Maintaining Triangle Queries under Updates

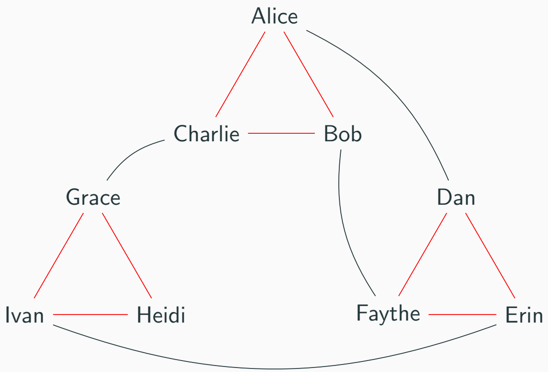
Ahmet Kara, Milos Nikolic, Hung Q. Ngo, Dan Olteanu, Haozhe Zhang

Daniel Schmitt

January 12, 2022

Motivation

Motivation



- Graphs as models for, e.g., social networks, the internet,. . .
- Triangles often appear in social networks
- Triangle counts: community detection, local clustering coefficient (Δ_1), transitivity ratio (Δ_0),. . .
- Social networks are dynamic

Outline

Introduction

Database

Definition (Schema)

A schema $\mathbf{X} = (X_1, \ldots, X_n)$ is a tuple of variables with discrete domain $Dom(\mathbf{X}) = Dom(X_1) \times \ldots \times Dom(X_n)$.

Definition (Relation)

A relation K is a function $K : Dom(\mathbf{X}) \mapsto \mathbb{Z}$.

- $x \in K \iff K(x) \neq 0$
- $|K| = |\{x \mid x \in K\}|$

Definition (Database)

A database D is a set of relations. $|D| = \sum_{K \in D} |K|$

Relational Algebra

Definition (Projection)

$\pi_F x$ is the projection of x onto the variables in the tuple F.

Definition (Selection)

$\sigma_{F=t} K = \{x \in K \mid \pi_F x = t\}$

Queries

Relations $R[A, B], S[B, C], T[C, A]$.

Ternary triangle query:

$$\Delta_3(a, b, c) = R(a, b) \cdot S(b, c) \cdot T(c, a)$$

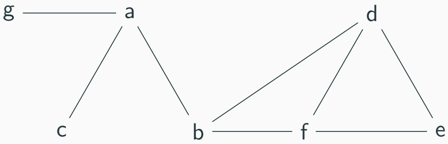
Binary: $\Delta_2(a, b) = \sum_{c \in Dom(C)} R(a, b) \cdot S(b, c) \cdot T(c, a)$

Unary: $\Delta_1(a) = \sum_{b \in Dom(B)} \sum_{c \in Dom(C)} R(a, b) \cdot S(b, c) \cdot T(c, a)$

Nullary:
$\Delta_0() = \sum_{a \in Dom(A)} \sum_{b \in Dom(B)} \sum_{c \in Dom(C)} R(a, b) \cdot S(b, c) \cdot T(c, a)$

Result: $\Delta_{\dots}(\dots)$ for all free variables (if $\neq 0$)

Example Queries ($R = S = T$)



Δ_0
2

a	b	c	Δ_3
b	d	f	1
d	e	f	1

a	Δ_1
b	1
d	2
f	2
e	1

a	b	Δ_2
b	d	1
b	f	1
d	f	2
d	e	1
e	f	1

Incremental View Maintenance (for Δ_0)

For a single-tuple update $\delta R = \{(\alpha, \beta) \mapsto m\}$ ($m \in \mathbb{Z}^*$):

$$\Delta_0() + \delta\Delta_0() = \Delta_0() + \delta R(\alpha, \beta) \cdot \underbrace{\sum_{c \in Dom(C)} S(\beta, c) \cdot T(c, \alpha)}_{\mathcal{O}(|D|) \text{ if linear \#C values, could be precomputed}}$$

Precompute auxiliary view

$$V_{ST}(b, a) = \sum_{c \in Dom(C)} S(b, c) \cdot T(c, a)$$

- $\mathcal{O}(1)$ delta query computation
- $\mathcal{O}(|D|)$ view maintenance
- $\mathcal{O}(|D|^2)$ space for V_{ST}

Key idea: Partiton relation s.t. $\#C$-values is sublinear

$\mathbf{IVM}^{\epsilon}$ for $\Delta_0()$

Partitions

Definition (Partition)

For relation K over $\mathbf{X}$, X in $\mathbf{X}$, threshold θ, K is partitioned in K^H, K^L, if

union $K(x) = K^H(x) + K^L(x), x \in Dom(\mathbf{X})$

domain partiton $\pi_X K^H \cap \pi_X K^L = \emptyset$

heavy part for any X-value: $|\sigma_{X=x} K^H| \geq \frac{1}{2}\theta$

light part for any X-value: $|\sigma_{X=x} K^L| < \frac{3}{2}\theta$

Maintaining Δ_0

We want to maintain

$$\begin{aligned}\Delta_0() &= \sum_{a,b,c} R(a,b) \cdot S(b,c) \cdot T(c,a) \\ &= \sum_{r,s,t \in \{H,L\}} \underbrace{\sum_{a,b,c} R^r(a,b) \cdot S^s(b,c) \cdot T^t(c,a)}_{=\Delta_0^{rst}()} \\ &= \sum_{r,s,t \in \{H,L\}} \Delta_0^{rst}()\end{aligned}$$

where $R[\mathbf{A}, B], S[\mathbf{B}, C], T[\mathbf{C}, A]$ are partitioned on A, B, C.

Maintain Δ_0^{rst} using different strategies:

- Compute $\delta\Delta_0^{rst}$ directly
- Compute $\delta\Delta_0^{rst}$ using auxiliary materialized views

IVM^ϵ State

Definition (IVM^ϵ State)

For $D = \{R, S, T\}$, $\epsilon \in [0, 1]$, an IVM^ϵ state is (ϵ, N, P, V) with:

1. $\frac{1}{4}N \leq |D| < N$ ($N = \Theta(|D|)$)
2. P: set of partitions of R, S, T with $\theta = N^\epsilon$
3. V: set of materialized views

Insight

- At most $\frac{N}{\frac{1}{2}N^\epsilon} = 2N^{1-\epsilon}$ distinct A-values can exist in R^H
- Any A-value in R^L appears less than $\frac{3}{2}N^\epsilon$ times

Maintaining $\delta\Delta_0^{rst}$

Update $\delta R^r = \{(\alpha, \beta) \mapsto m\}$ affects either R^H or R^L, i.e., only four partitions $\Delta_0^{rHH}, \Delta_0^{rHL}, \Delta_0^{rLH}, \Delta_0^{rLL}$ are maintained.

$$\delta\Delta_0^{rHH} = m \cdot \sum_c S^H(\beta, c) \cdot T^H(c, \alpha)$$

$\leq 2N^{1-\epsilon}$ distinct C-values, summing takes $\mathcal{O}(N^{1-\epsilon}) = \mathcal{O}(|D|^{1-\epsilon})$

$$\delta\Delta_0^{rLL} = m \cdot \sum_c S^L(\beta, c) \cdot T^L(c, \alpha)$$

$< \frac{3}{2}N^{\epsilon}$ tuples have given β, summing takes $\mathcal{O}(N^{\epsilon}) = \mathcal{O}(|D|^{\epsilon})$

$\delta\Delta_0^{rLH}$: like $\delta\Delta_0^{rHH}$ or $\delta\Delta_0^{rLL}$, i.e., $\mathcal{O}(|D|^{\min\{\epsilon, 1-\epsilon\}})$

What about $\delta\Delta_0^{rHL}$?

Problem: Number of C-values could be linear in $|D|$ for $\delta\Delta_0^{rHL}$

Solution:

- Materialized view $V_{ST}(b,a) = \sum_c S^H(b,c) \cdot T^L(c,a)$
- $\delta\Delta_0^{rHL} = m \cdot V_{ST}(\beta,\alpha)$ in $\mathcal{O}(1)$

$\delta S^H = \{(\beta,\gamma) \mapsto m\}$ and $\delta T^L = \{(\gamma,\alpha) \mapsto m\}$ require view maintenance

Visualization of $\delta\Delta_0^{rHL}$ maintenance

B	C	m
$\dots$	$\dots$	$\dots$
β	γ	m
$\dots$	$\dots$	$\dots$

(a) S^H

C	A	m
$\dots$	$\dots$	$\dots$
γ	$\dots$	$\dots$
γ	$\dots$	$\dots$
γ	$\dots$	$\dots$
$\dots$	$\dots$	$\dots$

(b) T^L

Table 1: Update $\delta S^H = \{(\beta, \gamma) \mapsto m\}$

δS^H: fixed γ, at most $\frac{3}{2}N^\epsilon$ tuples have $C = \gamma$ in T^L

Visualization of $\delta\Delta_0^{rHL}$ maintenance

B	C	m
$\dots$	$\dots$	$\dots$
$\dots$	γ	$\dots$
$\dots$	γ	$\dots$
$\dots$	γ	$\dots$
$\dots$	$\dots$	$\dots$

(a) S^H

C	A	m
$\dots$	$\dots$	$\dots$
γ	α	$\dots$
$\dots$	$\dots$	$\dots$

(b) T^L

Table 2: Update $\delta T^L = \{(\gamma, \alpha) \mapsto m\}$

δT^L: fixed γ, at most $2N^{1-\epsilon}$ distinct B-values in S^H

Space Complexity for IVM^{ϵ} state (ϵ, N, P, V)

Obviously, $\epsilon, N = \mathcal{O}(1)$, $|P| = |D|$

Three auxiliary materialized views are used:

1. $V_{ST}(b, a) = \sum_c S^H(b, c) \cdot T^L(c, a)$
2. $V_{RS}(a, c) = \ldots$
3. $V_{TR}(c, b) = \ldots$

For V_{ST}:

$$|V_{ST}| \leq \min\{N \cdot \frac{3}{2} N^{\epsilon}, N \cdot 2N^{1-\epsilon}\}$$
$$= \mathcal{O}(|D|^{1+\min\{\epsilon, 1-\epsilon\}})$$

Summary for $\Delta_0()$

Theorem

For a database D, $\epsilon \in [0, 1]$*, IVM*$^\epsilon$ *maintains* Δ_0 *for single-tuple updates with:*

preprocessing $\mathcal{O}(|D|^{\frac{3}{2}})$ *[2, 3, 4]*

update time $\mathcal{O}(|D|^{\max\{\epsilon, 1-\epsilon\}})$

space $\mathcal{O}(|D|^{1+\min\{\epsilon, 1-\epsilon\}})$

enumeration delay $\mathcal{O}(1)$

IVM$^{\epsilon}$ for $\Delta_3(a, b, c)$ (sketch)

Decomposing Δ_3

We want to maintain

$$\begin{aligned}\Delta_3(a,b,c) &= R(a,b)\cdot S(b,c)\cdot T(c,a)\\ &= \Delta_3^{HHH}(a,b,c) + \Delta_3^{LLL}(a,b,c)\\ &+ \Delta_3^{\boxminus HL}(a,b,c) + \Delta_3^{H\boxminus L}(a,b,c)\\ &+ \Delta_3^{HL\boxminus}(a,b,c)\end{aligned}$$

We focus on Δ_3^{HHH} and $\Delta_3^{\boxminus HL}$.

Maintaining Δ^{HHH}

Δ^{HHH} is *materialized*.

A	B	C	Δ_3
$\dots$	$\dots$	$\dots$	$\dots$
α	β	c_1	$\dots$
α	β	c_2	$\dots$
α	β	$\dots$	$\dots$
α	β	c_k	$\dots$
$\dots$	$\dots$	$\dots$	$\dots$

Update $\delta R^H = \{(\alpha, \beta) \mapsto m\}$ for
$\Delta_3^{HHH}(a, b, c) = R^H(a, b) \cdot S^H(b, c) \cdot T^H(c, a)$

For fixed α, T^H has at most $\frac{3}{2}N^{1-\epsilon}$ C-values, i.e., $\mathcal{O}(|D|^{1-\epsilon})$

Space complexity: $\mathcal{O}(|D|^{\frac{3}{2}})$

Maintaining $\Delta^{\boxminus HL}$

Update $\delta R^H = \{(\alpha, \beta) \mapsto m\}$ for
$\Delta_3^{\boxminus HL}(a,b,c) = \sum_{r \in \{H,L\}} R^r(a,b) \cdot S^H(b,c) \cdot T^L(c,a)$

Direct Computation: Possibly $\mathcal{O}(|D|)$ affected rows.

Auxiliary View: $V_{ST}(b,c,a) = S^H(b,c) \cdot T^L(c,a)$ does not help

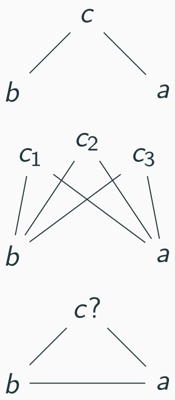
Solution: factorized evaluation

Maintaining $\Delta^{\boxminus HL}$ using hierarchical views

$$V_{ST}(b, c, a) = S^{H}(b, c) \cdot T^{L}(c, a)$$

$$\hat{V}_{ST}(b, a) = \sum_{c} V_{ST}(b, c, a)$$

$$V^{\boxminus HL}(a, b) = \sum_{r \in \{H, L\}} R^{r}(a, b) \cdot \hat{V}_{ST}(b, a)$$



Enumeration: For all $(a, b) \in V^{\boxminus HL}$, find c in V_{ST}, $\mathcal{O}(1)$ delay

Maintenance: $\mathcal{O}(|D|^{\max\{\epsilon, 1-\epsilon\}})$ (like $\Delta_0()$)

Space: $\mathcal{O}(|D|^{1+\min\{\epsilon, 1-\epsilon\}})$ (like $\Delta_0()$)

Summary for $\Delta_0()$ and $\Delta_3(a, b, c)$

Theorem

For a database D, $\epsilon \in [0, 1]$, IVM^ϵ maintains Δ_0 and Δ_3 for single-tuple updates with $\mathcal{O}(|D|^{\frac{3}{2}})$ preprocessing time, $\mathcal{O}(|D|^{\max\{\epsilon,1-\epsilon\}})$ update time, $\mathcal{O}(1)$ enumeration delay, and space

Δ_0	$\mathcal{O}(|D|^{1+\min\{\epsilon,1-\epsilon\}})$
Δ_3	$\mathcal{O}(|D|^{\frac{3}{2}})$

Results for Δ_1, Δ_2 are similar.

Rebalancing and Amortized Analysis (sketch)

Major Rebalancing

Reminder

For $D = \{R, S, T\}$, $\epsilon \in [0, 1]$, an IVM^{ϵ} state is (ϵ, N, P, V) with:

1. $\frac{1}{4}N \leq |D| < N$ $(N = \Theta(|D|))$
2. P: a set of partitions of R, S, T with $\theta = N^{\epsilon}$
3. $\ldots$

Updates might change $|D|$; repartitioning and preprocessing takes $\mathcal{O}(|D|^{\frac{3}{2}})$.

Halving and doubling trick: major rebalancing at most every $\approx \frac{1}{4}N = \Theta(|D|)$ updates.

"$\frac{\mathcal{O}(|D|^{\frac{3}{2}})}{\Theta(|D|)} = \mathcal{O}(|D|^{\frac{1}{2}})$" amortized update time

Minor Rebalancing

Reminder

For relation K over $\mathbf{X}$, X in $\mathbf{X}$, threshold $\theta = N^{\epsilon}$, K is partitioned in K^H, K^L, if

heavy part for any $x \in \pi_X K^H$: $|\sigma_{X=x} K^H| \geq \frac{1}{2}\theta$

light part for any $x \in \pi_X K^L$: $|\sigma_{X=x} K^L| < \frac{3}{2}\theta$

- Updates might change $|\sigma_{X=x} K^H|$ and $|\sigma_{X=x} K^L|$
- Rebalance: $\mathcal{O}(|D|^{\epsilon})$ tuples are deleted/reinserted
- Each update takes $\mathcal{O}(|D|^{\max\{\epsilon, 1-\epsilon\}})$
- At least $\frac{1}{2}\theta = \Theta(|D|^{\epsilon})$ updates between rebalances

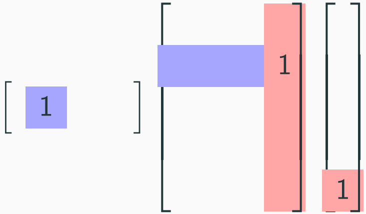
"$\frac{\mathcal{O}(|D|^{\epsilon + \max\{\epsilon, 1-\epsilon\}})}{\Theta(|D|^{\epsilon})} = \mathcal{O}(|D|^{\max\{\epsilon, 1-\epsilon\}})$" amortized update time

Optimality (sketch)

Optimality

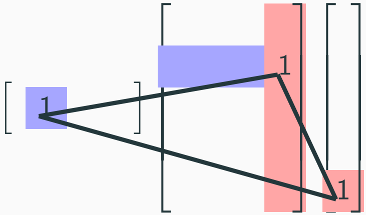
Online Vector-Matrix-Vector Multiplication Conjecture [1]

Given n pairs of n-dimensional boolean vectors (u_k, v_k) and a $n \times n$ matrix M, $(u_k)^T M v_k$ can not be computed one after the other in $\mathcal{O}(n^{3-\gamma})$ $(\gamma > 0)$



$$(u_i)^T M v_i = 1 \Leftrightarrow \exists i, j : u_k(i) = M(i, j) = v_k(j) = 1$$

Optimality



$$(u_i)^T M v_i = 1 \Leftrightarrow \exists i, j : u_k(i) = M(i, j) = v_k(j) = 1$$
$$\Leftrightarrow \exists i, j : R(a, i) \cdot S(i, j) \cdot T(j, a) = 1$$

Unless *OMv* fails, there is no algorithm that maintains $\Delta_{\dots}$ with $\mathcal{O}(|D|^{\frac{1}{2}-\gamma})$ update time and $\mathcal{O}(|D|^{1-\gamma})$ enumeration delay ($\gamma > 0$).

Conclusion

Conclusion

Theorem

For a database D, $\epsilon \in [0, 1]$*, IVM*$^\epsilon$ *maintains* Δ_0 *and* Δ_3 *with* $\mathcal{O}(|D|^{\frac{3}{2}})$ *preprocessing time,* $\mathcal{O}(|D|^{\max\{\epsilon, 1-\epsilon\}})$ *amortized update time,* $\mathcal{O}(1)$ *enumeration delay, and space*

$$\Delta_0 \quad \mathcal{O}(|D|^{1+\min\{\epsilon, 1-\epsilon\}})$$
$$\Delta_3 \quad \mathcal{O}(|D|^{\frac{3}{2}})$$

Furthermore, IVM$^\epsilon$ *is Pareto worst-case optimal for* $\epsilon = \frac{1}{2}$*, unless OMv fails.*

References

[1] Monika Henzinger et al. "Unifying and Strengthening Hardness for Dynamic Problems via the Online Matrix-Vector Multiplication Conjecture". In: *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*. STOC '15. Portland, Oregon, USA: Association for Computing Machinery, 2015, pp. 21–30. ISBN: 9781450335362. DOI: 10.1145/2746539.2746609. URL: https://doi.org/10.1145/2746539.2746609.

[2] Hung Q Ngo et al. "Worst-case optimal join algorithms". In: *Journal of the ACM (JACM)* 65.3 (2018), pp. 1–40.

[3] Todd L Veldhuizen. “Leapfrog triejoin: A simple, worst-case optimal join algorithm”. In: *arXiv preprint arXiv:1210.0481* (2012).

[4] Todd L. Veldhuizen. “Triejoin: A Simple, Worst-Case Optimal Join Algorithm”. In: *ICDT*. 2014.