The Fine-Grained Complexity of Boolean Conjunctive Queries and Sum-Prod Problems

Austen Z. Fan†, Paraschos Koutris†, Hangdong Zhao†

†University of Wisconsin - Madison

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

Boolean Conjunctive Queries

A **conjunctive query** q is an expression of the form

$$q(x_1, \dots, x_k) : - R_1(\vec{y}_1), \dots, R_n(\vec{y}_n).$$

It is called **Boolean** if its head is empty. Conjunctive queries capture the Select-Project-Join (SPJ) expressions in database system. For example,

$$q(x, y, z) : - R(x, y), S(y, z), T(z, x)$$

is listing 3-cycles, while

$$q() : - R(x, y), S(y, z), T(z, x)$$

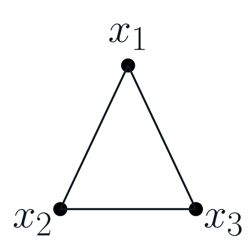
is detecting (the existence of) a 3-cycle.

Hypergraphs

For every CQ q, we associate a **hypergrpah** $\mathcal{H}$ to it, where the vertices are variables and the hyperedges are atoms. For example, the BCQ

$$q() : - R(x_1, x_2), S(x_2, x_3), T(x_3, x_1)$$

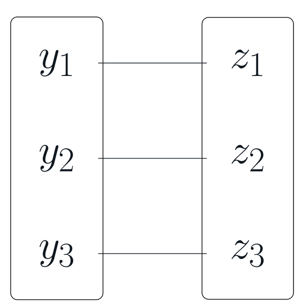
is associated with



while the BCQ

$$q() : - R(y_1, z_1), S(y_2, z_2), T(y_3, z_3), U(y_1, y_2, y_3), V(z_1, z_2, z_3)$$

is associated with



Sum-of-Product over Semiring

Green, Karvounarakis and Tannen observed that database queries can be written as Sum-of-Product computation over semirings [1].

$$q() : - R_1(\vec{x}_1), R_2(\vec{x}_2), \dots, R_n(\vec{x}_n)$$

$$q(I) := \bigvee_{v:\text{valuation}} \bigwedge_{i=1}^{n} R_i(v(\vec{x}_i))$$

$$q(I) := \bigoplus_{v:\text{valuation}} \bigotimes_{i=1}^{n} R_i(v(\vec{x}_i))$$

Then, $(\{\text{TRUE}, \text{FALSE}\}, \vee, \wedge) \leftrightarrow$ set semantics while $(\mathbb{N}, +, *) \leftrightarrow$ bag semantics.

Given an edge-weighted graph $G = (V, \text{weight})$:

Compute $\bigvee_{V' \subseteq V, |V'| = k} \bigwedge_{\{v,w\} \in V'} \text{weight}(\{v, w\}) \leftrightarrow$ Boolean k-clique

Compute $\sum_{V' \subseteq V, |V'| = k} \prod_{\{v,w\} \in V'} \text{weight}(\{v, w\}) \leftrightarrow$ Counting k-clique

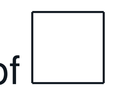
Compute $\min_{V' \subseteq V, |V'| = k} \sum_{\{v,w\} \in V'} \text{weight}(\{v, w\}) \leftrightarrow$ Minimum k-clique

PANDA

The state-of-the-art algorithm to answre a BCQ is called Proof-Assisted eNtropic Degree-Aware, or **PANDA** [2].

Theorem [Abo Khamis, Ngo & Suciu]: Any BCQ q can be solved in time $\tilde{O}(N^{\text{subw}(q)})$.

The **submodular width** of a hypergraph is $\text{subw}(\mathcal{H}) := \max_b \min_{(\mathcal{T}, \chi)} \max_{t \in V(\mathcal{T})} b(\chi(t))$.

For example, the submodular width of  is $\frac{3}{2}$.

Fine-Grained Complexity

An emerging field in theoretical computer science which aims to classify problems according to their exact running time, or "hardness in easy problems."

The followings can be found in [3]:

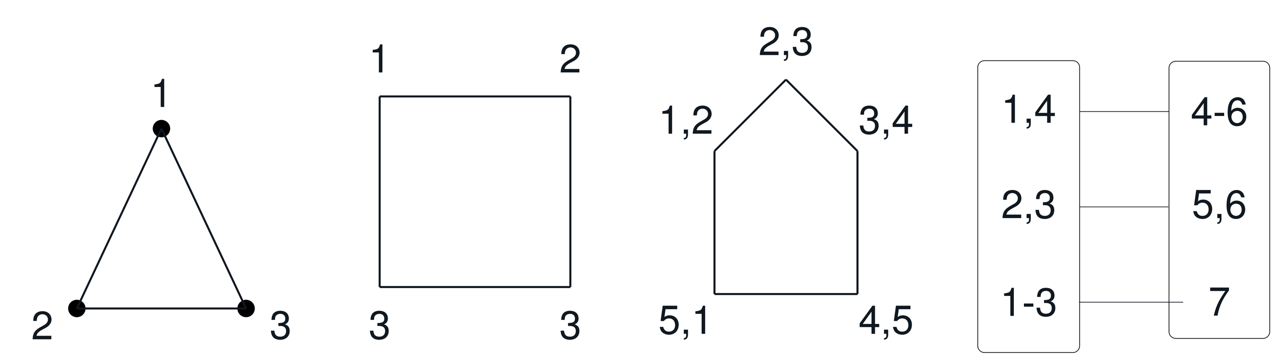
Hypothesis [Lincoln, Vassilevska-Williams & Williams]: Any combinatorial algorithm to detect a k-clique in a graph with n nodes requires $n^{k-o(1)}$ time on a Word RAM model.

Hypothesis [Lincoln, Vassilevska-Williams & Williams]: Any randomized algorithm to find a k-clique of minimum total edge weight requires $n^{k-o(1)}$ time on a Word RAM model.

Clique Embedding Power

A k-**clique embedding** from C_k to $\mathcal{H}$ is a mapping ψ from $v \in [k]$ to a non-empty subset $\psi(v) \subseteq \mathcal{V}$ such that (1) $\forall v, \psi(v)$ induces a connected subhypergraph and (2) $\forall \{v, u\}$, $\psi(v), \psi(u)$ touch in $\mathcal{H}$.

Examples of clique embeddings:



Below we summarize the **clique embedding power** and submodular width for some classes of queries.

	emb	subw
Acyclic	1	1
Chordal	$=$	$=$
ℓ-cycle	$2 - 1/\lceil \ell/2 \rceil$	$2 - 1/\lceil \ell/2 \rceil$
$K_{2,\ell}$	$2 - 1/\ell$	$2 - 1/\ell$
$K_{3,3}$	2	2
A_ℓ	$(\ell - 1)/2$	$(\ell - 1)/2$
$\mathcal{H}_{\ell,k}$	ℓ/k	ℓ/k
Q_b	$17/9$	2
Q_{hb}	$7/4$	2

Our Main Results

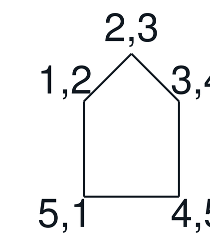
Theorem 1 ([4]) *For any $\mathcal{H}$, $\text{CSP}(\mathcal{H})$ cannot be computed via a combinatorial algorithm in time $O(|I|^{\text{emb}(\mathcal{H}) - \epsilon})$ unless the Combinatorial k-Clique Conjecture is false.*

Our reduction is **semiring oblivious** in the following sense:

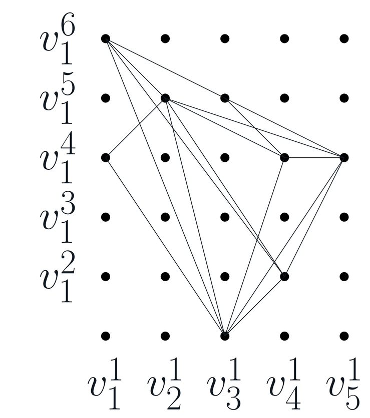
Theorem 2 ([4]) *For any $\mathcal{H}$, $\text{CSP}(\mathcal{H})$ over tropical semiring cannot be computed via any randomized algorithm in time $O(|I|^{\text{emb}(\mathcal{H}) - \epsilon})$ unless the Min Weight k-Clique Conjecture is false.*

Proof by Picture

To reduce 5-cycle BCQ to 5-clique, consider the 5-clique-embedding of 5-cycle.



WLOG, assume the input graph for detecting 5-clique is 5-partite.



Construct the database instance as follows:

x_1	x_2
$\langle v_1^6, v_2^5 \rangle$	$\langle v_2^5, v_3^1 \rangle$
$\langle v_1^4, v_2^5 \rangle$	$\langle v_2^5, v_3^1 \rangle$

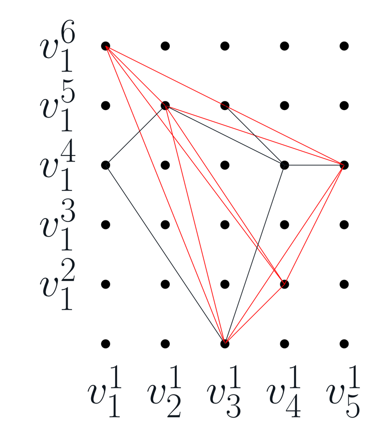
x_2	x_3
$\langle v_2^5, v_3^1 \rangle$	$\langle v_3^1, v_4^2 \rangle$

x_3	x_4
$\langle v_3^5, v_4^4 \rangle$	$\langle v_4^4, v_5^4 \rangle$
$\langle v_3^1, v_4^2 \rangle$	$\langle v_4^2, v_5^4 \rangle$
$\langle v_3^1, v_4^4 \rangle$	$\langle v_4^4, v_5^4 \rangle$

x_4	x_5
$\langle v_4^2, v_5^4 \rangle$	$\langle v_5^4, v_1^6 \rangle$

x_5	x_1
$\langle v_5^4, v_1^6 \rangle$	$\langle v_1^6, v_2^5 \rangle$

The 5-clique in the input graph is in one-to-one correspondence to a 5-cycle in the database instance.



Acknowledgements

This research was supported in part by National Science Foundation grant IIS-1910014.

References

[1] Todd J. Green, Gregory Karvounarakis, and Val Tannen. "Provenance semirings". In: *PODS*. ACM, 2007, pp. 31–40.

[2] Mahmoud Abo Khamis, Hung Q. Ngo, and Dan Suciu. "What Do Shannon-type Inequalities, Submodular Width, and Disjunctive Datalog Have to Do with One Another?" In: *PODS*. ACM, 2017, pp. 429–444.

[3] Andrea Lincoln, Virginia Vassilevska Williams, and R. Ryan Williams. "Tight Hardness for Shortest Cycles and Paths in Sparse Graphs". In: *SODA*. SIAM, 2018, pp. 1236–1252.

[4] Austen Z. Fan, Paraschos Koutris, and Hangdong Zhao. "The Fine-Grained Complexity of Boolean Conjunctive Queries and Sum-Product Problems". In: *ICALP*. Vol. 261. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 127:1–127:20.