Linear Hotspot Discovery on All Simple Paths: A Summary of Results

Xun Tang
University of Minnesota, Twin Cities
Minneapolis, Minnesota
tangx456@umn.edu

Jayant Gupta
University of Minnesota, Twin Cities
Minneapolis, Minnesota
gupta423@umn.edu

Shashi Shekhar
University of Minnesota, Twin Cities
Minneapolis, Minnesota
shekhar@umn.edu

ABSTRACT

Spatial hotspot discovery aims at discovering regions with statistically significant concentration of activities. It has shown great value in many important societal applications such as transportation engineering, public health, and public safety. This paper formulates the problem of Linear Hotspot Detection on All Simple Paths (LHDA) which identifies hotspots from the complete set of simple paths enumerated from a given spatial network. LHDA overcomes the limitations of existing methods which miss hotspots that naturally occur along linear simple paths on a road network. To address the computational challenges, we propose a novel algorithm named bi-directional fragment-multi-graph traversal (ASP_FMGT) and two path reduction approaches ASP_NR and ASP_HD. Experimental analyses show that ASP_FMGT has substantially improved performance over state-of-the-art approach (ASP_Base) while keeping the solution complete and correct. Moreover, a case study on real-world datasets showed that ASP_FMGT outperforms existing approaches.

CCS CONCEPTS

• **Information systems → Geographic information systems**; **Data mining**.

KEYWORDS

Spatial data mining, spatial networks, spatial hotspot detection

ACM Reference Format:
Xun Tang, Jayant Gupta, and Shashi Shekhar. 2019. Linear Hotspot Discovery on All Simple Paths: A Summary of Results. In *27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '19), November 5–8, 2019, Chicago, IL, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3347146.3359100

1 INTRODUCTION

Spatial hotspot discovery finds regions of interest which have statistically significant concentration of activities (e.g., traffic accidents). It has been applied in many societal applications including transportation engineering, public health, and public safety over the years. This paper studies the problem of Linear Hotspot Discovery on All Simple Paths (LHDA) which identifies all the simple paths (i.e. paths with no loops) in a given spatial network (e.g., road network) that have statistically significant density of activities.

Figure 1: (a) Pedestrians at risk on road [2]. (b) Queens Boulevard with new paved sidewalk and road separators [4].

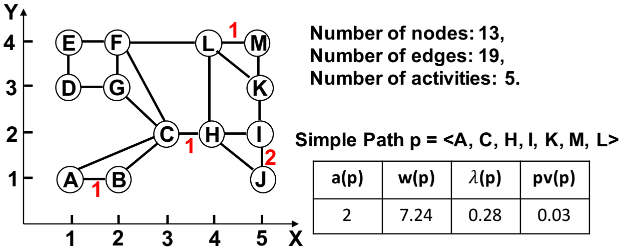
Application Domains: Improving infrastructure for pedestrian safety is a critical task in transportation engineering. For example, Figure 1(a) shows pedestrians at risk walking in a motor vehicle lane since the sidewalk is covered by snow [2]. In contrast, as shown in Figure 1(b), Queens Boulevard in New York City once called the "Boulevard of Death", has become dramatically safer because of newly built dividers between pedestrians, bicycles, and motorcycles [4]. LHDA provides an efficient data-driven approach to help identify accident-prone routes (i.e., linear hotspots) in cities which need urgent attention. In public health and epidemiology, LHDA helps identify disease hotspots which are critical for disease prevention, preparation and reduction [1]. Also, LHDA helps identify streets prone to crimes to help deploy police force efficiently.

Challenges The computational challenges of LHDA come from the following aspects. First, the size of studied spatial networks can be very large. The number of hotspots can also be beyond polynomial in the worst case since a complete graph containing $|N|$ nodes can generate $N \times |N|!$ distinct simple paths.

Related Work Traditionally, clustering algorithms (e.g., DBScan [3]) are used for detecting regions with high activity concentration. However, without a statistical significance test, they tend to output false positive hotspots formed just by chance. False positives (e.g., false crime hotspots) are unacceptable in many societal applications due to the potential severe consequences. To eliminate the false positives, approaches with statistical significance test have been developed to detect hotspots modeled in Euclidean space. They discover hotspots in a collection of regular shapes such as circle, ellipse [5], and rectangle [7]. For modeling activities along roads, some recent works focus on spatial network models [10, 11]. Network iso-distance hotspot detection [11] discovers hotspots in iso-distance sub-networks. Another approach, Linear Hotspot Detection on Shortest Paths (LHDSP) [10] discovers linear hotspots as shortest paths given the trade off between solution completeness and computational tractability. There are also, bottom-up approaches that consider flexible paths rather than shortest paths [6], but these miss the hotspots that have overall high concentration but sparse sub-paths.

SIGSPATIAL '19, November 5–8, 2019, Chicago, IL, USA

ACM ISBN 978-1-4503-6909-1/19/11.
https://doi.org/10.1145/3347146.3359100



Number of nodes: 13,
Number of edges: 19,
Number of activities: 5.

Simple Path p = <A, C, H, I, K, M, L>

a(p)	w(p)	λ(p)	pv(p)
2	7.24	0.28	0.03

Figure 2: Illustration of LHDA (best in color)

Contribution This paper formulates the problem of Linear Hotspot Discovery on All Simple Paths (LHDA) which detects novel hotspot patterns which have not been studied before. It proposes a novel prune-and-refine algorithm based on bi-directional fragment-multi-graph traversal (ASP_FMGT) and two path reduction approaches ASP_NR and ASP_HD to overcome the computational challenges. A case study on a real-world dataset shows that the proposed approaches located linear hotspots with high accuracy and discover patterns that were completely missed before. Experimental evaluations show the proposed approaches achieve substantial improvement in scalability over the baseline approach.

Scope and Organization of This Paper Our focus is discrete activity modeled as points. Modeling of other types of activities such as trajectories is not considered. Also, a thorough analysis of the underlying incentives of hotspots can only be done by domain experts and fall outside the scope of this paper. Approaches based on spatial statistics (e.g., Getis-Ord statistic) have been used extensively, and comparative analysis with such methods [13] is also beyond the scope of this paper. The rest of this paper is organized as follows: Section 2 formulates the problem of LHDA. Section 3 proposes a baseline algorithm ASP_Base. Section 4 proposes novel algorithms ASP_FMGT, ASP_NR, and ASP_HD. Section 5 presents a case study on a real-world dataset. Section 6 discusses the experimental evaluation of the proposed approaches. Section 7 concludes the paper and previews future work.

2 PROBLEM STATEMENT

2.1 Basic Concepts

Definition 1. *A **spatial network** $G = (N, E)$ consists of a node set N and an edge set E, where each node $n \in N$ is at 2-D location $n = (x, y)$. Each edge $e = (n_i, n_j) \in E$ connects nodes n_i and n_j.*

Definition 2. *A **simple path** p is a sequence of k nodes $p = (n_1 \dots n_k)$ with no repeated nodes (i.e. no loops), and two consecutive nodes n_i and n_j are connected by the edge between them (n_i, n_j).*

Definition 3. *An **activity set** A is a collection of activities, and each activity $a \in A$ is associated with an edge $e \in E$.*

Definition 4. *The **activity coverage** and **weight** of path p: $a(p)$ and $w(p)$ are the number of activities and weight of p, respectively.*

Definition 5. *The **density of path** p, $\lambda(p) = \frac{a(p)}{w(p)}$.*

Lemma 1. *Given a path p concatenated by k mutually exclusive subpaths $\{p_i^{sub}, i = 1 \dots k\}$, $\min\{\lambda(p_i^{sub})\} \le \lambda(p) \le \max\{\lambda(p_i^{sub})\}$.*

This lemma allows bounding the density of a path using the information from its subpaths (Proof seen in full paper [12]).

Definition 6. *An **active edge** is an edge $e \in E$ that has at least one activity. The end nodes of an active edge are **active nodes**.*

2.2 Problem Statement

Given:

(1) A spatial network $G = (N, E)$ with a set of activities A, each associated with an edge,
(2) A density threshold, θ_λ,
(3) A p-value threshold, θ_v,
(4) A number of Monte Carlo simulations m.

Find: Simple paths $p \in R$ where $\lambda(p) \ge \theta_\lambda$ and p—value$(p) \le \theta_v$

Objective: Computational efficiency

Constraints:

(1) Each $p \in R$ is longer than or equal to a minimum weight threshold θ_w,
(2) Each $p \in R$ starts and ends with active nodes,
(3) Results are correct and complete.

Using the example in Figure 2 as input, a density threshold $\theta_\lambda = 0.25$, a p-value threshold $\theta_s = 0.05$, a minimum weight threshold $\theta_w = 5$, Path $p = (A, C, H, I, K, M, L)$ is a linear hotspot with density $\lambda(p) = 0.28$, p—value$(p) = 0.03$, and weight $w(p) = 7.24$.

3 ASP_BASE: BASELINE APPROACH

Step 1:ASP_Base enumerates all-simple-paths (ASPs) from the input spatial network. It takes each active node as the root and recursively traverses the network in a pre-order and depth-first manner. In each step of the traversal, a simple path p starting at the root node is enumerated and evaluated whether it is a hotspot candidate (i.e. $\lambda(p) \ge \theta_\lambda$ and $w(p) \ge \theta_w$). By running this algorithm for every active node at the root, we can find all the hotspot candidates in the input network G.

Step 2: This step evaluates the statistical significance of each candidate hotspot using Monte Carlo simulations. A hotspot candidate with a density lower than l out of m highest densities from simulations has a p-value of $\frac{l+1}{m+1}$. Hotspot candidates whose p-values are less than or equal to the given p-value threshold (e.g., 0.05) are deemed statistically significant hotspots.

4 ASP_FMGT: NOVEL APPROACH

A large portion of simple paths can be very sparse or even empty in the real world. Based on this observation, we employ a novel prune-and-refine algorithm which eliminates a large collection of sparse paths for a small computational cost.

4.1 Building Fragment-multi-graph Trees

ASP_FMGT prunes the network by traversing an FMG-tree built based on the partitioned network.

Network partitioning: At first, the input spatial network is partitioned into evenly-sized two-dimensional rectangular grids. The ASPs within each fragment are computed and the following necessary information is stored: (1) maximum density from each boundary node to all other nodes ($\lambda_{ba}^{max}(N)$); (2) maximum density between each pair of boundary nodes ($\lambda_{bb}^{max}(N_1, N_2)$).

Building Fragment-multi-graph: The original spatial network is now condensed to an abstract fragment-multi-graph (FMG) where each FMG-node represents a fragment. An FMG is much smaller in size than the original network and thus allows traversal and pruning with substantially less cost.

Building Fragment-multi-graph trees: Now that we have the FMG, we build Fragment-multi-graph trees (FMG-trees) which

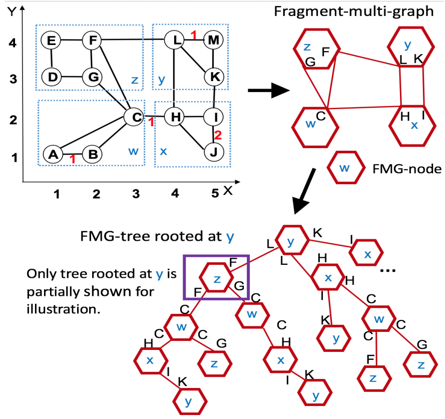


Figure 3: Example of network partitioning and fragment-multi-graph (best in color)

cover the ASPs in the original network. Similar to ASP_Base but applied on the FMG, an FMG-tree roots at an FMG-node and recursively grows towards its neighbors. Note that the simple paths covered in the FMG-trees are a superset of the actual ASPs, which ensures completeness of the solution with no adverse effects on correctness thanks to the refine phase at the end of the algorithm. An illustrative example can be seen in Figure 3.

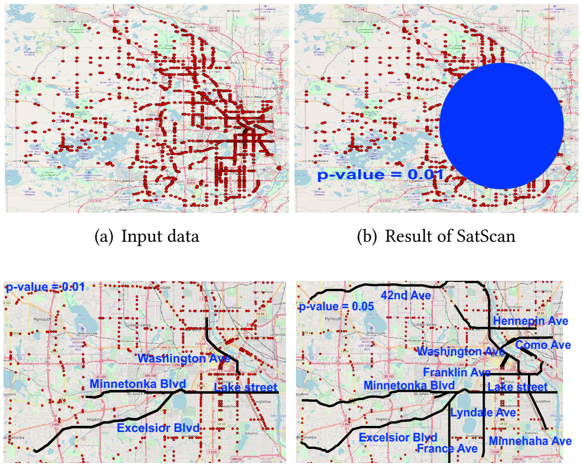
4.2 FMG-tree Pruning

Bottom-up traversal for information gathering: The pseudo-code of the recursive bottom-up traversal is shown in Algorithm 1. To start, each FMG-node considers the subpaths which pass through it and then end within one of its child FMG-nodes. According to Lemma 1, a "local" upper-bound density of this subpath λ^{max}_{local} is the maximum among the following three values: (1) the maximum density between the inbound node to all outbound nodes λ^{max}_{within} (Line 3). This value is the maximum value of λ^{max}_{bb} for the inbound node pre-computed during network partitioning; (2) the maximum density of the connecting edges λ^{max}_{edge} (Line 4); (3) and the maximum density of the paths within a child FMG-node from the inbound node λ^{max}_{child}(Line 5). This is equal to $\lambda^{max}_{ba}(N_{inbound})$ pre-computed in network partitioning. Note that if the investigated FMG-node is a leaf, then this value is 0 (Line 1-2). Now each FMG-node has a "local" upper-bound density λ^{max}_{local} (Line 6). In order to know the maximum of these "local" densities among the subtrees rooted at it $\lambda^{max}_{subtree}$, each FMG-node recursively collects the "local" upper-bounds from its children (Line 7). The bottom-up traversal uses this upper-bound as well as the density of the paths already traversed to make a final decision on whether a subtree should be pruned.

Top-down traversal for pruning: At each stop at an FMG-node during the traversal, we decide whether we can prune the subtree rooted at this FMG-node or need to continue the traversal using the maximum value of the upper-bound densities of two parts: (1) already visited paths and (2) unvisited paths. The first part is the concatenation of a sequence of FMG-nodes and their connecting edges. Thus, its upper-bound density can be determined as the maximum density among the FMG-nodes and connecting edges in this visited path. The second part is computed during the bottom-up phase ($\lambda^{max}_{subtree}$). Now, we can prune the entire FMG-subtree rooted at an FMG-node being visited if the corresponding upper-bound density is smaller than the density threshold θ_{λ}.

Algorithm 1 Bottom-up traversal (recursive)

```
Input:
    1) FMG-node $N^{FMG}$
    2) FMG-tree $T^{FMG}$
1: if $N^{FMG}$ is a leaf in $T^{FMG}$ then
2:     return 0
3: $\lambda^{max}_{within} \leftarrow \max\{\lambda^{max}_{bb}(N_{inbound}, N_{outbound})$ for $N^{FMG}\}$
4: $\lambda^{max}_{edge} \leftarrow \max\{\lambda_{connectingedge}\}$ for $N^{FMG}$
5: $\lambda^{max}_{child} \leftarrow \lambda^{max}_{ba}(N_{inbound})$ for $N^{FMG}$
6: $\lambda^{max}_{local} \leftarrow \max\{\lambda^{max}_{within}, \lambda^{max}_{edge}, \lambda^{max}_{child}\}$
7: $\lambda^{max}_{subtree} \leftarrow \max\{\lambda^{max}_{local}, \max_{N^{FMG}_{child}} bottom\text{-}up(N^{FMG}_{child}, T^{FMG})\}$
8: return $\lambda^{max}_{subtree}$
```



(a) Input data (b) Result of SatScan

(c) Linear hotspots (p-value = 0.01) (d) Linear hotspots (p-value = 0.05)

Figure 4: Case study on traffic accidents in Hennepin County, Minneota [9]. (Best in color)

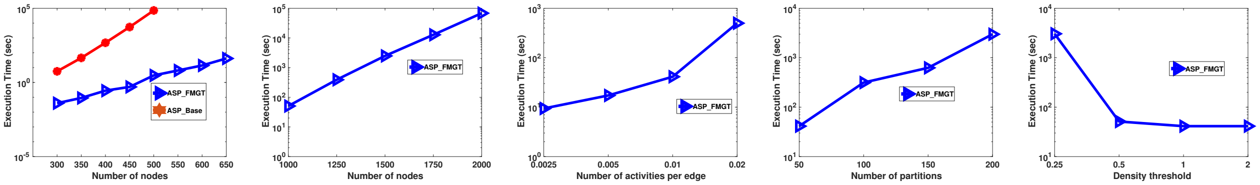
4.3 Refine phase

The refine phase applies ASP_base on the pruned FMG-trees. Lastly, ASP_FMGT conducts Monte Carlo simulations for computing the statistical significance of each hotspot candidates.

4.4 Path reduction

Non-repeated fragment path reduction (ASP_NR): This approach enforces that each fragment is passed only once in an FMG-tree. It aims at eliminating the simple paths that repeatedly pass through the same region (i.e., FMG-node) back-and-forth via different boundary nodes.

Highest density paths in FMG-node path reduction (ASP_HD): ASP_HD selects the only path that connects the inbound and outbound boundary nodes for a passed-by FMG-node to retain representative simple paths that are concatenated by a sequence of sub-paths with the highest density in each corresponding FMG-node. On the other hand, for root and leaf FMG-nodes which have only either outbound or inbound bounary nodes, it picks the path with the highest density starting from the boundary node.



(a) Network size (small) (b) Network size (large) (c) Activity density (d) Number of partitions (e) Density threshold

Figure 5: Experimental results under different parameters for ASP_Base and ASP_FMGT (y-axis in log scale)

5 CASE STUDY ON REAL-WORLD DATASET

We conducted a case study comparing the linear hotspots discovered by the proposed path reduction approaches (i.e., ASP_NR+_ASP_HD) to the result returned by SaTScan [5] on a real-world traffic accident dataset which contained 1345 traffic accidents that occurred on major roads in Hennepin County (shown in Figure 4(a)), Minnesota, USA from 2010 to 2015 [9]. As shown in Figure 4(b), SatScan returns only one significant hotspot (p-value = 0.01) covering most of the metro area. This result does not provide useful information about individual roads prone to high traffic accident rates. In contrast, ASP_NR+_ASP_HD was able to locate the streets with statistically significant high accident density. Figures 4(c) and 4(d) are the maps of the linear hotspots longer than 500 meters using 0.01 and 0.05 as the p-value threshold, respectively. Previous studies [8] can validate the detected roads, which for example, identify “Lake St” and “Lyndale Ave” as streets whose intersections have the highest pedestrian total number. We also detected some new accident hotspots (e.g., Como Ave) that were previously unknown to domain experts.

6 EXPERIMENTAL EVALUATION

Experimental Setup: Each experiment varied one of the four parameters, namely network size (default = 650 nodes), activity density (default = 0.01), number of partitions (default = 50), and density ratio (default = 1.0) while setting the rest to their default values. The spatial networks used in the experiments were synthetic networks whose nodes were distributed following complete spatial randomness and 95% of the nodes were 2-degree nodes, mimicking real-world road networks. Monte Carlo simulations are not included in the costs as they simply multiply the total cost by a constant (e.g., 99). The algorithms were implemented in Java and were executed on an Intel Core i7 2.5 GHz CPU and 32 GB RAM.

Experimental Results: *Effect of network size (smaller networks):* As shown in Figure 5(a), ASP_FMGT was approximately 10^2 times faster than ASP_Base at start and the speedup got larger and reached 10^5 faster with 500 nodes. *Effect of network size (larger networks):* Figure 5(b) shows that the costs increased near linearly with the network size. In similar execution time (e.g, 10^5 seconds), ASP_FMGT handled a network 4 times larger than ASP_Base did. *Effect of activity density:* As shown in Figure 5(c), the cost first grew slowly and then faster as the density was larger. The reason was that the pruning rate remained very high at first and then dropped as the density increased further. *Effect of number of partitions:* As shown in Figure 5(d), the cost first increased since the increased number of partitions led to larger FMG-trees and a larger cost in the pruning phase. *Effect of density threshold:* Figure 5(e) shows that the cost dropped dramatically at first and then decreased slowly. This happened because the pruning rate significantly increased when the density threshold increased from 0.25 to 0.5.

7 CONCLUSION AND FUTURE WORK

This paper investigated the problem of linear hotspot detection on all simple paths (LHDA) that is important in many societal applications (e.g., transportation engineering, public health). To solve this problem, we proposed ASP_FMGT based on bi-directional fragment-multi-graph tree traversal and two path reduction approaches, ASP_NR and ASP_HD. Experiments demonstrated that ASP_FMGT achieves substantial improvement in scalability compared to the baseline ASP_Base without any loss of completeness or correctness. A case study on a real-world dataset confirmed that ASP_FMGT locate patterns with statistical significance and accuracy, and find previously missed statistically significant patterns.

In future, we plan to study other metrics beside density such as density ratio and log likelihood ratio [5] in terms of computational cost and result quality. We also plan to discover spatio-temporal linear hotspots which may potentially reveal the life-cycle and moving trends of hotspots using the additional temporal information.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grants No. 1901099, IIS-1320580, the National Geospatial-Intelligence Agency under Grant No. HM0210-13-1-0005. We would also like to thank Kim Koffolt and the spatial computing research group for their helpful comments and refinements.

REFERENCES

[1] 2017. Surveillance, Epidemiology, and End Results Program. https://https://seer.cancer.gov/, National Cancer Institute.
[2] Michelle Ernst, M Lang, and S Davis. 2011. Dangerous by design: solving the epidemic of preventable pedestrian deaths. *Transportation for America: Surface Transportation Policy Partnership, Washington, DC.* (2011).
[3] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *Kdd*, Vol. 96. 226–231.
[4] Winnie Hu. 2017. No Longer New York City‘s ‘Boulevard of Death’. https://nyti.ms/2AtYAWr, New York Times.
[5] Martin Kulldorff, Katherine Rand, Greg Gherman, Gray Williams, and David DeFrancesco. 1998. SaTScan v 2.1: Software for the spatial and space-time scan statistics. *Bethesda, MD: National Cancer Institute* (1998).
[6] Becky PY Loo and Shenjun Yao. 2013. The identification of traffic crash hot zones under the link-attribute and event-based approaches in a network-constrained environment. *Computers, Environment and Urban Systems* 41 (2013), 249–261.
[7] Daniel B Neill and Andrew W Moore. 2004. Rapid detection of significant spatial clusters. In *Proceedings of the tenth ACM SIGKDD*. ACM, 256–265.
[8] City of Minneapolis. 2017. Pedestrian Crash Study 2017. https://bit.ly/2lVtVxR
[9] Minnesota Department of Transportation. 2018. Minnesota Crash Mapping Analysis Tool. https://www.dot.state.mn.us/stateaid/crashmapping.html
[10] Xun Tang, Emre Eftelioglu, Dev Oliver, and Shashi Shekhar. 2017. Significant linear hotspot discovery. *IEEE Transactions on Big Data* 3, 2 (2017), 140–153.
[11] Xun Tang, Emre Eftelioglu, and Shashi Shekhar. 2017. Detecting Isodistance Hotspots on Spatial Networks: A Summary of Results. In *SSTD*. Springer, 281–299.
[12] Xun Tang, Jayant Gupta, and Shashi Shekhar. 2019. *Linear Hotspot Discovery on All Simple Paths: A Summary of Results.* Technical Report. 19-009, University of Minnesota, Computer Science and Engineering.
[13] Long Tien Truong and Sekhar VC Somenahalli. 2011. Using GIS to identify pedestrian-vehicle crash hot spots and unsafe bus stops. *Journal of Public Transportation* 14, 1 (2011), 6.