An Efficient Transformation for Klee’s Measure Problem in the Streaming Model

Gokarna Sharma, Costas Busch, Ramachandran Vaidyanathan, Suresh Rai, and Jerry Trahan

Louisiana State University
Outline of the talk

• Introduction

• Motivation and Results

• General Area Estimation Algorithm

• Transformation

• Analysis Overview

• Conclusion
Introduction

• **Klee’s Measure Problem (KMP):** Given m axis-aligned rectangles, how quickly can we compute the area of their union?

• This problem has been studied extensively

• Best bound for time requirement: \( O(m \log m) \)

• Best bound for workspace requirement: \( O(m) \)
  – i.e., linear in the size of the input

• We consider Klee’s measure problem in the streaming model
Introduction (2)

- We have **limited workspace** and the input rectangles can be seen just **one time**
  - rescanning is not feasible

- The stream of rectangles are over a **discrete space**

- The task is to estimate **at any time** the area occupied by the **rectangles that have arrived so far**
  - i.e., the total number of distinct points

- Generally referred to as the zeroth frequency moment (denoted by $F_0$)

- We focus on the problem in 2-dimensions
Motivation

• Due to spatial and temporal data the arise in many applications, e.g.,
  – VLSI processing
  – sensor networks

• OpenGIS spatial database
  – a large collection of relatively simple geometric objects
  – rectangles are the most basic types

• Query processing in constraint database models
  – computation over a set of geometric objects

• Suitable in online scenarios
  – rescanning of dataset is not possible
Related Work

- Deterministic **exact** solutions via sweeping a vertical line across the area

- existing lower bounds

- Recent focus on approximation algorithms
  - Bringmann and Friedrich’s $(1 \pm \varepsilon)$-approximation (CGTA 2010)
  - Workspace is still $O(m)$

- Computing $F_0$ exactly requires space linear in the distinct input values [Alon et al., STOC’96]

- Recent work in $F_0$ of KMP [Tirthapura and Woodruff, PODS’12]
  - $(\varepsilon, \delta)$-approximation
  - Both processing time and workspace $O(1/\varepsilon \log(mU/\delta))^{O(1)}$
    $U$ is the universe of input rectangles
Results

• **Fat rectangles**
  - $1/c \leq b/a \leq c$, $c \geq 1$
  - $(\varepsilon, \delta)$-approximation for $F_0$
  - Processing time per rectangle $O(1/\varepsilon \log(U/\varepsilon) \log 1/\delta)$
  - Workspace $O(1/\varepsilon^2 \log U \log(1/\delta))$ bits
  - Query time $O(1)$

✓ An algorithm is said to **$(\varepsilon, \delta)$-approximate** $F_0$ if the estimated output $F_0'$ satisfies $\Pr[|F_0' - F_0| < \varepsilon F_0] > 1 - \delta$, $0 < \varepsilon, \delta < 1$
Results (2)

• **Arbitrary rectangles** (any ratio of side lengths)
  – $O(\sqrt{\log U})$-approximation for $F_0$
  – $\Omega(1/\sqrt{\log U}) \leq \text{est}(A)/A \leq O(\sqrt{\log U})$
  – Processing time per rectangle $O(\log U \log\log U)$
  – Workspace $O(\log^2 U \log\log U)$ bits
  – Query time $O(\log U)$

Comparing to [Tirthapura and Woodruff, PODS’12]
– Our algorithm is very simple
– Bounds do not depend on $m$
– Provides different tradeoffs
Solution Overview

• Transform each input rectangle to an interval (i.e. range in one dimension)
  – Proximity transformation based on a Z-ordering

• Apply one dimensional streaming approximation algorithm in ranges to provide estimation for the stream of rectangles
Generic $F_0$ Estimation Algorithm

Input: a stream $X$ of rectangles
Output: estimate $\text{est}(A)$ of the distinct points $A$ covered

- Define $\chi$ buckets of normal rectangles

- **When a new rectangle $x$ arrives:**
  - Normalize it into a sequence of normal rectangles and assign each to some appropriate bucket
    - One dimensional mapping
      - Map each normal rectangle to a range
    - Apply a range-efficient $(\varepsilon, \delta)$-approximation algorithm to update the estimate
  - Maintain $E_{\text{max}}$ and $E_{\text{sum}}$

- **When an estimate for $F_0$ is asked for:**
  - Return $\text{est}(A) = \sqrt{E_{\text{max}} \cdot E_{\text{sum}}}$
Normalization

• **Ranges**
  - Let \( Z_n = \{0, 1, \ldots, 2^n - 1\} \) be a one dimensional space of \( 2^n \) grid points
  - A range \( r = [p_1, p_2] \), where \( 0 \leq p_1 \leq p_2 < 2^n \)

• **Rectangles**
  - A rectangle is a subset of \( U = Z_n^2 \)
  - \( 1 \leq |x|/|x'| \leq 1/16 \)
  - \( 1 \leq a/a' \leq 4 \) and \( 1 \leq b/b' \leq 4 \).

• **Normal rectangle**
  - \( x' \) is a normal rectangle
  - *Both sides some power of 2*

• **Aspect ratio**
  - Aspect ratio of \( x \) is \( b/a \)
Union of rectangles

- Given a rectangle $x$ we compute $y$ which is
  - either a normal rectangle or it is a rectangle which consists of multiple normal rectangles

- $X = x_0 \cup x_1 \cup \ldots$
- $Y = y_0 \cup y_1 \cup \ldots$

- We estimate the area $X$ based on the area $Y$

- We prove that $|X| = \gamma |Y|$
  - $1 \leq \gamma \leq (2c_v + 2c_h - 3) \cdot (2.\min\{c_v, c_h\} - 1)$
  - $c_v, c_h$ are maximum ratio $a/a'$, $b/b'$ among rectangles
Proximity Transformation

- maps normalized rectangles to one-dimensional ranges

- preserves intersection properties of rectangles
Fat Rectangles (One Bucket)

- **A stream of squares**
  - For a square $x$, there is an internal square $x'$ which consists of at most $c/n$ normal squares and
  - $a(1-n) \leq a' \leq a, \ 0 < n \leq 1$

- $E_{\text{max}} = E_{\text{sum}}$ for squares
  - direct output of the one-dimensional streaming algorithm

- project each of them to one-dimensional space using proximity transformation

- Apply one dimensional streaming algorithm to return $(\varepsilon', \delta)$-approximation

- $c_v, c_h \leq 1/(1-n)$

- substituting $\varepsilon'$ and $n$ by a function of $\varepsilon$, we obtain $(\varepsilon, \delta)$-approximation for $A$

**A stream of fat rectangles:** a fat rectangle with aspect ratio $h$ can be converted to at most $h+1$ squares that cover the same area
Arbitrary Rectangles (Many Buckets)

- Let \( Y=\{x_0, x_1, \ldots, x_{m-1}\} \) be a set of \( m \) rectangles

- Partition \( Y \) arbitrary into \( \chi \) subsets \( \{Y_0, Y_1, \ldots, Y_{\chi-1}\} \)

- Let:
  - \( A_i \): the area of the union of rectangles in \( Y_i \)
  - \( A_i' \): the area of the union of rectangles in \( Y_i' \) (a set of normalized rectangles of \( B_i \))
  - \( p = |\text{est}(A) - A|/A \) (the relative error)

- \( A \epsilon_1 \leq \text{est}(A) \leq A \epsilon_2 \) with probability \( (1- \delta)^\chi \)
  - \( \epsilon_1 = \Delta_1 (1- \epsilon)/\sqrt{\chi} \) and
  - \( \epsilon_2 = \Delta_2 (1+ \epsilon)\sqrt{\chi} \)
  - \( 0 < \Delta_1 \leq \Delta_2 \leq 1 \)
Arbitrary Rectangles (2)

- **Approximation**
  - Project each bucket $B_i$ to $Z_{2n}$
  - Apply a range-efficient algorithm for $F_0$ in each bucket $B_i$
  - For arbitrary rectangles, $\gamma \leq 91$
  - Setting $\Delta_1 = 1/\gamma$, $\Delta_2 = 1$, and $\chi = 2n + 1 = \log U + 1$
    - $\epsilon_1 = (1- \epsilon)/(91\sqrt{\log U + 1})$ and
    - $\epsilon_2 = (1+ \epsilon)\sqrt{\log U + 1}$ with probability $(1- \delta)^{\log U + 1}$

  - Setting $\epsilon = 1/2$ and $\delta = 1/(\log U + 1)$
    - $\Omega(1/\sqrt{\log U}) \leq \text{est}(A)/A \leq O(\sqrt{\log U})$ with constant probability
Arbitrary Rectangles (3)

• A $O(\sqrt{\log U})$ hides large constants

• We can bring them to 1 by tiling each rectangle $r$ with poly-log number of rectangles

• We prove that:
  – A rectangle $r$ can be $(1-n)$–approximately tiled with $4 \log^2(1/n^2)$ normal rectangles
    – It increases the time complexity by the same poly-log factor
  – However, it can be traded-off for the enhanced accuracy
Conclusions

• A randomized approximation algorithm with poly-log bounds on time and space requirements

• An \((1 \pm \varepsilon)\)-approximation for fat rectangles

• An \(O(\sqrt{\log U})\)-approximation for general rectangles

• **Future work:**
  – \(O(1)\) or \((1 \pm \varepsilon)\)-approximation algorithm for general rectangles
  – Efficient bounds for higher dimensional scenarios
  – Experimental evaluation in a real-time setting
Thank you!