Journal of Uncertain Systems
Vol.2, No.3, pp.212-222, 2008
Online at: www.jus.org.uk

An Algorithm for Discovery of Fuzzy Inclusion Dependencies in Fuzzy Databases

A.K. Sharma [1,*], A. Goswami [2], D.K. Gupta [2]
[1] *M.M.M. Engineering College,Gorakhpur, UP, India*
[2] *Indian Institute of Technology, Kharagpur, WB, India*

Received 15 October 2007; Accepted 2 February 2008

Abstract

Fuzzy inclusion dependencies $(FID_\alpha s, \alpha \in [0,1])$ express subset-relationships between fuzzy databases and are thus important indicators for redundancies between fuzzy databases. In general, the discovery of $FID_\alpha s$ will be beneficial in any effort to integrate unknown fuzzy databases. The problem of searching $FID_\alpha s$ between two fuzzy relations is NP-hard. Therefore, we have mapped the $FID_\alpha s$ searching problem to a weighted hypergraph to reduce it to a clique finding problem in a collection of *k*-hypergraphs. Correctness and complexity of the algorithm are also discussed. By reducing the problem to a weighted hypergraph problem, we achieved a significant improvement in performance over the naive algorithm. Our algorithm uses a NP complete graph algorithm (clique-finding), but a test implementation shows that most of the real world problems can be solved with our approach.
© 2008 World Academic Press, UK. All rights reserved.

Keywords: fuzzy inclusion dependency, fuzzy databases, hypergraph, clique-finding

1 Introduction

Usually, meta information about databases, such as the semantics of schema objects, functional dependencies, or relationships between different databases, is not explicitly available for database integration. A functional dependency is a constraint on a set of attributes $(A_1, A_2, \cdots A_k, X)$ in a relation R, specifying that for any two tuples t_1 and t_2 from R, the following conditions holds:

$$t_1[A_1, A_2, \cdots A_k] = t_2[A_1, A_2, \cdots A_k] \Rightarrow t_1[X] = t_2[X]$$

The derivation of functional dependencies through inference rules has been treated extensively by (Casanova *et al.*, 1982), (Mitchell, 1983), (Missaoui, and Godin, 1990), and (Kantola *et al.*, 1992). The problem of finding evidence for functional dependencies from the extent of relations has also been considered. Several projects deal with the question how to efficiently find candidates for functional dependencies from among the attributes of a relation (Savnik and Flach, 1993), (Bell and Brockhausen, 1995). Functional dependencies and inclusion dependencies are related but have some important differences. In particular, functional dependencies generally are defined only within one relation, whereas the natural purpose of inclusion dependencies is to define relationships across two different relations. Mitchell (1983) also considers inclusion dependencies within one relation. Functional and inclusion dependencies are related in the sense that they both constrain possible valid database states and are thus helpful in database design. However, for our purpose of discovering information about relationships across unknown fuzzy relational databases the case of fuzzy inclusion dependencies is more useful. A reliable algorithm to discover $FID_\alpha s$ will enable an integration system to incorporate new fuzzy relational databases that would not have been used previously since their relationships with existing fuzzy relational database was not known. A simple algorithm compares fuzzy attributes, compares fuzzy relations, and, finally, compares fuzzy databases to discover $FID_\alpha s$ between them. We have already proposed the idea of $FID_\alpha s$ and a naive algorithm for its discovery in a pair of

* Corresponding author. Email: akscse@rediffmail.com (A.K. Sharma)

fuzzy relations, which has recently been published (Sharma *et al.*, 2004). Since the problem is NP-hard, we propose an algorithm to map the problem to a graph problem. Further, we discuss searching for $FID_\alpha s$ using clique finding algorithms and propose an algorithm which uses those clique-finding algorithms (Bron/ Kerbosch and HYPERCLIQUE) to find fuzzy inclusion dependencies.

Rest of the paper is organized as follows: Section 2 gives brief definitions of the concepts used in developing the algorithm in question. Section 3 introduces and discusses an algorithm to map the problem of Discovery of Fuzzy Inclusion Dependencies ($FID_\alpha s$) in *Fuzzy Databases* to the problem of *Finding Cliques in Hypergraphs*. The algorithm for the Discovery of $FID_\alpha s$ is developed and discussed in Section 4 and Section 5 concludes.

2 Definitions, Concepts & Background

Definition 2.1: Fuzzy Value Equivalent (FVEQ): Let *A* and *B* be two fuzzy sets defined on universe of discourse *U*, with their membership functions μ_A and μ_B, respectively. A fuzzy value $a \in A$ is said to be equivalent to some other fuzzy value $b \in B$, iff $b \in \mu_B(x)$, for some $x \in S$, where S is the set of crisp values that are returned by $\mu_A^{-1}(a)$, where μ_A^{-1} is the inverse of the membership function of fuzzy set *A*.

Example 2.1: Consider the Figure 1, where membership functions representing the fuzzy sets child, *young*, *mid*, and *old* are used to identify the age of a person in relations *Emp* and *Staff*. These relations are under different DBAs, hence, the membership functions that correspond to these relations are shown to be non-identical being designed by different domain experts. Now let μ_A and μ_B represent membership functions of the fuzzy set *young* used in fuzzy relations *Emp* and *Staff* respectively and μ_C represent the membership function of the fuzzy set *mid* used in fuzzy relations Staff. μ_A and μ_B are not identical because of individual differences in domain experts. Let there be a fuzzy value $(0.5/young) = a \in A$, then $\mu_A^{-1}(a) = \{25,35\} = S$. If age *x* of a person is 25 years, then $\mu_B(x) = (1.0/young) = b \in B$. Therefore, the fuzzy value (0.5=young) in fuzzy set *A* is said to be FVEQ to fuzzy value $(0.5/young)$ in fuzzy set *B*. Similarly, if age *x* of a person is 35 years, then $\mu_C(x) = (0.5/mid) = b \in C$, hence the fuzzy value $(0.5/young)$ in fuzzy set *A* is said to be FVEQ to fuzzy value $(0.5/mid)$ in fuzzy set *C*.

2.1 Notations used for Fuzzy Relational Databases

The fuzzy relational data model as given by Buckles & Petry (1982) and its derivatives are considered here, however, throughout these work notations similar to that in Casanova *et al.* (1982) will be used. Set variables will be denoted by capital letters and variables denoting elements of a set will be denoted by small letters. "*k*-subset of *X*" means a subset of *X* with cardinality *k*, while a "*k*-set" is simply a set with cardinality *k*. A fuzzy value is an element of data stored in a fuzzy relation's extent. Examples include .6/good, .5/old, or .8/high etc. A domain *D* is a finite set of fuzzy values. A fuzzy attribute is a bag (multiset) of fuzzy values. A fuzzy relational schema is a pair (*Rel*,*U*) where *Rel* is name of the fuzzy relation and $U = (a_1, a_2, \cdots, a_n)$ is a finite ordered *n*-tuple labels, that is known to be fuzzy attribute names. A fuzzy relation is a 3-tuple *R* = (*Rel*,*U*,*E*) with *Rel* and *U* as above and $E \subseteq D_1 \times D_2 \times \cdots \times D_n$ the fuzzy relation extent. The sets $D_1, D_2, \cdots, D_n$ are called the domains of *R*'s fuzzy attributes. A fuzzy tuple in fuzzy relation *R* is an element of *E*. An operator $t[a_1, a_2, \cdots, a_n]$ returns the projection *t* on the fuzzy attributes named $a_1, a_2, \cdots, a_n$.

2.2 Fuzzy Inclusion Dependencies (FIDs)

Fagin R. (1981) introduced and formally defined the inclusion dependency (IND) that can be derived across two relations. Similarly, fuzzy inclusion dependency (FID) (Sharma *et al.* (2004)) has been introduced and formally defined that can be derived across two fuzzy relations as given below.

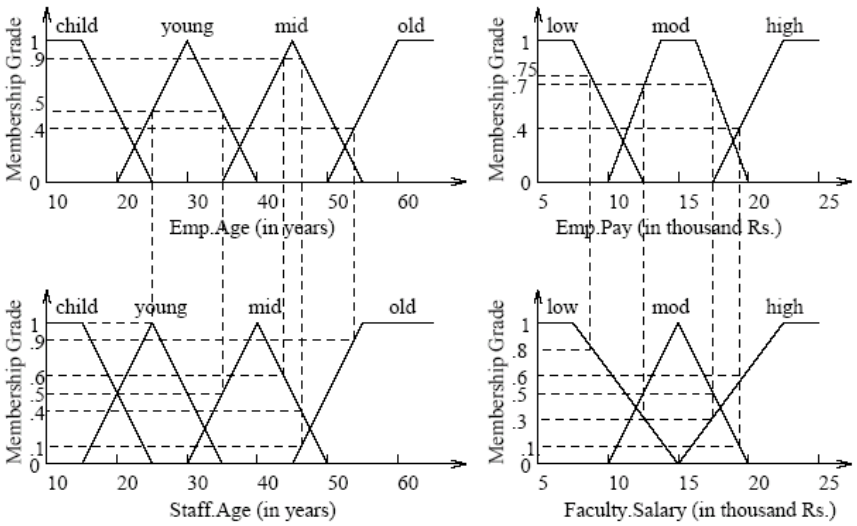
Definition 2.2 (FID): Let $R[a_1, a_2, \cdots, a_n]$ and $S[b_1, b_2, \cdots, b_m]$ be (projections on) two fuzzy relations. Let *X* be a sequence of *k* distinct fuzzy attribute names from *R*, and *Y* be a sequence of *k* distinct fuzzy attribute names from *S*,

with $1 \le k \le \min(n,m)$. Then, fuzzy inclusion dependency FID is an assertion of the form $R[X] \subseteq S[Y]$, where all the Fuzzy Values under all the attribute names in $R[X]$ are Fuzzy Value Equivalent to some Fuzzy Values under respective attribute names in $S[Y]$, however, the vice versa may not hold.

Remark: The assertion $R[X] \subseteq S[Y]$ in the above definition indicates $\mu_X(u) \subseteq \mu_Y(u), \forall u \in U$ which may not be fully satisfied because, two different database designers may be having different perceptions about the same object, and may have used different membership functions to represent the same fuzzy set. Say for example in following Figure $Emp[Age]$ uses a fuzzy set "mid" with support (35-55) to identify the middle aged persons, whereas $Staff[Age]$ uses a fuzzy set "mid" with support (30-50) in the same context. This leads to the definition of partial fuzzy inclusion dependency FID_α as follows.

2.2.1 Definition Partial Fuzzy Inclusion Dependency (FID_α)

Let $R[a_1, a_2, \cdots, a_n]$ and $S[b_1, b_2, \cdots, b_m]$ be (projections on) two fuzzy relations. Let *X* be a sequence of *k* distinct fuzzy attribute names from *R*, and *Y* be a sequence of *k* distinct fuzzy attribute names from *S*, with $1 \le k \le \min(n,m)$. Then, a partial fuzzy inclusion dependency FID_α is an assertion of the form $R[X] \subseteq S[Y]$, such that the fuzzy subset-hood $\mathbb{S}(R[X], S[Y]) = |R[X] \overset{f}{\cap} S[Y]| / |R[X]| \ge \alpha$, where α is specified in the interval [0,1] and most of the Fuzzy Values under all the attribute names in $R[X]$ are Fuzzy Value Equivalent (FVEQ) to some Fuzzy Values under respective attribute names in $S[Y]$, however, the vice versa may not hold.



Emp

Name	Age	Pay	μ
Basu	.9/mid	.7/mod	.7
Datta	.9/mid	.75/low	.75
Dhar	.4/old	.4/high	.4
Gupta	.9/mid	.7/mod	.7
Roy	.9/mid	.7/mod	.7

Faculty

Name	Dept	Salary	μ
Basu	EE	.5/mod	.5
Datta	MA	.8/low	.8
Dhar	EC	.4/high	.4
Gupta	CS	.3/high	.3

Staff

Name	Age	μ
Basu	.6/mid	.6
Dhar	.9/old	.9
Roy	.1/old	.1

Figure 1: Fuzzy relations with respective membership functions & mappings

Definition 2.3 (Valid FID): A $FID_\alpha \quad \rho = (R[a_{i_1}, a_{i_2}, \cdots, a_{i_k}] \subseteq S[b_{i_1}, b_{i_2}, \cdots, b_{i_k}])$ is valid between two relations $R = (r, (a_1, a_2, \cdots, a_n), E_R)$ and $S = (r, (a_1, a_2, \cdots, a_n), E_S)$ if the sets of fuzzy tuple in E_R and E_S satisfy the assertion given by ρ. Otherwise, FID_α is called invalid for R and S. In other words, FID_α is said to be valid if $\mathbb{S}(R[X], S[Y]) = |R[X] \overset{f}{\cap} S[Y]| / |R[X]| \geq \alpha$ holds.

Example 2.2: Consider the fuzzy relations each belonging to different fuzzy relational databases and their respective membership functions & mapping as given in the Figure 1. It is observed that:
Emp[Age]={.9/mid, .9/mid, .4/mid, .9/mid, .9/mid}
Emp[Pay]={7/mod, .75/low, .4/high, .7/mod, .7/mod}
Staff[Age]={.6/mid, .9/old, .1/old}
Faculty[Salary]={.5/mod, .8/low, .4/high, .3/high}

Valid fuzzy inclusion dependencies (FID_α)

$Staff[Age] \overset{f}{\subseteq} Emp[Age]$	for $\alpha = 1.0$
$Faculty[Salary] \overset{f}{\subseteq} Emp[P]$	for α=0.75
$Staff[Name, Age] \overset{f}{\subseteq} Emp[Name, Age]$	for $\alpha = 1.0$
$Faculty[Name, Salary] \overset{f}{\subseteq} Emp[Name, Pay]$	for α=0.75

Since the fuzzy subset-hood $\mathbb{S}(Staff[Age], Emp[Age]) = |Staff[Age] \overset{f}{\cap} Emp[Age]| / |Staff[Age]|$, where all elements of the fuzzy set *Staff*[Age] are Fuzzy Value Equivalent to some element of fuzzy set *Emp*[Age], therefore,

$$Staff[Age] \overset{f}{\cap} Emp[Age] = Staff[Age],$$

thus,

$$\mathbb{S}(Staff[Age], Emp[Age]) = \frac{|\{.6/mid, .9/old, .1/old\}|}{|\{.6/mid, .9/old, .1/old\}|} = \frac{3}{3} = 1.$$

Hence the fuzzy inclusion dependency

$$\text{FID} = Staff[Age] \overset{f}{\subseteq} Emp[Age]$$

is valid.
Similarly, the fuzzy subset-hood

$$\mathbb{S}(Faculty[Salary], Emp[Pay]) = \frac{|Faculty[Salary] \overset{f}{\cap} Emp[Pay]|}{|Faculty[Salary]|}.$$

As indicated in the Figure, all elements (except one ".4/high") of the fuzzy set *Faculty*[Salary] are Fuzzy Value Equivalent to some element of fuzzy set *Emp*[Pay], therefore,

$$Faculty[Salary] \overset{f}{\cap} Emp[Pay] = \{.5/\text{mod}, .8/low, .3/high\}$$

Thus,

$$\mathbb{S}(Faculty[Salary], Emp[Pay]) = \frac{|\{.5/\text{mod}, .8/low, .3/high\}|}{|\{.5/\text{mod}, .8/low, .4/high, .3/high\}|} = \frac{3}{4} = .75,$$

hence, partial fuzzy inclusion dependency

$$FID_{\alpha=.75} = Faculty[Salary] \subseteq Emp[Pay]$$

is valid.

A fuzzy inclusion dependency is merely a statement about two fuzzy relations that may be true or false. A valid FID describes the fact that a fuzzy projection of one fuzzy relation *R* forms a fuzzy subset of another fuzzy projection (of the same number of fuzzy attributes) of a fuzzy relation S. Note that FIDs are defined over sequences of attributes, not sets, since the order of attributes is important (FIDs are not invariant under permutation of the attributes of only one side) and concept of Fuzzy Value Equivalent is used to measure the equality of two fuzzy values or two fuzzy tuples.

Definition 2.4 (Arity of a FID): Let *X*, *Y* be sequences of *k* fuzzy attributes, respectively and $\rho = R[X] \overset{f}{\subseteq} S[Y]$ be a FID. Then *k* is the arity of ρ, denoted by $|\rho|$, and ρ is called a *k*-ary FID. A similar definition holds for partial FID.

Example 2.3: In Figure 1, the partial fuzzy inclusion dependency

$$Faculty[Name, Salary] \overset{f}{\subseteq} Emp[Name, Pay]$$

has the arity 2, hence it is said to be a binary $FID_{\alpha=.75}$, whereas, the fuzzy inclusion dependency

$$Staff[Age] \overset{f}{\subseteq} Emp[Age]$$

has got the arity 1, hence it is said to be an unary FID.

2.3 Inference Rules for FIDs

Casanova *et al.* (1982) have provided some important insights into the IND problem. They have described a complete set of inference rules for INDs, in the sense that repeated application of their rules will generate all valid INDs that can be derived from a given set of valid INDs (i.e., those rules form an axiomatization for INDs). Those rules will be redefined from the view point of FIDs as given below.

Axiom 1: (Refexivity) $R[X] \overset{f}{\subseteq} R[X]$, if *X* is a sequence of distinct fuzzy attributes from *R*. Similarly, $\mathbb{S}(R[X], R[X]) \geq \alpha$ holds.

Axiom 2: (Projection and Permutation) hold for both FID and Partial FID (i.e. FID_α).

If $R[A_{i_1}, \cdots, A_{i_m}] \overset{f}{\subseteq} S[B_{i_1}, \cdots, B_{i_m}]$ is valid, then $R[A_{i_1}, \cdots, A_{i_k}] \overset{f}{\subseteq} S[B_{i_1}, \cdots, B_{i_k}]$ is valid for any sequence $(i_1, \cdots, i_k)$ of distinct integers from $(1, \cdots m)$.

Note that permutation refers to "synchronous" reordering of attributes on both sides, i.e.,

$$R[X,Y] \overset{f}{\subseteq} S[X,Y] \Rightarrow R[Y,X] \overset{f}{\subseteq} S[Y,X] \text{ but, } \neg(R[X,Y] \overset{f}{\subseteq} S[X,Y] \Rightarrow R[Y,X] \overset{f}{\subseteq} S[X,Y]).$$

Axiom 3: (Transitivity) holds for $FID_{\alpha=1}$. If $R[X] \overset{f}{\subseteq} S[Y]$ and $S[Y] \overset{f}{\subseteq} T[Z]$ are both valid, then $R[X] \overset{f}{\subseteq} T[Z]$ is valid.

Proof: It is sufficient to show that if a Fuzzy Values *x* is Fuzzy Value Equivalent to some Fuzzy Values *y* and the Fuzzy Value *y* is Fuzzy Value Equivalent to a Fuzzy Value z then x is Fuzzy Value Equivalent to *z*.

$(R[X] \overset{f}{\subseteq} S[Y] \Leftrightarrow \forall x \in X, y \in \mu_Y(q)$ for some $q \in Q$ where $Q = \mu_X{}^{-1}(x)$ and

$S[Y] \overset{f}{\subseteq} T[Z] \Leftrightarrow \forall y \in Y, z \in \mu_T(q)$ for some $q \in Q$ where $Q = \mu_Y{}^{-1}(y))$

$\Leftrightarrow (\forall x \in X, z \in \mu_T(q)$ for some $q \in Q$ where $Q = \mu_X{}^{-1}(x))$

$\Leftrightarrow X$ is Fuzzy Value Equivalent to *Z*.

(Transitivity) may not hold for $FID_{\alpha<1}$.

$(\mathbb{S}(R[X], S[Y]) \geq \alpha$ and $\mathbb{S}(S[Y], T[Z]) \geq \alpha) \not\Rightarrow \mathbb{S}(R[X], T[Z]) \geq \alpha$

Definition 2.5 (Derived FID): A valid FID ρ can be derived from a set $\overset{f}{\Sigma}$ of valid FIDs, denoted by $\overset{f}{\Sigma} \vDash \rho$, if ρ can be obtained by repeatedly applying the above axioms on some set of FIDs taken from $\overset{f}{\Sigma}$.

Similarly, a valid partial inclusion dependency FID_α ρ can be derived from a fuzzy set $\overset{f}{\Sigma}$ of valid FID_α denoted by $\overset{f}{\Sigma} \vDash \rho$, if ρ can be obtained by repeatedly applying the above axioms on some set of FIDs taken from $\overset{f}{\Sigma}$.

The membership function of the fuzzy set $\overset{f}{\Sigma}_k$ may be given as follows:

$$\mu_{\underset{\Sigma_k}{f}}(\rho) = \frac{|R[a_{i_1}, a_{i_2}, \cdots, a_{i_k}] \overset{f}{\bigcap} S[b_{j_1}, b_{j_2}, \cdots, b_{j_k}]|}{|R[a_{i_1}, a_{i_2}, \cdots, a_{i_k}]|} \geq \alpha.$$

where $\rho = R[a_{i_1}, a_{i_2}, \cdots, a_{i_k}] \overset{f}{\subseteq} S[b_{j_1}, b_{j_2}, \cdots, b_{j_k}]$ and $k = 1, 2, \cdots, \min(n, m)$, n and m are the cardinality of sets of fuzzy attribute names belonging to fuzzy relations R and S respectively. For example a fuzzy set of valid $FID_{\alpha=.6}$ of arity k may be given as,

$$\overset{f}{\Sigma}_k = \{.66/\rho_1, .77/\rho_2, \rho_3, \cdots\}.$$

Fuzzy inclusion dependency (FID) is a special case of partial fuzzy inclusion dependency ($FID_{\alpha=1}$). Therefore partial fuzzy inclusion dependency is a generalized approach that shall be used in further discussions.

Since FIDs are invariant under synchronous permutation of both sides (by Axiom 2), now equality of FIDs (which applies to both valid and invalid FIDs) will be defined.

Definition 2.6 (Equality of FIDs): Two FIDs $R[a_{i_1}, \cdots, a_{i_m}] \overset{f}{\subseteq} S[b_{i_1}, \cdots, b_{i_m}]$ and $R[c_{i_1}, \cdots, c_{i_m}] \overset{f}{\subseteq} S[d_{i_1}, \cdots, d_{i_m}]$ are equal iff there is a sequence $(i_1, \cdots, i_m)$ of distinct integers $1, \cdots, m$, such that

$$(a_{i_1} = c_{i_1} \wedge b_{i_1} = d_{i_1}) \wedge \cdots \wedge (a_{i_m} = c_{i_m} \wedge b_{i_m} = d_{i_m}).$$

A similar definition holds for the equality of partial fuzzy inclusion dependencies too. Note that equality according to this definition is an equivalence relation on FIDs. It is also clear that equivalence preserves validity, i.e., in a set of equal FIDs, the elements are either all valid or all invalid.

One very important observation on FID is that a k-ary FID with $k > 1$ naturally implies a set of unary FID. Let $\rho = R[X] \overset{f}{\subseteq} S[Y]$ be a k-ary FID. Let there be unary $R[x] \overset{f}{\subseteq} S[y]$ with $x \in X$ and $y \in Y$. Then, there exist a close relationship between ρ and $\overset{f}{\Sigma}_1$, as formalized in following Corollary.

Corollary 2.1: Let $\overset{f}{\Sigma}_k$ be the set of all possible k-ary FID_α between two given fuzzy relations R and S. Let $\overset{f}{\Sigma}_1^k$ be the fuzzy set whose elements are all k-sets of unary FID_α between R and S. Then, there is an isomorphism between $\overset{f}{\Sigma}_k$ and $\overset{f}{\Sigma}_1^k$. It is said that $\overset{f}{\Sigma}_1^k$ is implied by $\overset{f}{\Sigma}_k$.

This isomorphic mapping is possible since FID_α are invariant under permutations of their attribute pairs (such that there are exactly as many k-ary FID_α as there are k-subsets of unary FID_α), and each pair of single attributes in a k-ary FID_α ρ corresponds to one unary FID_α implied by ρ. Note that the isomorphism does not hold for valid FID_α since clearly the existence of k unary valid FID_α does not imply the existence of any higher-arity valid FID_α (i.e., only the direction $\overset{f}{\Sigma}_k \Rightarrow \overset{f}{\Sigma}_1^k$ holds for valid FID_α, not the converse).

Validity of FID_α is preserved under projections and permutation, by Axiom 2. In order to describe all fuzzy inclusion dependency information between two fuzzy relations it is, therefore, not necessary to list all FID_α between two fuzzy relations. Rather, a small set of FID_α from which all others can be generated will suffice, as formalized with the following definition.

Definition 2.7 (Generating set of FID_α): Consider a fuzzy set of valid partial fuzzy inclusion dependencies $\overset{f}{\Sigma} = \{v_1/\rho_2, v_2/\rho_2, \cdots, v_n/\rho_n, \}$. A generating set of $\overset{f}{\Sigma}$, denoted by $\mathcal{G}(\overset{f}{\Sigma})$, and is a set of valid FID_α with the following properties:

(1) $\forall \rho \in \overset{f}{\Sigma}: \overset{f}{\mathcal{G}}(\Sigma) \models \rho$,

(2) $\forall \rho \in \mathcal{G}(\overset{f}{\Sigma}): \neg((\mathcal{G}(\overset{f}{\Sigma}) \overset{f}{-} \rho) \models \rho)$,

where the symbol '$\overset{f}{-}$' stands for "fuzzy set-difference".

In words, the generating set $\mathcal{G}(\overset{f}{\Sigma})$ contains exactly those valid FID_α from which all valid FID_α in $\overset{f}{\Sigma}$ can be derived. The set is not empty for any $\overset{f}{\Sigma}$, since it can be constructed by first including all $\rho \in \overset{f}{\Sigma}$ into $\mathcal{G}(\overset{f}{\Sigma})$ and then removing all ρ for which property 2 does not hold. The set is minimal since removing any FID_α ρ from a $\mathcal{G}(\overset{f}{\Sigma})$, for which property 2 holds would by definition violate property 1. Therefore, generating sets contain all information about fuzzy inclusion dependencies between fuzzy relations in a minimal number of FID_α.

Definition 2.8: A *k*-uniform hypergraph (or a *k*-hypergraph) is a pair $G = (V, E)$ of the set *V* of nodes and the set *E* of edges. An element $e \in E$ is a set with cardinality *k* of pair-wise distinct elements from *V*, denoted by $\{v_1, v_2, \cdots, v_k\}$. An element $e \in E$ is called a k-hyperedge. k is called the rank of graph G.

Definition 2.9: Let $G = (V, E)$ be a graph. A clique of *G* is a set $C \subseteq V$ such that $\forall v_1, v_2 \in C : \{v_1, v_2\} \in E$. A single node with no adjacent edges is a clique of cardinality 1.

Definition 2.10: Let $G = (V, E)$ be a *k*-hypergraph. A hyperclique of *G* is a set $C \subseteq V$ such that for each *k*-subset *S* of distinct nodes from *C*, the edge implied by *S* exists in *E*. The cardinality of a hyperclique *C* is the number of nodes in *C*. A single node with no adjacent edges is a hyperclique of cardinality 1.

Definition 2.11: The degree of a node $v \in V$ in a *k*-hypergraph $G = (V, E)$ is the number of edges that have *v* as element. i.e. $\deg(v) = |\{e \in E \mid v \in e\}|$.

Example 2.4: Figure 2 gives an illustrative example of a 3-hypergraph with five nodes and six number of 3-hyperedges, viz {1,2,3}, {2,3,4}, {3,4,5}, {1,2,4}, {1,2,5}, {1,3,4}. Here, a node is represented as a numbered circle and a 3-hyperedge is represented as three different lines connected to a small black circle •.

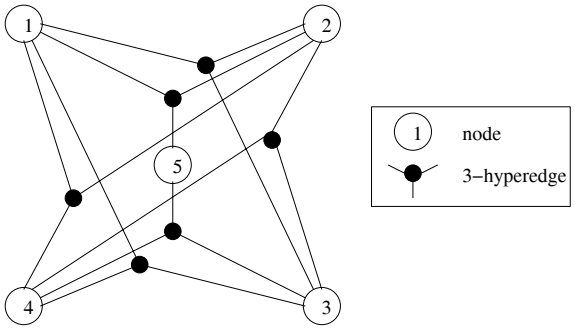


Figure 2: A 3-hypergraph with 5 nodes & 6 edges

3 Mapping to a Graph Problem

Let *R* and *S* be two fuzzy relations with k_S and k_R attributes such that $k_R < k_S$. Then an algorithm to map the set of $FID_\alpha s$ of the form $R[a_i] \overset{f}{\subseteq} S[b_i]$ that may exist between the fuzzy relations k_S and k_R is as follows.

Algorithm 3.1: FID_α -TO-GRAPH

Create a fuzzy set *V* corresponding to the fuzzy set $\overset{f}{\sum_1}$ of all valid unary $FID_\alpha s$ between *R* and *S* such that $v \in V$ correspond to $\rho \in \overset{f}{\sum_1}$.

Create a weighted graph (2-Hypergraph) as follows:

Create a fuzzy set E_2 corresponding to the fuzzy set $\sum_2^f$ of all valid binary $FID_\alpha s$ between *R* and *S* such that $e \in E_2$ correspond to $\rho_{ij} \in \sum_2^f$. Each ρ_{ij} can be seen as $\rho_i, \rho_j \in \sum_1^f$ that corresponds to $v_i, v_j \in V$, where $i, j \in \{1, 2, \cdots, |V|\}$ by corollary 2.

Create a weighted graph $G_2 = (V, E_2)$ such that $\forall v \in V$ there exists a valid unary FID_α ρ between *R* and *S* whose membership grade $\mu_{\sum_1^f}(\rho)$ correspond to the weight of the node $w(v)$ and $\forall e \in E_2$ there exists a valid binary FID_α ρ_{ij} between *R* and *S* whose membership grade $\mu_{\sum_2^f}(\rho_{ij})$ correspond to the weight of the edge $w(e) = \min(w(v_i), w(v_j))$, where $i, j \in \{1, 2, \cdots, |V|\}$.

Create Hypergraphs as follows:

Create a fuzzy set E_k corresponding to the fuzzy set $\sum_k^f$ of all valid *k*-ary $FID_\alpha s$ between *R* and *S* such that the $e \in E_k$ corresponds to $\rho_{i_1 \cdots i_k} \in \sum_k^f$ for $k \in \{1, 2, \cdots, |V|\}$.

Create a weighted *k*-Hypergraph $G_k = (V, E_k)$ for each $k \in \{1, 2, \cdots, |V|\}$ such that $\forall e \in E_k$ there exists a *k*-ary FID_α $\rho_{i_1 \cdots i_k}$ between *R* and *S* whose membership grade $\mu_{\sum_k^f}(\rho_{i_1 \cdots i_k})$ correspond to the weight of the *k*-Hyperedge $w(e) = \min(w(v_{i_1}), \cdots, w(v_{i_k}))$, where $i_1, \cdots, i_k \in \{1, 2, \cdots, |V|\}$

Now the FID_α-finding problem may be viewed as a problem of constructing the above graphs.

Lemma 3.1: A *k*-ary valid FID_α implies $\binom{k}{m}$ *m*-ary valid $FID_\alpha s$, for any $1 \le m \le k$.

The following theorem is observed by Lemma 3.1:

Theorem 3.1: Given the two fuzzy relations *R* and *S* with k_R and k_S attributes, respectively, consider a collection of *k*-Hypergraphs $\{G_2, \cdots, G_{k_s}\}$ representing the $FID_\alpha s$ between *R* and *S* (as defined in Algorithm 3.1). Furthermore, let ρ_k be a *k*-ary valid FID_α between *R* and *S*. For a number *m*, with $m < k$, construct a fuzzy set E_m of all the *m*-ary $FID_\alpha s$ implied by ρ_k, which are all *m*-Hyperedges in G_m. Then the set of all nodes that are elements of any edge in E_m forms an *m*-Hyperclique in G_m, or alternatively, E_m is the set of edges of an *m*-Hyperclique in G_m.

Proof: By Definition 2.10, the set of edges of a hyperclique *C* in a *m*-hypergraph $G_m = (V, E_m)$ correspond to exactly all m-subsets of nodes from C. Also, the *m*-ary $FID_\alpha s$ implied by a FID_α ρ_k (with $m < k$) are exactly all *m*-subsets of the set of nodes implied by ρ_k. If *C* is the set of nodes implied by ρ_k, clearly there is a trivial isomorphic mapping between the edges of *C* and the *m*-ary $FID_\alpha s$ implied by ρ_k. q.e.d.

The problem of searching for $FID_\alpha s$ is now reduced to the problem of finding hypercliques in a collection of *k*-hypergraphs.

4 An Algorithm to Search for $FID_\alpha s$

We now present the algorithm $GSEARCH_2$ which uses those clique-finding algorithms (Bron/Kerbosch and HYPERCLIQUE) to find fuzzy inclusion dependencies. $GSEARCH_2$ takes as input two relations *R* and *S* with k_R and k_S attributes, respectively and returns a generating set of fuzzy inclusion dependencies between attributes from *R* and *S*. The schema of both relations must be known, and it must be possible to perform a test for validity for any fuzzy inclusion dependency between two given sets of attributes from *R* and *S*. *R* and *S* do not necessarily have to have the same number of attributes.

We first establish the relationship between hypercliques and the generating set of $FID_\alpha s$ that we are trying to find. Intuitively, Theorem 4.1 shows that a clique-finding algorithm is a sensible approach to finding maximal $FID_\alpha s$ between relations. More specifically, we show that the $FID_\alpha s$ generated through a clique-finding algorithm are a (relatively small) superset of the generating set $G(\sum^f)$ and are thus a starting point for a complete and fast solution of the $FID_\alpha s$ finding problem.

Theorem 4.1: Consider the $FID_\alpha s$ searching problem between relations $R[A]$ and $S[B]$ with solution $G(\sum^f)$ (i.e., generating set of valid $FID_\alpha s$). Let V be the set of unary valid $FID_\alpha s$ between R and S. Let E_k with $1 < k \le \min(|A|,|B|)$, be the set of k-ary valid $FID_\alpha s$ between R and S. Recall that the elements of E_k can then be seen as edges in a k-Hypergraph $G_k = (V, E_k)$ by Theorem 3.1. Now consider the set C of all maximal cliques in the k-hypergraph G_k, obtained by the above clique-finding algorithms (Bron/Kerbosch and HYPERCLIQUE).

The following properties hold for any $c \in C$:

(1) If ρ_c corresponding to c is valid, then it is part of the generating set $G(\sum^f)$ of $FID_\alpha s$ between R and S.

(2) If ρ_c is invalid, then some of its subsets are in $G(\sum^f)$.

Furthermore, all $\rho \in G(\sum^f)$ are subsets of or equal to some ρ_c as above.

Proof: By Theorem 3.1, a valid FID_α implies a k-hyperclique in a k-hypergraph G_k constructed for this FID_α-finding problem. Also, a correct clique finding algorithm returns a set of maximal cliques. Property (1) must hold since if there was an FID_α larger than ρ_c, a clique corresponding to that larger FID_α would have been found. Property (2) is true since we assumed valid unary and k-ary $FID_\alpha s$ to make up graph G_k. If ρ_c is not valid but its unary and binary sub-$FID_\alpha s$ are, then some sub-$FID_\alpha s$ of ρ_c must be part of $G(\sum^f)$. Property (3) holds since any FID_α implies some (not necessarily maximal) complete subgraph of G_k, and by the definition of a clique, all such complete subgraphs are subsumed by the set of cliques found in G_k.

Algorithm 4.1: $GSEARCH_2$

INPUT: Relations R and S with attributes k_R and k_S $(k_R \le k_S), \alpha \in [0,1]$.

OUTPUT: Set result, containing a generating set of $FID_\alpha s$ for R and S.

1: Set $V \leftarrow generateValidUnaryFID_\alpha s(R,S,\alpha)$, $E \leftarrow generateValidBinaryFID_\alpha s(R,S,V,\alpha)$;

2: Set Graph $G_2 \leftarrow (V,E)$, $F \leftarrow generateCliquesAndVerifyAsFID_\alpha s(G_2,\alpha)$, $result \leftarrow \{c \mid c \in F \wedge |c| = 1\}$;

3: for $m \leftarrow 3 \cdots k_S$, KHyergraph $G_m = (V, \phi)$;

4: Set $C_{tmp} = \phi$;

5: for all $c \in F$, if (c is valid $\wedge |c| \ge (m-1)$) $result \leftarrow result \cup c$), if ($c$ is invalid $\wedge |c| \ge m$) $C_{tmp} \leftarrow C_{tmp} \cup c$;

6: $E_m = generateKAryFID_\alpha sFromCliques(m, C_{tmp}, \alpha)$;

7: If $(E_m = \phi)$ return $result$;

8: Set $result \leftarrow result \cup generateSubFID_\alpha s(m-1, E_m, result)$;

9: Set KHypergraph $G_m = (V, ValidFID_\alpha s(E_m))$, $F \leftarrow generateCliquesAndVerifyAsFID_\alpha s(G_m,\alpha)$;

10: Return $result$.

Algorithm 4.2 $VERIFY(R,S,A,B,\alpha)$, where A, B are attribute lists from fuzzy relations R, S.

Query Q_1: select $count$ $(distinct$ $R.A_{i_1}, R.A_{i_2})$ from R, S, where $R.A_{i_1}$ $FVEQ$ $S.B_{i_1}$ and $R.A_{i_2}$ $FVEQ$ $S.B_{i_2}$.

Query Q_2: select $count(distinct$ $A_{i_1}, A_{i_2})$ from R;

Step 1: $c_1 \leftarrow$ Execute Query Q_1;

Step 2: If $(c_1 = 0)$ return $false$; // FID_α does not exist

Step 3: $c_2 \leftarrow$ Execute Query Q_2;

Step 4: $Mu = c_1 / c_2$; // $Mu = \mu_{\sum_k^f}(\rho)$, where, $\sum_k^f$ is fuzzy set of *k*-ary $FID_\alpha s$

Step 5: If $(Mu \geq \alpha)$ return $(Mu, true)$, // FID_α is valid
else return $(Mu, false)$ // FID_α is invalid.

Subroutines of $GSEARCH_2$

$generateValidUnaryFID_\alpha s(R,S,\alpha)$: This function generates all $k_R \cdot k_S$ unary $FID_\alpha s$ and verifies their validity against the database. It returns the set V of the $FID_\alpha s$ valid in the database, which are the nodes for all subsequent graphs and hypergraphs.

$generateValidUnaryFID_\alpha s(R,S,\alpha)$: This function generates all those $FID_\alpha s$ whose implied binary $FID_\alpha s$ are elements of E. The function then checks all those $FID_\alpha s$ against the database and returns the set E of all those valid binary $FID_\alpha s$.

$generateCliquesAndVerifyAsFID_\alpha s(G_k,\alpha)$: This function accepts a *k*-hypergraph. It returns a set of all hypercliques in G_k, together with a Boolean value for each element in the set in the following manner:
For $k > 2$, this function calls algorithm HYPERCLIQUE on *k*. If $k = 2$, the Bron/ Kerboschalgorithm is run. The function then tests each generated (hyper) cliques implied FID_α against the database. It returns a set of all those $FID_\alpha s$ with more than *k* nodes (i.e., at most one FID_α for each clique discovered), regardless of their validity, but marks each FID_α as valid or invalid according to its state in the database and computes its membership grade to respective class of set of FID_α.

generate K Ary $FID_\alpha s$ *From Cliques* (k, C_{tmp}, α)**:** This function accepts a number *k* and a set of $FID_\alpha s$. Input set *E* is assumed to be composed of invalid $FID_\alpha s$ and to correspond to cliques found by a clique finding algorithm on a $(k-1)$-Hypergraph. This function now generates all *k*-ary $FID_\alpha s$ implied by each FID_α in E, tests each one of them, and returns the union of those $FID_\alpha s$ for all elements of *E*. Similarly to function *generate Cliques and Verify as* $FID_\alpha s$ (G_k, α), both valid and invalid $FID_\alpha s$ are returned, and each FID_α is marked as valid or invalid according to its state in the database.

generate SubFID$_\alpha s$ $(k-1, E_k, result)$**:** This function accepts a number *k*, a set of (valid or invalid) $FID_\alpha s$, and set result, which is the set of $FID_\alpha s$ already included in the solution earlier (lines 5 and 10). The function now generates all $FID_\alpha s$ that satisfy the following three conditions:

1. ρ is a *k*-subset of an invalid FID_α from E.
2. ρ is not a subset of any valid FID_α from E.
3. ρ is not a subset of any FID_α already in result.

The function returns the set of all ρ that meet the above conditions.

Valid $FID_\alpha s$ (E_k)**:** Given a set of $FID_\alpha s$ marked as valid or invalid, this function simply returns the valid $FID_\alpha s$ from E_k.

5 Conclusions

The discovery of inclusion dependencies is a hard problem, with an inherent NP-complexity (Kantola *et al*, 1992), and so is the case with discovery of fuzzy inclusion dependencies. By reducing the problem to a weighted graph

problem, we achieved a significant improvement in performance over the naïve algorithm. Our algorithm uses an NP-complete graph algorithm (clique-finding), but a test implementation shows that most of the real world problems can be solved with our approach. Application of fuzzy inclusion dependencies lies for example in fuzzy schema integration, an important phase of fuzzy database integration process. As our algorithm discovers interrelationships between fuzzy relational databases, it helps in the identification of fuzzy relational databases that are useful for a variety of purposes, such as the identification of fuzzy relational database duplicates, system integration support, or the purpose of identifying backup fuzzy data.

References

[1] Bell, S., and P. Brockhausen, Discovery of data dependencies in relational databases, *Technical Report*, University of Dortmund, 1995.

[2] Buckles, B.P., and F.E. Petry, Uncertainity models in information and database systems, *Information Sciences*, vol.11, pp.77-87, 1985.

[3] Casanova, M.A., R. Fagin, and C.H. Papadimitriou, Inclusion dependencies and their interaction with functional dependencies, *Proceedings of ACM Conference on Principles of Database Systems*, pp.171-176, 1982.

[4] Fagin, R., A normal form for relational databases that is based on domains and keys, *ACM Transactions on Database Systems*, vol.6, no.3, pp.387-415, 1981.

[5] Kantola, M, H. Mannila, *et al.*, Discovering functional and inclusion dependencies in relational databases, *International J. of Intelligent Systems*, vol.7, pp.591-607, 1992.

[6] Missaoui, R., and R. Godin, The implication problem for inclusion dependences: A graph approach, *SIGMOD Record*, vol.19, no.1, pp.36-40, 1990.

[7] Mitchell, J.C., Inference rules for functional and inclusion dependencies, *Proc. of ACM Symposium on Principles of Database Systems*, Atlanta, Georgia, vol.21-23, pp.58-69, 1983.

[8] Savnik, I., and P.A. Flach, Bottom-up induction of functional dependencies from relations, *Proc. of AAAI-93 Workshop: Knowledge Discovery in Databases*, pp.174-185, 1993.

[9] Sharma, A.K., A. Goswami, and D.K. Gupta, Discovery of fuzzy inclusion dependencies in fuzzy relational databases, *Proc. of The Ninth IEEE Symposium on Computers and Communications Alexandria*, Egypt, pp.128-133, 2004.