Part 1: Data Exchange

Consider the following task of reverse query processing [1].

**Input:**
- A schema $S = (R, \Sigma)$ where $\Sigma$ is a set of integrity constraints;
- A $k$-ary query $Q$ over $S$;
- A $k$-ary relation $r$.

**Goal:** Construct an instance $J$ over $S$ such that $Q(J) = r$, or determine that no such $J$ exists.

**Question 1.1.** Show “simple and interesting” examples of input where $Q$ is a Conjunctive Query (CQ) and $\Sigma$ consists of Functional Dependencies (FDs) such that:

1. No solution $J$ exists, but if we eliminate the FDs then a solution does exist.
2. There are no FDs, and yet, no solution $J$ exists.

**Question 1.2.** Assume that all of the following hold:

- $\Sigma$ consists of only FDs;
- $Q$ is a CQ of hypertree width bounded by some constant $b$;
- $Q$ has no self joins\(^1\) and no contants.

Devise a polynomial-time (combined-complexity) algorithm for solving reverse query processing under these terms. Prove the correctness of your algorithm, and explain why it terminates in polynomial time.

**Note:** We use combined complexity here, which means that you cannot assume that $S$ and $Q$ are fixed. In particular, you cannot assume that, given a candidate instance $J'$, you can construct $Q(J')$ in polynomial time!

Part 2: Consistent Query Answering

Consider the Boolean CQ $Q() := R(x, y), S(y, x, z), T(z, w)$.

**Question 2.1.** Using the trichotomy theorem of Koutris and Wijsen (studied in class), show that $\text{Consistent}^Q_\Sigma$ cannot be phrased in First-Order Logic.

\(^1\)A self join of a CQ is pair of atoms over the same relation symbol; hence, “no self joins” means that every relation symbol appears exactly once.
**Question 2.2.** Devise a polynomial-time algorithm for computing $\text{Consistent}_\Sigma^Q(I)$ on a given input instance $I$. (Hint: adapt the strategy for $R(x, y), S(y, x)$ we saw in class.)

**Part 3: Aggregates on Repairs**

Consider a schema $(S, \Sigma)$ where $S$ consists of a single relation schema $R(A, B)$ and $\Sigma$ is the FD set \{A $\rightarrow$ B, B $\rightarrow$ A\}. The following are aggregate queries in SQL that produce a number as a result.

- $Q_1$: SELECT COUNT(*) FROM R
- $Q_2$: SELECT MIN(B) FROM R

Given an inconsistent instance $I$ over $(S, \Sigma)$, we are interested in the extremal (minimal and maximal) values for each query $Q$:

\[
Q_{\min}^i(I) \overset{\text{def}}{=} \min \{ Q(J) \mid J \in \text{Repairs}_\Sigma(I) \}
\]
\[
Q_{\max}^i(I) \overset{\text{def}}{=} \max \{ Q(J) \mid J \in \text{Repairs}_\Sigma(I) \}
\]

For each of the two queries $Q_i$, give an efficient algorithm for, or prove hardness of, computing $Q_{\min}^i(I)$ and $Q_{\max}^i(I)$.

Hint: find out about the following concepts (and you can use known results about them):

- Maximum matching;
- Minimum maximal matching.

Good luck!

**References**