Tạp chí Tin học và Điều khiển học, T. 17, S. 3 (2001), 53–59

ON THE DESIRABILITY OF Ω-ACYCLIC DATABASE SCHEMES

NGUYEN VAN DINH

Abstract. In this paper we study a subclass of acyclic database schemes, the *ω-acyclic database schemes* and some closely related problems. We first prove that with this class given here, the notion of *acyclic hypergraphs* used by graph theorists is equivalent to the notion, in the sense relevant to database theories. In the last of the paper, new characterizations for the class of the ω-acyclic database schemes are also given.

Tóm tắt. Trong bài báo này, chúng tôi nghiên cứu một lớp con của các lược đồ cơ sở dữ liệu, đó là lớp các lược đồ CSDL *ω-phi chu trình.* Chúng tôi đã chứng minh được rằng với lớp này thì khái niệm *phi chu trình* của các siêu đồ thị được định nghĩa trong lý thuyết đồ thị và trong lý thuyết CSDL là tương đương. Phát triển các kết quả của lý thuyết đồ thị, chúng tôi đã đưa ra những đặc trưng mới cho lớp các lược đồ này.

1. INTRODUCTION

Since 1979, Namibar K. K. is the first one, who presented the idea of using hypergraph as a tool for the design of relational database schemes [8]. A database scheme is naturally viewed as a hypergraph. If $\mathcal{R}$ is a database scheme over U, then $\mathcal{R}$ may be viewed as a hypergraph $\langle U, \mathcal{R}\rangle$. That is, the attributes in $\mathcal{R}$ are the nodes in the hypergraph and the relation schemes of $\mathcal{R}$ are the hyperedges.

For the first time, since 1981, the notion of acyclic database schemes was appeared in the study of *semijoins* and the existence of a *full reducer* for a system for distributed databases (SDD-1) [10]. Then *pairwise consistency* (PC), *total consistency* (TC), the connection of *join tree* and *full reducer* of the database schemes were also studied [1], [3], [4].

These studies showed that if a database scheme is cyclic then the management is difficult and the cost is high. In addition, a cyclic database scheme may has *redundancies* and *lossy joins*, but an acyclic scheme has no above problems. In addition, it appears that queries whose hypergraph are acyclic have a number of optimization algorithms that are simpler and more efficient than those one in the general case. Thus, the *acyclicity* plays an important role on the database schemes; it is a desirable property of database schemes.

There are many equivalent definitions for the notion of acyclic hypergraph, in the sense relevant to database systems. However, none of these definitions is equivalent to the one generally used by graph theorists. Hence, the direct application of results of graph theory for the database schemes is very difficult. Some authors presented the new notions of acyclic hypergraphs to study a subclass of database schemes, such as the *γ-acyclic database schemes* [5]. In this paper, we consider a special subclass of database schemes, in which request that the intersection of nondisjoinst pair of relation schemes has only one attribute. We call this class the *ω-acyclic database schemes.* We prove that for this class the notion of *acyclic hypergraphs* used by graph theorists is equivalent to the definitions for this notion, used in the database theories.

Up to the present, many characteristics of acyclic database schemes were found and there exists some algorithms to test cyclicity of the database schemes, such as Graham algorithm, GYO algorithm ... [7], [9], [12].

In the last section of this paper, basing on the results of graph theory, we proved equivalence of the new characterizations for the ω-acyclic database schemes. The new characterizations showed the relation between the number of attributes and the number of relation schemes on the ω-acyclic database schemes.

2. HYPERGRAPHS AND DATABASE SCHEMES BACKGROUND

Some preliminary concepts about *hypergraphs* and *acyclic database schemes* presented in [2], [7], [9], [12] are summarized in this part.

2.1. Hypergraphs and cycles in a hypergraph

Definition 2.1. Let $X = \{x_1, x_2, \dots, x_n\}$ be a finite set, and let $\mathcal{E} = \{E_1, E_2, \dots, E_m\}$ be a family of subsets of X. The family $\mathcal{E}$ is said to be a hypergraph on X if:

(1) $E_i \neq \emptyset$ $(i \in I = \{1, 2, \dots, m\})$;

(2) $\bigcup_{i \in I} E_i = X$.

The pair $H = \langle X, \mathcal{E} \rangle$ is called a *hypergraph.* The elements $x_1, x_2, \dots, x_n$ are called the *vertices* (or *nodes*) and the sets $E_1, E_2, \dots, E_m$ are called the hyperedges.

H is *reduced* if no edge in $\mathcal{E}$ properly contains another edge and every node is in some edge. The *reduction of* H, written RED(H), is H with any contained edges and non-edge nodes removed.

If it is clear when dealing with hypergraphs, we may use "edges" for "hyperedges".

Definition 2.2. In a hypergraph $H = \langle X, \mathcal{E} \rangle$, a *cycle of length* q is defined to be a sequence $(x_1, E_1, x_2, E_2, \dots, x_q, E_q, x_{q+1})$ such that:

(1) $x_1, x_2, \dots, x_q$ are all distinct vertices of H.

(2) $E_1, E_2, \dots, E_q$ are all distinct edges of H.

(3) $x_k, x_{k+1} \in E_k$ for $k = 1, 2, \dots, q$.

(4) $q > 1$ and $x_{q+1} = x_1$.

If only first three conditions of the definition are satisfied, this sequence is called *a chain of length* q.

A hypergraph $H = \langle X, \mathcal{E} \rangle$ is an *acyclic hypergraph* if H does not have a cycle; otherwise it is a *cyclic hypergraph.*

Example 2.1. A cyclic hypergraph with a unique cycle of length 4: $(x_1, E_1, x_2, E_2, x_3, E_3, x_4, E_4, x_1)$

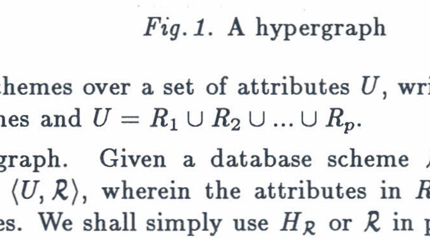

Fig. 1. A hypergraph

2.2. Acyclic Database Schemes

A *database scheme* is defined to be a set of relation schemes over a set of attributes U, written $\mathcal{R} = \{R_1, R_2, \dots, R_p\}$, wherein $R_1, \dots, R_p$ are relation schemes and $U = R_1 \cup R_2 \cup \dots \cup R_p$.

A database scheme is naturally viewed as a hypergraph. Given a database scheme $\mathcal{R} = \{R_1, R_2, \dots, R_p\}$ over U, its hypergraph , denoted $H_{\mathcal{R}} = \langle U, \mathcal{R} \rangle$, wherein the attributes in R are the nodes and the relation schemes of $\mathcal{R}$ are the hyperedges. We shall simply use $H_{\mathcal{R}}$ or $\mathcal{R}$ in place of $H_{\mathcal{R}} = \langle U, \mathcal{R} \rangle$ when dealing with the hypergraph that $\mathcal{R}$ represents.

We shall be concerned mainly with database schemes that have no proper partition into two sets of the relation schemes, such that they are disjoint. That mean its hypergraphs consist of a single connected component and it is called *connected hypergraph.*

Example 2.2. In drawing hypergraphs, nodes are represented by their labels and hyperedges are represented by closed curves around the nodes. The hypergraph for $\mathcal{R}_a = \{ABC, ADE, BE\}$ and $\mathcal{R}_b = \{ABC, AFE, EDC, AEC\}$ are given in figures 2 and 3.

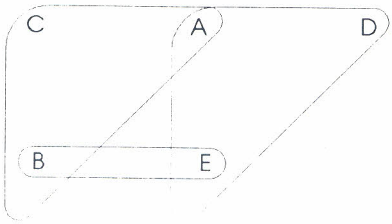

Fig. 2. Hypergraph for $H_{\mathcal{R}_a}$

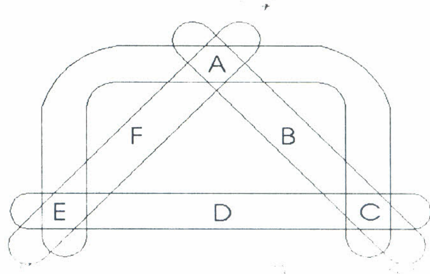

Fig. 3. Hypergraph for $H_{\mathcal{R}_b}$

Definition 2.3. Let $H = \langle X, \mathcal{E}\rangle$ and $H' = \langle X', \mathcal{E}'\rangle$ be hypergraphs, wherein $X' \subseteq X$ and $\mathcal{E}' \subseteq \mathcal{E}$, then H' is a *subhypergraph* of H.

The *X'-induced hypergraph* for H, denoted $H_{X'}$, is the reduction of hypergraph $\langle X', \mathcal{E}_{X'}\rangle$, where:

$$\mathcal{E}_{X'} = \{E_i \cap X' | E_i \in \mathcal{E}\}.$$

Note that, $H_{X'}$ is not necessarily a subhypergraph of H, since $\mathcal{E}_{X'}$ may contain edges not in $\mathcal{E}$.

Definition 2.4. Let $H = \langle X, \mathcal{E}\rangle$ be a hypergraph. A set $F \subseteq X$ is an *articulation set* for H if $F = E_1 \cap E_2$ for some pair of edges $E_1, E_2 \in \mathcal{E}$, and the induced hypergraph $H_{\{X-F\}}$ has more connected components than H.

A *block of hypergraph* H is an induced hypergraph of H with no articulation set. A block is trivial if it has only one edge.

Definition 2.5. Let $H = \langle X, \mathcal{E}\rangle$ be a hypergraph, H is acyclic if it is reduced and has no *nontrivial* blocks; otherwise it is cyclic.

A database scheme $R = \{R_1, R_2, ..., R_p\}$ is cyclic or acyclic precisely when its hypergraph $H_{\mathcal{R}}$ is.

Example 2.3. Consider the database scheme $\mathcal{R}_a = \{ABC, ADE, BE\}$, its hypergraph shown in figure 2, is a block, since it contains no articulation set. We conclude $H_{\mathcal{R}_a}$ is cyclic. Precisely, the database scheme $\mathcal{R}_a$ is cyclic.

The database scheme $\mathcal{R}_b = \{ABC, AFE, EDC, AEC\}$ has its hypergraph which is acyclic (figure 3), so $\mathcal{R}_b$ is an acyclic database scheme.

Algorithm 2.1. The Graham Reduction Algorithm [6]
The Graham reduction algorithm consists of repeated application of two reduction rules to hypergraphs until neither can be applied further. Let $H = \langle X, \mathcal{E}\rangle$ be a hypergraph. The two reduction rules are:

(1) *rE. (edge removal):* If E and F are edges in $\mathcal{E}$ such that E is properly contained in F, remove E from $\mathcal{E}$. (when, said, E is *removable edge in favor of* F).

(2) *rN. (node removal):* If A is a node in X, and A is contained in at most one edge in $\mathcal{E}$, remove A from X and also from all edges in $\mathcal{E}$ in which it appears.

We say the Graham reduction *succeeds* on hypergraph H if the result of applying the Graham reduction algorithm to H is an empty hypergraph.

Theorem 2.1. The Equivalence Theorem for Acyclic Database Schemes [7]
Let $\mathcal{R}$ is a connected database scheme, the following conditions are equivalent:

(1) *$\mathcal{R}$ is acyclic;*
(2) *Graham reduction succeeds on $\mathcal{R}$;*
(3) *$\mathcal{R}$ has a join tree;*
(4) *$\mathcal{R}$ has a full reducer;*
(5) *PC (pair wise consistency) implies TC (total consistency) for $\mathcal{R}$;*

(6) $\mathcal{R}$ *has the running intersection property;*
(7) $\mathcal{R}$ *has the increasing join property;*
(8) RED($\mathcal{R}$) *is a unique 4NF decomposition;*
(9) *The maximum weight spanning tree for* $\mathcal{R}$ *is a join tree;*
(10) MVD($\mathcal{R}$) $\models$ $*[\mathcal{R}]$.

The proof of this theorem will proceed via a series of lemmas, and can be found in [7].

The first two equivalent conditions of this theorem show that hypergraph $H_{\mathcal{R}}$ is acyclic if and only if Graham reduction succeeds on $H_{\mathcal{R}}$. Thus, we can use condition (2) as a definition for acyclic property of a hypergraph $H_{\mathcal{R}}$ of database schemes $\mathcal{R}$.

Example 2.4. Applying the Graham reduction algorithm to hypergraphs

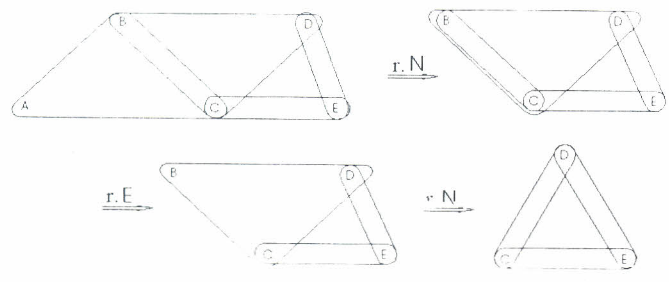

Fig. 4. $\mathcal{R}_c = \{ABC, BCD, CE, DE\}$

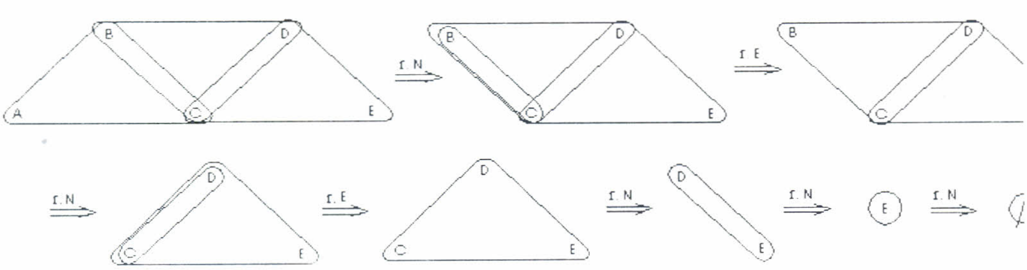

Fig. 5. $\mathcal{R}_d = \{ABC, BCD, CDE\}$

The result of the Graham reduction algorithm to $H_{\mathcal{R}_c}$ is a nonempty hypergraph; thus this hypergraph is cyclic (Fig. 4). Otherwise, hypergraph $H_{\mathcal{R}_d}$ is acyclic, since Graham reduction succeeds on it. (Fig. 5).

3. THE Ω-ACYCLIC DATABASE SCHEMES

In this section, we define a subclass of hypergraphs, the *ω-acyclic hypergraphs*, and we shall prove that with this class the notion of acyclic hypergraph used by graph theorists is equivalent to this notion that used by database theorists. In the last of this section, basing on the results of graph theory, we can prove two new characterizations for the ω-acyclic hypergraphs.

We first prove lemmas and present examples to show that the notion of acyclic hypergraphs used in graph theory is not equivalent to that one used in database theories.

Lemma 3.1. *Let* $H = \langle X, \mathcal{E} \rangle$ *be a hypergraph. H is acyclic (in the sense of Definition 2.2) only if* $|E_i \cap E_j| \leq 1$ *for every pair of edges* $E_i, E_j \in \mathcal{E}$.

Proof. Suppose that H is acyclic, and assume the contrary, that there exists a pair of edges $E_i \neq E_j$, $E_i, E_j \in \mathcal{E}$, such that, $|E_i \cap E_j| > 1$. Assume that $\{x_i, x_j\} \subseteq E_i \cap E_j$, thus there are x_i, x_j in

$E_i \cap E_j$. Consider the sequence $(x_i, E_i, x_j, E_j, x_i)$. It is clear that this sequence satisfies conditions (1) through (4) of Definition 2.2. Hence, it is a *cycle of length* 2. Thus H is cyclic. This contradicts the hypothesis. The proof is completed. □

Lemma 3.2. *Let $H_{\mathcal{R}} = \langle X, \mathcal{E} \rangle$ be the connected hypergraph for a database scheme $\mathcal{R}$. If the Graham reduction algorithm does not succeed on $H_{\mathcal{R}}$ then the result of the Graham reduction algorithm on $H_{\mathcal{R}}$ (the remaining part of $H_{\mathcal{R}}$) has at least three distinct hyperedges and three distinct nodes.*

Proof. Suppose the contrary, the remaining part of $H_{\mathcal{R}}$ has only two edges $E_{i_1} \neq E_{i_2}$. Thus there exists $x_{i_1} \in E_{i_1}$, $x_{i_1} \notin E_{i_2}$ and we can remove x_{i_1} by rN. rule. Now $E_{i_1} \subseteq E_{i_2}$, and E_{i_1} can be removed by rE. rule. The remaining part of $H_{\mathcal{R}}$ has only one hyper edge and we can remove it. So, $H_{\mathcal{R}}$ is empty, which contradicts the fact that the Graham reduction algorithm does not succeed on $H_{\mathcal{R}}$.
Otherwise, if the remaining part of $H_{\mathcal{R}}$ has only two nodes, it can not have three distinct hyperedges. The lemma is proved. □

Lemma 3.3. *Let $H_{\mathcal{R}} = \langle X, \mathcal{E} \rangle$ be the connected hypergraph for a database scheme $\mathcal{R}$. If $H_{\mathcal{R}}$ is acyclic according to the Definition 2.2 (said, G-definition) then it is acyclic according to the definition in relational database theories (said, R-definition).*

Proof. Suppose that $H_{\mathcal{R}}$ is acyclic according to Definition 2.2 (*G-definition*), we have only to prove that the Graham reduction succeeds on $H_{\mathcal{R}}$, i.e. it is acyclic according to *R-definition*.

Assume the contrary, that the Graham reduction does not succeed on $H_{\mathcal{R}}$. According to the Lemma 3.2, the remaining part of $H_{\mathcal{R}}$ has at least three hyperedges and three nodes, namely $E_{i1} \neq E_{i2} \neq E_{i3}$ and $x_{i1} \neq x_{i2} \neq x_{i3}$ (If it has more than three, the proof is similar). Each node x_{ij} should be in at least two hyperedges, because if not so, this node can be removed by the *rN. rule* of Graham reduction. We always can build a sequence $(x_{i1}, E_{i1}, x_{i2}, E_{i2}, x_{i3}, E_{i3}, x_{i4})$ wherein $x_{ij}, x_{ij+1} \in E_{ij}$ $(j = 1, 2, 3)$. Thus we have $x_{i2} \in E_{i1} \cap E_{i2}$ and $x_{i3} \in E_{i2} \cap E_{i3}$. Since $H_{\mathcal{R}}$ is acyclic (by *G-definition*), then by Lemma 3.1 applied to connected hypergraph $H_{\mathcal{R}}$ there exists $|E_i \cap E_j| = 1$ for every pair of edges of $\mathcal{E}$. However, there is only x_{i2} in $E_{i1} \cap E_{i2}$ and only x_{i3} in $E_{i2} \cap E_{i3}$, so x_{i1}, x_{i4} should be in $E_{i1} \cap E_{i3}$, once again apply Lemma 3.1, we have $x_{i1} = x_{i4}$. We see that the above sequence satisfies the conditions of the Definition 2.2, thus it is a cycle of length 3. This contradicts the hypothesis that $H_{\mathcal{R}}$ is acyclic.
The proof is completed. □

Example 3.1. Consider the hypergraph $H_{\mathcal{R}_b}$ (Fig. 2) for the database scheme $\mathcal{R}_b = \{ABC, AFE, EDC, AEC\}$. Since this hypergraph has the cycle of length 3 $(A, \{AFE\}, E, \{EDC\}, C, \{CBA\}, A)$, thus it is cyclic according to the Definition 2.2. On the other hand, it is easy to verify that the Graham reduction succeeds on $H_{\mathcal{R}}$. Hence, the notion of *acyclic hypergraphs* used by graph theorists is not equivalent to the definitions for the notion, used in the database theories.

Definition 3.1. Let $H = \langle X, \mathcal{E} \rangle$ be a hypergraph. H is called *ω-hypergraph* if $|E_i \cap E_j| \leq 1$ for every pair of distinct edges $E_i, E_j \in \mathcal{E}$.

If an *ω-hypergraph* H is acyclic (cyclic, respectively) then H is called *ω-acyclic* (*ω-cyclic*, respectively) hypergraph.

A database scheme $\mathcal{R}$ is *ω-acyclic* (*ω-cyclic*, respectively) if the hypergraph for $\mathcal{R}$ is *ω-acyclic* (*ω-cyclic*, respectively).

The following theorem will show that with *ω-hypergraph* the notion of *acyclic* used by graph theorists is equivalent to that used by database theorists.

Theorem 3.1. *Let $H = \langle X, \mathcal{E} \rangle$ be a ω-hypergraph, then the two following conditions are equivalent:*
(1) H is acyclic according to the G-definition in graph theory;
(2) H is acyclic according to the R-definition in database theories.

Proof. The proof will proceed via following steps:

(1) $\Rightarrow$ (2) The proof is immediate from Lemma 3.3.
(2) $\Rightarrow$ (1) Suppose that H is acyclic according to the *R-definition*, thus the Graham reduction succeeds on hypergraph H. We have to prove that H is also acyclic according to the *G-definition*, i.e. H does not have a cycle. Consider an arbitrary chain $(x_1, E_1, x_2, E_2, \dots, x_q, E_q, x_{q+1})$ of H, we need only show that $x_1 \neq x_{q+1}$. Suppose the contrary, that $x_1 = x_{q+1}$. This chain should satisfies the conditions (1), (2), (3) of the Definition 2.2, so we have:

$$x_i \in E_{i\text{-}1} \cap E_i\,, \text{ for } i = 2, 3, \dots, q.$$

Otherwise,

$$x_{q+1} = x_1 \in E_1\,,$$
$$x_1 = x_{q+1} \in E_q\,.$$

Hence, we get

$$x_1 \in E_1 \cap E_q\,.$$

It is clear that each x_i $(i = 1, 2, \dots, q)$ belongs to at least two edges, thus no x_i can be removed from this chain. This contradicts the hypothesis that the Graham reduction succeeds on hypergraph H. The theorem is proved. □

The next theorem will be fundamental in this paper.

Theorem 3.2. *Let $\mathcal{R} = \{R_1, R_2, \dots, R_p\}$ be a connected database scheme over the set of the attributes U. The following conditions are equivalent:*
(1) *$\mathcal{R}$ is ω-acyclic;*
(2) $\sum_{1 \leq i \leq p} (|R_i| - 1) = |U| - 1$;
(3) $\left| \bigcup_{i \in J} R_i \right| > \sum_{i \in J} (|R_i| - 1)$, for any $J \subset I = \{1, 2, \dots, p\}$, $J \neq \emptyset$.

Proof. Let $H_{\mathcal{R}}$ be the hypergraph for database scheme $\mathcal{R}$. The proof will proceed via following steps:

(1) $\Leftrightarrow$ (2) Consider the bipartite graph $G(H_{\mathcal{R}})$ whose nodes represent the nodes and hyperedges of $H_{\mathcal{R}}$, wherein the nodes that representing $x_j \in U$ is joined to the nodes representing R_i if and only if $x_j \in R_i$. Hence, the number of the nodes of $G(H_{\mathcal{R}})$ is $\sum_{1 \leq i \leq p} |R_i|$. For example, let $\mathcal{R} = \{AB, BCD, CE\}$, said x_1, x_2, x_3, x_4, x_5 are nodes which represent the attributes A, B, C, D, E and e_1, e_2, e_3 are nodes which represent the relation schemes $R_1 = (A\,B)$, $R_2 = (B\,C\,D)$, $R_3 = (C\,E)$. Then the bipartite graph $G(H_{\mathcal{R}})$ for $H_{\mathcal{R}}$ is:

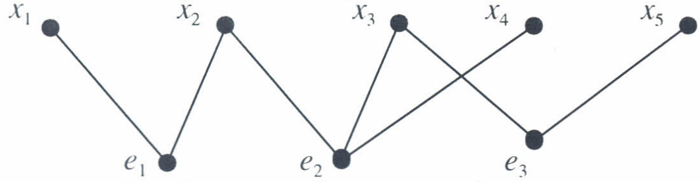

Fig. 6. Graph $G(H_{\mathcal{R}})$

It is clear that hypergraph $H_{\mathcal{R}}$ is acyclic if and only if $G(H_{\mathcal{R}})$ is a tree, this condition is equivalent to the following condition

$$\sum_{1 \leq i \leq p} |R_i| = |U| + p - 1.$$

i.e.

$$\sum_{1 \leq i \leq p} (|R_i| - 1) = |U| - 1.$$

(1) $\Leftrightarrow$ (3) +(if) Suppose that the condition (3) is satisfied, we have to prove $H_{\mathcal{R}}$ is acyclic. Assume the contrary, that $H_{\mathcal{R}}$ is cyclic, i.e. it has a *cycle of length* q $(q < p)$ $(x_1, R_1, x_2, R_2, \dots, x_q, R_q, x_{q+1})$, wherein $x_1 = x_{q+1}$, let $J = \{1, 2, \dots, q\}$. We have:

$$|\bigcup_{i\in J} R_i| = |\bigcup_{i\in J}(R_i - \{x_i\})| \leq \sum_{i\in J} |R_i - \{x_i\}| = \sum_{i\in J}(|R_i| - 1).$$

This inequality conflicts with the condition (3), so $H_{\mathcal{R}}$ is acyclic.
+(only if) Now we suppose that $H_{\mathcal{R}}$ is acyclic. Therefore, an arbitrary subhypergraph $\{R_i | i \in J\} \subset \mathcal{R}$, is acyclic. According to the condition (2), we have:

$$\sum_{i\in J}(|R_i| - 1) = |\bigcup_{i\in J} R_i| - 1 < |\bigcup_{i\in J} R_i|, \ J \subset I.$$

The theorem is proved. □

Example 3.2. Consider the database scheme $\mathcal{R}_a = \{ABC, ADE, BE\}$. Its hypergraph is showed in figure 2. We have $|U| = 5$; $R_1 = (ABC)$, $R_2 = (ADE)$, $R_3 = (BE)$. It is clear that $\mathcal{R}_a$ is connected and $|R_i \cap R_j| \leq 1$ for $i \neq j$. Otherwise, we have $\sum(|R_i| - 1) = 2 + 2 + 1 = 5 > |U| - 1$, so the condition (2) of Theorem 3.2 is not satisfied. Hence, $\mathcal{R}_a$ is cyclic.

Example 3.3. Consider the database scheme $\mathcal{R}_e = \{AB, BCD, DE, CF\}$. We have $|U| = 6$, $R_1 = (AB)$, $R_2 = (BCD)$, $R_3 = (DE)$, $R_4 = (CF)$. It is clear that $\mathcal{R}_a$ is connected and $|R_i \cap R_j| \leq 1$ for $i \neq j$. Otherwise, we have $\sum(|R_i| - 1) = 1 + 2 + 1 + 1 = 5 = |U| - 1$, so the condition (2) of Theorem 3.2 is satisfied. Hence, $\mathcal{R}_a$ is acyclic.

REFERENCES

[1] Aho A. V., Beeri C., and Ullman J. D., The theory of Joins in relational databases, *ACM Transactions on Database System* **4** (3) (1979) 297–314.
[2] Berge C., *Graphs and Hypergraphs*, North Holland, Amsterdam, The Netherlands, 1973.
[3] Bernstein P. A. and Chiu D. M., Using Semi-joins to solve relational queries, *JACM* **28** (1) (1981) 25–40.
[4] Bernstein P. A. and Goodman N., *Full Reducers for Relational Queries using Multi-Attribute Semi-Joins*, IEEE Computer Network Symp., 1979.
[5] Edward P. F. Chan, Hector J. Hernandez, On the desirability of γ-acyclic BCNF database schemes, *Proceedings of ICDT*, Italy, 1986.
[6] Graham M. H., On the universal relation, *Computer Systems Research Group Report*, Univ. of Toronto, Canada, 1979.
[7] Maier D., *The Theory of Relational Databases*, Computer Science Press, 1982.
[8] Namibar K. K., Some analytic tools for the design of relational database system, VLDB V, Rio de Janeiro, Brazil; *ACM, IEEE* (1979) 417–428.
[9] Nguyen Van Dinh, On the acyclic database schemes, *Proceedings of National Workshop on Informatics and Technology*, Hue, June 2000, Science-Technique Press, Hanoi, 2001, p. 44–55.
[10] Rothnie J. B., Bernstein P. A., et al., Introduction to a system for distributed databases (SDD-1), *ACM TODS* **5** (1) (1981) 1–17.
[11] S. Nguyen, D. Pretolani, and L. Markenzon, Some Path problems on oriented hypergraphs, *Theoretical Informatics and Applications*, Elsevier, Paris, **32** (1-2-3) (1998).
[12] Ullman Jeffrey D., *Principles of Database and Knowledge-Base Systems*, Computer Science Press, USA, 1989.
[13] Ho Thuan and Nguyen Van Dinh, Hypergraph representation of a join-expression of relations and determination of a full reducer, *National Workshop on Informatics and Technology*, Hai Phong, June 2001.

Received April 20, 2001
Revised July 31, 2001

United Nations International School, Hanoi