of some database models to deal with complex objects and data redundancy together.

The technique of employing subsets of values for tuple components to express imprecision in relational databases was proposed by Buckles and Petry in their classic papers on the fuzzy relational database model [1]-[4].

For various types of reflexive fuzzy relations the reader is referred to [5]. The standard definition of a fuzzy reflexive relation μ in A demands μ (a, a) = 1, which we have seen to be too strong. We have proposed positive values for all μ (a, a) and μ (u, v) ≤ μ (a, a) for all u ≠ v, and a in A. We have shown that with this definition of a fuzzy reflexive relation the redefined fuzzy equivalence relation supersedes most of the theorems proposed by Murali [6].

Since Zadeh introduced the definition of a fuzzy relation from A to B as a fuzzy subset of A×B [7], the theory of fuzzy relations has been developed [5], [8], [9]. Dubois and Prade provided an account of fuzzy relations [10]. More recently, Nemitz has studied fuzzy relations connected with fuzzy relations and fuzzy functions [8]. Murali defined the fuzzy equivalence relation on a set and showed that there exists a correspondence between fuzzy equivalence relations and certain classes of fuzzy subsets [6]. In this equivalence relation, the equivalence class and partition were introduced. The concept of a fuzzy equivalence class was introduced by Zadeh as a natural generalization of the concept of an equivalence class [11]. Ovchinnikov and De Baets et al. defined a fuzzy partition as the set of all fuzzy equivalence classes of some fuzzy equivalence relation [12], [13].

II. OUR CONTRIBUTION

In this paper, an attempt has been made to reduce data redundancy over a database relation. Reduced dada redundancy spanning multiple relations forms an interesting extension to previous work. The data redundancy is described in basis of equivalence relation. Partition and equivalence classes are also used to find out the redundancy easily and efficiently. In that way, a data base without error is described. It is efficient in practice and also applicable in much larger datasets.

III. EQUIVALENCE RELATION

Definition [5]: A fuzzy relation μ in a set A is a fuzzy subset of A×A. μ is reflexive in A if μ(a, a) = 1 and μ is symmetric in A if μ(a, b) = μ(a, b) for all a, b in A.

In mathematics, we often investigate relationships between certain objects (numbers, functions, sets, figures, etc.). If an element 'a' of a set A is related to an element 'b' of a set B, we might write:

a is related to b

or shortly,

a related b

or even more shortly,

a R b.

The essential point is that we have two objects, a and b, that are related in some way.

Manuscript received on May, 2013.

Sujoy Dutta, Computer Science and Engineering, Kalinga Institute of Industrial Technology (KIIT University), Bhubaneswar, India.

Dr. Laxman Sahoo, Computer Science and Engineering, Kalinga Institute of Industrial Technology (KIIT University), Bhubaneswar, India.

Debasis Dwibedy, Computer Science and Engineering, Kalinga Institute of Industrial Technology (KIIT University), Bhubaneswar, India.

Also, we say "a is related to b", not "b is related to a", so the order of a and b is important. In other words, the ordered pair (a, b) is distinguished by the relation. This observation suggests the following formal definitions of a relation.

A. Definition

Let A and B be two sets. A relation R from A into B is a subset of the Cartesian product A×B.

If A and B happen to be equal, we speak of a relation on A instead of using the longer phrase "a relation from A into A".

Equivalence relations constitute a very important type of relations on a set.

B. Definition

Let *A* be a nonempty set. A relation R on A (that is, a subset *R* of A×A) is called an equivalence relation on A if the following hold.

(i) (a,a) ∈ R for all a ∈A,

(ii) if (a,b) ∈R, then (b,a) ∈R (for all a,b∈ A)

(iii) if (a,b) ∈R and (b,c) ∈R, then (a,c) ∈R (for all a,b,c∈ R).

This definition presents the logical structure of an equivalence relation very clearly, but we will almost never use this notation. We prefer to write a~b, or a≈b, or a≡b or some similar symbolism instead of (a,b) ∈ R in order to express that a,b are related by an equivalence relation R. Here, a~ b can be read as "a is equivalent to b". Our definition then assumes the form below.

C. Definition

Let A be a nonempty set. A relation R on A (that is, a subset R of A× A) is called an equivalence relation on A if the following hold.

(i) a∼a for all a∈ A,

(ii) if a∼ b, then b ∼a (for all a,b∈ A),

(iii) if a ∼b and b ∼c then a∼ c (for all a,b,c∈ A).

A relation ~ that satisfies the first condition (i) is called a reflexive relation; one that satisfies the second condition (ii) is called a symmetric relation and one that satisfies the third condition (iii) is called a transitive relation. An equivalence relation is therefore a relation which is reflexive, symmetric and transitive. Notice that symmetry and transitivity requirements involve conditional statements (if..., then...). In order to show that ~ is symmetric, for example, we must make the hypothesis a ~ b and use this hypothesis to establish b ~ a. On the other hand, in order to show that ~ is reflexive, we have to establish a ~ a for all a∈ A, without any further assumption.

Examples: (a) Let A be a nonempty set of numbers and let equality = be our relation. Then = is certainly an equivalence relation on A since

(i) a = a for all a ∈ A,

(ii) if a = b, then b = a (for all a,b ∈ A),

(iii) if a = b and b =c, then a = c (for all a,b,c ∈ A).

(b) Let A be the set of all points in the plane except the origin. For any two points P and R in A, let us put P∼R if R lies on the line through the origin and P.

(i) P∼P for all points P in A since any point lies on the line through the origin and itself. Thus ∼ is reflexive.

(ii) If P ∼R, then R lies on the line through the origin and P; therefore the origin, P, R lie on one and the same line; therefore P lies on the line through the origin and R; and R∼ P. Thus ∼is symmetric.

(iii) If P ∼R and R ∼T, then the line through the origin and R contains the points P and T, so T lies on the line through the origin and P, so we get P T. Thus∼ is transitive. This proves that is an equivalence relation on A.

D. Definition (equivalence relation)

A binary relation R on a set A is an equivalence relation if and only if

(1) R is reflexive

(2) R is symmetric, and

(3) R is transitive.

Example 1: The equality relation (=) on a set of numbers such as {1, 2, 3} is an equivalence relation.

Example 2: The congruent modulo *m* relation on the set of integers i.e. {<a, b>| a≡b (mod *m*)}, where *m* is a positive integer greater than 1, is an equivalence relation.

Note that the equivalence relation on hours on a clock is the congruent mod 12, and that when m = 2, i.e. the congruent mod 2, all even numbers are equivalent and all odd numbers are equivalent. Thus the set of integers are divided into two subsets: evens and odds.

Example 3: Taking this discrete structures course together this semester is another equivalence relation.

Equivalence relations can also be represented by a digraph since they are a binary relation on a set. For example, the digraph of the equivalence relation congruent mod 3 on {0, 1, 2, 3, 4, 5, 6} is shown in Fig. 1. It consists of three connected components.

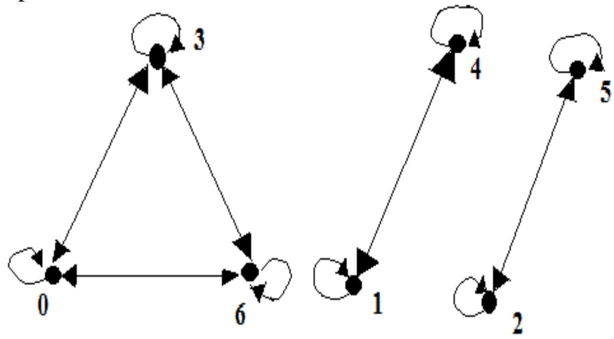

Fig. 1 Example digraph of the equivalence relation

IV. EQUIVALENCE CLASSES FOR EQUIVALENCE RELATION

The set of even numbers and that of odd numbers in the equivalence relation of congruent mod 2, and the set of integers equivalent to a number between 1 and 12 in the equivalence relation on hours in the clock example are called an equivalence class. Formally it is defined as follows:

Let ~ be an equivalence relation on a nonempty set A, and let a be an element of A. The equivalence class of a is defined to be the set of all elements of A that are equivalent to a The equivalence class of a will be denoted by [a] (or by class(a), cl(a), ba or by a similar symbol): [a] = { x ∈ A : x ~ a }.

An element of an equivalence class X ⊆ A is called a representative of X. Notice that x∈ [a] and x ~ a have exactly the same meaning. In particular, we have a∈[a] by reflexivity. So, any a∈A is a representative of its own equivalence class.

In mathematics, given a set X and an equivalence relation ~ on X, the equivalence class of an element n in X is the subset of all elements in X which are equivalent to n. Equivalence classes among elements of a structure are often used to produce a smaller structure whose elements are the classes, distilling

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

a relationship every element of the class shares with at least one other element of another class. This is known as modding out by the class. The class may assume the identity of one of the original elements, as when fractions are put in reduced form.

(1) Let R be an equivalence relation on A and let a ∊ A. The set [a] = {x|aRx} is called the equivalence class of a.

(2) The element in the bracket in the above notation is called the Representative of the equivalence class.

Every element x of X is a member of the equivalence class [x]. Every two equivalence classes [x] and [y] are either equal or disjoint. Therefore, the set of all equivalence classes of X forms a partition of X: every element of X belongs to one and only one equivalence class. Conversely every partition of X comes from an equivalence relation in this way, according to which x ~ y if and only if x and y belong to the same set of the partition. It follows from the properties of an equivalence relation that x ~ y if and only if [x] = [y].

In other words, if ~ is an equivalence relation on a set X, and x and y are two elements of X, these statements are equivalent:

x ~ y,

[x] = [y], and

[x]∩[y] ≠ Ø

Example: Let X be the set of ordered pairs of integers (a,b) with b not zero, and define an equivalence relation ~ on X according to which (a,b) ~ (c,d) if and only if ad = bc. Then the equivalence class of the pair (a,b) can be identified with the rational number a/b, and this equivalence relation and its equivalence classes can be used to give a formal definition of the set of rational numbers. The same construction can be generalized to the field of fractions of any integral domain.

V.EQUIVALENCE CLASSES FOR SIMILARITY RELATIONS

A similarity relation allows us to measure nearness of domain elements. For each domain D**,** a similarity relation **s** is defined over its domain elements, s: D x D→[0, 1] [11]. A similarity relation is the generalization of an equivalence relation. If x, y, z ∊ D, s can be defined as,

reflexive: s(x,x) = 1.0

symmetric: s(x,y) = s(y, x)

transitive: s(x,z) = Max[Min(S(x,y), S(y,z))] for all y∊D

Reflexive, Symmetric, Transitive: implies an equivalence relation.

A similarity-based fuzzy relational database is defined as a set of relations consists of tuples [14]. Fuzzy tuple is any member of a fuzzy relation. For interpretation of a fuzzy tuple, it is essential to select any one element from each set of the tuple. The space of interpretations is the cross product (D1 × D2× … × Dn). Let ti represents the i-th tuple of a relation R in the form (ti1,ti2,..., tim), where tij is defined on the domain set D, 1≤ j ≤ m. Allowing tuple component tij to be a subset of the domain Dj means that fuzzy information can be represented. This leads to the definition that Fuzzy Relation is essentially the subset of the cross product P(D1) × P(D2) × … × P(Dn), where P(D) is the power set of the domain D.

Different degrees of similarity to the elements in each domain are introduced and compared with similarity relation for the representation of "fuzziness" in the fuzzy object-oriented data model based on fuzzy similarity database model. Table I illustrates a simple fuzzy object-oriented database representing NINE EMPLOYERS OF SCHOOL. Fig. 2 shows the similarity relations for domains attributing to JOB-POSITION**,** EXPERIENCE and SALARY respectively [15].

TABLE I A fuzzy object-oriented database relation

EMP#	JOB-POSITION	EXP	SALARY
1	Principal	7	60k
2	Dean	5	60k
3	Clark	2	10k
4	Teacher of Software Engineering	6	42k
5	Teacher of Operating system	3	42k
6	Clark	2	8k
7	Teacher of Software Engineering	3	35k
8	Teacher of Networking	5	60K
9	Teacher of Operating system	2	35k

	Principal	Dean	Clark	Teacher of Software Engineering	Teacher of Operating system	Teacher of Networking
Principal	1	0.5	0.7	0.5	0.4	0.2
Dean	0.5	1	0.5	0.7	0.4	0.2
Clark	0.7	0.5	1	0.5	0.4	0.2
Teacher of Software Engineering	0.5	0.7	0.5	1	0.4	0.4
Teacher of Operating system	0.4	0.4	0.4	0.4	1	0.4
Teacher of Networking	0.25	0.7	0.5	0.7	0.4	1

JOB-POSITION = {Principal, Dean, Teacher of Software Engineering, Teacher of Operating system, Teacher of Networking}

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

	2	3	5	6	7
2	1	0.8	0.6	0.6	0.4
3	0.8	1	0.6	0.6	0.4
5	0.6	0.6	1	0.8	0.6
6	0.6	0.6	0.8	1	0.6
7	0.4	0.4	0.6	0.6	1

EXPERIENCE

	8	10	35	42	60
8	1	0.7	0.5	0.2	0.2
10	0.7	1	0.5	0.2	0.2
35	0.5	0.5	1	0.2	0.2
42	0.2	0.2	0.2	1	0.4
60	0.2	0.2	0.2	0.4	1

SALARY

Fig. 2 Reduce data redundancy using Similarity relation for attribute domains

In Fuzzy, a *similarity relation, S*, is a fuzzy relation which is reflexive, symmetric, and transitive. Let *x, y* be elements of a set *X* and $\mu_s(x,y)$ denotes the grade of membership of the ordered pair (*x,y*) in *S*. Then *S* is a similarity relation in *X* if and only if, for all *x, y, z* in *X*,

$\mu_s(x,x) = 1$ (reflexivity),

$\mu_s(x,y) = \mu_s(y,x)$ (symmetry), and

$\mu_s(x,z) \geqslant \forall\ (\mu_s(x,y)\ Å\ \mu_s(y,z))$ (transitivity),

Where, ∀ and Å denote max and min respectively. Fig. 3 illustrates an example of similarity relation. A partition tree is shown in Fig. 4.

Sim(x, y)	A	B	C	D	E
A	1.0	0.8	0.4	0.5	0.8
B	0.8	1.0	0.4	0.5	0.9
C	0.4	0.4	1.0	0.4	0.4
D	0.5	0.5	0.4	1.0	0.5
E	0.8	0.9	0.4	0.5	1.0

Fig. 3 An example of similarity relation

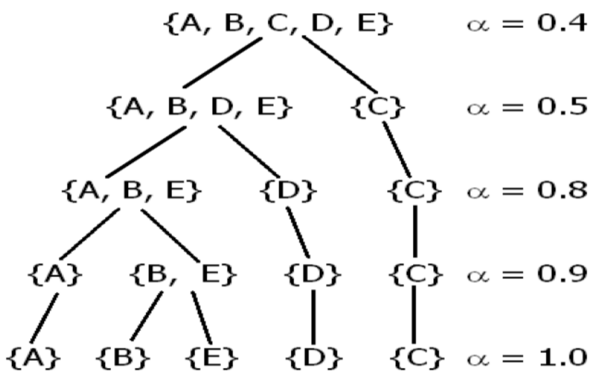

Fig. 4 Partition Tree

The α-level set $S_\alpha = \{(x,y)|Sim(x,y) \geq \alpha\}$ is an equivalence relation, e.g., $S_{0.8} = \{\{A,B,E\}, \{C\}, \{D\}\}$

VI. PARTITIONS AND EQUIVALENCE RELATION

An equivalence relation R on a set A, every element of A is in an equivalence class. For if an element, say b, does not belong to the equivalence class of any other element in A, then the set consisting of the element b itself is an equivalence class. Thus the set A is in a sense covered by the equivalence classes. Another property of equivalence class is that equivalence classes of two elements of a set A are either disjoint or identical, that is either $[a] = [b]$ or $[a] \cap [b] = \emptyset$ for arbitrary elements a and b of A. Thus the set A is partitioned into equivalence classes by an equivalence relation on A. This is formally stated as a theorem below after the definition of partition. An equivalence relation partitions a set into several disjoint subsets as shown in Fig. 5.

Let A be a set and $A_1, A_2, ..., A_n$ be subsets of A. Then $\{A_1, A_2, ..., A_n\}$ is a partition of A, if and only if,

(1) $\bigcup^n_{i=1} A_i = A$, and

(2) $A_i \cap A_j = \emptyset$ for $A_i \neq A_j$, $1 \leq i, j \leq n$

A partition of a set S is a family F of non-empty subsets of S such that

(i) if A and B are in F, then either A=B or A⋂B = Ø and

(ii) $\bigcup_{A \in F} A = S$

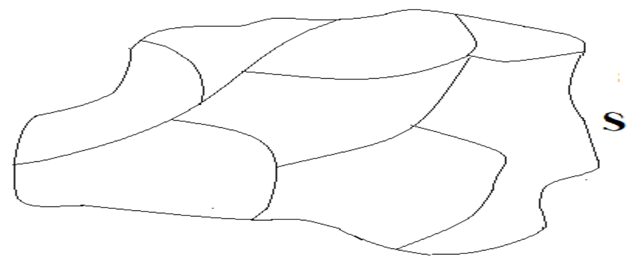

Fig. 5 An equivalence relation partitions a set into several disjoint subsets

If S is a set with an equivalence relation R, it is easy to see the equivalence classes of R form a partition of the set S.

Since F is a partition, for each x in S there is one (and only one) set

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

of F which contains x. Thus, x R x for each x in S (R is reflexive) If there is a set containing x and y then x R y and y R x both hold. (R is symmetric).If xRy and yRz, then there is a set of F containing x and y, and a set containing y and z. Since F is a partition, and these two sets both contain y, they must be the same set. Thus, x and z are both in this set and xRz (R is transitive). Thus, R is an equivalence relation.

In Fuzzy we construct a fuzzy equivalence relation associated with a class of fuzzy subsets on S. We constructed the class of fuzzy subsets $\{\mu_X\}$, x∈ S, on S associated with a fuzzy equivalence relation p. We call this collection as the fuzzy partition of S with respect to μ. It is uniquely determined by μ. Let $\{\mu_j\}_{j\in J}$,J an indexing set be a class of fuzzy subsets on S. It is called a fuzzy partition of S_X, In the follows [6]:

(i) For each x∈ S, there is a unique j∈J such that $\mu_j(S) = 1$.

(ii) For each α∈ [0, 1], W{(μ_j, α)} (j∈J) form a crisp partition of S.

(iii) For each α such that 0< α ≤1,{W(μ_j, α), (j∈J) } is a refinement of {W(μ_j, β), j∈J }With 0≤β≤α≤1.

In this case, we can associate a unique fuzzy equivalence relation μ on S as follows:

$$\mu(x, y) = \sup_{j\in J} (\mu_j(x) \wedge \mu_j(y)) \text{ for all } x, y \in S.)$$

VII. CONCLUSION

The major objective of this paper is to reduce data redundancy over databases. Different degrees of similarity to the elements in each domain are introduced and compared with similarity relation for the representation of "fuzziness" in the fuzzy object-oriented database .An attempt has been used equivalence relation to reduce data redundancy in fuzzy object oriented databases. This approach is based on considering partitions of the relation and equivalence class that deriving valid dependencies from the relation. Partition and equivalence classes are also used to find out the redundancy easily and efficiently. In that way, a data base without error is described. This kind of facility will certainly improve the cooperative nature of objected-oriented databases and enhance the user friendliness of the database systems.

REFERENCES

[1] B. P. Buckles and F. E. Petry, "A fuzzy representation of data for relational databases," Fuzzy Sets and Systems, 7(3), 1982, pp. 213-226.

[2] B. P. Buckles and F. E. Petty, "Information-theoretic characterization of fuzzy relational databases," IEEE Trans. Systems Man Cybernet, 13(1), 1983, pp. 74-77.

[3] B. P. Buckles and F. E. Petry, "Fuzzy databases and their applications," in: M.M. Gupta and E. Sanchez, Eds., Fuzzy Information and Decision Processes, North-Holland, Amsterdam, 1982, pp. 361-371.

[4] B. P. Buckles and F. E. Petry, "Extending the fuzzy database with fuzzy numbers," Inform. Sci., 34(2), 1984, pp. 145-155.

[5] M. Chakraborty and S. Das, On fuzzy equivalence 1, Fuzzy Sets and Systems, Vol. 11, 1983, pp. 185-193.

[6] V. Murali, "Fuzzy equivalence relations," Fuzzy Sets and Systems, Vol. 30, 1989, pp. 155-163.

[7] L. A. Zadeh, "Fuzzy Sets," Inform. and Control, Vol. 8, 1965, pp. 338-353.

[8] W. C. Nemitz, "Fuzzy relations and fuzzy functions," Fuzzy Sets and Systems, Vol. 19, 1986, pp. 177-191.

[9] H. Ounalli, A. Jaoua, "On fuzzy difunctional relations," Inform Sci., Vol. 95, 1996, pp. 219-232.

[10] D. Dubois, H. Prade, "Fuzzy Sets and System Theory and Applications," Academic Press, New York, 1980.

[11] L. A. Zadeh, "Similarity relations and fuzzy orderings," Inform Sci., 3(1), March 1971, pp.177-200.

[12] S. Ovchinnikov, "Similarity relations, fuzzy partitions, and fuzzy orderings," Fuzzy Sets and Systems, Vol. 40, 1991, pp. 107–126.

[13] B. De Baets, G. De Cooman and E. Kerre, "The construction of possibility measures from samples on T-semi-partitions," Inform. Sci., Vol. 106, 1998, pp. 3–24.

[14] B. P. Buckles, and F. E. Petry, "A fuzzy representation of data for relational databases," Fuzzy Sets and Systems, Vol. 7, 1982, pp. 213-226.

[15] Sujoy Dutta, Dr. Laxman Sahoo and Debasis Dwibedy, "Mining Full Functional Dependency to Answer Null Queries and Reduce Imprecise Information Based on Fuzzy Object Oriented Databases," International Journal of Computer Science & Engineering Technology (IJCSET), 4 (03), March 2013, pp. 285-293.

Sujoy Dutta received his B.Tech degree in 2010 from Saroj Mohan Institute of Technology (Techno India) under West Bengal University of Technology, West Bengal and currently pursuing M. Tech in Computer Science and Engineering from KIIT University, Bhubaneswar. He was involved in Java application based project 'Online Examination System' during his B. Tech. His recent research area includes Database systems, Fuzzy Object Databases, Object Modeling and soft computing.

Major Publication:

Sujoy Dutta, Dr. Laxman Sahoo and Debasis Dwibedy, "Mining Full Functional Dependency to Answer Null Queries and Reduce Imprecise Information Based on Fuzzy Object Oriented Databases," *International Journal of Computer Science & Engineering Technology (IJCSET)*, 4 (03), March 2013, pp. 285-293.

Dr. Laxman Sahoo received his Ph.D in 1987 from IIT Kharagpur, West Bengal. Presently, he is professor and head of Database Engg. Group at KIIT University, Odisha. He served as Director/Coordinator/Head of Department in BITS Pilani, BITS Ranchi and Lucknow Indian Engg. College. He has guided over 400 Master Degree students. He is associated with many professional bodies as a member/chair person of technical committee and conference. He has published and presented many Research papers in national and international Journals. He is also author of a good number of computer related books. His research area includes VLSI design, DBMS, AI and Fuzzy Expert Systems.

Debasis Dwibedy received his B.Tech degree in 2010 from BIET, Bhadrak affiliated to BPUT University, Odisha and currently pursuing M.Tech in Computer Science and Engineering from KIIT University, Bhubaneswar. He was involved in Java application based project "Banking automation System" during his B.Tech. He has the research aptitude towards Fuzzy object database modeling and soft computing.