Acta Cybernetica 12 (1995) 167-180.

Some Remarks On Generating Armstrong And Inferring Functional Dependencies Relation*

János Demetrovics† Vu Duc Thi†

Abstract

The main purpose of this paper is to give some results concerning algorithms for generating Armstrong relation and inferring functional dependencies (FDs for short). Firstly, we present some algorithms for solving these two problems. In the second part of the paper some NP-complete problems related to generating Armstrong relation and inferring FDs are given.

Key Words and Phrases: relation, relational datamodel, functional dependency, relation scheme, generating Armstrong relation, dependency inference, minimal key, antikey.

1 Introduction

Problems that construct a relation r such that r is an Armstrong relation of a given relation scheme (generating Armstrong relation) and a relation scheme s such that FDs of s hold in a given relation (inferring FDs) have been applied for for database design, query optimization, and artificial intelligence. These problems have been investigated in a lot of papers [3,9,12,16,17,18].

In this paper we give some results related to generating Armstrong relation and inferring FDs. The paper is structured as follows. In Section 2, we present some characterizations of the Armstrong relation of a given relation scheme, and construct an algorithm for finding all minimal transversals of a given hypergraph. From these and the results, presented in [9], we construct algorithms for generating Armstrong relation and inferring FDs.

Section 3 gives some NP-complete problems related to generating Armstrong relation and inferring FDs.

Let us give some necessary definitions and results that are used in the next sections. The concepts given in this section can be found in [1,3,4,6,7,8,10,11,13,19].

Let $R = \{a_1, \ldots, a_n\}$ be a nonempty finite set of attributes. A functional dependency is a statement of the form $A \to B$, where $A, B \subseteq R$. The FD $A \to B$

*Research supported by Hungarian Foundation for Scientific Research Grant 2575.

†Computer and Automation Institute Hungarian Academy of Sciences P.O.Box 63, Budapest, Hungary, H-1502

holds in a relation $r = \{h_1, \ldots, h_m\}$ over R if $\forall h_i, h_j \in r$ we have $h_i(a) = h_j(a)$ for all $a \in A$ implies $h_i(b) = h_j(b)$ for all $b \in B$. We also say that r satisfies the FD $A \to B$.

Let F_r be a family of all FDs that hold in r. Then $F = F_r$ satisfies

(1) $A \to A \in F$,

(2) $(A \to B \in F, B \to C \in F) \Longrightarrow (A \to C \in F)$,

(3) $(A \to B \in F, A \subseteq C, D \subseteq B) \Longrightarrow (C \to D \in F)$,

(4) $(A \to B \in F, C \to D \in F) \Longrightarrow (A \cup C \to B \cup D \in F)$.

A family of FDs satisfying (1)-(4) is called an f-family (sometimes it is called the full family) over R.

Clearly, F_r is an f-family over R. It is known [1] that if F is an arbitrary f-family, then there is a relation r over R such that $F_r = F$.

Given a family F of FDs, there exists a unique minimal f-family F^+ that contains F. It can be seen that F^+ contains all FDs which can be derived from F by the rules (1)-(4).

A relation scheme s is a pair $< R, F >$, where R is a set of attributes, and F is a set of FDs over R. Denote $A^+ = \{a: A \to \{a\} \in F^+\}$. A^+ is called the closure of A over s. It is clear that $A \to B \in F^+$ iff $B \subseteq A^+$.

Clearly, if $s =< R, F >$ is a relation scheme, then there is a relation r over R such that $F_r = F^+$ (see, [1]). Such a relation is called an Armstrong relation of s.

Let r be a relation, $s =< R, F >$ be a relation scheme. Then A is a key of r (a key of s) if $A \to R \in F_r$ ($A \to R \in F^+$). A is a minimal key of $r(s)$ if A is a key of $r(s)$ and any proper subset of A is not a key of $r(s)$.

Denote $K_r(K_s)$ the set of all minimal keys of $r(s)$.

Clearly, K_r, K_s are Sperner systems over R, i.e. $A, B \in K_r$ implies $A \not\subseteq B$.

Let K be a Sperner system over R. We define the set of antikeys of K, denoted by K^{-1}, as follows:

$$K^{-1} = \{A \subset R : (B \in K) \Longrightarrow (B \not\subseteq A) \, and \, (A \subset C) \Longrightarrow (\exists B \in K)(B \subseteq C)\}.$$

It is easy to see that K^{-1} is also a Sperner system over R.

Let R be a nonempty finite set, $P(R)$ its power set, and $I \subseteq P(R)$, $R \in I$, and $A, B \in I \Longrightarrow A \cap B \in I$. I is called a meet-semilattice over R. Let $M \subseteq P(R)$. Denote $M^+ = \{\cap M' : M' \subseteq M\}$. We say that M is a generator of I if $M^+ = I$. Note that $R \in M^+$ but not in M, by convention it is the intersection of the empty collection of sets.

Denote $N = \{A \in I : A \neq \cap\{A' \in I : A \subset A'\}\}$.

It can be seen that N is the unique minimal generator of I.

2 Algorithms

It is known [3,9,17] that the worst-case time complexities of generating Armstrong relation and inferring FDs are exponential. In this section we present some characterizations of the Armstrong relation of a given relation scheme. An effective algorithm finding all minimal transversals of a given hypergraph is also given. These results and the results, presented in [9], are used to construct algorithms for generating Armstrong relation and inferring FDs.

Let $s =< R, F >$ be a relation scheme. A FD $A \to \{a\} \in F^+$ is called the primitive maximal dependency (PMD for short) of s if $a \notin A$ and for all $A' \subseteq A : A' \to \{a\} \in F^+$ implies $A = A'$.

Denote $T_a = \{A : A \to \{a\}$ is a PMD of $s\}$. It can be seen that $\{a\}, R \notin T_a$, and T_a is a Sperner system over R. It is possible that $T_a = \emptyset$.

Let $s =< R, F >$ be a relation scheme, $a \in R$. Set $K_a = \{A \subseteq R$: $A \to \{a\}$, $\nexists B$: $(B \to \{a\})(B \subset A)\}$. K_a is called the family of minimal sets of the attribute a.

Clearly, $R \notin K_a$, $\{a\} \in K_a$ and K_a is a Sperner system over R. It is easy to see that $K_a - \{a\} = T_a$.

Based on the results, presented in [9], we show some characterizations of the Armstrong relation of a given relation scheme.

Lemma 2.1 [9] Let F be an f-family over R, $a \in R$. Denote $L_F(A) = \{a \in R$: $(A, \{a\}) \in F\}$, $Z_F = \{A$: $L_F(A) = A\}$. Clearly, $R \in Z_F$, $A, B \in Z_F \Longrightarrow A \cap B \in Z_F$. Denote by N_F the minimal generator of Z_F. Set $M_a = \{A \in N_F$: $a \notin A$, $\nexists B \in N_F$: $a \notin B$, $A \subset B\}$. Then $M_a = MAX(F, a)$, where $MAX(F, a) = \{A \subseteq R : A$ is a nonempty maximal set such that $(A, \{a\}) \notin F\}$.

Let r be a relation over R. Clearly, F_r is an f-family over R. Denote $L_{F_r}(A) = \{a \in R : A \to \{a\} \in F_r\}$, $Z_{F_r} = \{A : L_{F_r}(A) = A\}$. Put

$E_r = \{E_{ij} : 1 \leq i < j \leq |r|\}$, where $E_{ij} = \{a \in R : h_i(a) = h_j(a)\}$. E_r is called the equality set of r.

From E_r we compute $N = \{A \in E_r : A \neq \cap\{A' \in E_r : A \subset A'\}\}$. It can be seen that N is the minimal generator of Z_{F_r}. Then for each $a \in R$ we have

$M_a = \{A \in N : a \notin A, \nexists B \in N : A \subset B\}$.

It can be seen that $M_a = \{A \in E_r : a \notin A, \nexists B \in E_r : A \subset B\}$.

It is known [5] that an arbitrary full family of FDs can be uniquely determined by its primitive maximal dependencies.

From the result, presented in [9] (see, Remark 2.9), and Lemma 2.1 we obtain $K_a^{-1} = M_a$ for all $a \in R$. Clearly, if K is a Sperner system, then K and K^{-1} are uniquely determined by each other. Consequently, the next proposition is clear

Proposition 2.2 Let s be a relation scheme, and r a relation over R. Then r is an Armstrong relation of s if and only if for every $a \in R$

$$K_a^{-1} = M_a.$$

Now we present the concept of hypergraph that is in [4].

Let R be a nonempty finite set and $P(R)$ its power set. The family $H = \{E_i : E_i \in P(R), i = 1, \ldots, m\}$ is called a hypergraph over R if $E_i \neq \emptyset$. (In [4] author requires that the union of E_is is R, in this paper we do not).

A hypergraph H is simple if $E_i \subset E_j$ implies $i = j$, i.e., H is a Sperner system over R.

The elements of R are called vertices, and the sets $E_1, \ldots, E_m$ are the edges of the hypergraph H.

It is easy to see that a simple graph is a simple hypergraph with $|E_i| = 2$.

Let $H = \{E_1, \ldots, E_m\}$ be a hypergraph over R. Set

$m(H) = \{E_i \in H : \nexists E_j \in H : E_j \subset E_i\}$.

It can be seen that $m(H)$ is a simple hypergraph, and the family H uniquely determines the family $m(H)$.

Let H be a hypergraph over R. A set $A \subseteq R$ is called a transversal of H (sometimes it is called a hitting set) if $E \in H$ implies $A \cap E \neq \emptyset$.

The family of all minimal transversals of H is called the transversal hypergraph of H, and denoted by $tr(H)$. Clearly, $tr(H)$ is a simple hypergraph.

Remark 2.3 Let K be a Sperner system over R. Based on the definitions of K^{-1} and $tr(K)$ we can see that $tr(K) = \{R - A : A \in K^{-1}\}$.

Denote $N_a = \{R - A : A \in M_a\}$. From Proposition 2.2 and Remark 2.3 we have

Proposition 2.4 Let r be a relation, and s a relation scheme over R. Then r is an Armstrong relation of s iff for all $a \in R$

$$tr(K_a) = N_a.$$

It is known [4] that if H, H' are two simple hypergraph over R, then $H = tr(H')$ iff $H' = tr(H)$. From this and Remark 2.3, we can see that if K is a Sperner system, then $tr(\{R - A : A \in K^{-1}\}) = K$. According to the definitions of the set of all antikeys, the family of all minimal transversals, and Proposition 2.2 we obtain

Proposition 2.5 Let r be a relation, and s a relation scheme over R. Then r is an Armstrong relation of s iff for all $a \in R$

$$N_a{}^{-1} = \{B : R - B \in K_a\}.$$

Clearly, from Proposition 2.4 we have

Proposition 2.6 Let r be a relation, and s a relation scheme over R. Then r is an Armstrong relation of s iff for all $a \in R$

$$K_a = tr(N_a).$$

It is obvious that $a \in R - A$, where $A \in M_a$. Clearly, $T_a = K_a - \{a\}$. Thus, from the definition of the transversal hypergraph we obtain $T_a = tr(\{(R-a) - A : A \in M_a\})$ for all $a \in R(*)$.

Let r be a relation over R. A FD $A \to \{a\} \in F_r$ is called the primitive maximal dependency of r if $a \notin A$ and for all $A' \subseteq A : A' \to \{a\} \in F_r$ implies $A = A'$.

Denote $V_a = \{A : A \to \{a\}$ is a PMD of $r\}$, and $N'_a = \{(R-a) - A : A \in M_a\}$. By (*) and according to the definitions of F_r, and F^+ we have

Proposition 2.7 Let r be a relation over R. Then for all $a \in R$, $V_a = tr(N'_a)$.

Proposition 2.7 was independently discovered in [18].

In this paper, we consider the comparison of two attributes as an elementary step of algorithms. Thus, if we assume that subsets of R are represented as sorted lists of attributes, then a Boolean operation on two subsets of R requires at most $|R|$ elementary steps.

Now we construct an algorithm that finds all minimal transversals of a given hypergraph.

Algorithm 2.8 (Finding all minimal transversals).

Input: Let $H = \{E_1, \ldots, E_m\}$ be a hypergraph over R.

Output: $tr(H)$.

Step 1: Set $L_1 = \{\{a\} : a \in E_1\}$. It is obvious that $L_1 = tr(\{E_1\})$.

Step q+1 $(q < m)$:

Assume that $L_q = S_q \cup \{B_1, \ldots, B_{t_q}\}$, where $B_i \cap E_{q+1} = \emptyset, i = 1, \ldots, t_q$ and $S_q = \{A \in L_q : A \cap E_{q+1} \neq \emptyset\}$.

For each i $(i = 1, \ldots, t_q)$ construct the set $\{B_i \cup b : b \in E_{q+1}\}$. Denote them by $A^i_1, \ldots, A^i_{r_i}$ $(i = 1, \ldots, t_q)$. Let

$$L_{q+1} = S_q \cup \{A^i_p : A \in S_q \Longrightarrow A \not\subset A^i_p, 1 \leq i \leq t_q, 1 \leq p \leq r_i\}.$$

Set $tr(H) = L_m$.

Theorem 2.9 For every q $(1 \leq q \leq m)$, $L_q = tr(\{E_1, \ldots, E_q\})$, i.e., $L_m = tr(H)$.

Proof. We prove this theorem by induction. It is obvious that $L_1 = tr(\{E_1\})$. We have to show that $L_{q+1} = tr(\{E_1, \ldots, E_{q+1}\})$. For this using the inductive hypothesis $L_q = tr(\{E_1, \ldots, E_q\})$.

Firstly, assume that D is the minimal subset of R such that $D \cap E_t \neq \emptyset\,(t = 1, \ldots, q+1)$. By the inductive hypothesis, there is a $X \in L_q$ such that $X \subseteq D$.

If $X \in S_q$, then $X \cap E_t \neq \emptyset$ for all $t = 1, \ldots, q+1$. Because D is the minimal subset of R such that $E_t \cap D \neq \emptyset\,(t = 1, \ldots, q+1)$, we have $X = D$. Hence, $D \in S_q$ holds. Consequently, we obtain $D \in L_{q+1}$.

If $X \cap E_{q+1} = \emptyset$, then $X = B_i$ holds for some i in $\{1, \ldots, t_q\}$. By $D \cap E_{q+1} \neq \emptyset$ we have $B_i \subset D$. Thus, $(D - B_i) \cap E_{q+1} \neq \emptyset$ holds. According to the construction of L_{q+1}, we have $A_p^i \subseteq D$ for some p in $\{1, \ldots, r_i\}$. Clearly, $A_p^i \cap E_l \neq \emptyset$ for all $l = 1, \ldots, q+1$, i.e., A_p^i is a transversal of the family $\{E_1, \ldots, E_{q+1}\}$. By $D \in tr(\{E_1, \ldots, E_{q+1}\})$ we obtain $D = A_p^i$. Because D does not contain the elements of S_q, we have $D \in L_{q+1}$.

Conversely, assume that $D \in L_{q+1}$. If $D \in S_q$, then $D \cap E_p \neq \emptyset$ $(p = 1, \ldots, q)$ and D is minimal for this property, and at the same time $D \cap E_{q+1} \neq \emptyset$. Consequently, we have $D \in tr(E_1, \ldots, E_{q+1})$.

Let $D \in L_{q+1} - S_q$. Clearly, there is an A_p^i $(1 \leq i \leq t_q$ and $1 \leq p \leq r_i)$ such that $D = A_p^i$. Our construction shows that $E_l \cap A_p^i \neq \emptyset$ for all $l = 1, \ldots, q+1$. By the construction of algorithm we obtain $A_p^i = B_i \cup \{b\}$ for some $b \in E_{q+1}$.

Suppose that C is a proper subset of A_p^i, and $C \in tr(\{E_1, \ldots, E_{q+1}\})$. Clearly, $b \in C$ holds. According to the definitions of the transversal and the family of all minimal transversals, C is a transversal of the collection $\{E_1, \ldots, E_q\}$. By the inductive hypothesis $(L_q = tr(\{E_1, \ldots, E_q\}))$, if there is $A \in S_q$ such that $A \subseteq C$, then we have $A \subset A_p^i$. This contradicts $A \not\subset A_p^i$ for all $A \in S_q$. If there is B_j $(1 \leq j \leq t_q)$ $B_j \cap E_{q+1} = \emptyset$ such that $B_j \subseteq C$, then $b \notin B_j$ and $B_j \subset B_i$. This conflicts with the fact that L_q is a simple hypergraph. Hence, $D \in tr(\{E_1, \ldots, E_{q+1}\})$ holds.

Thus, $L_{q+1} = tr(\{E_1, \ldots, E_{q+1}\})$. Hence, $L_m = tr(H)$ holds. The theorem is proved.

It can be seen that the hypergraph H uniquely determines the family $tr(H)$, and the determination of $tr(H)$ based on our algorithm does not depend on the order of $E_1, \ldots, E_m$.

Remark 2.10 Denote $L_q = S_q \cup \{B_1, \ldots, B_{t_q}\}$, and l_q $(1 \leq q \leq m-1)$ is the number of elements of L_q. It can be seen that the worst-case time complexity of our algorithm is

$$O(|R|^2 \sum_{q=0}^{m-1} t_q u_q),$$

where $l_0 = t_0 = 1$ and

$$u_q = \begin{cases} l_q - t_q & \text{if } l_q > t_q, \\ 1 & \text{if } l_q = t_q. \end{cases}$$

Clearly, in each step of our algorithm L_q is a simple hypergraph. It is known that the size of arbitrary simple hypergraph over R can not be greater than $C_n^{[n/2]}$, where $n = |R|$. $C_n^{[n/2]}$ is asymptotically equal to $2^{n+1/2}/(\pi \cdot n)^{1/2}$. From this, the worst-case time complexity of our algorithm can not be more than exponential in the number of attributes. In cases for which $l_q \leq l_m (q = 1, \ldots, m-1)$, it is easy to see that the time complexity of our algorithm is not greater than $O(|R|^2 |H| |tr(H)|^2)$. Thus, in these cases this algorithm finds $tr(H)$ in polynomial time in $|R|, |H|$ and

$|tr(H)|$. Obviously, if the number of elements of H is small, then this algorithm is very effective. It only requires polynomial time in $|R|$.

It can be seen that our algorithm is better than the algorithm, presented in [4], finding all minimal transversals.

We give the next example which illustrates our algorithm.

Example 2.11 Let $R = \{1,2,3,4,5,6\}$, and
$H = \{(1,2), (2,3,4), (2,4,5), (4,6)\}$.

From Algorithm 2.8 we obtain

$$L_1 = \{(1), (2)\};$$

$$L_2 = \{(1,3), (1,4), (2)\};$$

$$L_3 = \{(1,3,5), (1,4), (2)\};$$

$$L_4 = \{(2,6), (2,4), (1,3,5,6), (1,4)\}.$$

Clearly, $tr(H) = L_4$.

Now we give the algorithm, presented in [9], that finds K_a

Algorithm 2.12 [9] (Finding a minimal set of the attribute a).

Input: Let $s =< R, F >$ be a relation scheme, $A = \{a_1, \ldots, a_t\} \to \{a\}$.

Output: $A' \in K_a$.

Step 0: We set $L(0) = A$.

Step i+1: Set

$$L(i+1) = \begin{cases} L(i) - a_{i+1} & \text{if } L(i) - a_{i+1} \to \{a\}, \\ L(i) & \text{otherwise.} \end{cases}$$

Then set $A' = L(t)$.

Algorithm 2.13 [9] (Finding a family of all minimal sets of attribute a).

Input: Let $s =< R, F >$ be a relation scheme, $a \in R$.

Output: K_a.

Step 1: Set $L(1) = E_1 = \{a\}$.

Step i+1: If there are C and $A \to B$ such that $C \in L(i)$, $A \to B \in F$, $\forall E \in L(i) \Longrightarrow E \not\subseteq A \cup (C - B)$, then by Algorithm 2.12 construct an E_{i+1}, where $E_{i+1} \subseteq A \cup (C - B)$, $E_{i+1} \in K_a$. We set $L(i+1) = L(i) \cup E_{i+1}$. In the converse case we set $K_a = L(i)$.

It is shown [9] that there exists a natural number n such that $K_a = L(n)$.

It can be seen that the worst-case time complexity of algorithm is $O(|R||F||K_a|(|R| + |K_a|))$.

Thus, the time complexity of this algorithm is polynomial in $|R|, |F|$, and $|K_a|$.

Clearly, if the number of elements of K_a for a relation scheme $s =< R, F >$ is polynomial in the size of s, then this algorithm is effective. Especially, when $|K_a|$ is small.

Based on Proposition 2.4, Algorithms 2.8 and 2.13 we construct the next algorithm.

Algorithm 2.14 (Generating Armstrong relation).

Input: Let $s =< R, F >$ be a relation scheme.

Output: A relation r such that $F_r = F^+$.

Step 1: For each $a \in R$ by Algorithm 2.13 we compute K_a, and from Algorithm 2.8 find $tr(K_a)$.

Step 2: $N = \bigcup_{a \in R} tr(K_a)$

Step 3: Denote elements of N by $A_1, \ldots, A_t$ construct a relation $R = \{h_0, h_1, \ldots, h_t\}$ as follows:

For all $a \in R$, $h_0(a) = 0$, $\forall i = 1, \ldots, t$

$$h_i(a) = \begin{cases} i & \text{if } a \in A_i \\ 0 & \text{otherwise} \end{cases}$$

It is known [16] that if $s =< R, F >$ is a relation scheme. Denote $Z_s = \{A: A^+ = A\}$, and N_s is a minimal generator of Z_s. Then

$$N_s = \bigcup_{a \in R} MAX(F^+, a)$$

where

$$MAX(F^+, a) = \{A \subseteq R: A \to \{a\} \notin F^+, A \subset B \Longrightarrow B \to \{a\} \in F^+\}.$$

From this and the definitions of M_a, and N_a of the relation r we have $tr(K_a) = N_a$ for all $a \in R$. Consequently, by Proposition 2.4 we obtain $F_r = F^+$.

The estimation and the effectiveness of this algorithm are analogous to the algorithm, presented in [9] (see, Remark 2.12 in [9]), so its proof will be omitted.

Now we give the algorithm finding all antikeys, presented in [20].

Let $K = \{B_1, \ldots, B_m\}$ be a Sperner system over R.

For each $q = 1, \ldots, m$ we construct $K_q = \{B_1, \ldots, B_q\}^{-1}$ by induction.

Set $K_1 = \{R - \{a\} : a \in B_1\}$. It is obvious that $K_1 = \{B_1\}^{-1}$.

By the inductive hypothesis we have constructed $K_q = \{B_1, \ldots, B_q\}^{-1}$ for $(q < m)$.

We assume that $K_q = F_q \cup \{X_1, \ldots, X_{t_q}\}$, where $X_1, \ldots, X_{t_q}$ containing B_{q+1} and $F_q = \{A \in K_q : B_{q+1} \not\subseteq A\}$.

For all i $(i = 1, \ldots, t_q)$ construct the antikeys of $\{B_{q+1}\}$ on X_i in an analogous way as K_1. Denote them by $A_1^i, \ldots, A_{r_i}^i$ $(i = 1, \ldots, t_q)$. Let

$$K_{q+1} = F_q \cup \{A_p^i : A \in F_q \Longrightarrow A_p^i \not\subseteq A, 1 \leq i \leq t_q, 1 \leq p \leq r_i\}.$$

Set $K^{-1} = K_m$.

Denote $K_q = F_q \cup \{X_1, \ldots, X_{t_q}\}$ and l_q $(1 \leq q \leq m-1)$ is the number of elements of K_q.

Remark 2.15 [20] The time complexity of algorithm finding all antikeys is

$$O(|R|^2 \sum_{q=0}^{m-1} t_q u_q),$$

where

$$u_q = \begin{cases} l_q - t_q & \text{if } l_q > t_q \\ 1 & \text{if } l_q = t_q \end{cases}$$

According to Proposition 2.5 and the algorithm finding all antikeys we will construct the following algorithm.

Algorithm 2.16 (Inferring FDs).

Input: r be a relation over R.

Output: $s =< R, F >$ such that $F^+ = F_r$.

Step 1: From r compute the equality set E_r

Step 2: Set $N = \{A \in E_r : A \neq \cap\{B \in E_r : A \subset B\}\}$

Step 3: For each $a \in R$ find $M_a = \{A \in N_R : a \notin A, \nexists B \in N_R : a \notin B, A \subset B\}$. Compute $N_a = \{R - A : A \in M_a\}$.

Step 4: By the algorithm finding all antikeys, for each $a \in R$ construct N_a^{-1}.

Step 5: Construct $s =< R, F >$, where $F = \{R - B \to \{a\} : \forall a \in R, B \in N_a^{-1}, R - B \neq \{a\}\}$

By Proposition 2.5 we have $F_r = F^+$.

Remark 2.17 Clearly, for all $a \in R$ N_a is computed in polynomial time in the size of r. It can be seen that the complexity of Algorithm 2.16 is the complexity of step 4. By Remark 2.15, it is easy to see that the worst-case time complexity of Algorithm 2.16 is

$$O(n^2 \sum_{i=1}^{n} (\sum_{q=0}^{m_i - 1} t_{iq} u_{iq}))$$

where $R = \{a_1, \ldots, a_n\}$, $m_i = |N_{a_i}|$ and

$$u_{iq} = \begin{cases} l_{iq} - t_{iq} & \text{if } l_{iq} > t_{iq} \\ 1 & \text{if } l_{iq} = t_{iq} \end{cases}$$

Meaning of l_{iq}, t_{iq}, u_{iq} see Remark 2.15.

In cases for which $l_{iq} \leq l_{m_i}(\forall i, \forall q: 1 \leq q \leq m_i)$ the time complexity of our algorithm is $O(n^2 \sum_{i=1}^{n} |N_{a_i}||N_{a_i}{}^{-1}|^2)$. Thus, the complexity of Algorithm 2.16 is polynomial in $|R|$, $|N_{a_i}|$, $|N_{a_i}{}^{-1}|$. Clearly, in these cases if $|N_{a_i}{}^{-1}|$ is polynomial (Especially, it is small) in the size of r, then our algorithm is effective.

According to Proposition 2.6 and algorithm 2.8 we give the next algorithm for inferring FDs.

Algorithm 2.18 (Inferring FDs).

Input: r be a relation over R.

Output: $s =< R, F >$ such that $F^+ = F_r$.

Step 1: From r compute the set N_a for all $a \in R$.

Step 2: By Algorithm 2.8, construct $tr(N_a)$, for every $a \in R$.

Step 3: Construct $s =< R, F >$, where $F = \{A \to \{a\}: \forall a \in R, A \in tr(N_a), A \neq \{a\}\}$.

By Proposition 2.6 we have $F_r = F^+$.

The estimation of Algorithm 2.18 is analogous to Algorithm 2.16, so its proof will be omitted.

It can be seen that Algorithm 2.18 is similar to the algorithm inferring FDs, presented in [18]. However, it can be seen that Algorithm 2.8 is better than the algorithm, presented in [4], that is used in [18].

3 NP-complete Problems

In this section, we present some NP-complete problems related to PMDs, and the sets M_a. In Section 2, we show that these sets play important roles in generating Armstrong relation and inferring FDs.

Let $s =< R, F >$ be a relation scheme over R. Denote $L_a = \{A : A \to \{a\}, a \notin A\}$. It can be seen that L_a contains all PMDs concerning a, i.e., $T_a \subseteq L_a$.

Firstly, we introduce the following problem related to the set L_a.

Theorem 3.1 The following problem is NP-complete:

Let $s =< R, F >$ be a relation scheme over R, $a \in R$, and an integer m ($m \leq |R|$), decide whether there is an A such that $a \notin A$, $A \to \{a\}$, (i.e., $A \in L_a$), and $|A| \leq m$.

Proof. We nondeterministically choose a set A so that $|A| \leq m$, $a \notin A$, and decide whether $A \to \{a\}$ is an element of F^+. Clearly, by the polynomial time algorithm finding the closure (see [2]), our algorithm is nondeterministic polynomial. Thus, our problem lies in NP.

Now we shall show that our problem is NP-hard. It is known [15] that the problem deciding whether there exists a key having cardinality less than or equal to a given integer for relation scheme is NP-complete. Now we prove that this problem is polynomially reducible to our problem.

Let $s' =< P, F' >$ be a relation scheme over P. Now we construct the relation scheme $s =< R, F >$, as follows:

$R = P \cup a$, where $a \notin P$ and $F = F' \cup P \rightarrow \{a\}$.

It is obvious that s is constructed in polynomial time in the sizes of P and F'. Based on the construction of s and the definition of the minimal key we can see that if $A \in K_{s'}$, then $A \in K_s$. Conversely, if B is a minimal key of s, then by $R \rightarrow \{a\} \in F$ we have $a \notin B$. On the other hand, by the definition of the minimal key $B \in K_{s'}$. Thus, $K_{s'} = K_s$ holds. By $P \rightarrow \{a\} \in F$, and $a \notin P$, if $B \rightarrow \{a\}$ is a PMD of s, then $B \in K_s$. It can be seen that if $A \in K_{s'}$, then $A \rightarrow \{a\} \in F^+$. According to the definition of PMD, $A \rightarrow \{a\}$ is a PMD of s. Consequently, C is a key of s' if and only if $a \notin C$, and $C \rightarrow \{a\} \in F^+$. The theorem is proved.

Now we give the NP-complete problem concerning M_a, (see, Lemma 2.1).

Theorem 3.2 The following problem is NP-complete:

Let $s =< R, F >$ be a relation scheme over R, $a \in R$, and an integer $m\,(m \leq |R|)$, decide whether there is an A such that $a \notin A$, $A \rightarrow \{a\} \notin F^+$, and $m \leq |A|$.

Proof. By the proof of Theorem 3.1, it is clear that our problem lies in NP.

It is known [14] that the independent set problem is NP-complete :

Given integer m and a non-directed graph $G =< V, E >$, where V is the set of vertices and E is the set of edges. An independent set in G is a subset $A \subseteq V$ such that for all $a, b \in A$, the edge (a, b) is not in E. The independent set problem is deciding whether G contains an independent set A having cardinality greater than or equal to m.

We shall prove that the independent set problem is polynomially reducible to our problem.

Let $G =< V, E >$ be a non-directed graph, $m \leq |V|$. Set

$s' =< V, F' >$, where $F' = \{\{a_i, a_j\} \rightarrow V : (a_i, a_j) \in E\}$, and

$s =< R, F >$, where $R = V \cup \{a\}$, $a \notin V$, and $F = F' \cup V \rightarrow \{a\}$.

Clearly, s, s' are constructed in polynomial time in the size of G.

According to the definition of the set of edges, E is a simple hypergraph over V. From this, we can see that s' is in BCNF. Because E is the set of edges, and by the definition of the minimal key, we can see that if $(a_i, a_j) \in E$, then $\{a_i, a_j\}$ is a minimal key of s'. Conversely, if $B \in K_{s'}$, then there is an $\{a_i, a_j\}$ such that $\{a_i, a_j\} \subseteq B$. Because B is a minimal key, we have $\{a_i, a_j\} = B$. Hence, $K_{s'} = E$ holds.

Consequently, A is not a key of s' if and only if $\{a_i, a_j\} \notin A$ for all $(a_i, a_j) \in E$. Thus, A is not a key of s' if and only if A is an independent set of G.

On the other hand, by the proof of Theorem 3.1 C is a key of s' if and only if $C \rightarrow \{a\} \in F^+$, and $a \notin C$. Consequently, A is not a key of s' if and only if $a \notin A$, and $A \rightarrow \{a\} \notin F^+\}$.

Thus, A is an independent set of G if and only if A does not contain a, and $A \to \{a\} \notin F^+$. The theorem is proved.

Now we will show that Theorem 3.1 is still true for the relations.

Theorem 3.3 The following problem is NP-complete:

Let r be a relation over R, $a \in R$, and an integer $m\,(m \leq |R|)$, decide whether there is an A such that $a \notin A$, $A \to \{a\} \in F_r$, and $|A| \leq m$.N

Proof.

We nondeterministically choose a set A so that $|A| \leq m$, $a \notin A$, and decide whether $A \to \{a\} \in F_r$. Clearly, using the definition of the functional dependency, we can test in polynomial time that the functional dependency $A \to \{a\}$ holds or does not hold in r. It is obvious that our algorithm is nondeterministic polynomial. Thus, the problem lies in NP.

It is shown [14] that the the vertex cover problem is NP-complete:

Given integer m and a non-directed graph $G =< V, E >$, where V is the set of vertices and E is the set of edges, decide whether G has a vertex cover having cardinality not greater than m.

Let $G =< V, E >$ be a non-directed graph, $m \leq |V|$. Put $R = V \cup a$, where $a \notin V$.

Denote the elements of E by $E_1, \ldots, E_t$ construct a relation
$r = \{h_0, h_1, \ldots, h_t\}$, as follows:
For all $b \in R$, $h_0(b) = 0$, $\forall i = 1, \ldots, t$

$$h_i(b) = \begin{cases} i & \text{if } b \in E_i \text{ or } b = a \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, the set E is a Sperner system. From this and by the definition of N_a we can see that $N_a = \{\{a_i, a_j, a\} : (a_i, a_j) \in E\}$. Consequently, we obtain $N'_a = \{\{a_i, a_j\} : (a_i, a_j) \in E\}$. According to Proposition 2.7, $V_a = tr(\{\{a_i, a_j\} : (a_i, a_j) \in E\})$. On the other hand, by the definition of the vertex cover we can see that A is a vertex cover of G if and only if A does not contain a, and $A \to \{a\}$ is an element of F_r. The proof is complete.

Thus, for the relations Theorem 3.1 is still true. However, the next proposition shows that the problem, presented in Theorem 3.2, can be solved in polynomial time if the relation scheme is changed to the relation.

Proposition 3.4 Let r be a relation over R, $a \in R$, and an integer $m\,(m \leq |R|)$. Then the problem deciding whether there is an A such that $a \notin A$, $A \to \notin F_r$, and $m \leq |A|$ can be solved by a polynomial time algorithm.

Proof.

According to the difinitions of M_a and the antikey, and by Proposition 2.2 we can see that M_a is the family of all maximal sets A such that A doesn't contain a, and $A \to \{a\} \notin F_r$. Clearly, for every $a \in R$, we can compute the family M_a in polynomial time in the size of r.

Consequently, for relations, given an attribute a, and an integer m the problem deciding whether there is an A such that $a \notin A$, $A \rightarrow \{a\}$, and the cardinality of A is greater than or equal to m can be solved by a polynomial time algorithm. The proposition is proved.

References

[1] Armstrong W.W. Dependency Structures of Database Relationships. Information Processing 74, Holland Publ. Co. (1974) pp. 580-583.

[2] Beeri C., Bernstein P.A. Computational problems related to the design of normal form relational schemas. ACM Trans. on Database Syst. 4,1 (1979) pp. 30-59.

[3] Beeri C., Dowd M., Fagin R., Staman R. On the Structure of Armstrong relations for Functional Dependencies. J.ACM 31,1 (1984) pp. 30-46.

[4] Berge C. Hypergraphs : Combinatorics of Finite Sets (North - Holland, Amsterdam, 1989)

[5] Demetrovics J., Libkin L., Muchnik I.B. Functional dependencies and the semilattice of closed classes. Proceedings of MFDBS 87, Lecture Notes in Computer Science (1987) pp. 136-147.

[6] Demetrovics J., Thi V.D. Some results about functional dependencies. Acta Cybernetica 8,3 (1988) pp. 273-278.

[7] Demetrovics J., Thi V.D. Relations and minimal keys. Acta Cybernetica 8,3 (1988) pp. 279-285.

[8] Demetrovics J., Thi V.D. On Keys in the Relational Datamodel. Inform. Process. Cybern. EIK 24 (1988) 10, pp. 515-519.

[9] Demetrovics J., Thi V.D. Algorithms for generating Armstrong relation and inferring functional dependencies in the relational datamodel. Computers and Mathematics with Applications, Great Britain, 26, 4 (1993) pp. 43-55.

[10] Demetrovics J., Thi V.D. Some Problems concerning Keys for Relation Schemes and Relations in the Relational Datamodel. Information Processing Letters, North Holland 46, 4 (1993) pp. 179-183.

[11] Demetrovics J., Thi V.D. Some Computational Problems Related to the Functional Dependency in the Relational Datamodel. Acta Scientiarum Mathematicarum 57, 1-4 (1993) pp. 627-628.

[12] Demetrovics J., Thi V.D. Generating Armstrong relations for relation schemes and inferring functional dependencies from relations. International Journal on Information Theories and Applications, 1, 4 (1993) pp. 3-12.

[13] Demetrovics J., Thi V.D. Armstrong Relation, Functional Dependencies and Strong Dependencies. Comput. and AI, submitted for publication 1994.

[14] Garey M. R., Johnson D. S. Computers and Intractability : A Guide to the Theory of NP-Completeness. Bell Laboratories, W.H. Freeman and Company, San Francisco 1979.

[15] Lucchesi C.L., Osborn S.L. Candidate keys for relations. J. Comput. Syst. Scien. 17,2 (1978) pp. 270-279.

[16] Mannila H., Raiha K.J. Design by Example: An Application of Armstrong relations. J. Comput. Syst. Scien. 33 (1986) pp. 126-141.

[17] Mannila H., Raiha K.J. On the complexity of inferring functional dependencies. Discrete Applied Mathematics, North-Holland, 40 (1992) pp. 237-243.

[18] Mannila H., Raiha K.J. Algorithms for inferring functional dependencies from relations. Data and Knowledge Engineering, North Holland, 12 (1994) pp 83-99.

[19] Thi V.D. Investigation on Combinatorial Characterizations Related to Functional Dependency in the Relational Datamodel (in Hungarian). MTA-SZTAKI Tanulmanyok, Budapest, 191 (1986) pp. 1-157. Ph.D. Dissertation.

[20] Thi V.D. Minimal keys and Antikeys. Acta Cybernetica 7,4 (1986) pp. 361-371.

Received August, 1994