CS345 Notes for Lecture 10/16/96

Generalization to Unions of CQ's

$P_1 \cup P_2 \cup \cdots \cup P_k \subseteq Q_1 \cup Q_2 \cup \cdots \cup Q_n$ iff for all P_i there is some one Q_j such that $P_i \subseteq Q_j$.

Proof (If)

Obvious.

Proof (Only If)

Assume the containment holds.

- Let D be the canonical (frozen) database from CQ P_i.
- Since the containment holds, and $P_i(D)$ surely includes the frozen head of P_i, there must be some Q_j such that $Q_j(D)$ includes the frozen head of P_i.
- Thus, $P_i \subseteq Q_j$.

Union Theorem Just Misses Being False

Consider generalized CQ's allowing arithmetic-comparison subgoals.

P_1: `p(X) :- e(X) & 10 <= X & X <= 20`

Q_1: `p(X) :- e(X) & 10 <= X & X <= 15`
Q_2: `p(X) :- e(X) & 15 <= X & X <= 20`

- $P_1 \subseteq Q_1 \cup Q_2$, but $P_1 \subseteq Q_1$ and $P_1 \subseteq Q_2$ are both false.

CQ Contained in Recursive Datalog

Test relies on method of canonical DB's; containment mapping approach doesn't work (it's meaningless).

- Make DB D from frozen body of CQ.
- Apply program to D. If frozen head of CQ appears in result, then yes (contained), else no.

Example:

```
Q1: path(X,Y) :- arc(X,Z) &
        arc(Z,W) & arc(W,Y)
```

Q_2 is the value of *path* in the following recursive Datalog program:

```
r1: path(X,Y) :- arc(X,Y)
r2: path(X,Y) :- path(X,Z) & path(Z,Y)
```

- Freeze Q_1, say with 0, 1, 2, 3 as constants for X, Z, W, Y, respectively.

  $D = \{arc(0,1),\ arc(1,2),\ arc(2,3)\}$

- Frozen head is $path(0,3)$.

- Easy to infer that $path(0,3)$ is in $Q_2(D)$ — use r_1 three times to infer $path(0,1)$, $path(1,2)$, $path(2,3)$, then use r_2 to infer $path(0,2)$, $path(0,3)$.

Harder Cases

- Datalog program $\subseteq$ CQ: doubly exponential complexity. Reference: Chaudhuri, S. and M. Y. Vardi [1992]. "On the equivalence of datalog programs," *Proc. Eleventh ACM Symposium on Principles of Database Systems*, pp. 55–66.

- Datalog program $\subseteq$ Datalog program: undecidable.

CQ's With Negation

General form of conjunctive query with negation (CQN):

$$H \text{ :- } G_1 \ \&\ \ldots\ \&\ G_n\ \&$$
$$\texttt{NOT}\ F_1\ \&\ \ldots\ \&\ \texttt{NOT}\ F_m$$

- G's are *positive* subgoals; F's are *negative* subgoals.

- Apply CQN Q to DB D by considering all possible substitutions of constants for the variables of Q. If *all* the positive subgoals become facts in D and *none* of the negative subgoals do, then infer the substituted head.

- ☐ Set of inferred facts is $Q(D)$.

- Containment of CQ's doesn't change. $Q_1 \subseteq Q_2$ if for every database D, $Q_1(D) \subseteq Q_2(D)$.

Example:

```
C1: p(X,Z) :- a(X,Y) & a(Y,Z) &
              NOT a(X,Z)
C2: p(A,C) :- a(A,B) & a(B,C) &
              NOT a(A,D)
```

- Intuitively, Q_1 looks for paths of length 2 that are not "short-circuited" by a single arc from beginning to end.

- Q_2 looks for paths of length 2 that start from a node A that is not a "universal source"; i.e., there is at least one node D not reachable from A by an arc.

- We thus expect $Q_1 \subseteq Q_2$, but not vice-versa.

Levy-Sagiv Test

To test $Q_1 \subseteq Q_2$:

1. Construct the set of *basic canonical* databases that correspond to all the partitions of the set of variables of Q_1.

   ☐ That is, for each partition, assign a unique constant to each block of the partition.

   ☐ Create the basic canonical DB by replacing each variable by the constant of its block. The basic canonical DB is the set of resulting *positive* subgoals.

2. For each basic canonical DB D constructed in (1), check that:

   ☐ If $Q_1(D)$ contains the frozen head of Q_1, then so does $Q_2(D)$.

   Note that unlike ordinary CQ's, it is possible that $Q_1(D)$ does not contain Q_1's head, because D may make a negated subgoal false (i.e., D contains the frozen subgoal without the `NOT`).

3. If $Q_1(D)$ contains the frozen head of Q_1, we must then also consider the larger set of (*extended*) canonical DB's D' formed by adding to D other tuples that are formed from the same symbols as D, but not any of the tuples that are the negated subgoals of Q_1.

    - Check that if $Q_1(D)$ contains its frozen head, so does $Q_2(D')$.

4. If so, $Q_1 \subseteq Q_2$; if not, then not.

Example: Consider C_1 above. The variables are $\{X, Y, Z\}$.

- There are five partitions of the variables, shown in the table below.

	Partition	Basic Canonical DB D
1)	$\{X\}\{Y\}\{Z\}$	$\{a(0,1), a(1,2)\}$
2)	$\{X,Y\}\{Z\}$	$\{a(0,0), a(0,1)\}$
3)	$\{X\}\{Y,Z\}$	$\{a(0,1), a(1,1)\}$
4)	$\{X,Z\}\{Y\}$	$\{a(0,1), a(1,0)\}$
5)	$\{X,Y,Z\}$	$\{a(0,0)\}$

- In cases (2), (3), and (5), $C_1(D)$ does not contain its own frozen head.

    - E.g., in case (2), the only substitution that makes the positive subgoals of C_1 true is $X \to 0$, $Y \to 0$, and $Z \to 1$. But then, the negative subgoal `NOT` $a(X,Z)$ becomes false, since $a(X,Z) = a(0,1)$ and $a(0,1)$ is indeed in D.

- In cases (1) and (4), $C_1(D)$ contains C_1's frozen head, but so does $C_2(D)$ and any extended canonical DB $D' \supseteq D$.

    - E.g., in case (4), the frozen head of C_1 is $p(0,0)$. $C_2(D)$ contains $p(0,0)$, as we can see from the substitution $A \to 0$,$B \to 1$, $C \to 0$, $D \to 2$.

    - Moreover, adding tuples consisting of 0's, 1's and 2's to D cannot change things as long as we don't add $a(0,2)$, the frozen negative subgoal of C_1. Then, both

$C_1(D')$ and $C_2(D')$ contain C_1's frozen head.

Example: Consider a slightly different pair of CQ's:

```
C1: p(X,Z) :- a(X,Y) & a(Y,Z) &
              NOT a(X,Z)
C2: p(A,C) :- a(A,B) & a(B,C) &
              NOT a(C,C)
```

- C_1 is the same, so the basic canonical DB's are the same.
- However, consider the partition $\{X\}\{Y\}\{Z\}$.
- While for the resulting basic canonical DB $D = \{a(0,1), a(1,2)\}$, both $C_1(D)$ and $C_2(D)$ contain C_1's frozen head, the same is not true for the extended canonical DB $D' = \{a(0,1), a(1,2), a(2,2)\}$.