DEWS 2004 5-C-02

不完全情報を表現可能な bag-based データモデルの閉包性

山口　明也†　　石原　靖哲†　　藤原　融†

† 大阪大学大学院情報科学研究科
〒 560–8531 豊中市待兼山町 1–3
E-mail: †{a-yamaguti,ishihara,fujiwara}@ist.osaka-u.ac.jp

あらまし　不完全情報とは, 一部が曖昧であったり未確定であるような情報のことである. 実体化ビューを問い合わせの評価に再利用する場合や，秘密情報が漏洩しないことを保証したアクセス制御を行う場合に, 不完全情報を自然に表現できるデータモデルが有用であることが知られている. 筆者らは, 不完全情報を表現可能な, 多重集合 (bag) に基づいたデータモデル (EC-table と呼ぶ) を提案している. 本稿では, まず EC-table について述べ, その部分モデルを導入する. そして, これらのデータモデルに対して, 二つの意味定義 (OWA, CWA) を与え，それぞれのもとでの代数演算とその逆演算の閉包性を示す.

キーワード　不完全情報, 代数演算, 閉包性, 多重集合

Closure Properties of a Bag-based Data Model for Incomplete Information

Akinari YAMAGUCHI†, Yasunori ISHIHARA†, and Toru FUJIWARA†

† Graduate School of Information Science and Technology, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka, 560–8531 Japan
E-mail: †{a-yamaguti,ishihara,fujiwara}@ist.osaka-u.ac.jp

Abstract　Incomplete information is a kind of information including uncertain statements. Data models which can represent incomplete information are known to be useful for reusing materialized views to evaluate queries and for access control guaranteeing no secret information disclosed. We have proposed a bag-based data model (called EC-tables) which can represent incomplete information. In this paper, EC-tables are reviewed first. Then, a submodel of EC-tables, called restricted EC-tables, is introduced. Next, under two semantics called Closed World Assumption (CWA) and Open World Assumption (OWA) for these data models, the closure properties of algebraic operations and their inverses on each model are shown.

Key words　incomplete information, algebraic operation, closure property, bag

1. Introduction

Incomplete information is partial and ambiguous information such as “We know that the course Databases is given at room A01, but we *do not know* who teaches it” and “The room of the course Programming given by Ishihara is B02 *or* C03.” Information which we obtain in the real world is often incomplete. To handle the incompleteness strictly, we need data models which can represent incomplete information. Moreover, in some of the recent works on answering queries using materialized views, data models for incomplete information are demonstrated to be useful [1], [4]. Such data models are also useful for access control guaranteeing no secret information disclosed, as shown in the example below.

［Example 1］ Consider the idea of *Disclosure Monitor* proposed in [3]. When a database user issues a query , Disclosure Monitor maintains (a) the knowledge that the user has obtained from the database so far and (b) the knowledge that the user will obtain from the answer of the query. If some secret information can be derived from the knowledge (a) and (b), then Disclosure Monitor refuses to answer the query.

Data models for incomplete information are useful for representing the user's knowledge because a query and its answer can be regarded as incomplete information on the database. Therefore, if such incomplete information can be naturally treated, the idea of Disclosure Monitor will have more impact. □

Ordinary relational databases cannot represent incomplete information naturally. Therefore, some modifications of relational databases have been proposed. For example, in a *Codd table*, unknown values are represented by one special symbol, say *null*. In a *V-table*, unknown values are represented by variables. A *C-table* [5] is an extended V-table so that each tuple has a condition for the tuple to exist. Moreover, in [5], the complexity of operations on those models is investigated.

[Example 2] Tables 1 and 2 are examples of a V-table and a C-table, respectively. Table 1 shows that the course Databases is given at room A01, but the teacher is unknown. Table 2 contains a new attribute *con* representing the condition for the associated tuple to exist. For example, the tuple (Programming, Ishihara, y) exists if y is B02 or C03. □

The underlying data model of Codd tables, V-tables and C-tables is relational. However, most of the practical query languages are based on bags, i.e., the answer of a query may contain duplicate tuples. Therefore, in the situation of Example 1, relation-based data models for incomplete information cannot represent the user's knowledge precisely. Thus, a bag-based data model for representing incomplete information is desirable.

In [9], we have proposed a bag-based data model called *EC-tables*. EC-tables are an extension of C-tables and its underlying data model is based on bags.

[Example 3] Table 3 is an example of an EC-table. In this table, the number of tuples is given at the right of $\mapsto$. The first tuple says that some teacher x gives z courses at room A01. The second tuple says that the room where Ishihara gives two courses is B02 or C03. □

In this paper, we introduce a submodel of EC-tables in which no variables can be used for indicating the number of tuples. Hereafter the submodel is referred to as *restricted EC-tables*. We also provide two semantics of these models. One is called Closed World Assumption (CWA), which means that "invisible tuples do not exist." For example, under CWA, Table 3 says that any teacher other than Ishihara uses only room A01. The other semantics is called Open World Assumption (OWA), which means that "the existence of invisible tuples is open." For example, under OWA, Table 3

Table 1 V-table

Course	Teacher	Room
Databases	x	A01
Programming	Ishihara	y
Network	Ishihara	y

Table 2 C-table

Course	Teacher	Room	*con*
Databases	x	A01	true
Programming	Ishihara	y	$(y = \text{B02}) \vee (y = \text{C03})$
Network	Ishihara	y	$(y = \text{B02}) \vee (y = \text{C03})$

Table 3 EC-table

Teacher	Room	*con*		
x	A01	true	$\mapsto$	z
Ishihara	y	$(y = \text{B02}) \vee (y = \text{C03})$	$\mapsto$	2

does not say anything about the possibility that a teacher other than Ishihara uses a room other than A01. Then, in this paper, the closure properties of algebraic operations and their inverses on each model under these semantics are investigated. The closure properties of forward operations are important since the answer of a query to incomplete information should be represented under the same data model. The closure properties of inverse operations are also important when the incomplete information on a database is derived from a query to the database and its answer. For example, consider the situation in Example 1. Suppose that the user has issued a query q to the database D and obtained the answer A. Then, Disclosure Monitor computes the user's knowledge (a), i.e., the incomplete information on D determined by q and A. Roughly speaking, this computation is accomplished by applying the inverse of q to A. Therefore, in order for the incomplete information on D to be represented under a data model, the inverse of every operation in q is desirable to be closed on the same data model.

Table 4 shows our results. Y means that the operation is closed, while N means that the operation is not closed. "?" means that the closure property of the operation is unknown, although we conjecture that this is closed. From the table, the inverse of projection is not closed on EC-tables under either CWA or OWA. However, on restricted EC-tables, the inverse of projection is closed under both CWA and OWA.

There are some related works on bag-based data model for incomplete information. In [7], a partial order representing a degree of incompleteness between bags is introduced. Then, it is shown that the order is not expressible in a standard bag language called BQL [8]. In the model of [7], partial information such as "A's value is unknown" can be captured, but ambiguous information such as "A's value is B or C" cannot be represented naturally. On the other hand, in our model, ambiguous information can be naturally represented.

Table 4 Closure properties

(a) EC-tables, CWA

	Selection	Projection	Union	Product	Difference
Forward	Y*	Y*	Y*	Y*	Y
Inverse	N	N	Y	N	N

(b) EC-tables, OWA

	Selection	Projection	Union	Product	Difference
Forward	N	Y*	Y*	Y*	Y*
Inverse	Y*	N	Y	?	N

(c) Restricted EC-tables, CWA

	Selection	Projection	Union	Product	Difference
Forward	Y	Y	Y	Y	Y
Inverse	N	Y	Y	N	N

(d) Restricted EC-tables, OWA

	Selection	Projection	Union	Product	Difference
Forward	N	Y	Y	Y	Y
Inverse	Y	Y	Y	?	N

* means that it is a known result in [9].

In [6], aggregate queries on C-tables are defined. Then, it is shown that the aggregate queries on C-tables are closed. In that model, aggregate values are represented by the values of a special attribute. On the other hand, our model is purely a bag-based model.

The rest of this paper is organized as follows. Section 2 provides the definitions of bag-based databases and EC-tables. In Section 3 we prove the closure properties on EC-tables under both CWA and OWA. In Section 4 we prove the closure properties on restricted EC-tables under both CWA and OWA. Some of the proofs are omitted because of the space limitation. Lastly, in Section 5, we provide the summery and future work.

2. Definitions

2.1 Bag-based databases

In this section, we extend the definition of ordinary relational databases in [2] to bag-based databases. Although *bag*-based databases are not *relational*, we borrow the terminology of relational databases.

[Definition 1] A *relational schema* R is a set of attributes. For simplicity, we assume that the domain of every attribute in R is the set N of non-negative integers. A *tuple* t over R is a function from R to N. Let $t(A)$ denote the value of $A \in R$ in t. A *relational instance* D over R is a function from the set of tuples over R to N such that $\{t \mid D(t) \neq 0\}$ is a finite set. A *database schema* $\mathbf{R}$ is a finite sequence $\langle R_1, \cdots, R_n\rangle$ of relational schemas. A *database instance* over a database schema $\mathbf{R} = \langle R_1, \cdots, R_n\rangle$ is a sequence $\langle D_1, \cdots, D_n\rangle$, where each D_i is a relational instance of R_i. □

We define some notations. For a tuple t over R and $X \subseteqq R$, let $t[X]$ denote the function obtained by restricting the domain of t to X. Let $dom(D) = \{t \mid D(t) \neq 0\}$. If $D_1(t) \leqq D_2(t)$ for an arbitrary tuple t, then we write $D_1 \subseteqq D_2$.

Table 5 A relational instance D in Example 4

Birthday	Name	
June 10	Sato	$\mapsto 3$
May 5	Tanaka	$\mapsto 2$

[Example 4] Suppose that there are three people whose surnames and the dates of the birth are the same. The surname is Sato and the date is June 10. Moreover, assume that there are two other people whose surnames and the dates of the birth are the same. The surname is Tanaka and the date is May 5. Then, the relational instance D representing these facts is as follows (see Table 5):

$$D(t) = \begin{cases} 3 & \text{if } t = (\text{June } 10, \text{Sato}), \\ 2 & \text{if } t = (\text{May } 5, \text{Tanaka}), \\ 0 & \text{otherwise.} \end{cases}$$

In Table 5, the numbers of tuples are shown to the right of $\mapsto$. By definition, $dom(D) = \{(\text{June } 10, \text{Sato}), (\text{May } 5, \text{Tanaka})\}$. □

[Definition 2] We define *selection*, *projection*, *union*, *product* and *difference* as follows.

Selection $\sigma_C(D)$: Let D be a relational instance over a relational schema R and C be a condition. For every tuple t over R, we define $\sigma_C(D)(t)$ as follows:

$$\sigma_C(D)(t) = \begin{cases} D(t) & \text{if } t \text{ satisfies } C, \\ 0 & \text{otherwise.} \end{cases}$$

Projection $\pi_X(D)$: Let D be a relational instance over a relational schema R and let $X \subseteqq R$. For every tuple t over X, we define $\pi_X(D)(t)$ as follows:

$$\pi_X(D)(t) = \sum_{t' : t' = t[X]} D(t').$$

Union $D_1 \cup D_2$: Let D_1 and D_2 be relational instances over a relational schema R. For every tuple t over R, we define $(D_1 \cup D_2)(t)$ as follows:

$$(D_1 \cup D_2)(t) = D_1(t) + D_2(t).$$

Product $D_1 \times D_2$: Let D_1 and D_2 be relational instances over relational schemas R_1 and R_2, respectively, such that $R_1 \cap R_2 = \emptyset$. For every t_1 over R_1 and every t_2 over R_2, we define $(D_1 \times D_2)(t_1 t_2)$ as follows:

$$(D_1 \times D_2)(t_1 t_2) = D_1(t_1) \times D_2(t_2),$$

where $t_1 t_2$ denotes the tuple over $R_1 \cup R_2$ such that $t_1 t_2[R_1] = t_1$ and $t_1 t_2[R_2] = t_2$, and $\times$ in the right-hand side denotes the arithmetic multiplication.

Difference $D_1 - D_2$: Let D_1 and D_2 be relational instances over a relational schema R. For every tuple t, we define $(D_1 - D_2)(t)$ as follows:

$$(D_1 - D_2)(t) = \max(D_1(t) - D_2(t), 0),$$

where $-$ in the right-hand side denotes the arithmetic subtraction. □

2.2 EC-tables

2.2.1 Syntax

Let V be a set of *variables*. Let $\dot{-}$ denote the difference operation on non-negative integers, i.e.,

$$a \dot{-} b = \begin{cases} a - b & \text{if } a \geqq b, \\ 0 & \text{otherwise.} \end{cases}$$

A *non-negative integer expression* is an expression consisting of non-negative integers, variables, and operators $+$, $\times$ and $\dot{-}$. An *atomic conditional expression* is an expression in the form of $p = q$, where p and q are non-negative integer expressions. A *conditional expression* is an expression consisting of atomic conditional expressions and Boolean connectives $\neg$, $\wedge$ and $\vee$. A *C-tuple* u over R is a function defined over $R \cup \{con\}$ such that $u[R]$ is a total function from R to $N \cup V$ and $u(con)$ is a conditional expression. An *EC-table* E over R is a total function from the set of C-tuples over R to the set of non-negative integer expressions such that $\{u \mid E(u) \neq 0\}$ is finite. Let $dom(E) = \{u \mid E(u) \neq 0\}$. If the range of an EC-table E is N, then E is called a *restricted EC-table*.

2.2.2 Semantics

A *valuation* is a function from V to N. The domain of a valuation ν is extended as follows:

- For each constant $a \in N$, let $\nu(a) = a$.
- For non-negative integer expressions $x + y$, $x \times y$ and $x \dot{-} y$, let $\nu(x+y) = \nu(x) + \nu(y)$, $\nu(x \times y) = \nu(x) \times \nu(y)$ and $\nu(x \dot{-} y) = \nu(x) \dot{-} \nu(y)$.
- For an atomic conditional expression $l = m$, let $\nu(l = m) = (\nu(l) = \nu(m))$.
- For conditional expressions $\neg c$, $c \wedge d$ and $c \vee d$, let $\nu(\neg c) = \neg\nu(c)$, $\nu(c \wedge d) = \nu(c) \wedge \nu(d)$ and $\nu(c \vee d) = \nu(c) \vee \nu(d)$.
- For a C-tuple u over X, let $\nu(u)$ be a tuple over X satisfying that for each $A \in X$, $(\nu(u))(A) = \nu(u(A))$.
- For an EC-table E and a tuple t,

$$\nu(E)(t) = \sum_{u:\nu(u)=t, \nu(u(con))=true} \nu(E(u)).$$

We provide two semantics of EC-tables. The first semantics is *Closed World Assumption* (CWA), which means that "invisible tuples do not exist." The other is *Open World Assumption* (OWA), which means that "the existence of invisible tuples is unknown."

[Definition 3] The set $rep_C(\langle E_1, \cdots, E_n \rangle)$ of database instances represented by $\langle E_1, \cdots, E_n \rangle$ under *CWA* is defined as follows:

$$\begin{aligned} & rep_C(\langle E_1, \cdots, E_n \rangle) \\ & \quad = \{\langle D_1, \cdots, D_n \rangle \mid D_1 = \nu(E_1), \cdots, D_n = \nu(E_n) \\ & \qquad\qquad \text{for some valuation } \nu\}. \end{aligned}$$

The set $rep_O(\langle E_1, \cdots, E_n \rangle)$ of database instances represented by $\langle E_1, \cdots, E_n \rangle$ under *OWA* is defined as follows:

$$\begin{aligned} & rep_O(\langle E_1, \cdots, E_n \rangle) \\ & \quad = \{\langle D_1, \cdots, D_n \rangle \mid D_1 \supseteqq \nu(E_1), \cdots, D_n \supseteqq \nu(E_n) \\ & \qquad\qquad \text{for some valuation } \nu\}. \end{aligned}$$

If $n = 1$, then we write $rep_C(E_1)$ and $rep_O(E_1)$ instead of $rep_C(\langle E_1 \rangle)$ and $rep_O(\langle E_1 \rangle)$, respectively. □

[Definition 4] Let q be an operation on bag-based databases with n inputs and m outputs. The operation q is *closed* on EC-tables under CWA if for any sequence $\langle E_1, \cdots, E_n \rangle$ of EC-tables, there is a sequence $\langle E'_1, \cdots, E'_m \rangle$ such that

$$\begin{aligned} & rep_C(\langle E'_1, \cdots, E'_m \rangle) = \{q(\langle D_1, \cdots, D_n \rangle) \mid \\ & \qquad\qquad \langle D_1, \cdots, D_n \rangle \in rep_C(\langle E_1, \cdots, E_n \rangle)\}. \end{aligned}$$

The inverse of operation q is *closed* on EC-tables under CWA if for any $\langle E'_1, \cdots, E'_m \rangle$, there is $\langle E_1, \cdots, E_n \rangle$ such that

$$\begin{aligned} & rep_C(\langle E_1, \cdots, E_n \rangle) = \{\langle D_1, \cdots, D_n \rangle \mid \\ & \qquad\qquad q(\langle D_1, \cdots, D_n \rangle) \in rep_C(\langle E'_1, \cdots, E'_m \rangle)\}. \end{aligned}$$

□

Closure properties on EC-tables under OWA and on restricted EC-tables under CWA and OWA are defined in the same way.

3. Closure properties on EC-tables

3.1 Under CWA

In this section, we show that the inverse of projection is not closed (Theorem 1). Then, we show that both the inverse of union and difference are closed (Theorems 2 and 3).

[Theorem 1] The inverse of projection is not closed on EC-tables under CWA.

Proof : We assume that the inverse of projection is closed on EC-tables under CWA, and derive a contradiction. Consider an EC-table E' (Table 6 (a)) over a relational schema $\{A\}$ with $dom(E') = \{u\}$, $u(con) = true$ and $u(A) = E'(u) = x$, where x is a variable. From the assumption that the inverse of projection is closed, there is an EC-table E over relational schema $\{A, B\}$ such that

$$rep_C(E) = \{D \mid \pi_A(D) \in rep_C(E')\}.$$

Let m be the number of tuples in $dom(E)$, and let l be an integer such that $l > m$. Consider an instance D shown in

Table 6 E', D and $\pi_A(D)$ in Theorem 1

(a) E'

A	*con*	
x	*true*	$\mapsto x$

(b) D

A	B	
l	1	$\mapsto 1$
l	2	$\mapsto 1$
$\vdots$	$\vdots$	$\vdots$
l	l	$\mapsto 1$

(c) $\pi_A(D)$

A	
l	$\mapsto l$

Table 6 (b). Since $\pi_A(D)$ is an instance shown in Table 6 (c), we have $\pi_A(D) \in rep_C(E')$. Hence, we have $D \in rep_C(E)$. By the definition of CWA,

$$rep_C(E) = \{D \mid D = \nu(E) \text{ for some valuation } \nu\}.$$

Hence, there must be a valuation ν such that $D = \nu(E)$. Since m is the number of tuples in $dom(E)$, the number of tuples in $dom(D)$ is at most m. However, from Table 6 (b), the number of tuples in $dom(D)$ is l $(> m)$. This is a contradiction. □

Now, we prove that the inverse of union is closed.

[Definition 5] Let E be an EC-table over a relational schema R. We define $\cup^{-1}(E)$ as a pair $\langle E_1, E_2\rangle$ satisfying the following conditions, where E_1 and E_2 are EC-tables over R. For every u in $dom(E)$, introduce new variables x_u and y_u not appearing in $dom(E)$. Let Φ denote the following conditional expression:

$$\Phi = \bigwedge_{u \in dom(E)} (x_u + y_u = E(u)).$$

For each u in $dom(E)$, let $\hat{u}$ and $\tilde{u}$ be C-tuples over R such that

- $\hat{u}[R] = u[R]$, $\hat{u}(con) = u(con) \wedge \Phi$,
- $\tilde{u}[R] = u[R]$, $\tilde{u}(con) = u(con) \wedge (\neg\Phi)$.

Now, E_1 and E_2 are defined as follows:

$$E_1(u') = \begin{cases} x_u & \text{if } u'{=}\hat{u} \text{ for some } u \in dom(E), \\ 0 & \text{otherwise,} \end{cases}$$

$$E_2(u') = \begin{cases} y_u & \text{if } u'{=}\hat{u} \text{ for some } u \in dom(E), \\ E(u) & \text{if } u'{=}\tilde{u} \text{ for some } u \in dom(E), \\ 0 & \text{otherwise.} \end{cases}$$

□

Hereafter, we prove that $D_1 \cup D_2 \in rep_C(E)$ if and only if $\langle D_1, D_2\rangle \in rep_C(\cup^{-1}(E))$.

[Lemma 1] $D_1 \cup D_2 \in rep_C(E)$ if $\langle D_1, D_2\rangle \in rep_C(\cup^{-1}(E))$.

Proof : Let E be an EC-table over a relational schema R. Let $\langle E_1, E_2\rangle$ be $\cup^{-1}(E)$. Consider instances D_1 and D_2 such that $\langle D_1, D_2\rangle \in rep_C(\langle E_1, E_2\rangle)$. There must be a valuation ν' such that $\nu'(E_1) = D_1$ and $\nu'(E_2) = D_2$. Let $u \in dom(E)$. If $\nu'(\Phi)$ is true, then

$$\nu'(E_1(\hat{u})) + \nu'(E_2(\hat{u})) + \nu'(E_2(\tilde{u})) \\ = \nu'(x_u) + \nu'(y_u) + 0 = \nu'(E(u)).$$

Otherwise,

$$\nu'(E_1(\hat{u})) + \nu'(E_2(\hat{u})) + \nu'(E_2(\tilde{u})) \\ = 0 + 0 + \nu'(E(u)) = \nu'(E(u)).$$

Hence, we have $D_1 \cup D_2 = \nu'(E) \in rep_C(E)$. □

[Lemma 2] $D_1 \cup D_2 \in rep_C(E)$ only if $\langle D_1, D_2\rangle \in rep_C(\cup^{-1}(E))$.

Proof : Let E be an EC-table over a relational schema R, and D_1 and D_2 be arbitrary instances such that $D_1 \cup D_2 \in rep_C(E)$. Then, there is a valuation ν such that $\nu(E) = D_1 \cup D_2$. Let $\langle E_1, E_2\rangle$ be $\cup^{-1}(E)$. In what follows, we construct a valuation ν' such that $D_1 = \nu'(E_1)$ and $D_2 = \nu'(E_2)$.

ν' is defined on the variables appearing in $dom(E_1)$ or $dom(E_2)$ and the new variables x_u and y_u. For each variable z appearing in $dom(E_1)$ or $dom(E_2)$, let $\nu'(z) = \nu(z)$. For the new variables x_u and y_u, ν' satisfies the following equations:

(1) For each tuple $t \in dom(D_1)$,

$$\sum_{u:(u \in dom(E)) \wedge (\nu(u)=t) \wedge (\nu(u(con))=true))} \nu'(x_u) = D_1(t).$$

(2) For each tuple $t \in dom(D_2)$,

$$\sum_{u:(u \in dom(E)) \wedge (\nu(u)=t) \wedge (\nu(u(con))=true))} \nu'(y_u) = D_2(t).$$

(3) For each C-tuple $u \in dom(E)$,

$$\nu'(x_u) + \nu'(y_u) = \nu(E(u)).$$

Before proving the existence of such ν', we show that such ν' also satisfies $D_1 = \nu'(E_1)$ and $D_2 = \nu'(E_2)$. If ν' satisfies the equations of type (3), then $\nu'(\Phi) = true$. Therefore, $\nu'(\hat{u}(con)) = \nu(u(con))$ and $\nu'(\tilde{u}(con)) = false$ for each $u \in dom(E)$. Thus, the equation of type (1) becomes

$$D_1(t) = \sum_{\hat{u}:(\hat{u} \in dom(E_1)) \wedge (\nu'(\hat{u})=t) \wedge (\nu'(\hat{u}(con))=true)} \nu'(E_1(\hat{u})),$$

and therefore, $D_1 = \nu'(E_1)$. Similarly, we have $D_2 = \nu'(E_2)$.

Now we show the existence of ν'. First, consider a C-tuple $u \in dom(E)$ such that the equation of type (1) involving x_u is $\nu'(x_u) = D_1(t)$ (i.e., the equation contains only one variable x_u). Then, the equation of type (2) involving y_u must be $\nu'(y_u) = D_2(t)$. Hence, $\nu'(x_u) + \nu'(y_u) = D_1(t) + D_2(t) = (D_1 \cup D_2)(t) = \nu(E(u))$. Next, consider C-tuples $u_1, \cdots, u_n \in dom(E)$ such that the equation of type (1) involving x_{u_1} is $\nu'(x_{u_1}) + \cdots + \nu'(x_{u_n}) = D_1(t)$. Then, the equation of type (2) involving y_{u_1} must be $\nu'(y_{u_1}) + \cdots + \nu'(y_{u_n}) = D_2(t)$. Also, $\nu(E(u_1)) + \cdots + \nu(E(u_n)) =$

$(D_1 \cup D_2)(t)$ since $D_1 \cup D_2 = \nu(E)$. We can choose the value of $\nu'(x_{u_i})$ $(1 \leqq i \leqq n)$ so that $0 \leqq \nu'(x_{u_i}) \leqq \nu(E(u_i))$ and $\sum_i \nu'(x_{u_i}) = D_1(t)$. Let $\nu'(y_{u_i}) = \nu(E(u_i)) - \nu'(x_{u_i})$. These $\nu'(x_{u_i})$'s and $\nu'(y_{u_i})$'s satisfy all the equations. □

From Lemmas 1 and 2, we obtain the following theorem.

[Theorem 2] The inverse of union is closed on EC-tables under CWA. □

Next, we prove that the difference is closed.

[Definition 6] Let E_1 and E_2 be EC-tables over a relational schema R. For each pair of $F_1 \subseteqq dom(E_1)$ and $F_2 \subseteqq dom(E_2)$, let S and T be $dom(E_1) \cup dom(E_2)$ and $F_1 \cup F_2$, respectively, and $u_{F_1F_2}$ be a C-tuple satisfying the following conditions:

- $u_{F_1F_2}[R] = u[R]$ for some $u \in F_1 \cup F_2$;
- $u_{F_1F_2}(con)$ is the conjunction of the following conditional expressions:
  - $\bigwedge_{u_1 \in F_1} u_1(con)$,
  - $\bigwedge_{u_2 \in F_2} u_2(con)$,
  - $\displaystyle\bigwedge_{u' \in T} \bigwedge_{A \in R} (u_{F_1F_2}(A) = u'(A))$,
  - $\displaystyle\bigwedge_{u' \in S-T} \bigvee_{A \in R} \neg(u_{F_1F_2}(A) = u'(A))$.

Now, for each C-tuple u over R, $(E_1 - E_2)(u)$ is defined as follows:

$$
\begin{aligned}
&(E_1 - E_2)(u) \\
&= \begin{cases} \sum_{u_1 \in F_1} F_1(u_1) \dot{-} \sum_{u_2 \in F_2} F_2(u_2) & \\ \qquad \text{if } u = u_{F_1F_2} \text{ for some } F_1 \subseteqq dom(E_1) & \\ \qquad \text{and } F_2 \subseteqq dom(E_2), & \\ 0 \qquad \text{otherwise.} & \end{cases}
\end{aligned}
$$

□

Hereafter, we prove $rep_C(E_1 - E_2) = \{D_1 - D_2 \mid \langle D_1, D_2 \rangle \in rep_C(\langle E_1, E_2 \rangle)\}$.

[Lemma 3] $rep_C(E_1 - E_2) \supseteqq \{D_1 - D_2 \mid \langle D_1, D_2 \rangle \in rep_C(\langle E_1, E_2 \rangle)\}$.

Proof : Let E_1 and E_2 be EC-tables over a relational schema R. Consider an arbitrary pair $\langle D_1, D_2 \rangle \in rep_C(\langle E_1, E_2 \rangle)$. Then, there must be ν such that $\nu(E_1) = D_1, \nu(E_2) = D_2$. Hereafter, we prove $\nu(E_1 - E_2) = D_1 - D_2$.

- $\nu(E_1 - E_2) \supseteqq D_1 - D_2$

Consider an arbitrary tuple t such that $t \in D_1 - D_2$. Let F_1 and F_2 be $\{u_1 \mid u_1 \in dom(E_1), \nu(u_1(con)) = true, \nu(u_1[R]) = t[R]\}$ and $\{u_2 \mid u_2 \in dom(E_2), \nu(u_2(con)) = true, \nu(u_2[R]) = t[R]\}$, respectively. We prove that $\nu(u_{F_1F_2}(con))$ is *true*. $\nu(u_{F_1F_2}(con))$ is equal to the conjunction of the following four expressions:

(1) $\bigwedge_{u_1 \in F_1} \nu(u_1(con))$,

(2) $\bigwedge_{u_2 \in F_2} \nu(u_2(con))$,

(3) $\displaystyle\bigwedge_{u' \in T} \bigwedge_{A \in R} (\nu(u_{F_1F_2}(A)) = \nu(u'(A)))$,

(4) $\displaystyle\bigwedge_{u' \in S-T} \bigvee_{A \in R} \neg(\nu(u_{F_1F_2}(A)) = \nu(u'(A)))$,

(1) is *true* because $\nu(u_1(con))$ is *true* in F_1. (2) is also *true* by the same reason. By the definition of F_1 and F_2, for each C-tuple $u' \in T$, we have $\nu(u'[R]) = t[R]$. From Definition 6, since $u_{F_1F_2}[R]$ is equal to some C-tuple in T, for each C-tuple $u' \in T$ we have $\nu(u'[R]) = \nu(u_{F_1F_2}[R])$. Hence, (3) is *true*. By the definition of F_1 and F_2, for each C-tuple $u' \in S - T$, we have $\nu(u_{F_1F_2}[R]) \not= \nu(u'[R])$. Hence, (4) is *true*. Therefore, $\nu(u_{F_1F_2}(con))$ is *true*. The number of tuples which are equal to $\nu(u_{F_1F_2}[R])$ is $\sum_{u_1 \in F_1} \nu(E_1(u_1))$ in D_1, and is $\sum_{u_2 \in F_2} \nu(E_2(u_2))$ in D_2. Since

$$
\begin{aligned}
&\nu((E_1 - E_2)(u_{F_1F_2})) \\
&= \sum_{u_1 \in F_1} \nu(E_1(u_1)) \dot{-} \sum_{u_2 \in F_2} \nu(E_2(u_2)) \\
&= (D_1 - D_2)(t),
\end{aligned}
$$

we obtain $\nu(E_1 - E_2) \supseteqq D_1 - D_2$.

- $\nu(E_1 - E_2) \subseteqq D_1 - D_2$

Consider an arbitrary C-tuple $u_{F_1F_2} \in dom(E_1 - E_2)$ such that $\nu(u_{F_1F_2}(con))$ is *true*. Then, by the definition of $u_{F_1F_2}(con)$, each of the four expressions (1)–(4) above is *true*. From (1) and (3), if $u_1 \in F_1$ then $\nu(u_1[R]) = \nu(u_{F_1F_2}[R])$. From (4), if $u_1 \notin F_1$ then $\nu(u_1[R]) \not= \nu(u_{F_1F_2}[R])$. Therefore, we have $F_1 = \{u_1 \mid u_1 \in dom(E_1), u_1(con) = true, \nu(u_1[R]) = \nu(u_{F_1F_2}[R])\}$. In the same way, $F_2 = \{u_2 \mid u_2 \in dom(E_2), u_2(con) = true, \nu(u_2[R]) = \nu(u_{F_1F_2}[R])\}$ is derived. Hence, the number of tuples which are equal to $\nu(u_{F_1F_2}[R])$ is $\sum_{u_1 \in F_1} \nu(E_1(u_1))$ in D_1, and is $\sum_{u_2 \in F_2} \nu(E_2(u_2))$ in D_2. Since

$$
\begin{aligned}
&\nu((E_1 - E_2)(u_{F_1F_2})) \\
&= \sum_{u_1 \in F_1} \nu(E_1(u_1)) \dot{-} \sum_{u_2 \in F_2} \nu(E_2(u_2)),
\end{aligned}
$$

we have $\nu(E_1 - E_2) \subseteqq D_1 - D_2$. □

[Lemma 4] $rep_C(E_1 - E_2) \subseteqq \{D_1 - D_2 \mid \langle D_1, D_2 \rangle \in rep_C(\langle E_1, E_2 \rangle)\}$.

Proof : Let E_1 and E_2 be EC-tables over a relational schema R. Consider an arbitrary instance D' such that $D' \in rep_C(E_1 - E_2)$. Then, there must be ν' such that $D' = \nu'(E_1 - E_2)$. Hereafter, we prove $D' = \nu'(E_1) - \nu'(E_2)$.

- $D' \supseteqq \nu'(E_1) - \nu'(E_2)$

Consider an arbitrary tuple t such that $t \in dom(\nu'(E_1) - \nu'(E_2))$. Let F_1 and F_2 be $\{u_1 \mid u_1 \in dom(E_1), \nu'(u_1(con)) = true, \nu'(u_1[R]) = t[R]\}$ and $\{u_2 \mid u_2 \in dom(E_2), \nu'(u_2(con)) = true, \nu'(u_2[R]) = t[R]\}$, respectively. We prove $\nu'(u_{F_1F_2}(con))$ is *true*. $\nu'(u_{F_1F_2}(con))$ is the conjunction of the following four expressions:

(1) $\bigwedge_{u_1 \in F_1} \nu'(u_1(con))$,

(2) $\bigwedge_{u_2 \in F_2} \nu'(u_2(con))$,

(3) $\displaystyle\bigwedge_{u' \in T} \bigwedge_{A \in R} (\nu'(u_{F_1F_2}(A)) = \nu'(u'(A)))$,

(4) $\bigwedge_{u' \in S-T} \bigvee_{A \in R} \neg(\nu'(u_{F_1F_2}(A)) = \nu'(u'(A)))$.

(1), (2), (3) and (4) are *true* in the same way as Lemma 3. Hence, $\nu'(u_{F_1F_2}(con))$ is *true*. The number of tuples which are equal to $\nu'(u_{F_1F_2}[R])$ is $\sum_{u_1 \in F_1} \nu'(E_1(u_1))$ in $\nu'(E_1)$, and is $\sum_{u_2 \in F_2} \nu'(E_2(u_2))$ in $\nu'(E_2)$. Since

$$\begin{aligned}&\nu'((E_1 - E_2)(u_{F_1F_2}))\\ &= \sum_{u_1 \in F_1} \nu'(E_1(u_1)) \dot{-} \sum_{u_2 \in u_2} \nu'(E_2(u_2))\\ &= (\nu'(E_1) - \nu'(E_2))(t),\end{aligned}$$

we obtain $\nu'(E_1 - E_2) = D' \supseteqq \nu'(E_1) - \nu'(E_2)$.

- $D' \subseteqq \nu'(E_1) - \nu'(E_2)$

Consider an arbitrary C-tuple $u_{F_1F_2}$ such that $\nu'(u_{F_1F_2}(con))$ is *true* and $u_{F_1F_2} \in dom(E_1 - E_2)$. Then, by the definition of $u_{F_1F_2}(con)$, each of the four expressions (1)–(4) above is *true*. We have $F_1 = \{u_1 \mid u_1 \in dom(E_1), u_1(con) = true, \nu'(u_1[R]) = \nu'(u_{F_1F_2}[R])\}$ and $F_2 = \{u_2 \mid u_2 \in dom(E_2), u_2(con) = true, \nu'(u_2[R]) = \nu'(u_{F_1F_2}[R])\}$ in the same way as Lemma 3. The number of tuples which are equal to $\nu'(u_{F_1F_2}[R])$ is $\sum_{u_1 \in F_1} \nu'(E_1(u_1))$ in $\nu'(E_1)$, and is $\sum_{u_2 \in F_2} \nu'(E_2(u_2))$ in $\nu'(E_2)$. Since

$$\begin{aligned}&\nu'((E_1 - E_2)(u_{F_1F_2})))\\ &= \sum_{u_1 \in F_1} \nu'(E_1(u_1)) \dot{-} \sum_{u_2 \in F_2} \nu'(E_2(u_2)),\end{aligned}$$

we obtain $\nu'(E_1 - E_2) = D' \subseteqq \nu'(E_1) - \nu'(E_2)$. □

By Lemmas 3 and 4, we obtain the following theorem.

[Theorem 3] The difference is closed on EC-tables under CWA. □

3.2 Under OWA

In this section, we show that selection and inverse of projection are not closed (Theorems 4 and 5). Then, we show that the inverse of union is closed (Theorem 6).

[Theorem 4] Selection is not closed on EC-tables under OWA.

Proof : We assume that selection is closed on EC-tables under OWA. Then, for each EC-table E, there is an EC-table E' such that

$$rep_O(E') = \{\sigma_C(D) \mid D \in rep_O(E)\}. \quad (1)$$

On the other hand, by the definition of OWA, we have

$$rep_O(E') = \{D' \mid D' \supseteqq \nu(E') \text{ for some valuation } \nu\}.$$

Hence, for an arbitrary instance $D \in rep_O(E)$ and tuple t not satisfying the condition C, $(D \cup \{t\})$ must be in $rep_O(E')$. This contradicts equation (1). □

[Theorem 5] The inverse of projection is not closed on EC-tables under OWA.

Table 7 E', D, $\pi_A(D)$ and $\pi_A(\nu(E))$ in Theorem 5

(a) E'

A	*con*	
x	*true*	$\mapsto x$

(b) D

A	B	
l	1	$\mapsto 1$
l	2	$\mapsto 1$
$\vdots$	$\vdots$	$\vdots$
l	l	$\mapsto 1$

(c) $\pi_A(D)$

A	
l	$\mapsto l$

(d) $\pi_A(\nu(E))$

A	
l	$\mapsto m'$

Proof : We assume that the inverse of projection is closed on EC-tables under OWA, and derive a contradiction. Consider an EC-table E' (Table 7 (a)) over a relational schema $\{A\}$ with $dom(E') = \{u\}$, $u(con) = true$ and $u(A) = E'(u) = x$, where x is a variable. From the assumption that the inverse of projection is closed, there is an EC-table E over relational schema $\{A, B\}$ such that

$$rep_O(E) = \{D \mid \pi_A(D) \in rep_O(E')\}.$$

Let m be the number of tuples in $dom(E)$, and let l be an integer such that $l > m$. Consider an instance D shown in Table 7 (b). Since $\pi_A(D)$ is an instance shown in Table 7 (c), we have $\pi_A(D) \in rep_O(E')$. Hence, we have $D \in rep_O(E)$. By the definition of OWA,

$$rep_O(E) = \{D \mid D \supseteqq \nu(E) \text{ for some valuation } \nu\}.$$

Hence, there must be a valuation ν such that $D \supseteqq \nu(E)$. There are two cases:

(1) Suppose that $D = \nu(E)$. Since m is the number of tuples in $dom(E)$, the number of tuples in $dom(D)$ is at most m. However, from Table 7 (b), the number of tuples in $dom(D)$ is l ($> m$). This is a contradiction.

(2) Suppose that $D \supset \nu(E)$. Let m' be the number of tuples in $dom(\nu(E))$. Note that $m' < l$ and that $\pi_A(\nu(E))$ must be the instance shown in Table 7 (d). Since $\nu(E)$ is also in $rep_O(E)$, $\pi_A(\nu(E))$ must be in $rep_O(E')$. However, there is no valuation ν' such that $\pi_A(\nu(E)) \supseteqq \nu'(E')$ since $m' < l$. That is, $\pi_A(\nu(E)) \notin rep_O(E')$. This is a contradiction. □

Now, we prove the inverse of union is closed.

[Theorem 6] The inverse of union is closed on EC-tables under OWA.

Proof : We have already proved that

$$D_1 \cup D_2 = \nu(E) \text{ if and only if } D_1 = \nu'(E_1), D_2 = \nu'(E_2)$$

in Theorem 2. Therefore, we conclude that

$$D_1 \cup D_2 \supseteqq \nu(E) \text{ if and only if } D_1 \supseteqq \nu'(E_1), D_2 \supseteqq \nu'(E_2).$$

This implies that the theorem holds. □

4. Closure properties on restricted EC-tables

In this section, we prove the inverse of projection is closed on restricted EC-tables under CWA. Proofs under OWA are omitted because of the space limitation.

[Definition 7] Let X and Y be relational schemas such that $X \cap Y = \emptyset$. For each pair of $u \in dom(E)$ and $A \in Y$, introduce new variables $x^A_{u,1}, \cdots, x^A_{u,E(u)}$. Let $\hat{u}_i$ $(1 \leqq i \leqq E(u))$ denote the tuple such that

- $\hat{u}_i[X] = u$,
- $\hat{u}_i(A) = x^A_{u,i}$ for each $A \in Y$, and
- $\hat{u}_i(con) = u(con)$.

$\pi^{-1}_{X \cup Y}(E)(u')$ is a restricted EC-table over $X \cup Y$ such that

$$\pi^{-1}_{X \cup Y}(E)(u') = \begin{cases} 1 & \text{if } u' = \hat{u}_i \text{ for some } u \in dom(E) \\ & \text{and } i\ (1 \leqq i \leqq E(u)), \\ 0 & \text{otherwise.} \end{cases}$$ □

Hereafter, we prove $\pi_X(D) \in rep_C(E)$ if and only if $D \in rep_C(\pi^{-1}_{X \cup Y}(E))$.

[Lemma 5] $\pi_X(D) \in rep_C(E)$ only if $D \in rep_C(\pi^{-1}_{X \cup Y}(E))$.

Proof : Let E be a restricted EC-table over a relational schema X. Consider an arbitrary instance D over relational schema $X \cup Y$ satisfying $\pi_X(D) \in rep_C(E)$. Then, there must be ν such that $\nu(E) = \pi_X(D)$. Let E' be $\pi^{-1}_{X \cup Y}(E)$. We can construct a valuation ν' satisfying the following conditions:

(1) The domain of ν' is the set of variables in $dom(E')$.
(2) For each variable x in the domain of ν, $\nu'(x) = \nu(x)$.
(3) $\pi_Y(\nu'(E')) = \pi_Y(D)$.

The last condition holds because an arbitrary C-tuple u' in $dom(E')$ satisfies $E'(u') = 1$ and each new variable in $t'(Y)$ appears only once. Thus, we obtain $\nu'(E') = D$. Hence, $D \in rep_C(\pi^{-1}_{X \cup Y}(E))$ is derived.

[Lemma 6] $\pi_X(D) \in rep_C(E)$ if $D \in rep_C(\pi^{-1}_{X \cup Y}(E))$.

Proof : Let E be a restricted EC-table over a relational schema X. Let E' be $\pi^{-1}_{X \cup Y}(E)$. Consider an arbitrary instance D such that $D \in rep_C(E')$. There must be a valuation $\nu'(E') = D$. Then, let ν be a valuation obtained by restricting the domain of ν' to the set of variables appearing as attribute-values of $A \in X$. Hereafter, we prove that $\pi_X(\nu'(E')) = \nu(E)$.

- $\pi_X(\nu'(E')) \subseteqq \nu(E)$

For each C-tuple $\hat{u} \in dom(E')$, if $\nu'(\hat{u}(con))$ is *true* then $\nu(u(con))$ is also *true*. For each attribute $A \in X$, we have $\nu'(\hat{u}(A)) = \nu(u(A))$. Since $E(u)$ is equal to the number of C-tuples $\hat{u}$ we obtain $\pi_X(\nu'(E')) \subseteqq \nu(E)$.

- $\pi_X(\nu'(E')) \supseteqq \nu(E)$

For each C-tuple $u \in dom(E)$, consider C-tuples $\hat{u}_i$ $(1 \leqq i \leqq E(u))$ over $X \cup Y$. If $\nu(u(con))$ is *true* then $\nu'(\hat{u}(con))$ is also *true*. For each attribute $A \in X$, we have $\nu'(\hat{u}(A)) = \nu(u(A))$. Hence, since just $E(u)$ C-tuples u exists in $dom(E')$, we obtain $\pi_X(\nu'(E')) \supseteqq \nu(E)$.

Hence, $\nu(E) = \pi_X(D) \in rep_C(E)$. □

From Lemmas 5 and 6, we obtain the following theorem.

[Theorem 7] The inverse of projection is closed on restricted EC-tables under CWA. □

5. Conclusion and future work

In this paper, we have proved the closure properties on EC-tables under both CWA and OWA. Next, we have proposed the submodel of EC-tables, and proved the closure properties on the model under both CWA and OWA.

As a future work, we will prove the cases where the closure properties are open, although we conjecture that they are closed. Then, we will evaluate the computational complexity of the operations and the sizes of EC-table after operations. Also, we will define aggregate queries to EC-tables and restricted EC-tables.

Acknowledgment

The authors are grateful to Mr. Atsushi Sakamoto of Osaka University for his helpful comments. The authors would also like to thank the reviewers for their invaluable comments.

References

[1] S. Abiteboul and O. M. Duschka, "Complexity of answering queries using materialized views," *Proc. 17th ACM PODS*, pp. 254–263, 1998.
[2] S. Abiteboul, R. Hull, and V. Vianu, *Foundations of Databases*, Addison-Wesley, 1995.
[3] A. Brodsky, C. Farkas and S. Jajodia, "Secure Databases: Constraints, Inference Channels, and Monitoring Disclosures " *IEEE TKDE* 12, 6, pp. 900-919, 2000.
[4] A. Y. Halevy, "Answering queries using views: A survey," *VLDB Journal*, 10, pp. 270–294, 2001.
[5] T. Imielinski and W. Lipski, "Incomplete information in relational databases," *J. ACM*, 31, 4, pp. 761–791, 1984.
[6] J. Lechtenbörger, H. Shu, and G. Vossen, "Aggregate queries over conditional tables," *J. Intelligent Information Systems*, 19, 3, pp. 343–362, 2002.
[7] L. Libkin and L. Wong, "On representation and querying incomplete information in databases with bags," *Information Processing Letters*, 56, pp. 209–214, 1995.
[8] L. Libkin and L. Wong, "Query Languages for Bags and Aggregate Function," JCSS, 55, pp. 241-272, 1997.
[9] J. Tsuchida, Y. Ishihara and T. Fujiwara, "Extension of a Formulation of the Security against Inference Attacks on Relational Databases," *Proc. Symposium on Cryptography and Information Security*, pp. 567–572, 2002 (in Japanese).