Equivalences Between Triangle and Range Query Problems

Lech Duraj Krzysztof Kleiner **Adam Polak**

Jagiellonian University in Kraków

Virginia Vassilevska Williams

Massachusetts Institute of Technology

Symposium on Discrete Algorithms 2020

Salt Lake City, January 5th

EdgeTriangleCounting

In: an m-edge graph G

Out: for each edge $e \in G$ the number of triangles containing e

EdgeTriangleCounting

In: an m-edge graph G

Out: for each edge $e \in G$ the number of triangles containing e

RangeInversionsQuery

In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

EdgeTriangleCounting

In: an m-edge graph G

Out: for each edge $e \in G$ the number of triangles containing e

RangeInversionsQuery

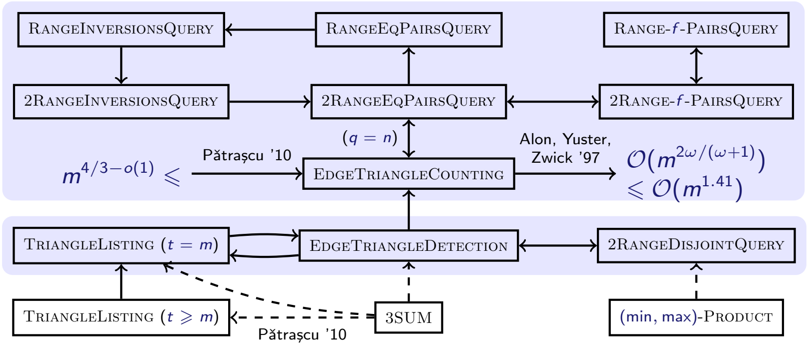
In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \dots, [l_q, r_q]$

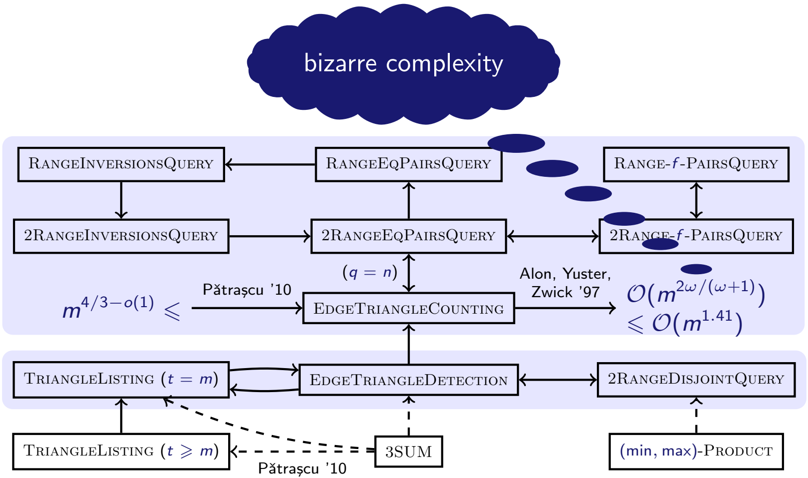
Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

EdgeTriangleCounting $\leftrightarrow$ RangeInversionsQuery

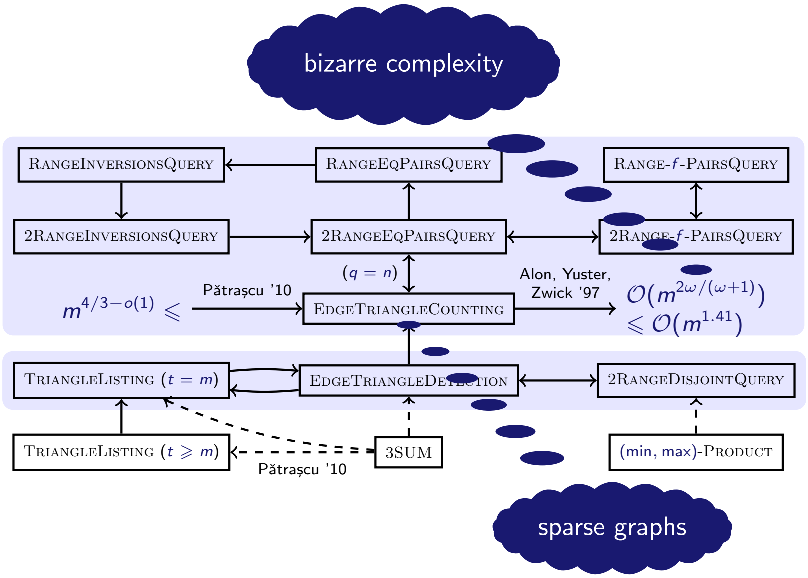
If EdgeTriangleCounting can be solved in $T(m)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n+q))$ time.

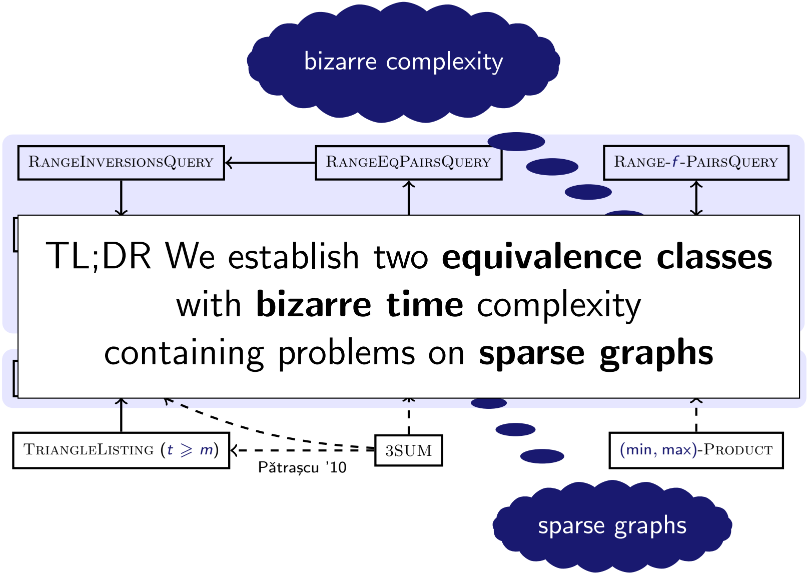
If RangeInversionsQuery can be solved in $T(n+q)$ time, then EdgeTriangleCounting can be solved in $\widetilde{\mathcal{O}}(T(m))$ time.


RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing $(t = m)$
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing $(t \geqslant m)$
3SUM
Pătrașcu '10
(min, max)-Product



bizarre complexity
RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)})$
$\leqslant \mathcal{O}(m^{1.41})$
TriangleListing $(t = m)$
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing $(t \geqslant m)$
3SUM
$(\min, \max)$-Product
Pătrașcu '10


bizarre complexity
RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)})$
$\leqslant \mathcal{O}(m^{1.41})$
TriangleListing $(t = m)$
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing $(t \geqslant m)$
3SUM
(min, max)-Product
Pătrașcu '10
sparse graphs



TL;DR We establish two **equivalence classes** with **bizarre time** complexity containing problems on **sparse graphs**

RangeInversionsQuery

In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

RangeInversionsQuery

In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

- Mo's algorithm: $\widetilde{\mathcal{O}}(n\sqrt{q})$ time

RANGEINVERSIONSQUERY

In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

- Mo's algorithm: $\widetilde{\mathcal{O}}(n\sqrt{q})$ time
- $\widetilde{\mathcal{O}}(n^{1.5})$ for $q = \Theta(n)$

RangeInversionsQuery

In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \dots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

- Mo's algorithm: $\widetilde{\mathcal{O}}(n\sqrt{q})$ time
- $\widetilde{\mathcal{O}}(n^{1.5})$ for $q = \Theta(n)$
- Is $\widetilde{\mathcal{O}}(n^{1.49})$ possible?

RangeInversionsQuery

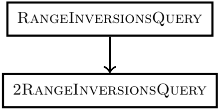
In: an integer array $A[1..n]$, and queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: how many inversions in subarray $A[l_i..r_i]$, i.e. $\#\{(x, y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] > A[y]\} = ?$

2RangeInversionsQuery

In: an integer array $A[1..n]$, and queries $([l_i', r_i'], [l_i'', r_i''])_{i \in [q]}$ $(r_i' < l_i'')$

Out: for each $i \in [q]$: $\#\{(x, y) : x \in [l_i', r_i'], y \in [l_i'', r_i''] \text{ and } A[x] > A[y]\} = ?$


RangeInversionsQuery
2RangeInversionsQuery

RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

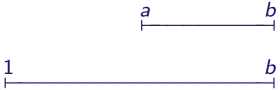
$\mathrm{inv}\left([a, b]\right) =$

RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

$\text{inv}\left([a, b]\right) =$

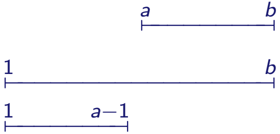
$\text{inv}\left([1, b]\right)$



RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

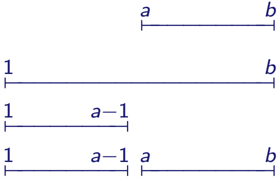
$$\mathrm{inv}\big([a, b]\big) = \mathrm{inv}\big([1, b]\big) - \mathrm{inv}\big([1, a-1]\big)$$



RangeInversionsQuery $\rightarrow$ 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

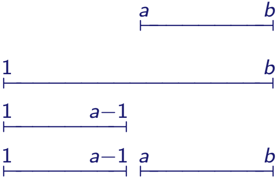
$$\begin{aligned} \operatorname{inv}\big([a, b]\big) = \\ \operatorname{inv}\big([1, b]\big) \\ - \operatorname{inv}\big([1, a-1]\big) \\ - \operatorname{inv}\big([1, a-1], [a, b]\big) \end{aligned}$$



RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

$$\begin{aligned}\operatorname{inv}([a, b]) = {} & \operatorname{inv}([1, b]) \\ & - \operatorname{inv}([1, a-1]) \\ & - \operatorname{inv}([1, a-1], [a, b])\end{aligned}$$

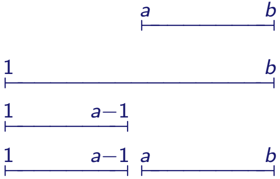


RangeInversionsQuery

RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n, q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n, q))$ time

$$\begin{aligned} \operatorname{inv}\big([a, b]\big) = & \\ \operatorname{inv}\big([1, b]\big) & \\ - \operatorname{inv}\big([1, a-1]\big) & \\ - \operatorname{inv}\big([1, a-1], [a, b]\big) & \end{aligned}$$



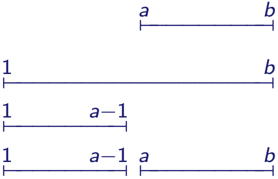
RangeInversionsQuery

preprocess $\operatorname{inv}\big([1, i]\big)$ for all i's, in $\mathcal{O}(n \log n)$

RangeInversionsQuery → 2RangeInversionsQuery

If 2RangeInversionsQuery can be solved in $T(n,q)$ time,
then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n,q))$ time

$$\begin{aligned} \mathrm{inv}\,([a,b]) = &\ \mathrm{inv}\,([1,b]) \\ &- \mathrm{inv}\,([1,a-1]) \\ &- \mathrm{inv}\,([1,a-1],[a,b]) \end{aligned}$$



RangeInversionsQuery

preprocess $\mathrm{inv}\,([1,i])$ for all i's, in $\mathcal{O}(n \log n)$

2RangeInversionsQuery

RangeInversionsQuery → 2RangeInversionsQuery

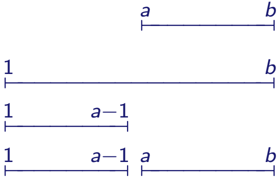
If 2RangeInversionsQuery can be solved in $T(n,q)$ time, then RangeInversionsQuery can be solved in $\widetilde{\mathcal{O}}(T(n,q))$ time

$\mathrm{inv}\left([a,b]\right) =$ — RangeInversionsQuery

$\mathrm{inv}\left([1,b]\right)$

$-\,\mathrm{inv}\left([1,a-1]\right)$

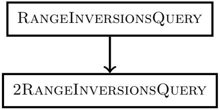
$-\,\mathrm{inv}\left([1,a-1],[a,b]\right)$

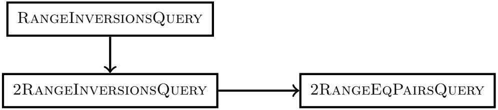


preprocess $\mathrm{inv}\left([1,i]\right)$ for all i's, in $\mathcal{O}(n \log n)$

2RangeInversionsQuery

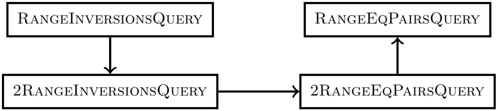
Total time: $\mathcal{O}(n \log n) + T(n,q) = \widetilde{\mathcal{O}}(T(n,q))$ □


RANGEINVERSIONSQUERY
2RANGEINVERSIONSQUERY



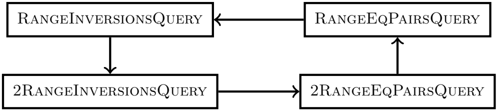
$$\#\{(x,y) : x \in [l'_i, r'_i], y \in [l''_i, r''_i] \text{ and } A[x] = A[y]\} = ?$$

$$\#\{(x,y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] = A[y]\} = ?$$

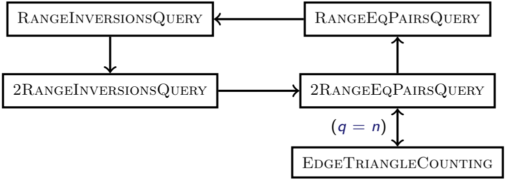


$$\#\{(x,y) : x \in [l'_i, r'_i], y \in [l''_i, r''_i] \text{ and } A[x] = A[y]\} = ?$$

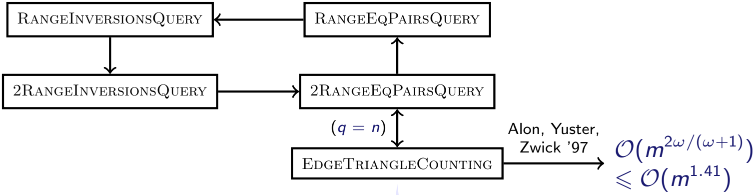
$$\#\big\{(x,y) : l_i \leqslant x < y \leqslant r_i \text{ and } A[x] = A[y]\big\} = ?$$



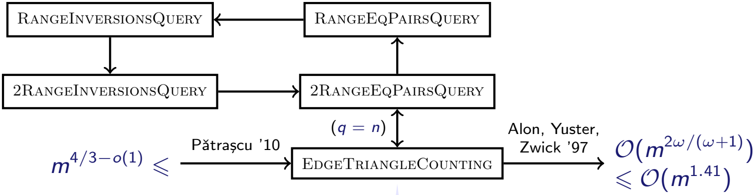
$$\#\big\{(x,y) : x \in [l_i', r_i'], y \in [l_i'', r_i''] \text{ and } A[x] = A[y]\big\} = ?$$



In: a (sparse) graph G
Out: for every edge $e \in G$
$\#$ of triangles containing e



In: a (sparse) graph G
Out: for every edge $e \in G$
$\#$ of triangles containing e

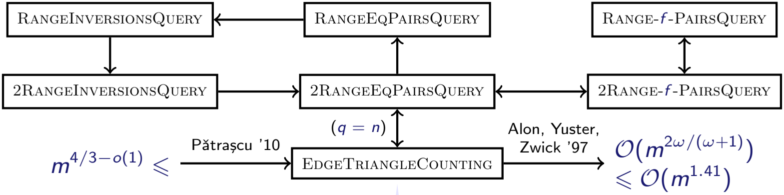


In: a (sparse) graph G

Out: for every edge $e \in G$

\# of triangles containing e

$\sum_{l_i \leqslant x < y \leqslant r_i} f(A[x], A[y]) = ?$



In: a (sparse) graph G

Out: for every edge $e \in G$ $\#$ of triangles containing e

RANGE-f-PAIRSQUERY

$f : \mathbb{Z}^2 \to \mathbb{Z}$, **piecewise polynomial non-axis-orthogonal**

In: an integer array $A[1..n]$, and
queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: $\sum_{l_i \leqslant x < y \leqslant r_i} f(A[x], A[y]) = ?$

RANGE-f-PAIRSQUERY

$f : \mathbb{Z}^2 \to \mathbb{Z}$, **piecewise polynomial non-axis-orthogonal**

In: an integer array $A[1..n]$, and
queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: $\sum_{l_i \leqslant x < y \leqslant r_i} f(A[x], A[y]) = ?$

Def. f is piecewise polynomial [LUW'19]

- $f(x, y) = \sum_k P_k(x, y) \cdot \mathbb{1}\left[A_k x + B_k y + C_k > 0\right]$

RANGE-f-PAIRSQUERY

$f : \mathbb{Z}^2 \to \mathbb{Z}$, **piecewise polynomial non-axis-orthogonal**

In: an integer array $A[1..n]$, and
queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

Out: for each $i \in [q]$: $\sum_{l_i \leqslant x < y \leqslant r_i} f(A[x], A[y]) = ?$

Def. f is piecewise polynomial non-axis-orthogonal [LUW'19]

- $f(x, y) = \sum_k P_k(x, y) \cdot \mathbb{1}\left[A_k x + B_k y + C_k > 0\right]$
- $\neg\forall_i \; A_i = 0 \vee B_i = 0$

RANGE-f-PAIRSQUERY

$f : \mathbb{Z}^2 \to \mathbb{Z}$, **piecewise polynomial non-axis-orthogonal**

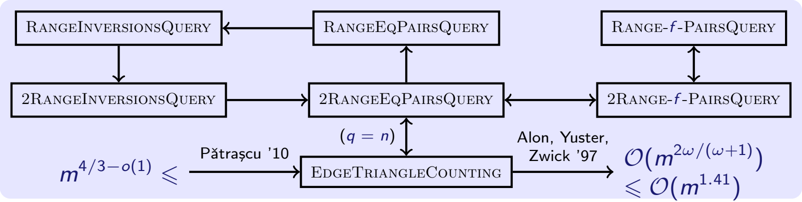
In: an integer array $A[1..n]$, and
queries $[l_1, r_1], [l_2, r_2], \ldots, [l_q, r_q]$

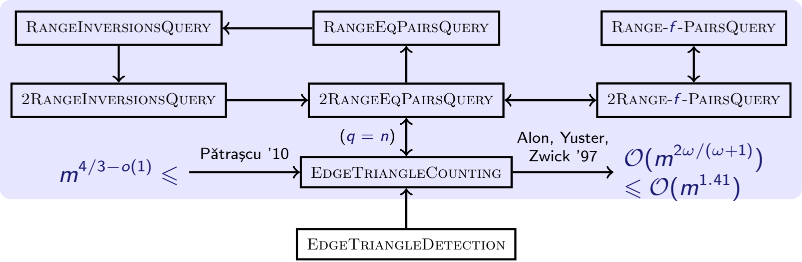
Out: for each $i \in [q]$: $\sum_{l_i \leqslant x < y \leqslant r_i} f(A[x], A[y]) = ?$

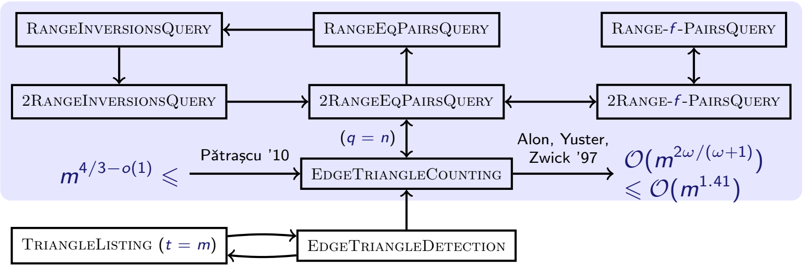
Def. f is piecewise polynomial non-axis-orthogonal [LUW'19]

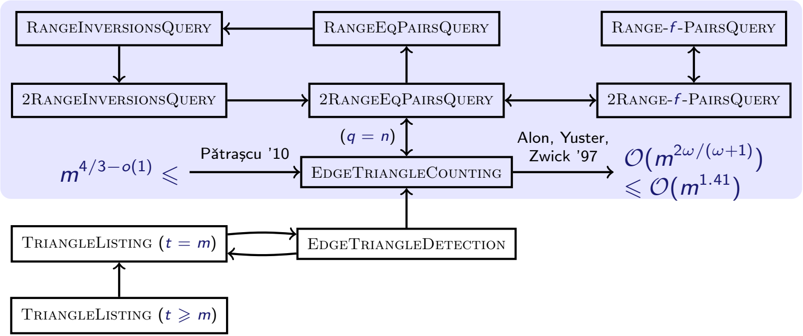
- $f(x, y) = \sum_k P_k(x, y) \cdot \mathbb{1}\left[A_k x + B_k y + C_k > 0\right]$
- $\neg\forall_i \; A_i = 0 \vee B_i = 0$

E.g. $=$, $\leqslant$, min, max, ℓ_1, ℓ_{2p+1}, $\mathbb{1}[x - y > \delta]$, $\max(0, x - y)$

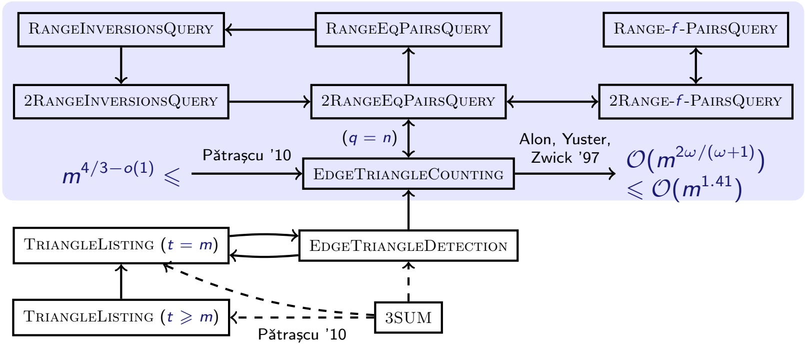

RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Pătraşcu '10
Alon, Yuster, Zwick '97
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$


RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Pătrașcu '10
Alon, Yuster, Zwick '97
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)})$
$\leqslant \mathcal{O}(m^{1.41})$
EdgeTriangleDetection

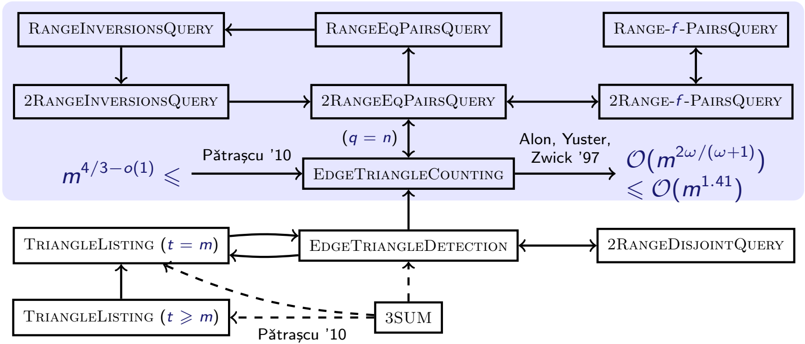

RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
(q = n)
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing (t = m)
EdgeTriangleDetection

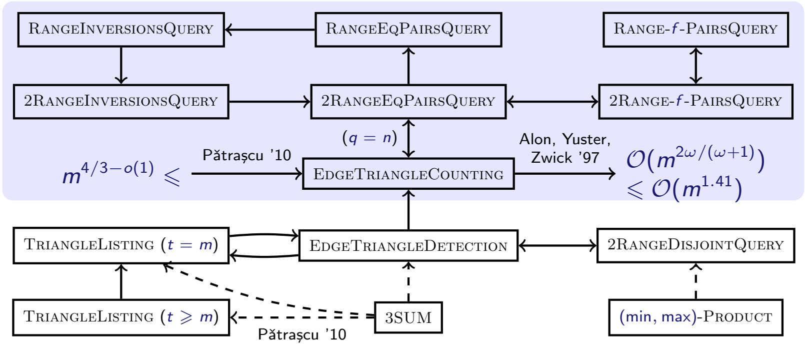


RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
($q = n$)
Pătrașcu '10
Alon, Yuster, Zwick '97
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing ($t = m$)
EdgeTriangleDetection
TriangleListing ($t \geqslant m$)

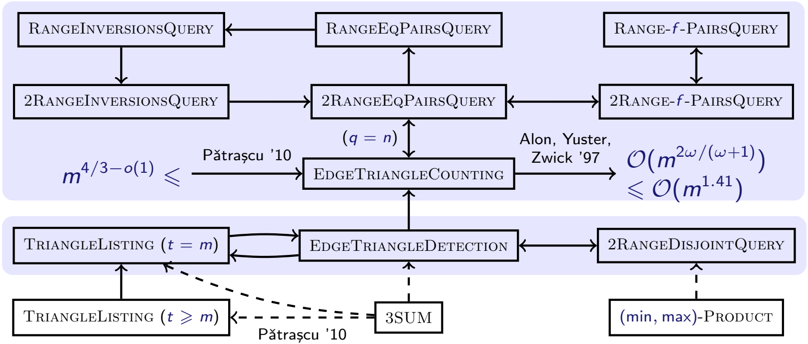


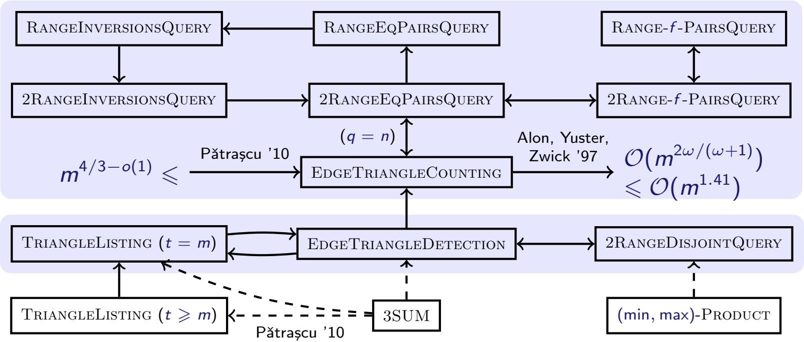
RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
($q = n$)
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing ($t = m$)
EdgeTriangleDetection
TriangleListing ($t \geqslant m$)
3SUM
Pătrașcu '10


RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
(q = n)
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing ($t = m$)
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing ($t \geqslant m$)
3SUM
Pătrașcu '10



RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing $(t = m)$
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing $(t \geqslant m)$
3SUM
(min, max)-Product
Pătrașcu '10


RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
(q = n)
Alon, Yuster, Zwick '97
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing (t = m)
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing (t ⩾ m)
3SUM
(min, max)-Product
Pătrașcu '10



RangeInversionsQuery
RangeEqPairsQuery
Range-f-PairsQuery
2RangeInversionsQuery
2RangeEqPairsQuery
2Range-f-PairsQuery
$(q = n)$
Pătrașcu '10
$m^{4/3-o(1)} \leqslant$
EdgeTriangleCounting
Alon, Yuster, Zwick '97
$\mathcal{O}(m^{2\omega/(\omega+1)}) \leqslant \mathcal{O}(m^{1.41})$
TriangleListing $(t = m)$
EdgeTriangleDetection
2RangeDisjointQuery
TriangleListing $(t \geqslant m)$
3SUM
Pătrașcu '10
(min, max)-Product

Thank you!