Gaifman-Locality of Queries
Notes for the Lecture of May 6
Toon Calders

The material presented in these notes is based on the series of papers [3, 4, 2, 1]. For an overview and a formal, yet relatively accessible treatment of expressibility results of SQL, especially [3]. Notice that the material presented in the papers [3, 4, 2, 1] covers much more than the lectures. Only the material covered in the lectures and in this note will be examined.

1 Assumptions and Preliminary Definitions

We assume the existence of an infinite, countable and unordered set of constants $C$. For reasons of simplicity we assume that all attributes in all relations we consider have $C$ as a domain. Hence, a relation $R$ of arity $k$ is a finite subset of $C^k$. For a tuple $\bar{t} \in C^k$, $t_i$, $1 \leq i \leq k$ denotes the $i$th component of $\bar{t}$; i.e., $\bar{t} = (t_1, \ldots, t_k)$.

For a relation $R$ of arity $k$, the active domain of $R$, denoted $\text{adom}(R)$, is defined as:

$$\bigcup_{i=1}^{k} \{ \bar{t} \mid \bar{t} \in R \}.$$ 

For a database $D$ with relations $R_1, \ldots, R_n$, $\text{adom}(D)$ denotes $\bigcup_{i=1}^{n} \text{adom}(R_i)$; i.e., $\text{adom}(D)$ is the set of all constants that appear in $D$.

2 Gaifman Locality: Definition

Intuitively, a query $Q$ will be called Gaifman local if there exists a bound $r$, such that $Q$ is “incapable” of looking further than a distance $r$. As such, if we have two tuples $\bar{t}$ and $\bar{t'}$, of which the “neighborhoods up to distance $r$” “look alike”, the query will have to treat both tuples in the exact same way; either both will be part of the result of the query, or both are not. Because of their similarity, the query $Q$ cannot separate them. Gaifman locality makes this definition formal.

First the notion of the Gaifman graph of a database is introduced. This graph represents how the elements of $C$ are “connected.”

Consider a database $D$ with relations $R_1, \ldots, R_n$, the arity of $R_i$ is $a_i$ for $i = 1 \ldots n$. The Gaifman graph of $D$, denoted $G(D)$, is defined as the undirected graph $G(\text{adom}(D), E)$ with $E$ the following set:

$$E := \bigcup_{i=1}^{n} \bigcup_{\bar{t} \in R_i} \{ \{ t_i, t_j \mid 1 \leq i, j \leq a_i \} \}.$$ 

That is, $(a, b)$ is in $E$ if and only if there exists a tuple in one of the relations of $D$ in which both $a$ and $b$ appear.
For $a, b \in \text{dom}(D)$, the distance between $a$ and $b$ in $D$, denoted $d_D(a, b)$, is defined as the length of the shortest path between $a$ and $b$ in $G(D)$. $d_D(a, a) = 0$ and $d_D(a, b) = \infty$ if there is no path between $a$ and $b$ in $G(D)$.

The sphere with radius $r$ around $a \in \text{dom}(D)$ in $D$ is defined as:
\[
S^D_r(a) := \{ b \in \text{dom}(D) \mid d(a, b) \leq r \},
\]
and the sphere around a tuple $t = (t_1, \ldots, t_k)$ as:
\[
S^D_r(t) := \bigcup_{i=1}^{k} S^D_r(t_i).
\]
The $r$-neighborhood of a tuple $t = (t_1, \ldots, t_k)$ in $D$ will now express what “can be seen at a distance $r$ from $t$”, and is defined as:
\[
N^D_r(t) := \langle R_1 \cap (S^D_r(t))^a_1, \ldots, R_n \cap (S^D_r(t))^a_n, t_1, \ldots, t_k \rangle.
\]

Let $x = (x_1, \ldots, x_k)$ and $y = (y_1, \ldots, y_k)$ be two tuples in $\text{dom}(D)^k$. $x$ and $y$ are said to be $r$-equivalent in $D$, denoted $x \approx^D_r y$, if $N^D_r(x) \cong N^D_r(y)$; that is: there exists a bijective function $\pi : \text{dom}(D) \to \text{dom}(D)$, such that:

- $\forall i = 1 \ldots n : \pi(R_i \cap (S^D_r(x))^a_i) = (R_i \cap (S^D_r(y))^a_i)$, and
- $\forall i = 1 \ldots k : \pi(x_i) = y_i$.

A query $Q$ is said to be Gaifman-local if for every database $D$, and every two tuples $x, y \in \text{dom}(D)^k$ it holds that if $x \approx^D_r y$, then $x \in Q(D)$ if and only if $y \in Q(D)$. That is, $Q$ cannot distinguish between tuples with isomorphic $r$-neighborhoods.

### 3 Main theorem

The following theorem is well-known in database theory. For a fully formal statement of what exactly is considered to be an SQL-query (i.e., which constructions are allowed), please see [3]. In short, the SQL-fragment taught in the Databases I course falls within the fragment described in the theorem.

**Theorem 1 ([3])** Every SQL-query (on unordered domains) that does not use constants is Gaifman-local, even in the presence of nesting and unnesting, any set of arithmetic operations and any set of aggregation functions.

**Note:** The theorem holds for all possible aggregation functions and every arithmetic operation, no matter how complex, as long as the domains used in the database are unordered. If the domains are ordered, the theorem no longer holds in full generality.

This theorem allows for an easy proof that transitive closure cannot be expressed in SQL, not even in the presence of nesting, unnesting, arithmetic operations, and aggregations: ... (yet to be completed)

### References

