Tạp chí Tin học và Điều khiển học, T. 18, S. 3 (2002), 290–294

ON THE DIRECT PRODUCT OF CHOICE FUNCTION

BINA RAMAMURTHY, VU NGHIA, VU DUC THI

Abstract. The closure operations and choice functions are equivalent descriptions of family of functional dependencies. The direct product of closure operations play very important role in theory of relational database, especially in combinatorics problems. The main goal of this paper is to define the direct product of choice functions. Some properties about the direct product of choice functions are presented in our paper.

Tóm tắt. Các toán tử đóng và các hàm chọn là những mô tả tương đương của họ các phụ thuộc hàm. Tích trực tiếp của các toán tử đóng có vai trò quan trọng trong lý thuyết cơ sở dữ liệu quan hệ, đặc biệt về tổ hợp. Mục đích chính của bài báo này là định nghĩa khái niệm tích trực tiếp của các hàm chọn và sau đó trình bày một số tính chất của nó.

1. INTRODUCTION

Direct product of decomposition of a closure operation plays an important role in the theory and pratice of relational database. We consider a relation of database as a matrix. A row contains the data of one individual. The estimation of the minimum cardinality of rows of such matrix is very valuable in pratice of relational database. The studies of estimation of the minimum cardinality of rows for direct product of decomposition of a closure operation can be found variously in [5, 7, 14]. In this paper we present the new notion and properties of direct product of decomposition of choice function.

In the next section some neccessary definitions and facts about relation database, some equivalent descriptions of family of functional dependencies besides choice function and closure operation theory are given.

2. BASIC DEFINITIONS

Let us give some formal definitions that are used in the next section. Those well-known concepts in relational database given in this section can be found in [2, 3, 4, 7, 10, 20].

A relational database system of the scheme $R(a_1, ..., a_n)$ is considered as a table, where columns corespond to the attributres a_i's, while the row are n-tuples of relation r. Let X and Y be nonempty sets of attributes in R. We say that instance r of R satisfies the functional dependency (FD) if two tuples agree on the values in attributes X, they must also agree on the values in attributes Y. Here is the formal mathematical definition of FDs.

Definition 2.1. Let $U = \{a_1, ..., a_n\}$ be a nonempty finite set of attributes. A functional dependency is a statement of the form $A \to B$, where $A, B \subseteq U$. The FD $A \to B$ holds in a relaion $R = \{h_1, ..., h_m\}$ over U 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.

Definition 2.2. 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) $(A \to B \in F, A \subseteq C, D \subseteq B) \Rightarrow (C \to D \in F)$,

(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 over U.

Clearly, F_R is an f-family over U. It is known [2] that if F is an arbitrary f-family, then there is a relation R over U such that $F_R = F$.

Given a family F of FDs over U, there exits 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).

Definition 2.3. A relation scheme s is a pair $\langle U, F\rangle$, where U is a set of attributes, and F is a set of FDs over U.

Denote $A^+ = \{a : A \to \{a\} \in F^+\}$. A^+ is called the closure of A over s.

It is clear that $A \to B \in F^+$ if $B \subseteq A^+$.

Clearly, if $s = \langle U, F\rangle$ is a relation scheme, then there is a relation R over U such that $F_R = F'$ (see [2]).

Definition 2.4. Let U be a nonempty finite set of attributes and $P(U)$ be its power set. A map $L : P(U) \to P(U)$ is called a closure operation (closure for short) over U if it satisfies the following conditions:

(1) $A \subseteq L(A)$ (Extensiveness Property),

(2) $A \subseteq B$ implies $L(A) \subseteq L(B)$ (Monotonicity Property),

(3) $L(L(A)) = L(A)$ (Closure Propery).

Let $s = \langle U, F\rangle$ be a relation scheme. Set $L(A) = \{a : A \to \{a\} \in F'\}$, we can see that L is a closure over U.

Theorem 2.1. ([2]) *If F is a f-family and if $L_F = \{a : a \in U$ and $A \to \{a\} \in F\}$, then L_F is a closure. Inversely, if L is a closure there exists only a f-family F over U such that $L = L_F$, and $F = \{A \to B : A, B \subseteq U, B \subseteq L(A)\}$.*

Let $L \subseteq P(U)$. L is called a meet-irreducible family over U (sometimes it is called a family of members which are not intersection of two other members) if $A, B, C \in L$, then $A = B \cap C$ implies $A = B$ or $A = C$.

Let $I \subseteq P(U)$, $U \in I$, and $A, B \in I \Rightarrow A \cap B \in I$, I is called a meet-semilattice over U. Let $M \subset P(U)$.

Denote $M^+ = \{\cap M' : M' \subseteq M\}$. We say that M is a generator of I if $M^+ = I$. Note that $U \in M^+$ but not in M, by convention it is the intersection of the empty collection of sets.

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

In [7] it is proved that N is the unique minimal generator of I.

It can be seen that N is a family of members which are not intersections of two other members.

Let L be a closure operation over U. Denote $Z(L) = \{A : L(A) = A\}$ and $N(L) = \{A \in Z(L) : A \neq \cap\{A' \in Z(L) : A \subset A'\}\}$. $Z(L)$ is called the family of closed sets of L. We say that $N(L)$ is the minimal generator of L.

It is shown [7] that if N is a meet-irreducible family then there is a closure L such that N is the minimal generator of it.

Theorem 2.2. ([2]) *There is an one-to-one correspondence between meet-irreducible families and f-families on U.*

Theorem 2.3. ([7]) *There is 1-1 correspondence between meet-irreducible families and meet-semilattices on U.*

Definition 2.5. Let $M \subseteq P(U)$, M is called a Sperner system over U if $A, B \in M$, then A is not a subset of B.

Definition 2.6. Let U be a nonempty finite set of attributes and $P(U)$ be its power set. A map $C : P(U) \to P(U)$ is called a choice function, if every $A \in P(U)$, then $C(A) \subseteq A$.

U is interpreted as a set of alternatives, A as a set of alternatives given to the decision-maker to choose the best and $C(A)$ as a choice of the best alternatives among A.

Let L be a closure operation, we define C and H associated with L as follows:

$$C(A) = U - L(U - A) \ (*), \quad \text{and} \quad H(A) = A \cap L(U - A) \ (**)$$

We can easily prove that $C(A)$ and $H(A)$ are two choice functions. And we name $C(A)$ choice function - I (for short, CF-I), and $H(A)$ choice function - II (for short, CF-II).

Theorem 2.4. *The relationship like (*) is considered as a 1-1 correspondence between closures and choice functions, which satisfies the following two conditions:*

For every $A, B \subseteq U$,

(1) *If $C(A) \subseteq B \subseteq A$, then $C(A) = C(B)$ (Out Casting Property),*
(2) *If $A \subseteq B$, then $C(A) \subseteq C(B)$ (Monotonicity Property).*

Theorem 2.5. *The relationship like (**) is considered as a 1-1 correspondence between closures and choice functions, which satisfies the following two conditions:*
For every $A, B \subseteq U$,
(1) *If $H(A) \subseteq B \subseteq A$, then $H(A) = H(B)$ (Out Casting Property).*
(2) *If $A \subseteq B$, then $H(B) \cap A \subseteq H(A)$ (Heredity Property).*

We also note that both C and H uniquely determine the closure L as the following

$$L(A) = U - C(U - A) \text{ and } H(A) = A \cup L(U - A).$$

For every $A \subseteq U$, the sets $C(A)$ and $H(A)$ form a partition of A, that is, $C(A) \cup H(A) = A$, and $C(A) \cap H(A) = \emptyset$.

Theorem 2.6. *There is a 1-1 correspondence between CFs-I and closure operations on U.*

Theorem 2.7. *There is a 1-1 correspondence between CFs-II and closure operations on U.*

3. RESULT

The direct product of closure operations play very important role in theory of relational database, especially in combinatorics problems. A plenty of properties related to direct product of closure operation can be found in [5], [14]. By relationship and interaction between closure operations and choice fuctions, we introduce the new definitions of direct product of choice function - Is as well as IIs. First of all, we have the following.

First of all, we are giving the formal definition of composition of functions.

Definition 3.1. Let f and g be two functions (e.g. closure operations, CFs - I, or II) on U, and we determine a map T as a composition of f and g the following:

$$T(X) = f(g(X)) = f.g(X) = fg(X) \text{ for every } X \subseteq U$$

Lemma 3.1. ([1]) *Let C_1 and C_2 be CFs -I on U, then following is equivalence:*
(1) $C_1 \subseteq C_2$,
(2) $C_1C_2 = C_1$.

Corollary 3.1. *If C is a CF-I on U, then* CC = C.

Theorem 3.1. ([14]) *Let L_1 and L_2 be closure operations on the disjoint ground sets U_1 and U_2 respectively. The direct product of closure operations $L_1 \times L_2$ is defined as following:*

$$(L_1 \times L_2)(X) = L_1(X \cap U_1) \cup L_2(X \cap U_2), \quad X \subseteq U_1 \cup U_2.$$

Then $(L_1 \times L_2)(X)$ is a closure operations on $U_1 \cup U_2$.

Here we give the generalization of above theorem.

Generalization 3.1. *Let $\{L_i \mid i = 1 \rightarrow n\}$ be closure operations on the disjoint ground sets $\{U_i\}$ respectively. The direct product of those closure operations $L_1 \times L_2 \times ... \times L_n$ is defined as following*

$$(L_1 \times L_2 \times ... \times L_n)(X) = \bigcup_{i=1}^{n} L_i(X \cap U_i)$$

with $X \subseteq U_1 \cup U_2 \cup ... \cup U_n$.

Then $(L_1 \times L_2 \times ... \times L_n)(X)$ *is a closure operation on* $U_1 \cup U_2 \cup ... \cup U_n$.

Theorem 3.2. *Let* C_1 *and* C_2 *be CFs -I on the disjoint ground sets* U_1 *and* U_2 *respectively. The direct product of CF-I,* $C_1 \times C_2$*, is defined as following:*

$$(C_1 \times C_2)(X) = C_1(X \cap U_1) \cup C_2(X \cap U_2), \quad X \subseteq U_1 \cup U_2.$$

Then $(C_1 \times C_2)(X)$ *is a CF- I on* $U_1 \cup U_2$.

Proof. For all $X \subseteq U_1 \cup U_2$, $(C_1 \times C_2)(X) = C_1(X \cap U_1) \cup C_2(X \cap U_2) \subseteq (X \cap U_1) \cup (X \cap U_2) \subseteq X \cap (U_1 \cup U_2) = X$. Thus, $(C_1 \times C_2)(X) \subseteq X$. For every X and $Y \subseteq U_1 \cup U_2$ and $X \subseteq Y$, then $X \cap U_1 \subseteq Y \cap U_1$, and $X \cap U_2 \subseteq Y \cap U_2$. By using Monotonicity Property of C_1 and C_2, we obtain $C_1(X \cap U_1) \subseteq C_1(Y \cap U_1)$ and $C_2(X \cap U_2) \subseteq C_2(Y \cap U_2)$. Hence $C_1(X \cap U_1) \cup C_2(X \cap U_2) \subseteq C_1(Y \cap U_1) \cup C_2(Y \cap U_2)$, that is $(C_1 \times C_2)(X) \subseteq (C_1 \times C_2)(Y)$ or $(C_1 \times C_2)$ satisfies Monotonicity Property. Now we need to show that $(C_1 \times C_2)(X)$ satisfies the Out Casting Property also. That is, for every $X, Y \subseteq U_1 \cup U_2$ and $(C_1 \times C_2)(X) = C_1(X \cap U_1) \cup C_2(X \cap U_2) \subseteq Y \subseteq X$, we need to show that $(C_1 \times C_2)(X) = (C_1 \times C_2)(Y)$. Since $Y \subseteq X$, we have $(C_1 \times C_2)(Y) \subseteq (C_1 \times C_2)(X)$. And it is obvious that $C_1(X \cap U_1) \subseteq C_1(X \cap U_1) \cup C_2(X \cap U_2) \subseteq Y$. Thus, we have $C_1(X \cap U_1) \cap U_1 \subseteq Y \cap U_1$ or $C_1(X \cap U_1) \subseteq Y \cap U_1$. Using Monotonicity Property of C_1, we have $C_1(C_1(X \cap U_1)) \subseteq C_1(Y \cap U_1)$ or $C_1(X \cap U_1) \subseteq C_1(Y \cap U_1)$ due to Corollary 3.1. Similarly, we obtain $C_2(X \cap U_1) \subseteq C_2(Y \cap U_1)$. Therefore $C_1(X \cap U_1) \cup C_2(X \cap U_2) \subseteq C_1(Y \cap U_1) \cup C_2(Y \cap U_2)$ or $(C_1 \times C_2)(X) \subseteq (C_1 \times C_2)(Y)$. Hence $(C_1 \times C_2)(X) = (C_1 \times C_2)(Y)$. The proof is completed. ■

Generalization 3.2 *Let* $\{C_i \mid i = 1, ..., n\}$ *be CFs-I with on the disjoint ground sets* $\{U_i\}$ *respectively. The direct product of CFs-I,* $C_1 \times C_2 \times ... \times C_n$*, is defined as following*

$$(C_1 \times C_2 \times ... \times C_n)(X) = \bigcup_{i=1}^{n} C_i(X \cap U_i)$$

with $X \subseteq U_1 \cup U_1 \cup U_1 \cup ... \cup U_n$.
Then $(C_1 \times C_2 \times ... \times C_n)(X)$ *is a* CFs-I *on* $U_1 \cup U_1 \cup U_1 \cup ... \cup U_n$.

Theorem 3.3. *Let* H_1 *and* H_2 *be CFs-II on the disjoint ground sets* U_1 *and* U_2 *respectively. The direct product of CFs -II,* $H_1 \times H_2$ *is defined as following*

$$(H_1 \times H_2)(X) = H_1(X \cap U_1) \cup H_2(X \cap U_2), \quad X \subseteq (U_1 \cup U_2)$$

Then $(H_1 \times H_2)(X)$ *is a CFs-II on* $U_1 \cup U_2$.

Proof. For all $X \subseteq (U_1 \cup U_2)$, $(H_1 \times H_2)(X) = H_1(X \cap U_1) \cup H_2(X \cap U_2) \subseteq (X \cap U_1) \cup (X \cap U_2) \subseteq X \cap (U_1 \cup U_2) = X$. Thus, $(H_1 \times H_2)(X) \subseteq X$. For every X and $Y \subseteq U_1 \cup U_2$ and $X \subseteq Y$, we need to prove that $(H_1 \times H_2)$ satisfies Heredity Property. Since $X \subseteq Y$, we have $X \cap U_1 \subseteq Y \cap U_1$, and $X \cap U_2 \subseteq Y \cap U_2$. By using Heredity Property of H_1 and H_2, we obtain $H_1(Y \cap U_1) \cap (X \cap U_1) \subseteq H_1(X \cap U_1)$ or $H_1(Y \cap U_1) \cap X \subseteq H_1(X \cap U_1)$. Similarly, we have $H_2(Y \cap U_2) \cap X \subseteq H_2(X \cap U_2)$. Hence, $(H_1(Y \cap U_1) \cap X) \cup (H_2(Y \cap U_2) \cap X) \subseteq H_1(X \cap U_1) \cup H_2(X \cap U_2)$, then $(H_1(Y \cap U_1) \cup H_2(Y \cap U_2)) \cap X \subseteq H_1(X \cap U_1) \cup H_2(X \cap U_2)$, that is, $(H_1 \times H_2)(Y) \cap X \subseteq (H_1 \times H_2)(X)$ or $(H_1 \times H_2)$ satisfies Heredity Property.

Now we need to show that $(H_1 \times H_2)(X)$ satisfies the Out Casting Property also. That is, for every X and $Y \subseteq U_1 \cup U_2$ and $(H_1 \times H_2)(X) = H_1(X \cap U_1) \cup H_2(X \cap U_2) \subseteq Y \subseteq X$, we need to show that $(H_1 \times H_2)(X) = (H_1 \times H_2)(Y)$. It is obvious that $H_1(X \cap U_1) \subseteq H_1(X \cap U_1) \cup H_2(X \cap U_2) \subseteq Y \subseteq X$. Then $H_1(X \cap U_1) \cap U_1 \subseteq Y \cap U_1 \subseteq X \cap U_1$ or $H_1(X \cap U_1) \subseteq Y \cap U_1 \subseteq X \cap U_1$. Using Out Casting Property of H_1, we obtain $H_1(X \cap U_1) = H_1(Y \cap U_1)$. Similarly, we attain $H_2(X \cap U_2) = H_2(Y \cap U_2)$. Therefore $H_1(X \cap U_1) \cup H_2(X \cap U_2) = H_1(X \cap U_1) \cup H_2(Y \cap U_2)$ or $(H_1 \times H_2)(X) = (H_1 \times H_2)(Y)$. The proof is completed. ■

Generalization 3.3. *Let* $\{H_i \mid i = 1, 2, ..., n\}$ *be CFs-II with on the disjoint ground sets* $\{U_i\}$ *respectively. The direct product of CFs-II,* $H_1 \times H_2 \times ... H_n$*, is defined as following with* $X \subseteq U_1 \cup U_2 \cup ... \cup U_n$.

$$(H_1 \times H_2 \times ... \times H_n)(X) = \bigcup_{i=1}^{n} H_i(X \cap U_i).$$

Then $(H_1 \times H_2 \times ... \times H_n)(X)$ *is a CF-II on* $U_1 \cup U_2 \cup ... \cup U_n$.

REFERENCES

[1] Arenas M., Bertossi L., Chomicki J., Consistent Query Answers in Inconsistent Databases, *Proc. ACM Symposium on Priciples of Database Systems* (1999) 68–79.

[2] Armstrong W.W., Dependency Structures of Database Relationships, *Information Processing* **74** (1974) 580–583.

[3] Beeri C., Bernstein P.A., Computational Problems Related to the Design of normal form Relation Relation Schemes, *ACM Trans. on Database Syst.* **1–4** (1979) 30–59.

[4] Beeri C., Dowd M., Fagin R., Staman R., On the Structure of Armstrong Relations for Functional Dependencies, *ACM* **(31)** (1) (1984) 30–46.

[5] Demetrovics J., Furedi Z., Katona G.O.H., Minimum Matrix Representation of closure operations, *Discrete Applied Mathematics* **11** (1985) 115–128.

[6] Demetrovics J., Katona G.O.H., Extremal Combinatorial Problems of Databases, *Symposium on Mathematical Fundamental of Database Systems*, Dresden, GDR, January, 1987.

[7] Demetrovics J., Katona G.O.H., A Survey of some Combinatorial Results Concerning Functional dependencies in Database Relations, *Annals of Mathematics and Artificial Intelligence*, **7** (1993) 63–82.

[8] Demetrovics J., Libkin L., Muchnik I.B., Functional Dependencies in Relational Databases: Alattice point of view, *Discrete Applied Mathematics* **40** (1992) 155–185.

[9] Demetrovics J., Hencsey G., Libkin L., Muchnik I.B., Normal Form Relation Schemes: A New Characterization, *Acta Cybernetica* **40** (1992) 155–185.

[10] Demetrovics J., Thi V.B., On Algorithms for Generating Armstrong Relations and Inferring Functional Dependencies in the Relational Datamodel, *Computers and Mathematics with Applications* **(26)** (4) (1993) 43–55.

[11] Demetrovics J., Thi V.B., Some Results about Nomal Forms for Functional Dependencies in the Relational Data Model, *Discrete Applied Mathematics* **69** (1996) 61–74.

[12] Demetrovics J., Thi V.B., Describing Candidate Keys by Hypergraphs, *Computer and Artificial Intelligence* **18** (2) (1999) 191–207.

[13] Demetrovics J., Thi V.B., Family of Functional Dependencies and it is Equivalent Descriptions, *Computer and Math with Application* **29** (4) (1995) 101–109.

[14] Libkin L., Direct Product Decomposition of Lattices, Closures and Relation Schemes, *Discrete Mathematics* **112** (1993) 119–138.

[15] Mannila H., Raiha K. J., On the Comlpexity of Inffering Functional Dependencies, *Discrete Applied Mathematics* **40** (1992) 237–243.

[16] Nghia Vu, Some Result about Choice Functions, *Journal of Computer Science and Cybernetics*, **17** (1) (2001) 35–39.

[17] Ramakrishnan R., Gehrke J., *Database Management Systems*, The McGraw- Hill, 2000.

[18] Schewe K.D., Thalheim B., Fundamental Concepts of Object Oriented Databases, *Acta Cybernetica* **11** (1&2) (1993) 49–83.

[19] Thayse A., *Modal Logic to Deductive Database*, John Wiley & Son, 1989.

[20] Ullman J., *Principles of Database and Knowledge Base Systems*, Computer Science Press, 1988.

Received February 1 - 2002.

Vu Duc Thi - Institute of Information Technology
Bina Ramamurthy Vu Nghia - State University of New York at Buffalo