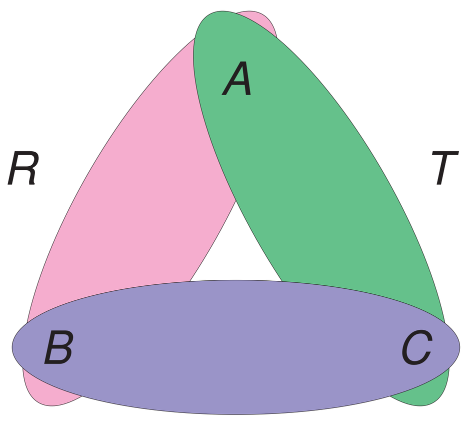
Counting Triangles under Updates

Ahmet Kara, Hung Q. Ngo, Milos Nikolic, Dan Olteanu, and Haozhe Zhang
fdbresearch.github.io

RelationalAI

Problem Setting

Maintain the triangle count Q
under single-tuple updates to R, S, and T!



Q counts the number of tuples
in the join of R, S, and T.

$$Q = \sum_{a,b,c} R(a,b) \cdot S(b,c) \cdot T(c,a)$$

*We write a instead of $a \in dom(A)$.

Relations and Single-Tuple Updates

- Relations are functions mapping tuples to multiplicities.
- A single-tuple update maps a tuple to a non-zero value (positive for insertions negative for deletions).

database **D**

R

A	B	
a_1	b_1	2
~~a_2~~	~~b_1~~	~~3~~
a_2	b_1	1

S

B	C	
b_1	c_1	2
b_1	c_2	1

T

C	A	
c_1	a_1	1
c_2	a_1	3
c_2	a_2	3

$R \cdot S \cdot T$

A	B	C	
a_1	b_1	c_1	$2 \cdot 2 \cdot 1 = 4$
a_1	b_1	c_2	$2 \cdot 1 \cdot 3 = 6$
~~a_2~~	~~b_1~~	~~c_2~~	~~$3 \cdot 1 \cdot 3 = 9$~~
a_2	b_1	c_2	$1 \cdot 1 \cdot 3 = 3$

$\delta R(a_2, b_1)$

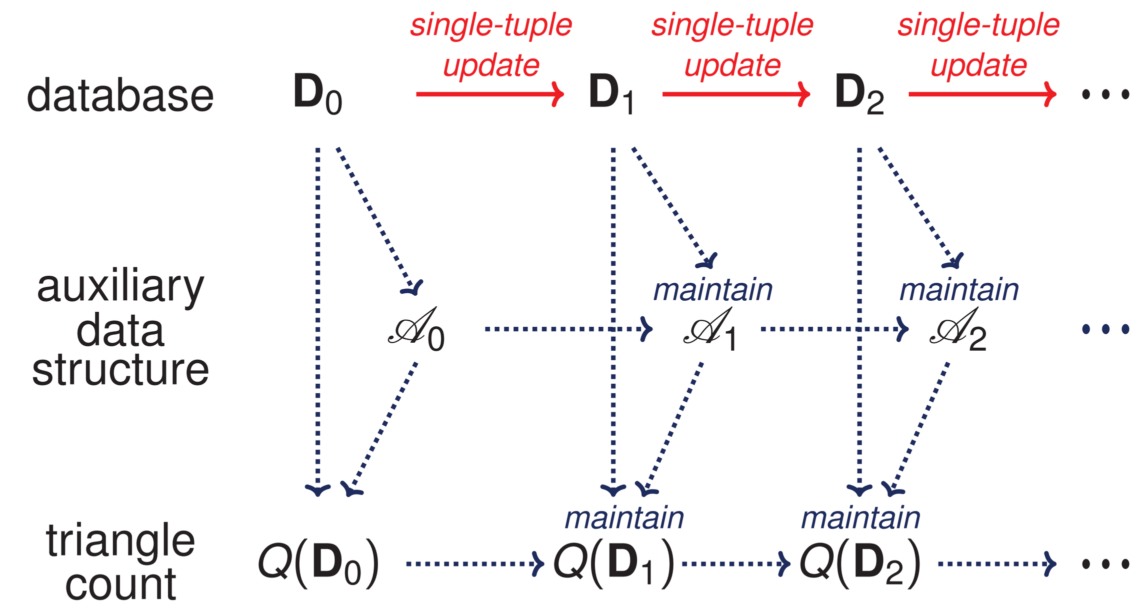
A	B	
a_2	b_1	-2

A single-tuple update to relation R

$Q(\mathbf{D})$

$\emptyset$	
~~()~~	~~$4+6+9=19$~~
()	$4+6+3=13$

The Maintenance Problem



Given a current database **D** and a single-tuple update, what are the time and space complexities for maintaining $Q(\mathbf{D})$?

Naïve Maintenance

"*Compute from scratch!*"

$$\sum_{a,b,c} \big[\underbrace{R(a,b) + \delta R(a',b')}_{newR}\big] \cdot S(b,c) \cdot T(c,a) = \sum_{a,b,c} newR(a,b) \cdot S(b,c) \cdot T(c,a)$$

Maintenance complexity

- Time: $O(|\mathbf{D}|^{1.5})$ by worst-case optimal join algorithms
- Space: $O(|\mathbf{D}|)$ to store the relations

Classical Incremental View Maintenance (IVM)

"*Compute the difference!*"

$$\sum_{a,b,c} \big[R(a,b) + \delta R(a',b')\big] \cdot S(b,c) \cdot T(c,a) = \sum_{a,b,c} R(a,b) \cdot S(b,c) \cdot T(c,a) + \sum_{c} \delta R(a',b') \cdot S(b',c) \cdot T(c,a')$$

Maintenance complexity:

- Time: $O(|\mathbf{D}|)$ to intersect C-values from S and T
- Space: $O(|\mathbf{D}|)$ to store the relations

Factorized Incremental View Maintenance (F-IVM)

"*Compute the difference by using pre-materialized views!*"

Pre-materialize $V_{ST}(b,a) = \sum_c S(b,c) \cdot T(c,a)$!

$$\sum_{a,b,c} \big[R(a,b) + \delta R(a',b')\big] \cdot S(b,c) \cdot T(c,a) = \sum_{a,b,c} R(a,b) \cdot S(b,c) \cdot T(c,a) + \delta R(a',b') \cdot V_{ST}(b',a')$$

Maintenance complexity:

- Time for updates to R: $O(1)$ to look up in V_{ST}
- Time for updates to S and T: $O(|\mathbf{D}|)$ to maintain V_{ST}
- Space: $O(|\mathbf{D}|^2)$ to store the relations and V_{ST}

IVM$^\varepsilon$ Closes the Maintenance Complexity Gap

Complexity bounds for maintaining the triangle count

Known upper bound
Time $O(|\mathbf{D}|)$, Space $O(|\mathbf{D}|)$
Known lower bound
Amortized time not $O(|\mathbf{D}|^{0.5-\gamma})$ for any $\gamma > 0$
(under reasonable complexity theoretic assumptions)

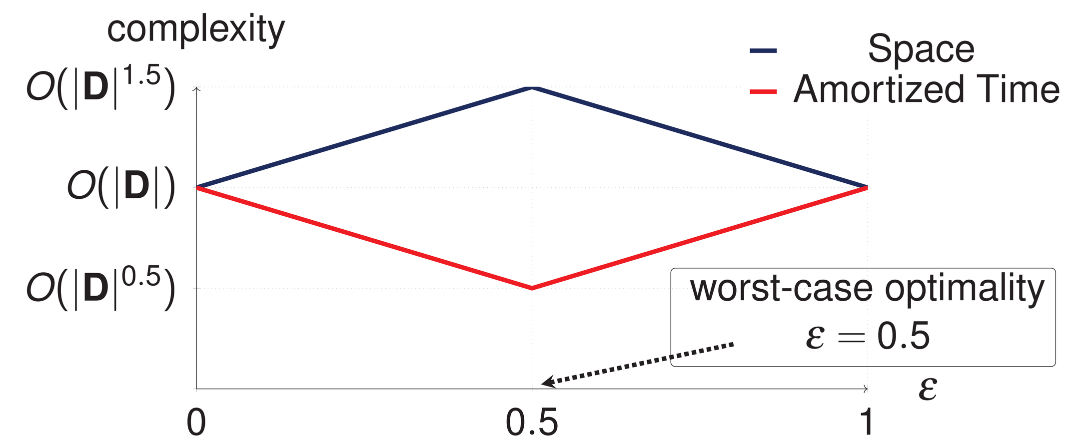
Can the triangle count be maintained in sublinear time?

Yes! We propose **IVM$^\varepsilon$!**
It maintains the triangle count in $O(|\mathbf{D}|^{0.5})$ amortized time.
This is **worst-case optimal!**

IVM$^\varepsilon$ Exhibits a Time-Space Tradeoff

Given $\varepsilon \in [0,1]$, IVM$^\varepsilon$ maintains the triangle count with

- $O(|\mathbf{D}|^{\max\{\varepsilon, 1-\varepsilon\}})$ amortized time and
- $O(|\mathbf{D}|^{1+\min\{\varepsilon, 1-\varepsilon\}})$ space.



- IVM$^\varepsilon$ recovers known maintenance approaches.

Summary and Future Work

Main ideas in IVM$^\varepsilon$

- Compute the difference like in Classical IVM!
- Materialize views like in F-IVM!
- New: Use adaptive processing based on data skew!

Generalization of IVM$^\varepsilon$

- IVM$^\varepsilon$ variants achieve amortized sublinear maintenance time for counting versions of Loomis-Whitney, 4-cycle, and 4-path.

Ongoing Work

- Using IVM$^\varepsilon$ to characterize the class of conjunctive count queries admitting sublinear maintenance time!
- Implementing IVM$^\varepsilon$ on top of DB-Toaster

Quick Look inside IVM$^\varepsilon$

Heavy-Light Partitioning

IVM$^\varepsilon$ partitions R into a light part R_L and a heavy part R_H based on the degrees of A-values:

- $R_L = \{t \in R \mid |\sigma_{A=t.A} R| < |\mathbf{D}|^\varepsilon\}$
- $R_H = R \setminus R_L$

Derived bounds on the two parts of R:

- for all A-values a: $|\sigma_{A=a} R_L| < |\mathbf{D}|^\varepsilon$
- $|\pi_A R_H| \leq |\mathbf{D}|^{1-\varepsilon}$

Analogous partitioning of S and T:

- $S = S_L \cup S_H$ (based on degrees of B-values)
- $T = T_L \cup T_H$ (based on degrees of C-values)

Adaptive Maintenance Strategy

- IVM$^\varepsilon$ rewrites the triangle count into a sum $\sum_{UVW \in \{L,H\}} Q_{UVW}$ of *skew-aware views* $Q_{UVW} = \sum_{a,b,c} R_U(a,b) \cdot S_V(b,c) \cdot T_W(c,a)$.
- It uses adaptive strategies for the maintenance of these skew-aware views.

Pre-materialized views maintained by IVM$^\varepsilon$

Pre-materialized view	Space
Q_{UVW} with $U, V, W \in \{L, H\}$	$O(1)$
$V_{RS}(a,c) = \sum_b R_H(a,b) \cdot S_L(b,c)$	$O(\lvert\mathbf{D}\rvert^{1+\min\{\varepsilon,1-\varepsilon\}})$
$V_{ST}(b,a) = \sum_c S_H(b,c) \cdot T_L(c,a)$	$O(\lvert\mathbf{D}\rvert^{1+\min\{\varepsilon,1-\varepsilon\}})$
$V_{TR}(c,b) = \sum_a T_H(c,a) \cdot R_L(a,b)$	$O(\lvert\mathbf{D}\rvert^{1+\min\{\varepsilon,1-\varepsilon\}})$

Computing the deltas of the pre-materialized views for an update $\delta R_*(a,b)$ to the heavy ($* = H$) or light ($* = L$) part of R.

Delta computation strategy from left to right	Time
$\delta Q_{*HH} = \delta R_*(a',b') \cdot \sum_c T_H(c,a') \cdot S_H(b',c)$	$O(\lvert\mathbf{D}\rvert^{1-\varepsilon})$
$\delta Q_{*HL} = \delta R_*(a',b') \cdot V_{ST}(b',a')$	$O(1)$
$\delta Q_{*LH} = \delta R_*(a',b') \cdot \sum_c T_H(c,a') \cdot S_L(b',c)$ or $= \delta R_*(a',b') \cdot \sum_c S_L(b',c) \cdot T_H(c,a')$	$O(\lvert\mathbf{D}\rvert^{\min\{\varepsilon,1-\varepsilon\}})$
$\delta Q_{*LL} = \delta R_*(a',b') \cdot \sum_c S_L(b',c) \cdot T_L(c,a')$	$O(\lvert\mathbf{D}\rvert^{\varepsilon})$
$\delta V_{RS}(a',c) = \delta R_H(a',b') \cdot S_L(b',c)$	$O(\lvert\mathbf{D}\rvert^{\varepsilon})$
$\delta V_{ST}(b,a) = \emptyset$	-
$\delta V_{TR}(c,b') = \delta R_L(a',b') \cdot T_H(c,a')$	$O(\lvert\mathbf{D}\rvert^{1-\varepsilon})$

Reference

Ahmet Kara, Hung Q. Ngo, Milos Nikolic, Dan Olteanu, and Haozhe Zhang. Counting triangles under updates in worst-case optimal time. http://arxiv.org/abs/1804.02780.