MIT/LCS/TM-160

ON THE COMPUTATIONAL COMPLEXITY OF CARDINALITY CONSTRAINTS IN RELATIONAL DATABASES

Paris C. Kanellakis

March 1980

On the Computational Complexity of Cardinality Constraints in Relational Databases

Paris C. Kanellakis

December 1979

Abstract

We show that the problem of determining whether or not a lossless join property holds for a database , in the presence of key dependencies and cardinality constraints on the domains of the attributes is NP-complete.

keywords: computational complexity, NP-completeness, relational databases, cardinality constraints

This work was supported by Grant NSF/ENG-77-19971 and partly by Grant MCS-7701193

1. Introduction

The theory of relational databases has attracted a great deal of attention recently [1],[2]. In this model data is arranged into one or more multi-column tables called relations. Each column corresponds to an attribute. The values of the entries in a column are chosen from a set called the domain of the corresponding attribute. A relation is therefore a finite subset of the Cartesian product of the domains of its attributes.

New relations can be constructed from given ones using certain algebraic operations on relations. If R is a relation, then $a(R)$ is its set of attributes. If R is a relation, then $D(A)$ is the domain of the attribute A. Thus $R \subseteq \prod_{A \in a(R)} D(A)$. If $t \in R$ and $X \in a(R)$ then t_X is a tuple t restricted to the attributes in X. The projection on X of R is $\pi_X(R) = \{t_X \mid t \in R\}$.

The natural join is an "inverse" of the projection operator, defined as follows:
Let $R_1, \ldots, R_m$ be relations, then their natural join is defined as

$$\bowtie_{i=1}^{m} R_i = \{t \in \prod_{A \in \bigcup_i a(R_i)} D(A) \mid t_{a(R_i)} \in R_i\} \tag{1}$$

A relation R is usually represented by a relational schema $S = (X, D)$. X is a subset of $2^{a(R)}$ such that $\bigcup_{X_j \in X} X_j = a(R)$, and D is a set of dependencies (predicates on relations) that the relation satisfies. Thus R is represented by its projections $\{\pi_{X_i}(R) : X_i \in X\}$.

There are many kinds of dependencies [4]. Among these we have functional dependencies. (F.D.'s) ($X \rightarrow Y$ meaning that for all $t, t' \in R$ $t_X = t'_X$ implies $t_Y = t'_Y$). Functional dependencies of the form $X \rightarrow a(R)$ - where $X' \not\rightarrow a(R)$ for all $X' \subseteq$ X - are called key dependencies. Other kinds of dependencies are multivalued dependencies (MVD's), join dependencies (JD's), etc [4]. A different kind of dependency introduced in [4] is the domain (or cardinality) dependency (DD) stating that $\mid D(A) \mid = c_A$ for some $A \in a(R)$.

In the following section we will deal with tuples (rows) w_i with symbols as entries ranging over the domains of the attributes (as the a's and b's of [1]). If two w's have the same symbols in the set of columns X, and $X \rightarrow Y$ is an FD then applying $X \rightarrow Y$ on this pair of rows means we equate the symbols in the columns of Y in these rows.

We say that a set Δ of predicates on relations logically implies a predicate δ if all relations R satisfying Δ also satisfy δ. If $S = (X, D)$ is a relational schema, we let D^+ be the set of all predicates implied by D. The following important predicate (the lossless join property) is a desirable feature of any relational schema (X, D):

$$\bowtie_{i=1}^{m} \pi_{X_i}(R) = R \Leftrightarrow LJ(X) \tag{2}$$

If (2) is satisfied then X is an appropriate way for representing R without loss of information [1]. It is a central problem in relational database theory, given $S = (X, D)$ to determine whether $LJ(X) \in D^+$ [6]. This can be done, in the case that D consists of FD's, in polynomial time. The purpose of this paper is to show that introducing DD's (as proposed in [4]) makes the problem much harder. In fact we show that determining whether $LJ(X) \in D^+$ is NP-complete even if D consists of only key dependencies and just one DD.

2. Complexity of Cardinality Constraints

Consider the following problems of testing for the lossless join property in the presence of constraints.

LOSSLESS JOIN: Given $S = (X, D)$ is there a "Counterexample" relation R satisfying D and not $LJ(X)$

Theorem 1: The LOSSLESS JOIN problem is NP-complete, if D consists of key dependencies and just one cardinality constraint $| Dom(A) | = 2$.

Proof: In order to prove that the problem is NP-complete, we will proceed by proving membership in NP and then reducing a known NP-complete problem to it.

To argue that the problem is in NP, we note that if $X = \{X_1, ..., X_m\}$ and R is a counterexample relation, then there exists a tuple t^* in $\bowtie_{i=1}^{m} \pi_{X_i}(R)$, which is not in R. If R has attributes $A_1, A_2, ..., A_l$ then from the definition of the join it follows that:

$\bowtie_{i=1}^{m} \pi_{X_i}(R) = \{a_1 a_2 ... a_l \mid$ there exist tuples $w_1, w_2, ... w_m$ in R such that w_i has a_j in position j if the attribute A_j is in X_i and an arbitrary value in position j otherwise $\}$ (3)

Let $w_1^*, w_2^*, ... w_m^*$ be the tuples corresponding to t^*. A relation consisting of these tuples is a short (at most m tuples) counterexample relation, which satisfies D (keys and cardinalities) but not $LJ(X)$.

Reduction: We shall now reduce the exact hitting set problem (EHS) to our problem.

EHS: Given a family $\{V_i\}$ $i = 1, \ldots, n$ of subsets of a set $T = \{t_1, t_2, \ldots t_p\}$, where $V_i = \{t_{i_1}, t_{i_2}, t_{i_3}\}$, is there "an exact hitting set", that is a set $W \subseteq T$ such that $| W \bigcap V_i | = 1, 1 \preceq i \preceq n$?

The EHS problem can be easily seen to be NP-complete (reduction from exact cover with three occurrences of each element see [5]). It was also used in [2].

Given an arbitrary instance of EHS consider the relation schema $S = (X, D)$ and the relation R it respresents with attributes $a(R)$:

$$a(R) = \{A, X, Y, V_i, V'_i, T^i_{12}, T^i_{21}, T^i_{13}, T^i_{31}, T^i_{23}, T^i_{32}, (1 \preceq i \preceq n)\}$$

constraints $D = \{d\} \bigcup K$, where d is the $DD = D(A) = \{T, F\}$ (all other attributes have countably infinite domains) and K is the set of key dependencies:

(1) $AV_iV'_iY \rightarrow a(R)$
(2) $AV_iYT^i_{12}T^i_{21} \rightarrow a(R)$
(3) $AV_iYT^i_{13}T^i_{31} \rightarrow a(R)$
(4) $AV_iYT^i_{23}T^i_{32} \rightarrow a(R)$
(5) $V_iYT^i_{12}T^i_{21}T^i_{13}T^i_{31}T^i_{23}T^i_{32} \rightarrow a(R)$
(6) $V'_iX \rightarrow a(R)$
where $1 \preceq i \preceq n$

and $X = \{X_1, X_2, \ldots X_p, X'_1, \ldots, X'_n, X'\}$ where:

$X' = a(R) - \{Y\}$

$X'_i = \{A, V_i, X, Y, T^i_{12}, T^i_{21}, T^i_{13}, T^i_{31}, T^i_{23}, T^i_{32}\} \quad 1 \preceq i \preceq n$

$X_j = \bigcup_{(t_j \in V_i) \wedge (t_j = t_{i_l},\, 1 \preceq l \preceq 3,\, g,h \neq l)} \{Y, V_i, V'_i, T^i_{lg}, T^i_{lh}\} \qquad 1 \preceq j \preceq p$

We use the notation $X'_i = X_{p+2}$ and $X' = X_{n+p+1}$. Note that $\bigcup_{j=1}^{n+p+1} X_j = a(R)$ as required and $A \notin X_j \quad 1 \preceq j \preceq p$, and $V'_i \notin X'_j$ Now let us prove, that there is an exact hitting set W iff there is a counterexample relation C:

a)"only if": Suppose that there is an EHS W. Consider the following m tuples $w_1 ..., w_m$.$(m = n+p+1)$: w_i has a_j in position j if the j^{th} attribute is in X_i, and otherwise w_j contains in position j an arbitrary value, which appears no where else (such unique values can be chosen for all attributes except A). For attribute A $(w_i)_A = T for\ p+1 \preceq i \preceq n+p+1$ and for $1 \preceq i \preceq p$ if $t_i \in W$ then $(w_i)_A = T = a_1$ else $(w_i)_A = F$. (Figure 1 has an example of an EHS and a tableau of the $w's$ where only the a entries are noted and the values of A). Obviously $t = Ta_2, ...a_{8n+3}$ is not among the m tuples described, yet it should be part of any instance containing these tuples, if this instance is to satisfy $LJ(X)$. We note that at this point FD's of types (2) to (6) are not applicable to any pair of tuples because type (6) requires equality of V'_i and X and all other types require equality of Y and T^i_{ke} and T^i_{ek} entries. Type (1) FD's can be applied to pairs of $w_i's$, which correspond to elements of T belonging to the same subset V_j and not belonging to W. Such an application does not make any of types (1) to (4) applicable, because the third element of V_j has a different entry in A and the w corresponding to X'_j has a different A entry also. Type (5) is also inapplicable, because all $T^i_{kl}'s$ for $k \neq l, \quad 1 \preceq k, \quad l \preceq 3$ do not have the same entries. Thus the tuples resulting from the application of the FD's on $w_1, w_2, ...w_m$, satisfy all constraints of $S = (X, D)$ and not $LJ(X)$, thus they form C. $LJ(X)$ is not satisfied because (6) can never be applied and $a_1a_2...a_{8n+3} \notin C$. C is the required counterexample relation.

b)"if": Suppose that a counterexample relation C exists. C should contain m tuples $w_1, w_2, ...w_m$ (not necessarily distinct) as those described in (3) and should not contain their corresponding $a_1a_2...a_l$ tuple. In fact, these tuples by themselves form a counterexample relation. Also we can replace all elements in these tuples, which do not correspond to a's and whose domains are unrestricted, with elements that are nowhere repeated in the tuples. These new tuples $w'_1, w'_2, ...w'_m$ also form a counterexample relation that we will call C'. Let us assume there is no exact hitting set and reach a contradiction. We know that if no EHS exists then for the tuples, which correspond to $X_1X_2, ...X_p$ and therefore to $t_1, t_2, ...t_p$ either:

(a) $\exists_{k \neq l}\, w'_k, w'_l \quad s.t. \quad (t_k, t_l \in V_j) \quad \bigwedge \quad (w'_k)_A = (w'_l)_A = T = a_1$

or

(b) $\exists_{\substack{k,l,q \\ distinct}}\, w'_k, w'_l, w'_q \quad s.t. \quad (t_k, t_l, t_q \in V_j) \quad \bigwedge \quad (w'_k)_A = (w'_l)_A = (w'_q)_A = F \neq a_1$

(If $a_1 = F$ the conditions are identical with T and F interchanged. If these conditions are not satisfied we can construct an EHS from the assignment of T, F values to the A column). If case (a) is true by applying type (1) dependencies and then one of types (2) to (4) we render the application of $V'_i X \rightarrow a(R)$ possible and prove that $a_1, a_2...a_{8n+3}$ must be one of the $w's$. In case (b) by repeatedly applying type (1) dependencies we make the application of a type (5) dependency possible and then that of $V'_i X \rightarrow a(R)$ with the same end effect as before (also in this case application of (5) would imply $a_1 = F$, although we assumed $a_1 = T$). Thus C' cannot be a counterexample relation, which is the desired contradiction. This concludes the proof of Theorem 1.

3. Discussion and Conclusions

We have proven that the LOSSLESS JOIN problem is NP-complete, even if D consists of only key dependencies and just one DD. This is directly connected with Domain Key Normal Form recognition [4].

If D contains only FD's the problem can be solved in polynomial time [1]. The best time bound to date is $0(n^2 logn)$ [3]. If D contains MVD's or JD's the problem is open and the general procedure ("chase"), which attempts to construct a template of all counterexample relations [6] is inefficient.

Acknowledgment The author would like to thank Prof. C.H. Papadimitriou for many helpful suggestions in the presentation of this result.

References:

[1] Aho, A.V., C. Beeri and J.D. Ullman: "The Theory of Joins in Relational Databases" Proc. 19th FOCS (Oct. 1977) pp. 107-113.

[2] Beeri,C. P.A. Bernstein: "Computational Problems Related to the Design of Normal Form Relational Schemas" TODS, Vol.4, No1 (March 1979) pp. 30-59.

[3] Downey, P.J., R. Sethi and R.E. Tarjan: "Variations of the Common Subexpression Problem" to appear in JACM.

[4] Fagin, R.: "A Normal Form for Relational Databases based on Domains and Keys" IBM Res. Rep. RJ2520 (May 1979).

[5] Garey, M.R., D.S. Johnson: "Computers and Intractability – A Guide to the Theory of NP-Completeness" Freeman (1979).

[6] Maier, D., A. Mendelzon, Y. Sagiv: "Testing Implications of Data Dependencies" Supplement to Proc. SIGMOD 1979, pp. 20-28.

	A	V_1	V_1'	V_2	V_2'	V_3	V_3'	T^1_{12}	T^1_{21}	T^1_{13}	T^1_{31}	T^1_{23}	T^1_{32}	T^2_{12}	T^2_{21}	T^2_{13}	T^2_{31}	T^2_{23}	T^2_{32}	T^3_{12}	T^3_{21}	T^3_{13}	T^3_{31}	T^3_{23}	T^3_{32}	X	Y
X_1	F	a_2	a_3			a_6	a_7	a_8		a_{10}										a_{20}		a_{22}					a_{27}
X_2	T	a_2	a_3	a_4	a_5	a_6	a_7		a_9			a_{12}		a_{14}		a_{16}					a_{21}			a_{24}			a_{27}
X_3	F	a_2	a_3	a_4	a_5						a_{11}		a_{13}		a_{15}			a_{18}									a_{27}
X_4	F			a_4	a_5	a_6	a_7										a_{17}		a_{19}				a_{23}		a_{25}		a_{27}
X_1'	T	a_2						a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}													a_{26}	a_{27}
X_2'	T			a_4										a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}							a_{26}	a_{27}
X_3'	T					a_6														a_{20}	a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}
X'	T	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}	a_{20}	a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	

$T = a_1$

EHS: $T = \{1,2,3,4\}$

$V_1 = \{1,2,3\}$

$V_2 = \{2,3,4\}$

$V_3 = \{1,2,4\}$

$W = \{2\}$

Figure 1