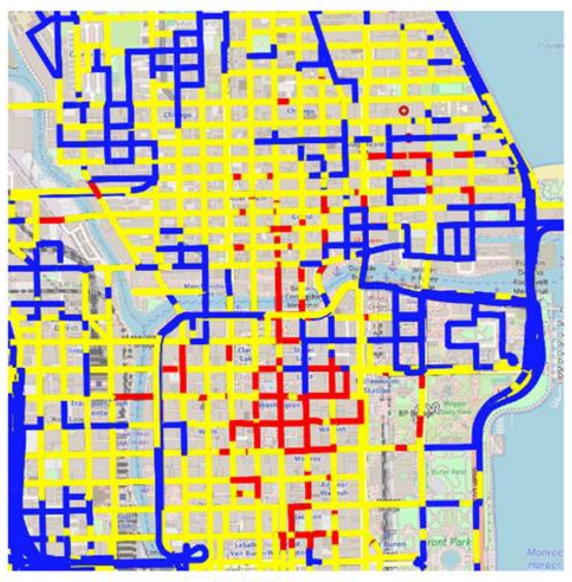
PLAN: Fast and Approximate Gaussian Kernel Density Visualization in Road Networks

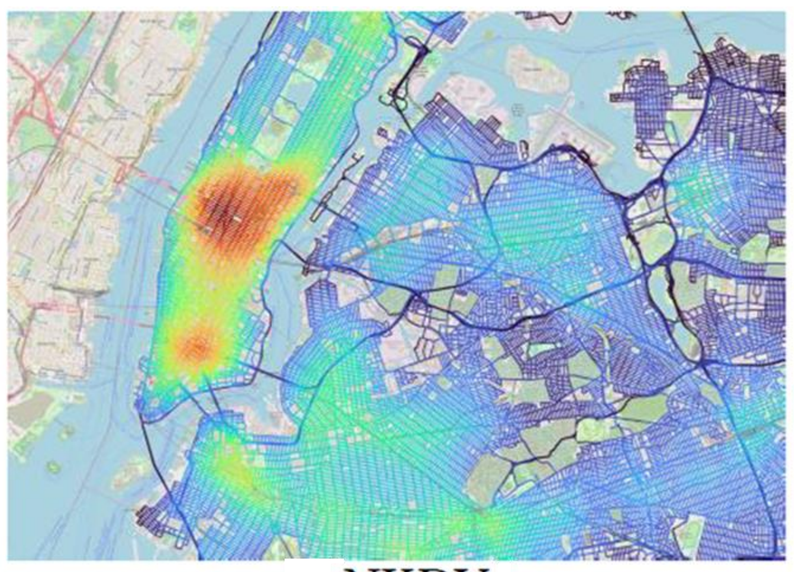
Tsz Nam Chan, Shenzhen University, edisonchan@szu.edu.cn
Hongwei Ye, Shenzhen University, yehongwei2023@email.szu.edu.cn
Bojian Zhu, Hong Kong Baptist University, csbjzhu@comp.hkbu.edu.hk
Leong Hou U, University of Macau, ryanlhu@um.edu.mo
Dingming Wu, Shenzhen University, dingming@szu.edu.cn
Ruisheng Wang, Shenzhen University, ruiswang@szu.edu.cn
Joshua Zhexue Huang, Shenzhen University, zx.huang@szu.edu.cn

Why Network Kernel Density Visualization (NKDV)?


Chicago crime data points


NKDV


New York traffic accident dataset


NKDV

- Identify hotspot/coldspot (i.e., red/blue) regions of a location dataset.
- Main applications
  - Traffic/Traffic accident hotspot detection
  - Crime hotspot detection
  - Urban planning
- Supported by many software packages (e.g., spNetwork, SANET, and PyNKDV)

What is NKDV?

- Given a location dataset $P = \{p_1, p_2, \dots, p_n\}$ and a road network $G = (V, E)$, we need to color each lixel q based on the network kernel density function $\mathcal{F}_P(q)$.

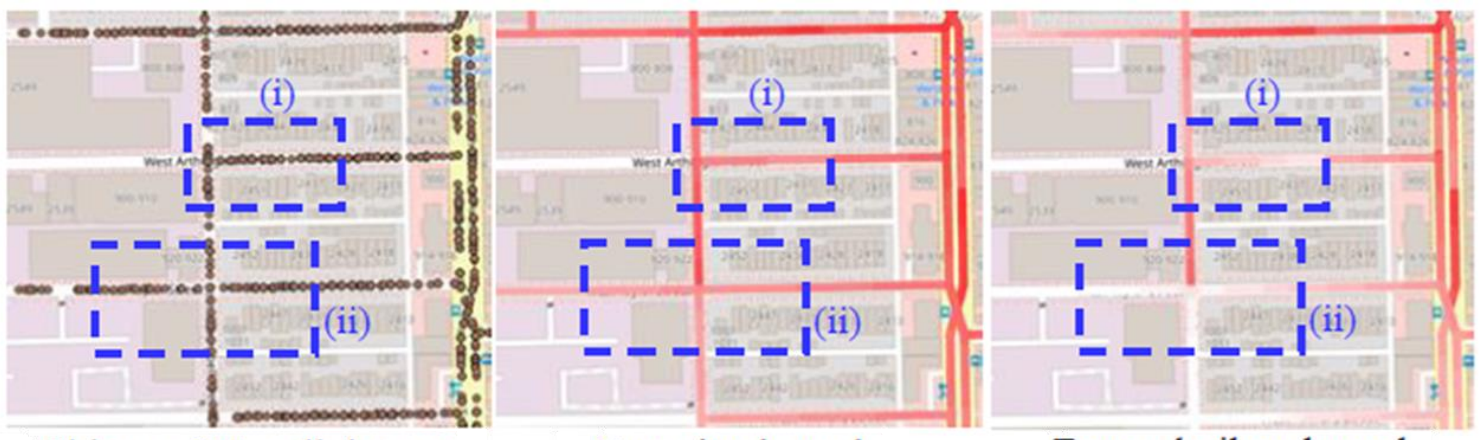
$$\mathcal{F}_P(q) = \frac{1}{|P|} \sum_{p \in P} K_G(q, p)$$

- Some commonly used kernel functions:

Kernel	Function
Epanechnikov	$\begin{cases} 1 - \frac{1}{b^2} d_G(q,p)^2 & \text{if } d_G(q,p) \le b \\ 0 & \text{otherwise} \end{cases}$
Gaussian	$\exp\left(-\frac{1}{b^2} d_G(q,p)^2\right)$

Location dataset — NKDV

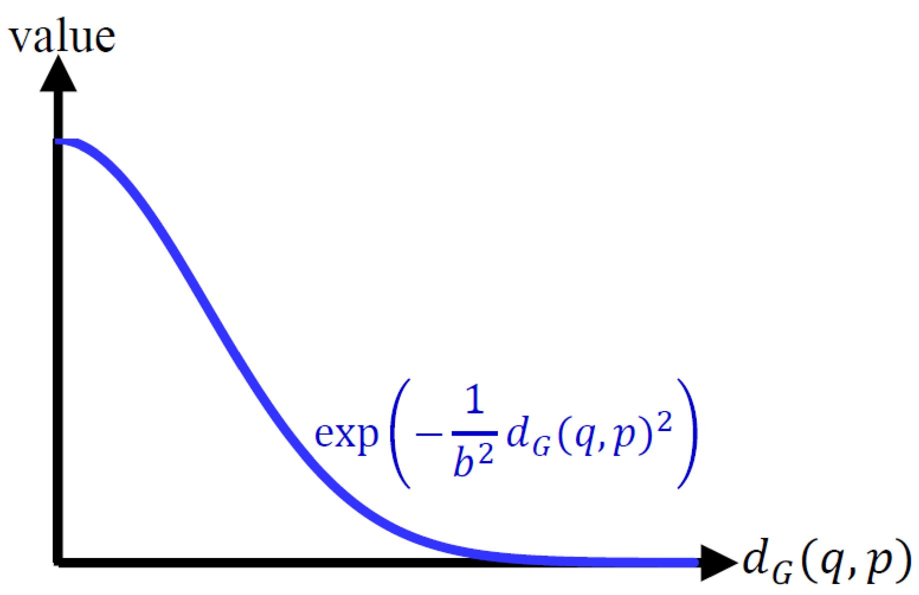
Why Gaussian Kernel?



Chicago 311-call dataset — Gaussian kernel — Epanechnikov kernel

- Gaussian kernel can provide smoother visualization.
- Gaussian kernel is the default one in many software packages.

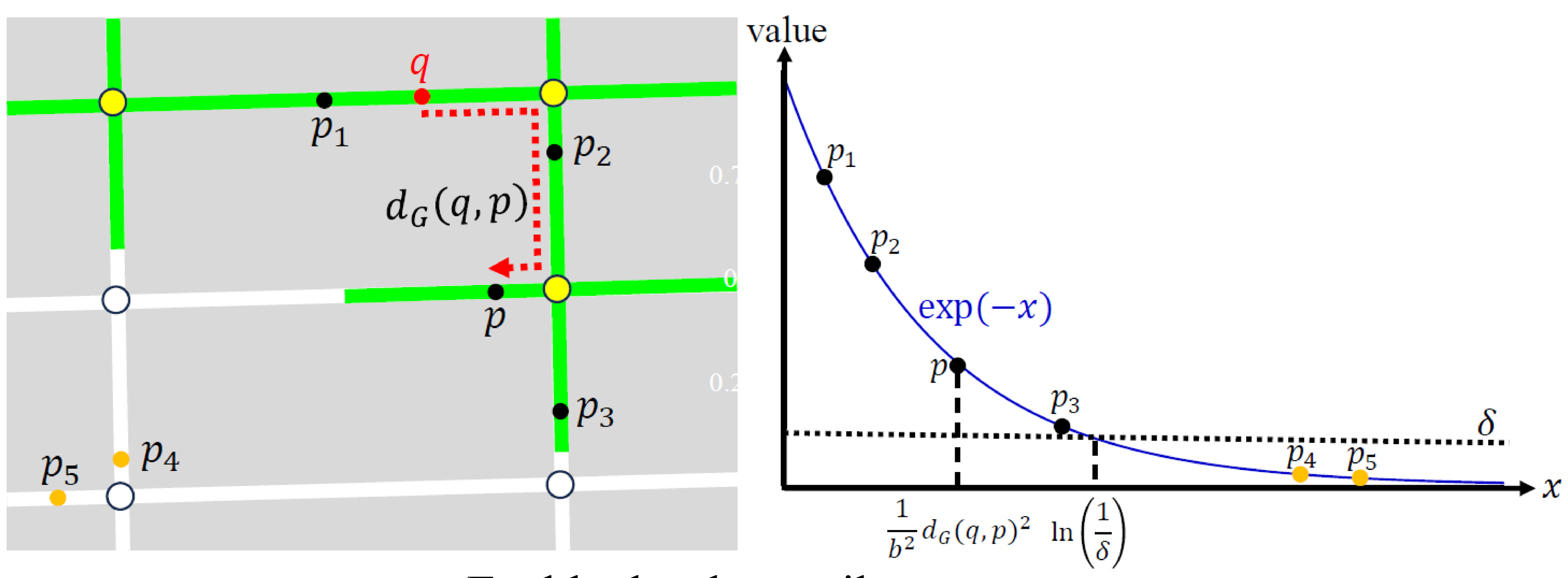
Challenges



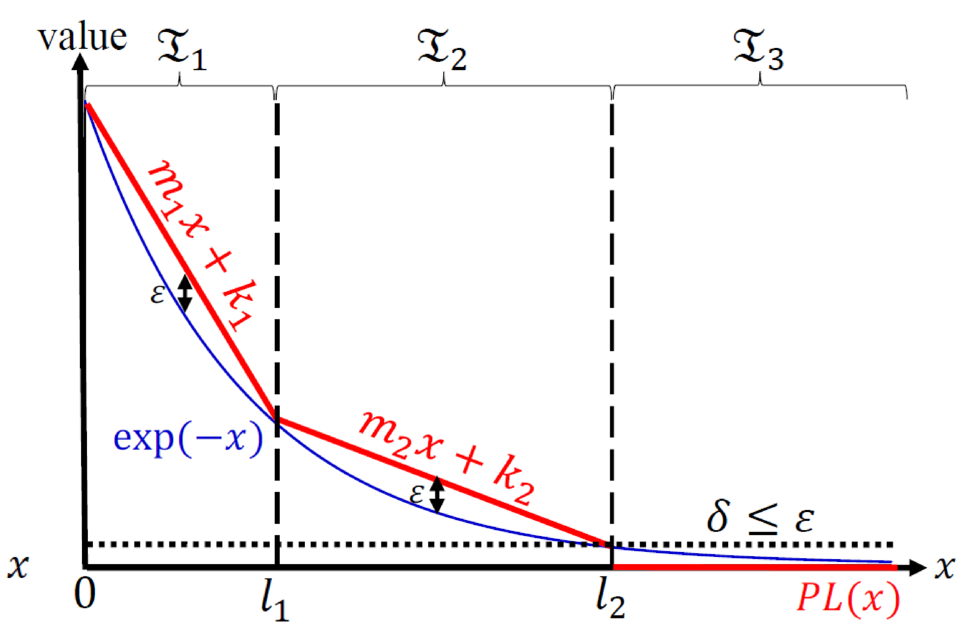
Gaussian kernel

- Does not have the short-tail property.
- Does not have the sum-of-distance property.
- The time complexity of the state-of-the-art solution is $O(|E|T_{SP} + |P|L)$.

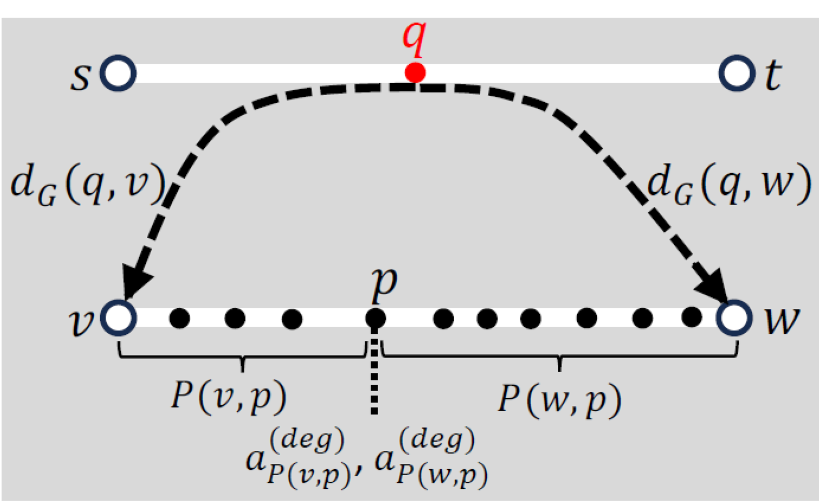
Piecewise-linear Approximate Solution (PLAN)



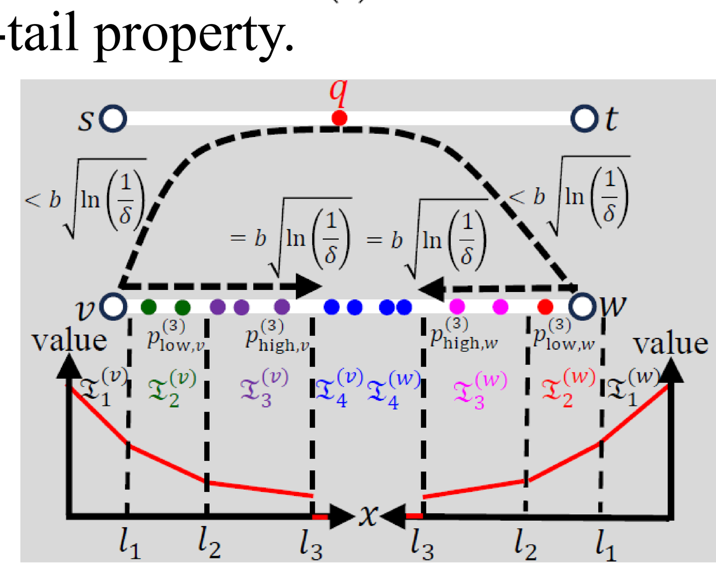
Enable the short-tail property.



Piecewise-linear approximation.



Augmentation


Use the binary search multiple times

- Use $A_P(q)$ to approximate $\mathcal{F}_P(q)$.

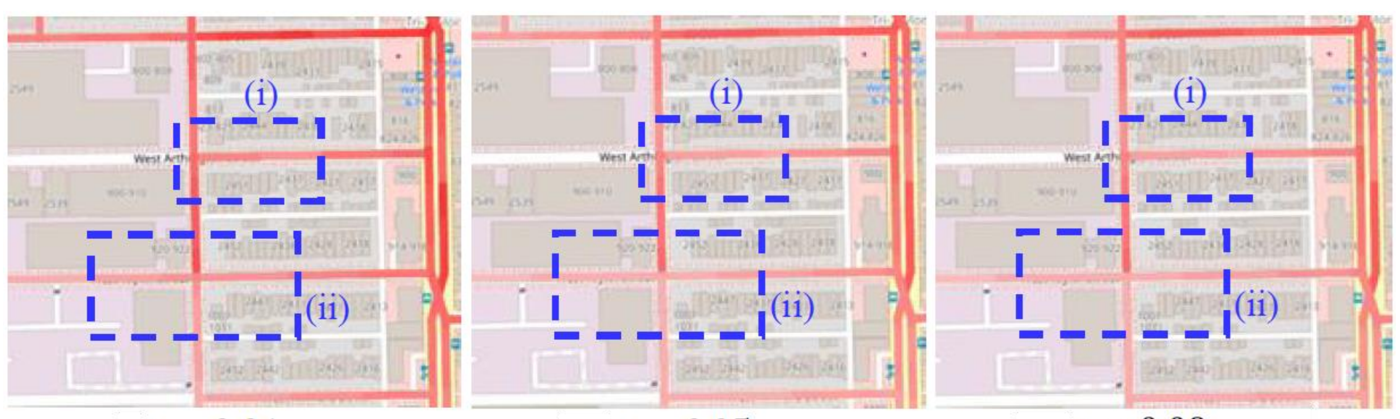
$$A_P(q) = \frac{1}{|P|} \sum_{p \in P} PL\left(\frac{1}{b^2} d_G(q,p)^2\right)$$

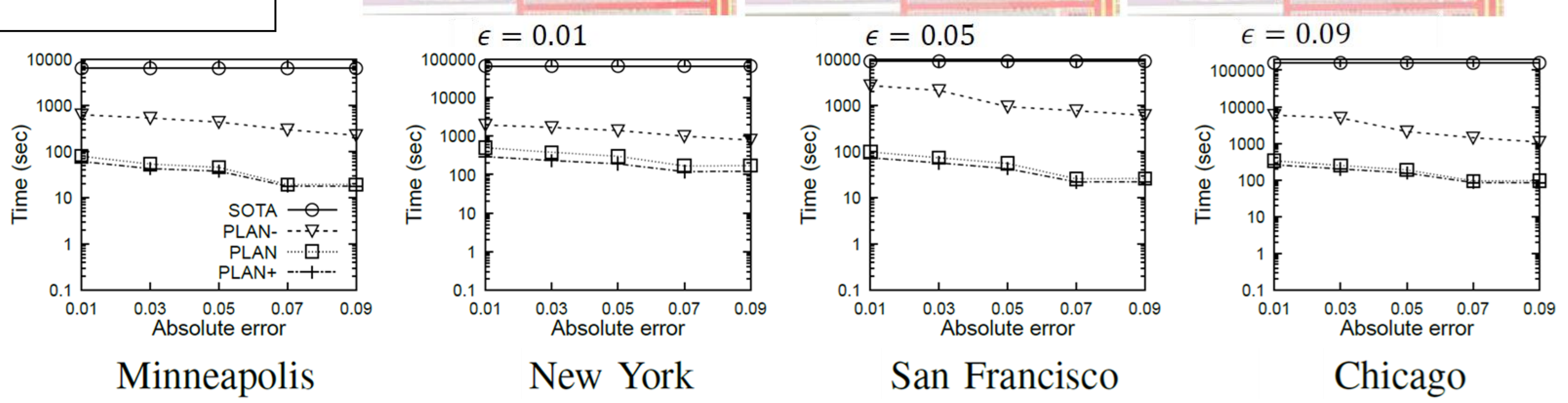
- Modify our previous ADA solution in VLDB 2021.

Theoretical and Experimental Results

Method	Time complexity	Space complexity	Accuracy guarantee
SOTA	$O(\lvert E\rvert T_{SP} + \lvert P\rvert L)$	$O(\lvert V\rvert + \lvert P\rvert + L + S_{SP})$	Exact
PLAN	$O\left(\lvert E\rvert T_{SP} + LM\lvert E\rvert \log\left(\frac{\lvert P\rvert}{\lvert E\rvert}\right)\right)$ (Theorem 1)	$O(\lvert V\rvert + \lvert P\rvert + L + S_{SP} + M)$ (Theorem 3)	ϵ-absolute error (Lemma 1)
PLAN+	$O\left(\lvert E\rvert T_{SP} + LM\lvert E\rvert \log\left(\frac{\lvert P\rvert}{M\lvert E\rvert}\right)\right)$ (Theorem 2)		

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert P\rvert$	Category
Minneapolis [36]	7,388	12,913	1,598,877	911 calls
New York [37]	71,019	120,623	1,897,418	Traffic accidents
San Francisco [34]	6,627	11,593	2,985,502	Crime events
Chicago [25]	41,111	70,067	10,893,129	311 calls





Minneapolis — New York — San Francisco — Chicago