Dynamic and Streaming Algorithms for Union Volume Estimation

Sujoy Bhore Karl Bringmann Timothy Chan Yanheng Wang

Dynamic and Streaming Algorithms for Union Volume Estimation

$X_1 \quad \cdots \quad X_n$

Dynamic and Streaming Algorithms for Union Volume Estimation

$X_1 \quad \cdots \quad X_n$

e.g., triangles, balls, polytopes

Dynamic and Streaming Algorithms for Union Volume Estimation

$$X_1 \cup \cdots \cup X_n$$

e.g., triangles, balls, polytopes

Dynamic and Streaming Algorithms for Union Volume Estimation

$$\mathrm{vol}(X_1 \cup \cdots \cup X_n) \approx ?$$

e.g., triangles, balls, polytopes

Dynamic and Streaming Algorithms for Union Volume Estimation

$$\mathrm{vol}(X_1 \cup \cdots \cup X_n) \approx ?$$

e.g., triangles, balls, polytopes
inserted and deleted on the fly

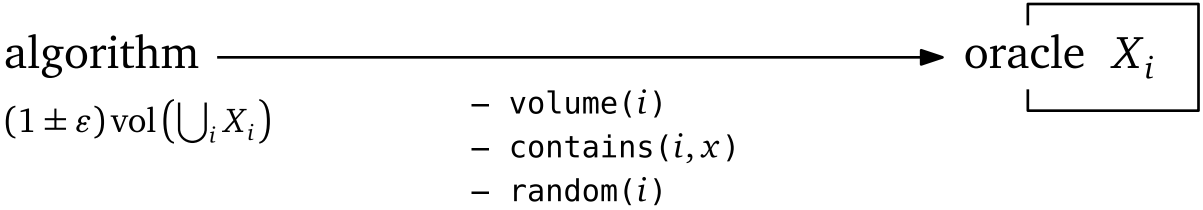
Algorithms in the Oracle Model [Karp, Luby ’83]

algorithm

$(1 \pm \varepsilon)\,\mathrm{vol}\left(\bigcup_i X_i\right)$

oracle X_i

Algorithms in the Oracle Model [Karp, Luby ’83]



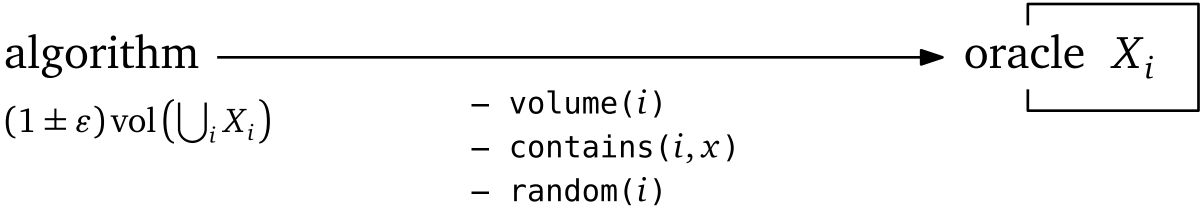
algorithm

$(1 \pm \varepsilon)\operatorname{vol}\left(\bigcup_i X_i\right)$

- `volume(`i`)`
- `contains(`i, x`)`
- `random(`i`)`

oracle X_i

Algorithms in the Oracle Model [Karp, Luby ’83]



algorithm $\longrightarrow$ oracle X_i

$(1 \pm \varepsilon)\,\mathrm{vol}\left(\bigcup_i X_i\right)$

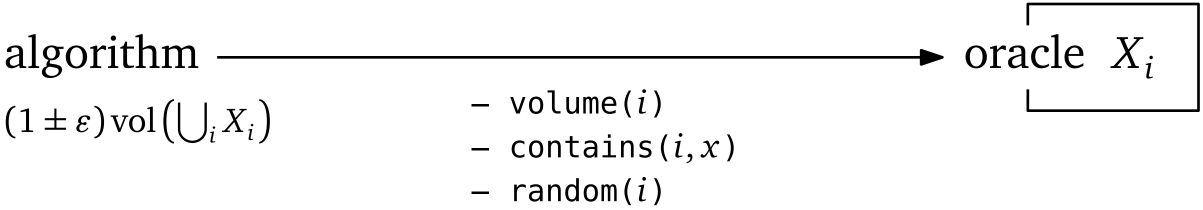
- `volume(`i`)`
- `contains(`i, x`)`
- `random(`i`)`

Classic:

$O(n/\varepsilon^2)$ algorithm [Karp, Luby, Madras ’89]

$\Omega(n/\varepsilon^2)$ lower bound [Bringmann, Larsen, Nusser, Rotenberg, Wang ’26]

Algorithms in the Oracle Model [Karp, Luby '83]



algorithm $\longrightarrow$ oracle X_i

$(1 \pm \varepsilon)\,\mathrm{vol}\left(\bigcup_i X_i\right)$

- `volume(`i`)`
- `contains(`i, x`)`
- `random(`i`)`

Classic:

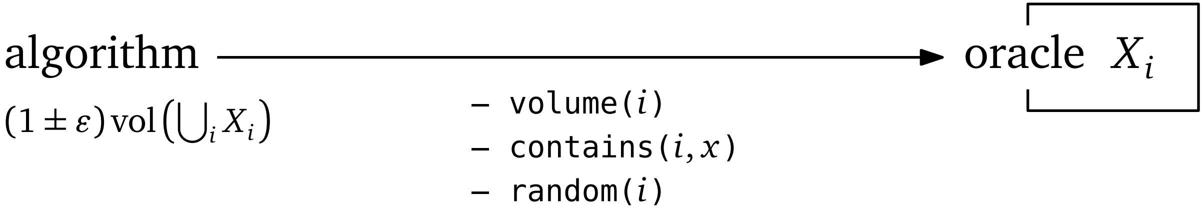
$O(n/\varepsilon^2)$ algorithm [Karp, Luby, Madras '89]

$\Omega(n/\varepsilon^2)$ lower bound [Bringmann, Larsen, Nusser, Rotenberg, Wang '26]

Incremental:

$\widetilde{O}(1/\varepsilon^2)$ update time and space [Meel, Vinodchandran, Chakraborty '21]

Algorithms in the Oracle Model [Karp, Luby ’83]



algorithm $\longrightarrow$ oracle X_i

$(1 \pm \varepsilon)\,\mathrm{vol}\left(\bigcup_i X_i\right)$

- `volume(`i`)`
- `contains(`i, x`)`
- `random(`i`)`

Classic:

$O(n/\varepsilon^2)$ algorithm [Karp, Luby, Madras ’89]

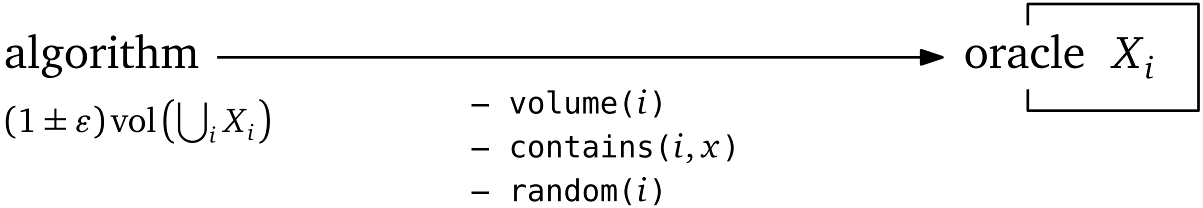
$\Omega(n/\varepsilon^2)$ lower bound [Bringmann, Larsen, Nusser, Rotenberg, Wang ’26]

Incremental:

$\widetilde{O}(1/\varepsilon^2)$ update time and space [Meel, Vinodchandran, Chakraborty ’21]

Decremental?

Algorithms in the Oracle Model [Karp, Luby ’83]



- `volume(`i`)`
- `contains(`i, x`)`
- `random(`i`)`

Classic:

$O(n/\varepsilon^2)$ algorithm [Karp, Luby, Madras ’89]

$\Omega(n/\varepsilon^2)$ lower bound [Bringmann, Larsen, Nusser, Rotenberg, Wang ’26]

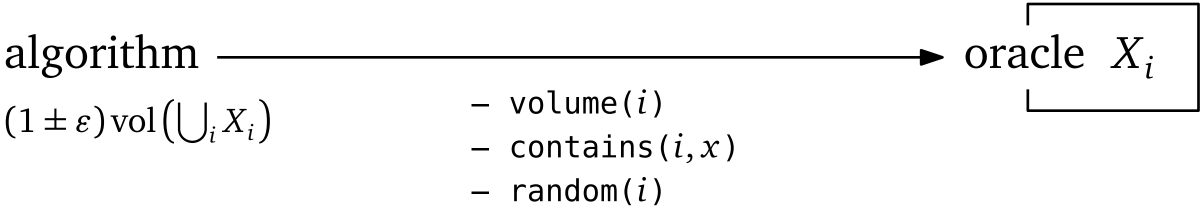
Incremental:

$\widetilde{O}(1/\varepsilon^2)$ update time and space [Meel, Vinodchandran, Chakraborty ’21]

Decremental?

Fully dynamic?

Algorithms in the Oracle Model [Karp, Luby ’83]



algorithm $\longrightarrow$ oracle X_i

$(1\pm\varepsilon)\,\mathrm{vol}\left(\bigcup_i X_i\right)$

- `volume`(i)
- `contains`(i,x)
- `random`(i)

Classic:

$O(n/\varepsilon^2)$ algorithm [Karp, Luby, Madras ’89]

$\Omega(n/\varepsilon^2)$ lower bound [Bringmann, Larsen, Nusser, Rotenberg, Wang ’26]

Incremental:

$\widetilde{O}(1/\varepsilon^2)$ update time and space [Meel, Vinodchandran, Chakraborty ’21]

Decremental?

Fully dynamic?

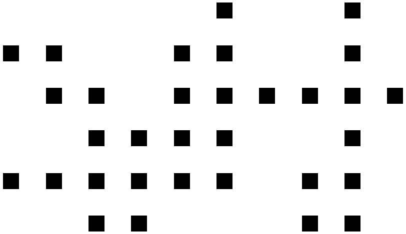
$\widetilde{O}(1/\varepsilon^2)$ update time and $\widetilde{O}(n+1/\varepsilon^2)$ space

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a finite set U

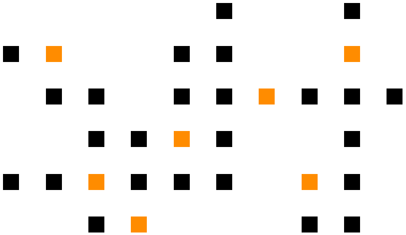
The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a finite set U



The Incremental Algorithm [MVC ’21]

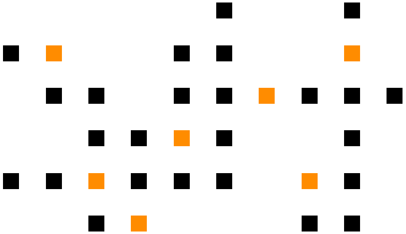
Definition: ℓ-sample of a finite set U



pick each point independently
with probability $2^{-\ell}$

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a finite set U

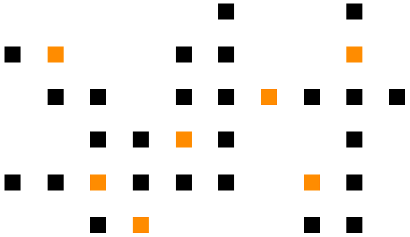


pick each point independently
with probability $2^{-\ell}$

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2|U|)$

The Incremental Algorithm [MVC '21]

Definition: ℓ-sample of a finite set U



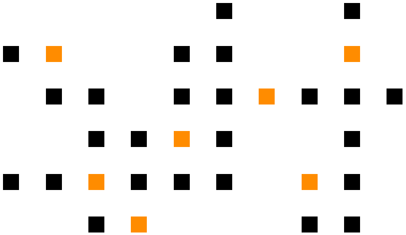
pick each point independently
with probability $2^{-\ell}$

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 |U|)$

Why? $N :=$ number of sampled points

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a finite set U



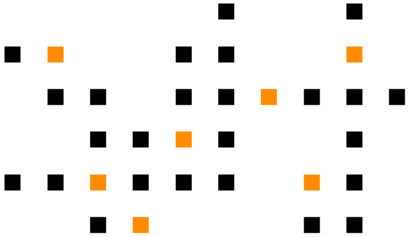
pick each point independently
with probability $2^{-\ell}$

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2|U|)$

Why? $N :=$ number of sampled points $\sim \mathrm{Bin}(|U|, 2^{-\ell})$

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a finite set U



pick each point independently
with probability $2^{-\ell}$

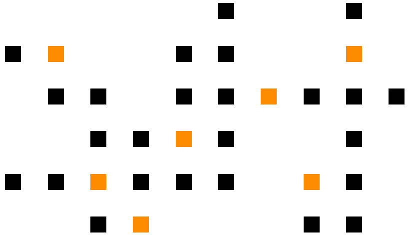
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 |U|)$

Why? $N :=$ number of sampled points $\sim \text{Bin}(|U|, 2^{-\ell})$

$\mathbb{E}[N] \approx 1/\varepsilon^2$

The Incremental Algorithm [MVC '21]

Definition: ℓ-sample of a finite set U



pick each point independently
with probability $2^{-\ell}$

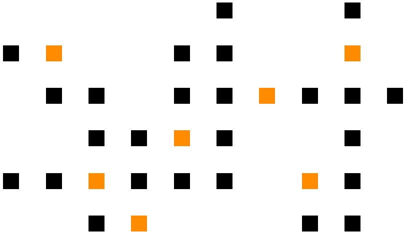
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 |U|)$

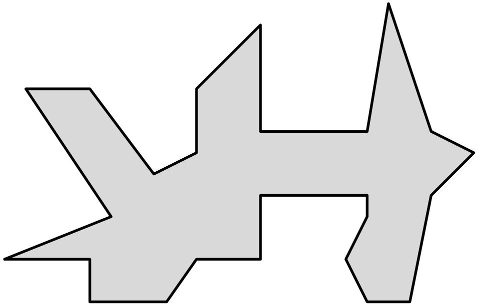
Why? $N :=$ number of sampled points $\sim \mathrm{Bin}(|U|, 2^{-\ell})$

$\mathbb{E}[N] \approx 1/\varepsilon^2$ $2^\ell N \in (1 \pm \varepsilon)|U|$ with high probability

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a ~~finite~~ set U





pick each point independently
with probability $2^{-\ell}$

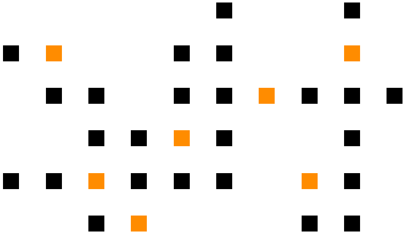
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2|U|)$

Why? $N :=$ number of sampled points $\sim \mathrm{Bin}(|U|, 2^{-\ell})$

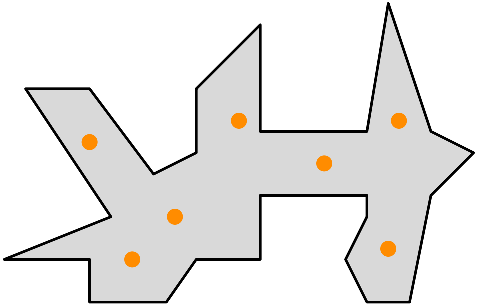
$\mathbb{E}[N] \approx 1/\varepsilon^2$ $\quad 2^\ell N \in (1 \pm \varepsilon)|U|$ with high probability

The Incremental Algorithm [MVC ’21]

Definition: ℓ-sample of a ~~finite~~ set U



pick each point independently with probability $2^{-\ell}$



pick each point independently with probability density $2^{-\ell}$

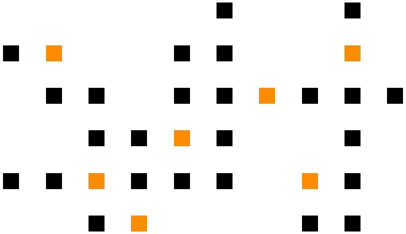
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 |U|)$

Why? $N :=$ number of sampled points $\sim \mathrm{Bin}(|U|, 2^{-\ell})$

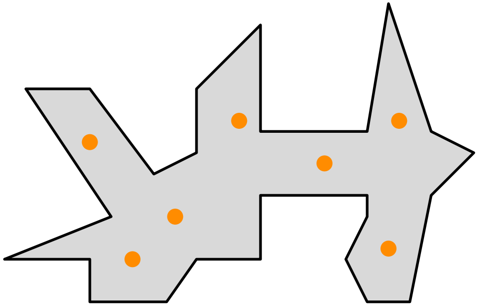
$\mathbb{E}[N] \approx 1/\varepsilon^2$ $\quad 2^\ell N \in (1 \pm \varepsilon)|U|$ with high probability

The Incremental Algorithm [MVC '21]

Definition: ℓ-sample of a ~~finite~~ set U



pick each point independently with probability $2^{-\ell}$



pick each point independently with probability density $2^{-\ell}$

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \overset{\mathrm{vol}(U)}{\cancel{|U|}})$

Why? $N :=$ number of sampled points $\sim \overset{\mathrm{Pois}(\mathrm{vol}(U),\,2^{-\ell})}{\cancel{\mathrm{Bin}(|U|, 2^{-\ell})}}$

$\mathbb{E}[N] \approx 1/\varepsilon^2 \qquad 2^\ell N \in (1 \pm \varepsilon)\underset{\mathrm{vol}(U)}{\cancel{|U|}}$ with high probability

The Incremental Algorithm [MVC ’21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

The Incremental Algorithm [MVC '21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

The Incremental Algorithm [MVC '21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \text{Pois}(\texttt{volume}(i)/2^\ell)$

$\texttt{random}(i)$ for N times

The Incremental Algorithm [MVC '21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \text{Pois}(\texttt{volume}(i)/2^\ell)$

$\texttt{random}(i)$ for N times

ℓ-sample from U:

The Incremental Algorithm [MVC '21]

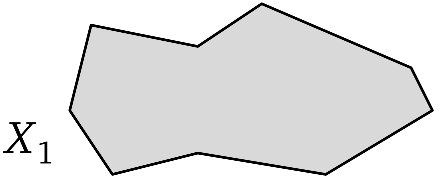
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \mathrm{Pois}(\texttt{volume}(i)/2^\ell)$

$\texttt{random}(i)$ for N times

ℓ-sample from U:



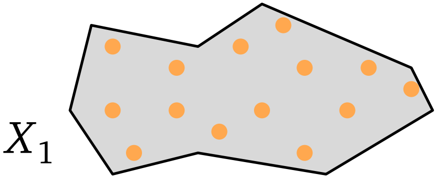
The Incremental Algorithm [MVC '21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \text{Pois}(\texttt{volume}(i)/2^\ell)$
$\texttt{random}(i)$ for N times

ℓ-sample from U:



The Incremental Algorithm [MVC '21]

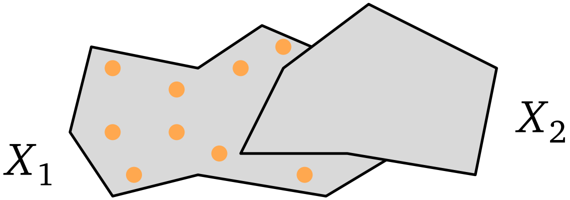
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \,\mathrm{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \mathrm{Pois}(\texttt{volume}(i)/2^\ell)$

`random`(i) for N times

ℓ-sample from U:



The Incremental Algorithm [MVC '21]

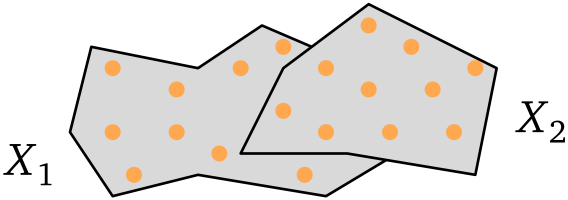
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \,\mathrm{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \mathrm{Pois}(\texttt{volume}(i)/2^\ell)$

`random`(i) for N times

ℓ-sample from U:



The Incremental Algorithm [MVC '21]

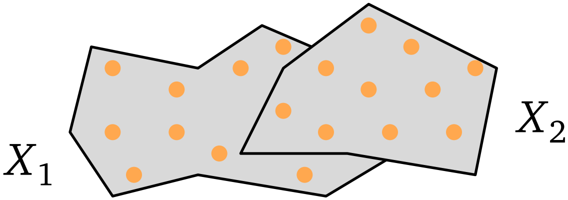
Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$

assume given

ℓ-sample from X_i: draw $N \sim \mathrm{Pois}(\texttt{volume}(i)/2^\ell)$

$\texttt{random}(i)$ for N times

ℓ-sample from U:



$S := \emptyset$

upon insertion of X_i,

$S := S \setminus \{x : \texttt{contains}(i, x)\}$

$S := S \cup (\ell\text{-sample from } X_i)$

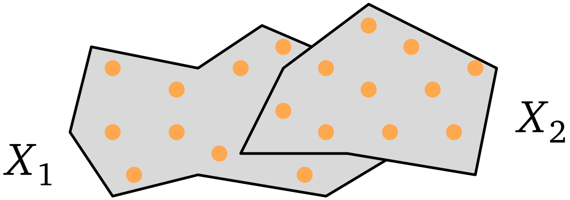
The Incremental Algorithm [MVC '21]

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \, \mathrm{vol}(U))$

initially $\ell := 0$
if $|S| \gg 1/\varepsilon^2$ then
  $\ell := \ell + 1$
  subsample S

ℓ-sample from X_i: draw $N \sim \mathrm{Pois}(\texttt{volume}(i)/2^\ell)$
`random`(i) for N times

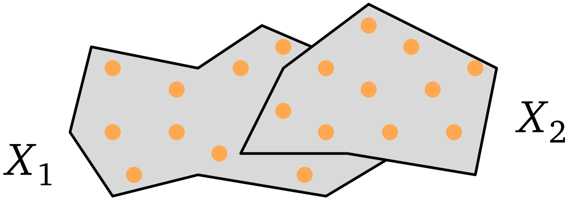
ℓ-sample from U:



```
$S := \emptyset$
upon insertion of $X_i$,
    $S := S \setminus \{x : \texttt{contains}(i, x)\}$
    $S := S \cup (\ell\text{-sample from } X_i)$
```

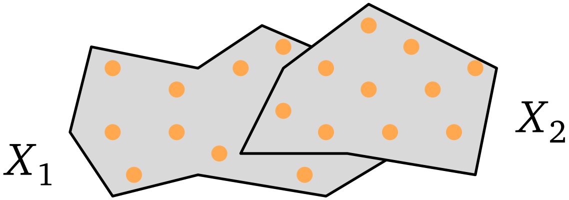
A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.



A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.

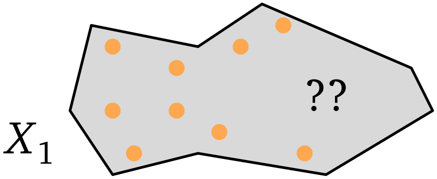


Challenges:

When X_i is deleted, not every point in X_i should be removed

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.

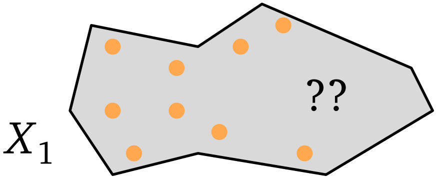


Challenges:

When X_i is deleted, not every point in X_i should be removed

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.

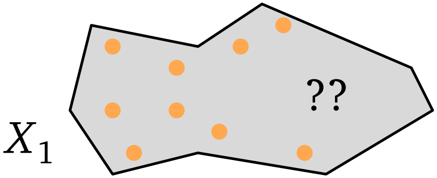


Challenges:

When X_i is deleted, not every point in X_i should be removed
→ Maintain a counter for each point

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.



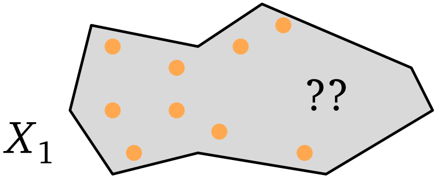
Challenges:

When X_i is deleted, not every point in X_i should be removed
→ Maintain a counter for each point

The number of points $|S|$ drops over time

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.



Challenges:

When X_i is deleted, not every point in X_i should be removed

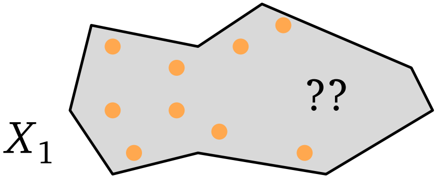
→ Maintain a counter for each point

The number of points $|S|$ drops over time

→ If $|S| \ll 1/\varepsilon$ then decrease ℓ and rebuild

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.



Challenges:

When X_i is deleted, not every point in X_i should be removed

→ Maintain a counter for each point

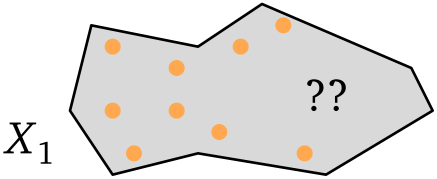
The number of points $|S|$ drops over time

→ If $|S| \ll 1/\varepsilon$ then decrease ℓ and rebuild

→ Bound the number of rebuilds by $O(\log \frac{\text{max volume}}{\text{min volume}})$

A Decremental Algorithm

Task: maintain an ℓ-sample of $U := \bigcup_i X_i$ for $\ell \approx \log(\varepsilon^2 \operatorname{vol}(U))$.



Challenges:

When X_i is deleted, not every point in X_i should be removed
→ Maintain a counter for each point

The number of points $|S|$ drops over time
→ If $|S| \ll 1/\varepsilon$ then decrease ℓ and rebuild
→ Bound the number of rebuilds by $O(\log \frac{\text{max volume}}{\text{min volume}}) = \widetilde{O}(1)$

A Decremental Algorithm

↓

A Fully Dynamic Algorithm?

A Decremental Algorithm

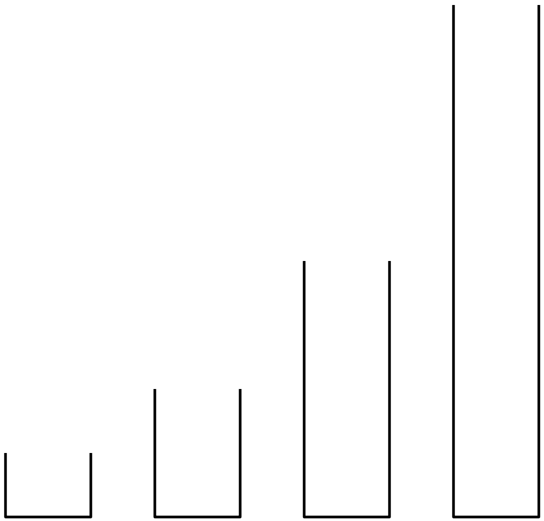
the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

A Decremental Algorithm

↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?



each being a decremental data structure

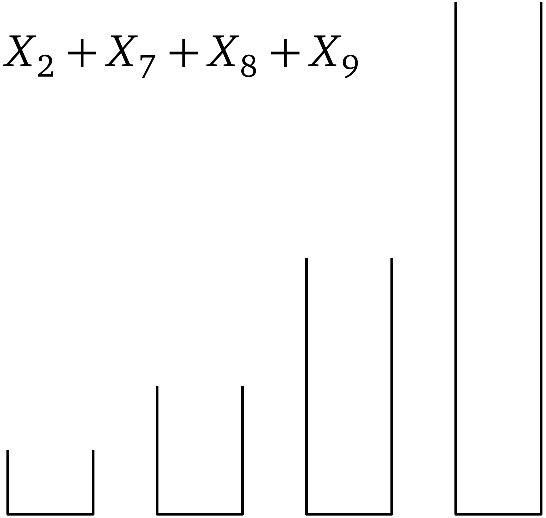
A Decremental Algorithm

↓ the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$



each being a decremental data structure

A Decremental Algorithm

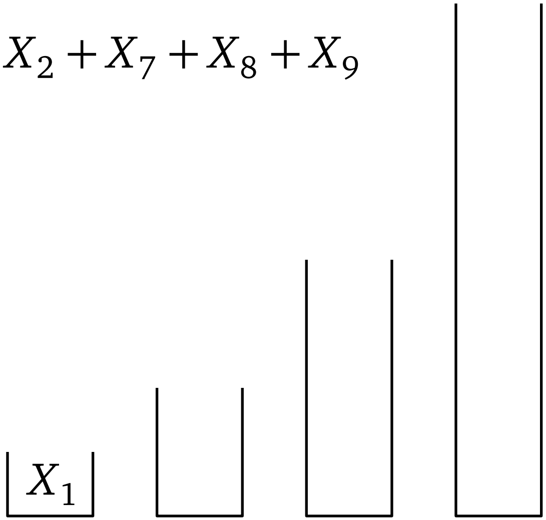
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

time



each being a decremental data structure

A Decremental Algorithm

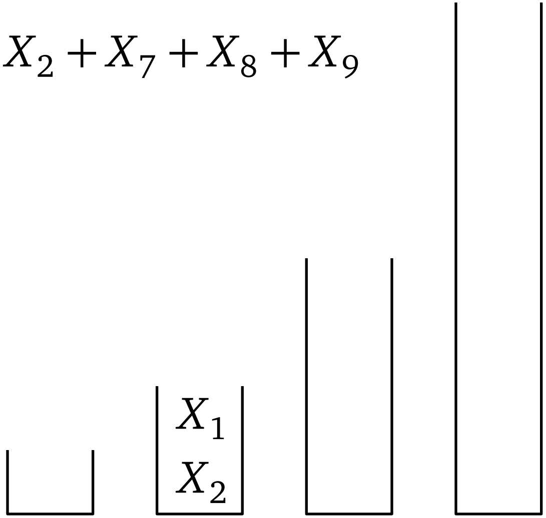
the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

Example:

$$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$$

time



each being a decremental data structure

A Decremental Algorithm

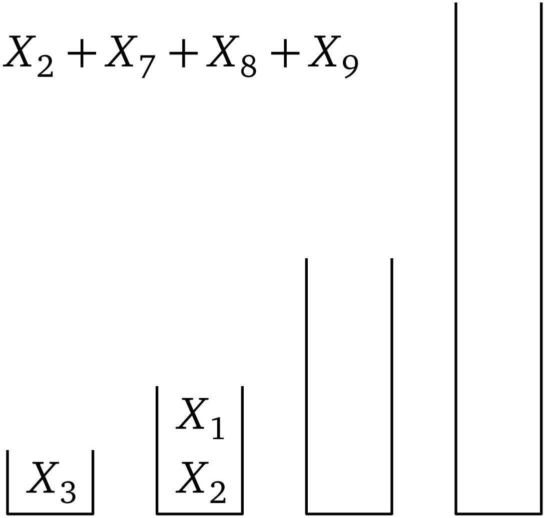
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

▲ time



each being a decremental data structure

A Decremental Algorithm

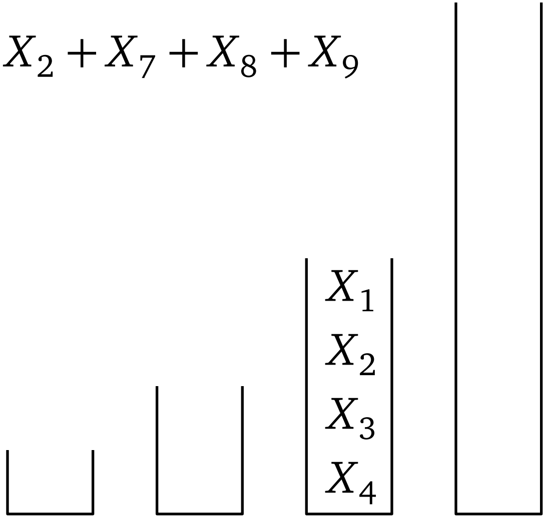
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

▲
time



each being a decremental data structure

A Decremental Algorithm

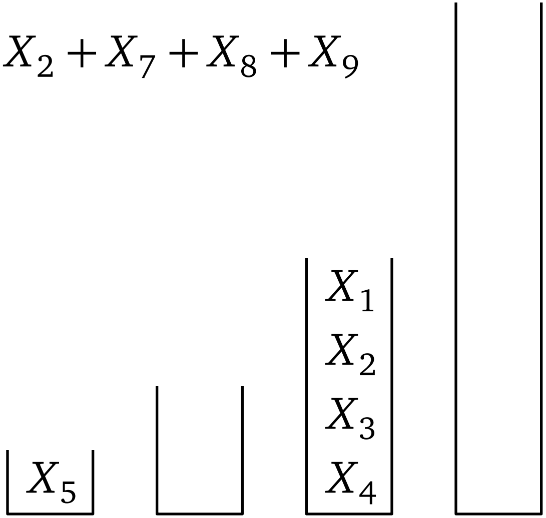
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$$

▲ time



each being a decremental data structure

A Decremental Algorithm

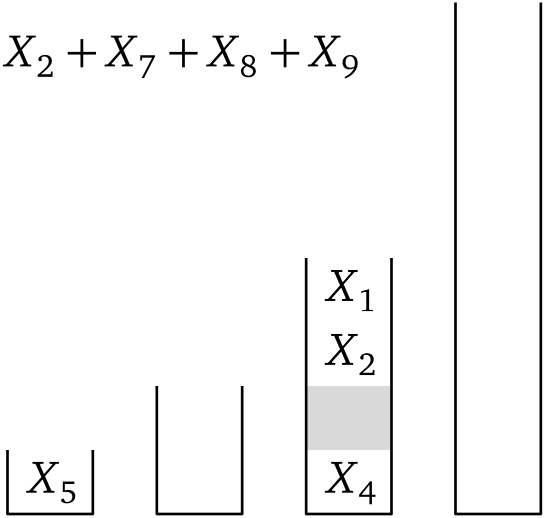
↓ the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

▲ time



each being a decremental data structure

A Decremental Algorithm

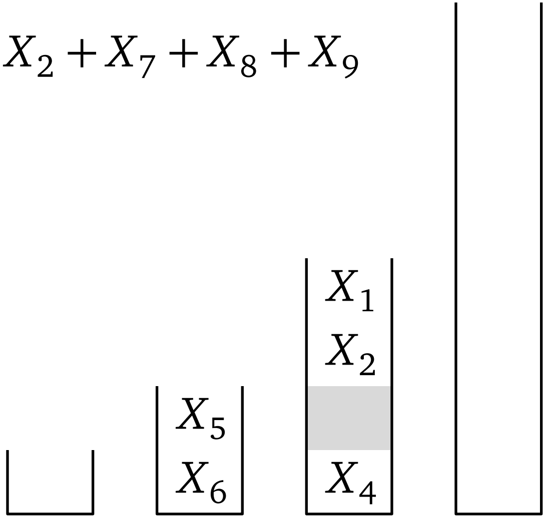
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

▲ time



each being a decremental data structure

A Decremental Algorithm

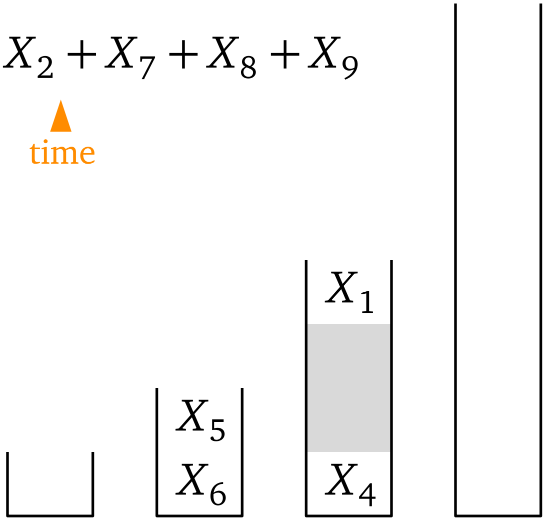
the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

time



each being a decremental data structure

A Decremental Algorithm

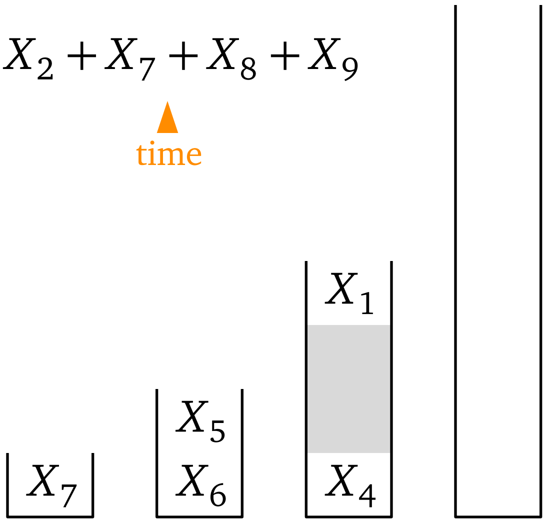
↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

▲ time



each being a decremental data structure

A Decremental Algorithm

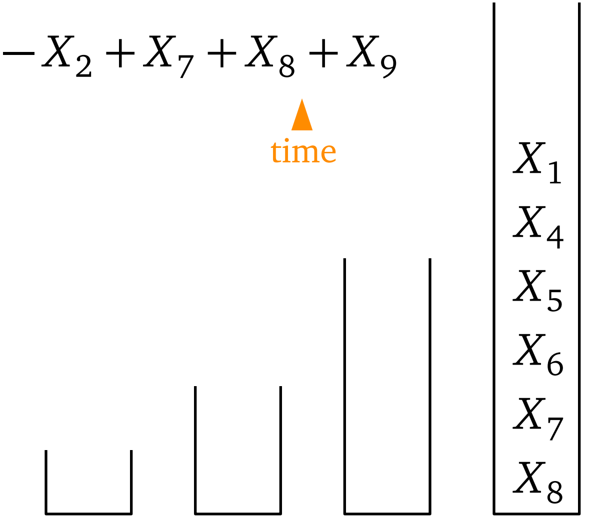
$\downarrow$ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

time



each being a decremental data structure

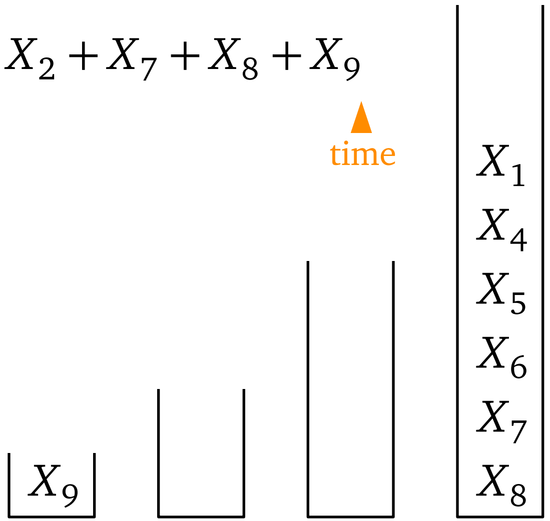
A Decremental Algorithm

↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$$



each being a decremental data structure

A Decremental Algorithm

↓ the logarithmic method [Bentley, Saxe ’80]

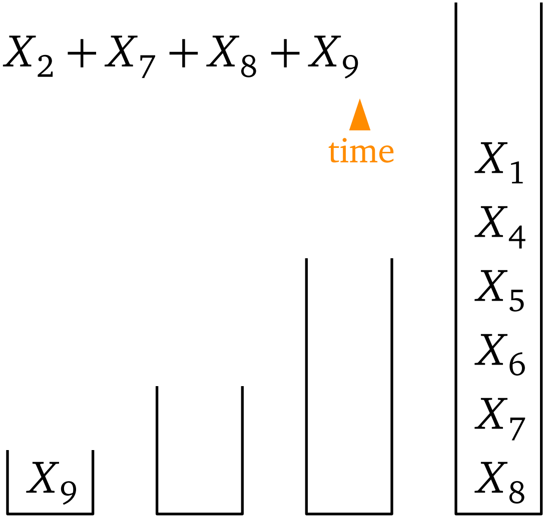
A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

time

amortized overhead $=$ polylog(n)



each being a decremental data structure

A Decremental Algorithm

↓ the logarithmic method [Bentley, Saxe '80]

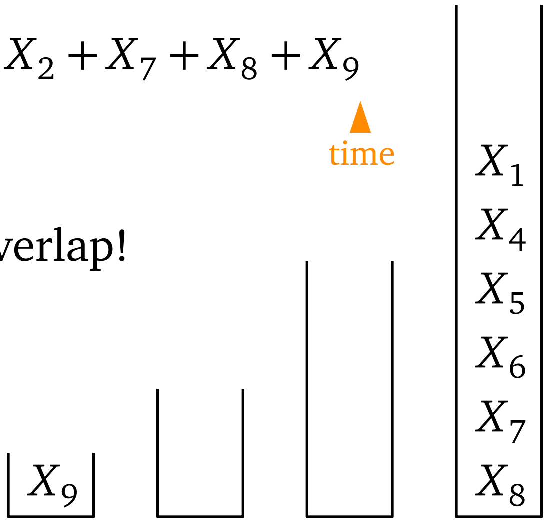
A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

amortized overhead $=$ polylog(n)

Issue: unions of different bins can overlap!



each being a decremental data structure

A Decremental Algorithm

a variant of ↓ the logarithmic method [Bentley, Saxe ’80]

A Fully Dynamic Algorithm?

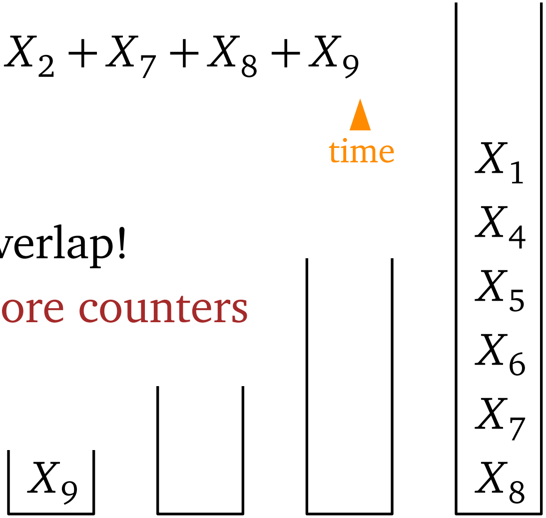
Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

amortized overhead $=$ polylog(n)

Issue: unions of different bins can overlap!

Solution: maintain ℓ-samples and more counters



each being a decremental data structure

A Decremental Algorithm

a variant of ↓ the logarithmic method [Bentley, Saxe '80]

A Fully Dynamic Algorithm?

Example:

$+X_1 + X_2 + X_3 + X_4 + X_5 - X_3 + X_6 - X_2 + X_7 + X_8 + X_9$

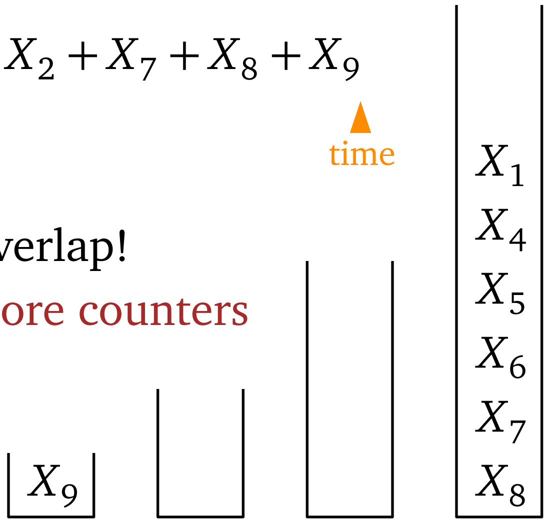
amortized overhead $=$ polylog(n)

Issue: unions of different bins can overlap!

Solution: maintain ℓ-samples and more counters

$\widetilde{O}(1/\varepsilon^2)$ amortized update time

$\widetilde{O}(n + 1/\varepsilon^2)$ space



each being a decremental data structure

Improving Space Complexity

Improving Space Complexity

1. **Sliding window**

   $X_1 \, X_2 \, X_3 \, X_4 \, X_5 \; \cdots$

Improving Space Complexity

1. Sliding window

$X_1 \; X_2 \; X_3 \; X_4 \; X_5 \; \cdots$

▲

now

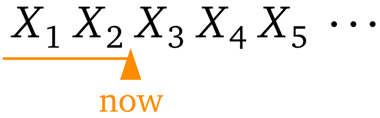
Improving Space Complexity

1. **Sliding window**

$$X_1\, X_2\, X_3\, X_4\, X_5\, \cdots$$

Improving Space Complexity

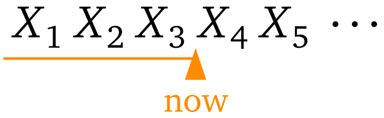
1. **Sliding window**



$X_1\ X_2\ X_3\ X_4\ X_5\ \cdots$

Improving Space Complexity

1. **Sliding window**



$X_1 \, X_2 \, X_3 \, X_4 \, X_5 \; \cdots$

Improving Space Complexity

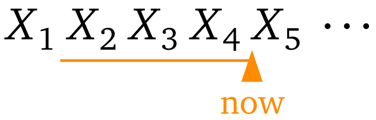
1. **Sliding window**

$$X_1\ \underline{X_2\ X_3}\ X_4\ X_5\ \cdots$$

now

Improving Space Complexity

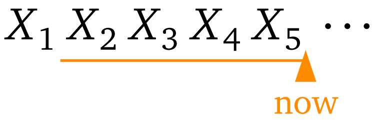
1. **Sliding window**



$X_1 \, X_2 \, X_3 \, X_4 \, X_5 \, \cdots$

Improving Space Complexity

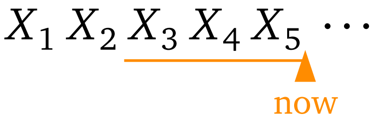
1. **Sliding window**



$$X_1\, X_2\, X_3\, X_4\, X_5\ \cdots$$

Improving Space Complexity

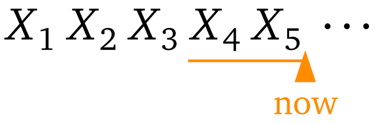
1. **Sliding window**



$X_1\ X_2\ X_3\ X_4\ X_5\ \cdots$

Improving Space Complexity

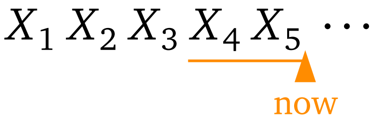
1. **Sliding window**



$X_1\; X_2\; X_3\; X_4\; X_5\; \cdots$

Improving Space Complexity

1. Sliding window

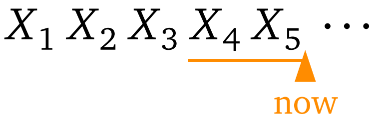


$X_1\, X_2\, X_3\, X_4\, X_5\ \cdots$

can achieve $\mathrm{polylog}(\frac{n}{\varepsilon})/\varepsilon^2$ update time *and* space

Improving Space Complexity

1. Sliding window



$X_1\ X_2\ X_3\ X_4\ X_5\ \cdots$

now

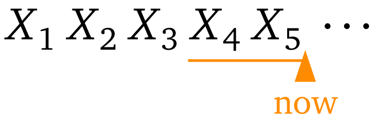
can achieve $\mathrm{polylog}(\frac{n}{\varepsilon})/\varepsilon^2$ update time *and* space

2. Convex bodies in $\mathbb{R}^d$

If each object contains an r-ball and is contained in an R-ball, then can achieve $\mathrm{polylog}(\frac{nR}{\varepsilon r})/\varepsilon^2$ update time and space

Improving Space Complexity

1. Sliding window



$X_1\ X_2\ X_3\ X_4\ X_5\ \cdots$

can achieve polylog$(\frac{n}{\varepsilon})/\varepsilon^2$ update time *and* space

2. Convex bodies in $\mathbb{R}^d$

If each object contains an r-ball and is contained in an R-ball, then can achieve polylog$(\frac{nR}{\varepsilon r})/\varepsilon^2$ update time and space

Questions?