Towards Understanding City-Level Routing using BGP Location Communities

THOMAS KRENC*, IIJ Research Laboratory, Japan
SHIVANI HARIPRASAD, UC San Diego, United States
MATTHEW LUCKIE, UC San Diego, United States
BENOIT DONNET, University of Liège, Belgium
KC CLAFFY, UC San Diego, United States

BGP communities are widely used by operators to encode routing metadata for traffic engineering, policy enforcement, and operational debugging. However, ≈90% of observed communities lack public documentation, limiting their utility for research and operational analysis. Among these, city-level communities offer valuable geographic insight into routing behavior, yet remain largely untapped. In this paper, we develop a scalable method to infer the geographic meaning of undocumented city communities using BGP data. We validate our approach against a ground truth dataset covering 1,482 city communities and through operator feedback. Applied to data from May 2025, our algorithm infers the locations of 80% of city communities with a precision of 70 km or better. We publish all code and datasets to support reproducibility and further research.

CCS Concepts: • **Networks** → **Routing protocols**; **Network measurement**.

Additional Key Words and Phrases: Border Gateway Protocol (BGP), BGP communities.

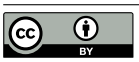
ACM Reference Format:
Thomas Krenc, Shivani Hariprasad, Matthew Luckie, Benoit Donnet, and kc claffy. 2025. Towards Understanding City-Level Routing using BGP Location Communities. *Proc. ACM Netw.* 3, CoNEXT4, Article 51 (December 2025), 13 pages. https://doi.org/10.1145/3768998

1 Introduction

One of the most critical metrics for understanding inter-domain routing is the Autonomous System Path (AS path), a sequence of ASes that agree to forward traffic toward a destination. The AS path is used by the Border Gateway Protocol (BGP) when selecting best paths, as BGP prefers routes with shorter AS paths. However, despite its utility, the AS path lacks geographic context which is a significant limitation given that neighboring ASes often peer at multiple, geographically distinct locations. To incorporate geographic context into routing decisions, many network operators tag routes with city-level information using BGP communities. These communities indicate where a route was learned and can be leveraged to optimize internal routing strategies. To facilitate debugging, some ASes export these communities and they are often captured by route collectors. However, while some networks publicly document their community values, many do not, leaving a significant portion of this data unexplored. Inferring the locations of unknown city communities can substantially improve our understanding of inter-domain routing.

*Work undertaken in part while a postdoctoral researcher at UC San Diego.

Authors' Contact Information: Thomas Krenc, IIJ Research Laboratory, Tokyo, Japan, tkrenc@iij.ad.jp; Shivani Hariprasad, UC San Diego, CAIDA, La Jolla, CA, United States, shariprasad@ucsd.edu; Matthew Luckie, UC San Diego, CAIDA, La Jolla, CA, United States, mjl@caida.org; Benoit Donnet, University of Liège, Liège, Belgium, benoit.donnet@uliege.be; kc claffy, UC San Diego, CAIDA, La Jolla, CA, United States, kc@caida.org.



ACM 2834-5509/2025/12-ART51
https://doi.org/10.1145/3768998

Towards being able to infer the city signaled in communities, in this paper we aim to improve our understanding of the properties of city communities, by investigating the following research question: Do prefixes tagged with a city community originate at the same location as the tagging router? As expected, we find that border routers tag prefixes on ingress originated from all over the world. However, more importantly, we observe a spatial correlation between large groups of prefixes tagged with city communities and the location of the tagging router. We leverage this property and devise an algorithm that identifies the largest geographically cohesive group for each city community. We validate our method against a ground truth data set covering 1,482 city communities (defined by 36 ASes) found in public BGP data from May 2025, where it inferred the location of 80% of city communities with an error of 70 km or less. We also contact network operators to verify our inferences for city communities for which we have no ground truth. We further investigate cases where our algorithm fails, derive limitations and conjecture the use of prefixes in locations different to their geolocated region, e.g., through remote peering. We make our code and datasets publicly available at https://github.com/tkrenc/conext25-artifacts.

2 Background

The Internet consists of over 80K Autonomous Systems (AS), each identified by an Autonomous System Number (ASN) [7]. ASNs were 16-bit integers, limiting the number of public ASes to less than 65535 ASes, so to accommodate more ASes participating in the Internet, the ASN space was expanded to 32 bits [40] in 2012. ASes exchange traffic based on business relationships, including peer-to-peer (P2P) and customer-provider (C2P) [12]. Two ASes can have different relationships at many different geographic locations [16]. In this work, we aim to infer those locations.

BGP Communities: The BGP communities attribute is a variable-length optional extension that augments BGP announcements with metadata [7]. A 32-bit BGP community has the form $\alpha:\beta$, where α identifies the ASN that defines the meaning of β. To accommodate 32-bit ASNs, the IETF introduced 48-bit extended communities [39] and 96-bit large communities, the latter of which follow the form $\alpha:\beta:\gamma$ [33]. With the exception of a few well-known communities (e.g., NO_EXPORT, BLACKHOLE [7, 22]), the values (α, β, and γ) are not standardized. Hence, operators assign semantics to otherwise arbitrary values, which other networks must interpret through documentation (e.g., [2] for AS1299 communities) in order to effectively use them. Operators of large, multi-regional networks typically use communities to implement complex routing policies that enable fine-grained control over ingress and egress traffic. Because the community attribute is transitive, it can be used by other ASes along the AS path. Based on the intended use of communities, operators distinguish them into two coarse-grained categories—*action* and *information* [33].

Action communities encode specific actions that the defining AS supports, such as exporting or suppressing a route, adjusting the local preference, or prepending the AS path. Additionally, they can define the intended target of these actions, such as a particular AS or groups of ASes within a specific location or region. For example, the action community `1299:2569` [2] can be used by a customer of AS1299 to instruct AS1299 to not export the tagged route to AS3356 in Europe.

Information communities are set by the defining AS and describe neighboring ASes, AS relationships, ROV status, or attributes of BGP sessions and routers to support functions such as preventing route leaks through relationship-aware policies. Geographic communities, the set of information communities that we focus on in this paper, identify peering facilities, cities, countries, or continents where a route is learned. For example, `3356:2073` [5] indicates that AS3356 received the route in London (GB). Operators can use geographic communities to implement cold-potato routing [36], where traffic is carried across the network to the location closest to the destination before handoff. In contrast, hot-potato routing hands traffic off at the nearest exit point. Cold-potato routing thus provides the key intuition for our city community inference method, which we elaborate on in §5.

3 Related Work

Over the past two decades, BGP communities have gained increasing attention from network operators. Between 2004 and 2007, the number of ASes defining communities grew from ≈400 to ≈1K, while unique communities increased from ≈3K to ≈7K [11]. This trend continued: from 2010 to 2018, ASes defining communities nearly doubled from ≈2.5K to ≈5K, while unique values grew from ≈19K to ≈63K [38]. This increase has attracted interest also in the research community for BGP-related studies [3, 10, 12, 15, 16, 18–20, 23, 27], infrastructure research [1, 18, 24, 30], and security [14, 17, 21, 38].

Efforts to compile a BGP community dictionary from published documentation date back to 2002, when Quoitin *et al.* [32] provided a taxonomy of early community uses based on the RIPE whois database. Later, Donnet *et al.* [11] manually built a dictionary of BGP communities using information published on network operator websites and in Internet Routing Registry records, which covered 22% of ≈7K communities observed in BGP data. In 2024, Liu *et al.* [26] proposed the first automated method to build a dictionary of BGP communities using public information. They reported that their dictionary covered 11% of ≈89K communities observed in BGP data, leaving significant potential for automatically inferring community semantics.

In an effort to automatically infer unknown BGP communities, Krenc *et al.* [25] and da Silva Jr *et al.* [8] introduced methods to distinguish information from action communities, using the observation that action communities often appear off-path. Similarly, da Silva Jr *et al.* [9] proposed a method to automatically infer *whether* a community signals geographic information. They coarsely label any such community with "location," covering cases from a specific link or router up to an entire continent. In contrast, our goal is to infer the specific location itself, at city-level.

4 Datasets and Preprocessing

In this work, we use three datasets: a dictionary of communities that signal city information, BGP routing data, and prefix geolocation data. We preprocess and combine these into a single dataset used to evaluate our inference method.

City communities dictionary (CITY): We manually compiled a list of city communities that encode city-level information. We searched registries [31], bgp.tools [6], the NLNOG Git repository [37], and operator websites for BGP community descriptions. We used available descriptions to determine whether a community signals a city or a PoP/peering facility within a city, and considered only those with descriptions that explicitly state city names. For example: we consider "1299:35400,Los Angeles (Customer)" a city community while "3356:70 Set BGP Local Pref to 70" not. For each city community, we recorded the city name and determined latitude and longitude coordinates. Our community dictionary comprises 3,133 city communities from 50 ASes, signaling 638 cities in 134 countries.

Routing data (BGP): We downloaded BGP data recorded by RouteViews [35] and RIPE RIS [34] collectors. Our method relies on RIB dumps (i.e., snapshots of stable routing state) taken at midnight UTC for each date considered in this work, since transient BGP churn (e.g., path exploration) can reveal less-preferred paths that are poorly correlated with the actual location where a prefix is learned. To avoid such transient noise, we exclude BGP updates and use only RIBs as cleaner input for route tagging. From each session in these RIB dumps, we extracted route entries containing only the prefix, AS path, and BGP communities. We focus on IPv4 routes, which comprise the majority of entries in BGP data, and leave IPv6 for future work. As of May 1, 2025, we collected BGP snapshots from 1,035 sessions, yielding a total of ≈439.5M routes. To test the stability of our algorithm over time, we also obtained yearly routing data dating back to 2017.

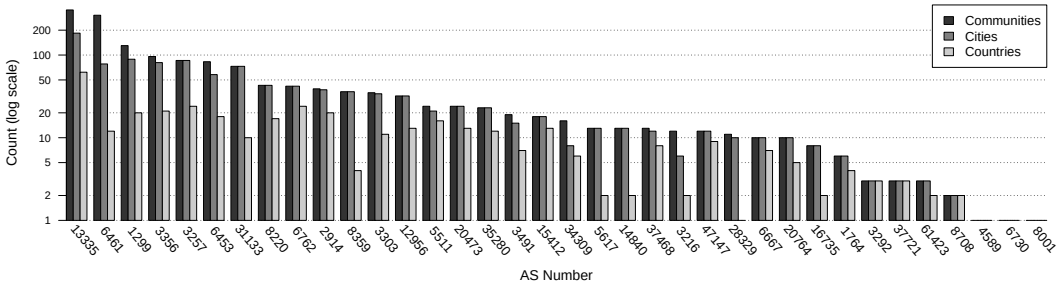


Fig. 1. Distribution of documented communities across selected autonomous systems (ASes) observed in BGP data from May 2025 (AGG). For each AS, bars indicate the number of unique communities, cities, and countries, highlighting operator-specific differences in how geographic granularity is encoded in BGP communities.

MaxMind's prefix geolocation data (GEO): We obtained the GeoLite2 geolocation database from MaxMind [29], a free, open, and regularly updated source of geolocation information that enables reproducibility. It provides geolocation data for the currently routed address space observed in BGP. The snapshot for May 2025 contains ≈3.36M non-overlapping IPv4 prefixes. For each prefix, the dataset includes geographic coordinates, i.e., a (latitude, longitude) pair, and an accuracy radius, among other fields. As with the routing data, we collected yearly snapshots of this database dating back to 2017. We acknowledge that GeoLite2 may be inaccurate at fine granularity (<50 km), and its accuracy radius information is not independently validated by peer-reviewed work. Accordingly, we treat MaxMind locations as heuristic approximations rather than ground truth.

Aggregated dataset (AGG): We combined the three datasets—CITY, BGP, and GEO—into a new dataset called AGG. CITY, our community ground truth dictionary, was used to filter routes in BGP containing city communities and assign each the corresponding latitude, longitude, and city name. GEO was then used to match prefixes in BGP and assign each the latitude, longitude, and accuracy radius. Because BGP and GEO may cover the same address space with different prefix lengths, we matched more specific prefixes in BGP with less specific ones in GEO, discarding BGP prefixes that were less specific than their GEO counterparts. The resulting AGG dataset, used to evaluate our inference method under different parameter settings in §6.2, was derived from ≈111.5M routes (25%) across 561 sessions (54%) and contained 936,918 geolocated IPv4 prefixes tagged with 1,595 unique city communities.

Fig. 1 shows the distribution of unique communities as well as the signaled cities and countries in the AGG dataset. The 1,595 city communities are defined by 36 ASes and signal 394 unique cities across 69 countries. AS13335 (Cloudflare, Content) and AS6461 (Zayo, Tier 1 Carrier/ISP) yield 351 and 303 unique cities, respectively, with both using multiple communities to signal identical cities. Almost half of the ASes are represented with 20 or more unique city communities. AS28329 (Megatelecom, Eyeball) shows city communities within a single country only (Brazil). Three ASes are represented with a single city community. Table 1 summarizes the network types and inference results for each AS.

5 Motivating Analysis

In this section, we investigate the spatial correlation between the geolocation of tagged prefixes and their tagging router, which could help infer the locations indicated by unknown city communities. Consider the scenario illustrated in Fig. 2. A customer network connects to its upstream provider, AS3356, via London ($R1$–$R7$) and Rome ($R2$–$R8$). Further, the customer originates the prefix P/16

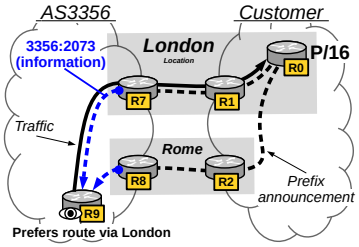


Fig. 2. *Intuition*: AS3356 learns prefix P/16 via *R*7 and *R*8. To offload the egress traffic close to the destination, router *R*9 selects the route learned via London by matching the community `3356:2073` with the geolocation of P/16.

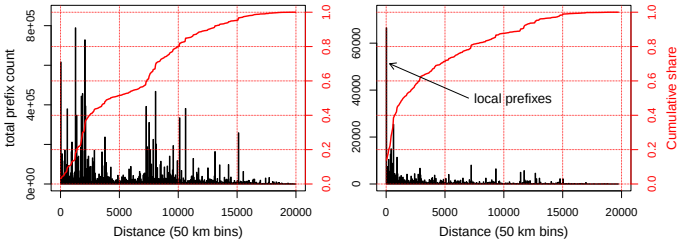


Fig. 3. *Motivating analysis*: Distances between tagging routers and tagged prefix origins grouped in 50 km bins, using ground truth city communities, prefix geolocation data, and BGP prefixes from May 2025. CDFs show relative cumulative share of those prefixes. **Challenge**: Tagged prefixes originate near and far in relation to the tagging router (left). We devise and apply our method to accentuate *local prefixes* from other prefixes (right).

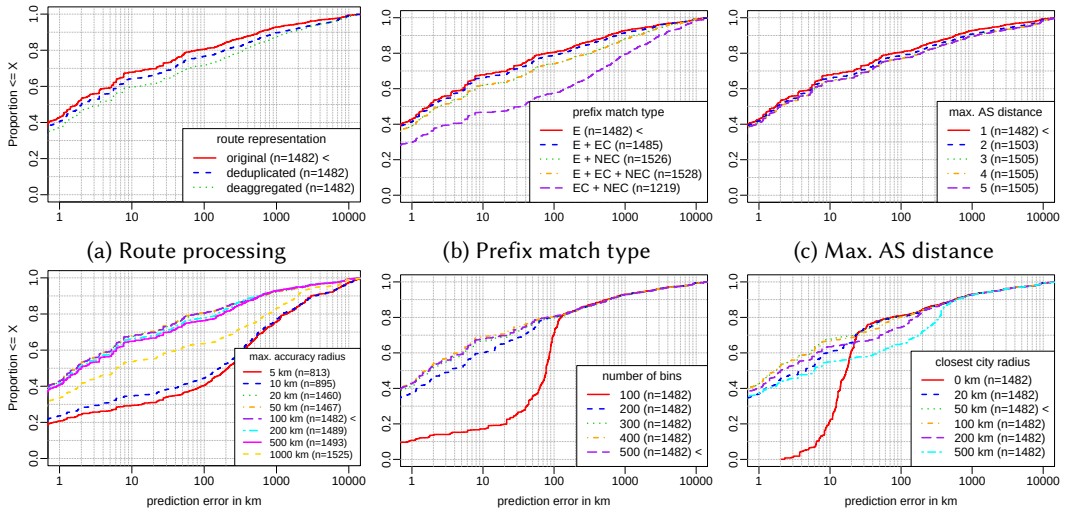
in London (*R*0) and exports it to AS3356 at both peering locations, London and Rome, to ensure connectivity in case one link fails. The customer offloads the traffic sourced at router *R*0 at the nearest exit point, London, to minimize internal transit time and cost, following a hot-potato routing policy. To keep inbound traffic symmetric and cost-efficient, the customer wants return traffic to reach *R*0 via London rather than Rome. In this scenario, the customer can rely on cold-potato routing offered by AS3356. This can be achieved in two ways. 1) AS3356 can prefer the route that was learned the closest to the destination P/16, by matching P/16's geolocation against city communities set by *R*7 and *R*8. Since *R*7 is located closer to P/16, AS3356 will prefer that route. 2) If AS3356 does not perform the above by default, the customer can apply action communities, e.g., 3356:90 [5] to remotely decrease the local preference to 90 (default 100) in *R*8 for P/16. In BGP, the route with the highest local preference is preferred. In either case, router *R*9 will prefer prefix P/16 tagged with the city community 3356:2073 [5], which we will likely observe at route collectors:

Prefix	Session	Peer	AS Path	Community
P/16	R9	3356	3356 Customer	3356:2073

Next, we substantiate this intuition with empirical data. Fig. 3 shows histograms of the distances between tagged prefixes and their tagging routers, based on the aggregated dataset AGG (§4). We group distances into 50 km bins. The plots display both the frequency of tagged prefixes (left axis) and their relative cumulative share (right axis). The *left plot* shows prefixes from all routes available in AGG. While we observed that over 600K prefixes were located within 50 km of their tagging router (*local prefixes*), they made up less than 5% of all tagged prefixes. Overall, prefixes were geolocated across a wide range—from near the tagging router to as far away as 20,000 km. Notably, distance bins between 2,000 and 10,000 km contain between 400K and 800K tagged prefixes. The *right plot* shows prefixes from a subset of routes in AGG that, overall, are located closer to the tagging router. §6 details our method for selecting *local prefixes* and additional fine-tuning steps, which enabled us to geolocate 80% of city communities with ground truth locations at an error of 70 km or less.

6 Methodology

This section outlines our approach for inferring the locations signaled by city communities and evaluates the impact of different parameter settings.



(a) Route processing (b) Prefix match type (c) Max. AS distance

(d) Max. accuracy radius (e) Number of bins (f) Closest city radius

Fig. 4. Performance evaluation using precision (error distance in km) and recall (number of communities with results). Each parameter is tested with different values while the others are fixed at their *default setting*, indicated by the "<" symbol in the legends.

6.1 Approach Overview

We infer the city associated with each community by analyzing the geographic distribution of BGP prefixes tagged with it. The approach consists of three main steps.

(Step I) Obtain tagged and geolocated BGP routes: Obtain BGP route information and select those prefixes that are tagged with city communities and geolocated with geographic coordinates (i.e., latitude and longitude).

(Step II) Cluster geographic locations of tagged prefixes: Cluster the coordinates of prefixes associated with each city community using a two-dimensional geographic frequency matrix, where the axes represent latitude and longitude and each cell value corresponds to the number of prefixes at that location. Use equal-distance bins that account for the Earth's curvature and non-uniform degree spacing.

(Step III) Extract densest location: Identify the center coordinates of the densest cluster for each city community by selecting the element in the matrix with the highest frequency. These coordinates then serve as the basis for determining the closest city.

6.2 Individual Parameters

We evaluated the *performance* of our approach under different parameter settings for route representation (§6.2.1), prefix selection (§6.2.2), and city inference (§6.2.3). Performance is measured by *precision* (the error distance between the inferred and ground truth city locations) and *recall* (the number of inferred city communities out of 1,595 in AGG). Fig. 4 shows results for each parameter with the others fixed at their optimal *default settings* (indicated by the "<" symbol in the legends). Applied to data from May 1, 2025, our method achieves an 80th-percentile error of ≈70 km and a recall of 1,482 out of 1,595 communities.

6.2.1 Route representation. We evaluate how alternative representations of BGP routes in AGG affect prefix counts in the matrix and the resulting precision.

Deduplicating sessions: Since we consider all IPv4 sessions from the various RouteViews and RIPE RIS collectors, we observe prefixes tagged by the same community in different sessions. For example, the prefix P/16 tagged with community T:CITY can occur on multiple paths:

```
Prefix  | Session   | Peer | AS Path  | Community
=============================================
P/16    | 192.0.2.1 | A    | A B T O  | T:CITY
P/16    | 192.0.2.2 | B    | B T O    | T:CITY
P/16    | 192.0.2.3 | C    | C T O    | T:CITY
```

where O is the origin AS, T is the tagging AS, and A, B, C are three different peers. In this example, we count the prefix three times, once per session, reflecting our default approach. In addition, we test an approach that focuses only on distinct P/16-T:CITY pairs, assigning a count of 1 in the frequency matrix and effectively deduplicating sessions that provide the same information.

Deaggregating prefixes: Since our approach relies on the highest frequency cell in the matrix, simply counting tagged prefixes may be misleading when a network aggregates hundreds of prefixes into a single one, e.g., 256 /24s into a /16. Hence, we also evaluate performance when we instead count the number of /24s covered by the BGP prefixes.

Fig. 4a shows the results of deduplicating sessions (blue dashed line) as well as by deaggregating prefixes into /24 blocks (green dotted line), contrasted against the original routes (red solid line) provided in AGG. We find that both approaches improve the inferences for some communities, but the overall performance decreases. The error increases in particular at the upper percentiles, e.g., from 100 km to 500 km. We conclude that the precision of our approach relies on BGP routes without deduplication or deaggregation. The recall is not impacted.

6.2.2 Prefix selection. To select local prefixes, we test three parameters: prefix overlap, AS distance between the origin and tagging AS, and accuracy radius. These parameters influence recall by affecting the number of selected prefixes tagged with a city community that are available in AGG for inference.

Overlapping prefixes: MaxMind's geolocation database (GEO) provides geolocation for non-overlapping prefixes. We define three matching types for BGP prefixes:

```
BGP   | GEO   | match type
=======================================
P/16  | P/16  | Exact (E)
P/24  | P/16  | Exact / Covered (EC)
Q/24  | Q/16  | Not-Exact / Covered (NEC)
```

In the example, BGP contains the prefixes P/16, P/24, and Q/24, while GEO contains P/16 and Q/16. Here, P and Q denote two non-overlapping prefixes; P/16 covers P/24, and analogously Q/16 covers Q/24. P/16 in BGP is exactly matched by P/16 in GEO; the match type is E. P/24 in BGP is covered by P/16, which in turn matches P/16 in GEO; the match type is EC. Q/24 in BGP is covered by Q/16 in GEO, but Q/16 itself is not present in BGP; the match type is NEC. For the more specific BGP prefixes P/24 and Q/24, we simply transfer the geolocation information from the respective less specific GEO prefix. In AGG, E-matched prefixes account for 24% of all matched prefixes. We do not consider BGP prefixes that are less specific in relation to GEO (§4).

Fig. 4b shows a comparison of performance results for combinations of the three different match types. We find that using only E matches (default) yields the best performance, while adding NEC slightly decreases precision, but increases recall (+44). Further, using EC has no significant impact, while using only covered prefixes (EC+NEC) yields the worst precision and recall (-263).

Maximum AS distance: The AS distance corresponds to the number of AS hops from the origin AS O and the tagging AS T. Intuitively, the further away these two ASes are, the less likelier it is that the prefix is geolocated near the tagging router. Fig. 4c shows that the increase of the maximum AS distance 1 (default) to 2 increases the recall (+23), but decreases the precision of the upper percentiles, e.g., from ≈80 km to ≈150 km for the 80th percentile.

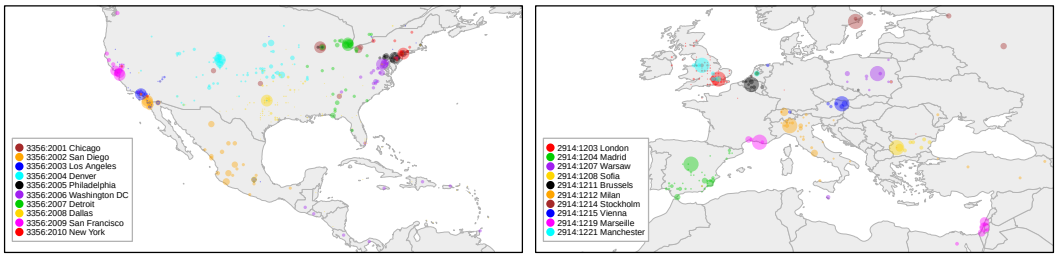


Fig. 5. Regional maps showing the functioning of our clustering approach using the geographic frequency matrix (step II), by the example of 10 city communities for AS3356 in North America (left) and AS2914 in Europe (right) each. The size of the circle is normalized per community and indicates the number of clustered prefixes, i.e., the frequency. We use the largest cluster to predict the location signaled in the respective community (step III).

Maximum accuracy radius: The accuracy radius is a parameter specific to MaxMind's geolocation database used along with the geolocation coordinates to constrain the mapped geolocation area [28]. The possible radii are 5, 10, 20, 50, 100, 200, 500, and 1000 km. For example, an accuracy radius of 20 km indicates that MaxMind believes the geolocation is constrained within a radius of 20 km from the geolocated coordinates. MaxMind assigns an accuracy radius of 1,000 km and uses the geographic coordinates of the country's center where the owning company is headquartered for prefixes it could not geolocate. Fig. 4d shows that a maximum accuracy radius of 100 km (default) yields the best performance. While higher radii increase the recall (e.g. +7 for 200 km), they decrease the precision. Interestingly, a maximum radius of 1000 km—which is equivalent to disregarding this parameter for prefix selection—performs better then a maximum radius of 5 or 10 km, which yield the worst performance. Our results suggest that MaxMind derives the 5 and 10 km radii using a methodology that does not align with our inference approach. The limitations of lower accuracy radii are examined further in §7, where we conduct a longitudinal study using historical MaxMind data.

6.2.3 City Inference. We test the last two parameters, i.e., the bin size of the frequency matrix and the closest city radius. While they can affect the precision, they do not affect the recall.

Frequency matrix binning: The binning size of the geographic space (step II) can impact the frequency and thus the selection of the densest areas (step III). We test different numbers of bins, i.e., 100, 200, 300, 400 and 500 (default). Increasing the number of bins decreases the area covered by each bin. Fig. 4e shows that increasing the number of bins improves inference precision up to approximately the 80th percentile (i.e., for errors of 200 km or less).

Finding the closest city: Because the clustering process can produce multiple dense regions within or near a city (step III), we implement a k-d tree [4] to efficiently locate the most populated city center within a given radius of the inferred coordinates. We use a global city database covering all cities with at least 1,000 inhabitants [13]. Fig. 4f shows that radii between 20 and 100 km yield the best precision (default is 50 km) compared with not applying any radius (0 km). Radii above 100 km are detrimental to performance.

7 Analysis

Prefix clusters: To examine the effectiveness of our clustering approach, we analyze prefixes tagged with ten selected city communities from Tier-1 ASes 3356 and 2914 (Fig. 5). The largest

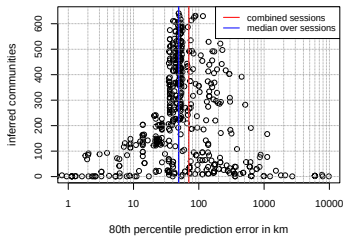


Fig. 6. Session comparison: 80th percentile error (x-axis) vs. number of inferred communities (y-axis) for individual sessions (circles). Half of the sessions yield error of 50 km or less.

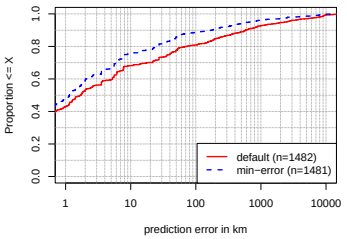


Fig. 7. Potential improvements of the 80th percentile prediction error from 70 km (combined sessions) to 25 km, if we considered the minimum error across individual sessions.

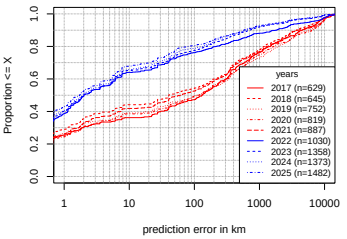


Fig. 8. Performance of our approach using data from 2017 to 2025. Changes in the MaxMind dataset have significant impact from 2021 to 2022.

clusters correspond to local prefixes that align with the geographic locations signaled by their respective communities. We also observe notable clusters geolocated in Mexico tagged as San Diego, in the Caribbean and Central America tagged as Washington DC, in Moscow or St. Petersburg tagged as Stockholm, and in Israel tagged as Marseille. These cases may be artifacts of the two networks' limited peering presence in those remote regions, with connectivity likely established through undersea cables or remote peering.

Individual BGP sessions: In §6.2.1, we found that our method benefits from combining many views from different vantage points and that deduplicating the routes decreases the precision. To further investigate the potential of individual sessions, we applied our method to each of the 561 individual sessions. Fig. 6 shows the number of unique communities (recall) on the x-axis and the 80th percentile error (precision) on the y-axis. No one session provided the necessary data to infer the location of all 1,482 communities. However, we find that half of the sessions yield an 80th percentile error of ≈50 km or less (blue line), compared to ≈70 km for the combined sessions (red line), highlighting the potential for more efficient use of individual sessions. Hence, we consider a hypothetical scenario in which the optimal session is known for each community, providing an upper-bound estimate of inference performance based on the datasets used in this study. In Fig. 7 we show the performance results for such a scenario where for each inferred community we pick the minimum error distance across all sessions (blue dashed line), and compare it against the results of our default approach (red solid line). This hypothetical scenario reduced the 80th percentile error from ≈70 km to ≈25 km.

Longitudinal Analysis: To investigate the stability of the algorithm, we evaluate its performance using different historical data sets with the same default parameter settings as for May 2025. We find that performance from 2022 to 2025 is comparable and gradually improves. For example, the 80th percentile prediction error decreases from ≈200 km to ≈70 km, while the number of inferred communities increases from 1,030 to 1,482. Surprisingly, we find that inference performance from 2017 to 2021 is significantly worse across all quantiles. Investigating possible root causes, we found that the accuracy radius distribution in MaxMind's geolocation database changed after 2021. We provide an overview of those changes in Table 2 of the appendix. We note that the databases included an accuracy radius of 1 km, which was discontinued in 2022. Subsequently, the number of entries with a 20 km accuracy radius increased. This aligns with our finding that a maximum accuracy radius of 10 yields the worst results (see Fig. 4), underscoring the importance for users of MaxMind's geolocation database to understand how accuracy radii are derived.

#	ASN	Type	Num	Median	P80	#	ASN	Type	Num	Median	P80
1	8708	Eyeball	2	0.33	0.53	18	8359	Eyeball	29	1.92	167.72
2	20473	Content	24	0.41	4.35	19	6667	Carrier	9	0.19	179.70
3	14840	Carrier	12	1.77	4.76	20	6730	Eyeball	1	190.09	190.09
4	2914	Tier-1	39	0.44	6.11	21	31133	Eyeball	64	1.72	190.52
5	35280	Carrier	23	0.63	7.09	22	3303	Eyeball	35	43.99	200.31
6	13335	Content	336	1.13	11.12	23	12956	Tier-1	30	4.50	220.43
7	20764	Carrier	9	0.66	18.61	24	6461	Tier-1	268	2.01	229.62
8	5511	Tier-1	22	0.93	25.21	25	28329	Eyeball	7	232.67	440.78
9	15412	Carrier	13	6.16	31.77	26	16735	Eyeball	8	22.49	519.94
10	3356	Tier-1	95	0.61	36.07	27	6762	Tier-1	40	2.58	595.13
11	8220	Carrier	42	1.21	38.41	28	37721	Eyeball	2	399.07	637.51
12	1764	Carrier	6	1.26	40.42	29	6453	Tier-1	78	7.71	926.18
13	3257	Tier-1	84	0.70	45.01	30	3292	Eyeball	2	617.33	987.63
14	5617	Eyeball	13	0.41	128.45	31	37468	Carrier	13	6.07	1055.07
15	1299	Tier-1	123	1.27	142.24	32	47147	Carrier	11	40.42	1186.15
16	61423	Content	3	40.42	162.18	33	3491	Tier-1	18	929.26	2877.41
17	3216	Eyeball	5	0.42	166.32	34	34309	Content	16	340.67	5264.34

Table 1. Breakdown of results by ASN, ordered by 80th percentile error in km.

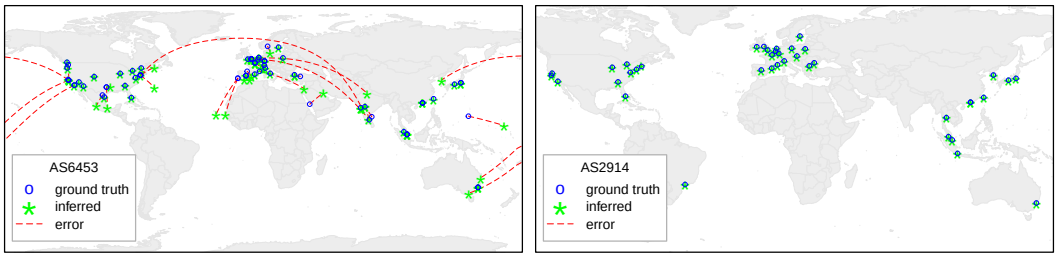


Fig. 9. Inference results for city communities of two Tier-1 ISPs, AS6453 and AS2914. Green asterisks show inferred locations; blue circles indicate ground truth (CITY); red dashed lines (great circles) represent inference errors.

Per ASN Analysis: Table 1 presents the inference results for 1,482 city communities, broken down by individual ASNs and ordered by 80th percentile error. Notably, no specific network type consistently performs particularly well or poorly. Extreme deviations for some ASes (e.g., 80th percentile > 1,000 km) highlight limitations for inferring unknown city locations, but when analyzed alongside ground truth data they can also reveal special routing configurations within the network. Fig. 9 illustrates this with AS6453 (Tier-1), which receives large clusters of prefixes from cities such as Melbourne and Brisbane (Australia) and Seoul (South Korea) via the US west coast, as well as prefixes from Praia and Dakar (West Africa) via Lisbon, and from Jerusalem via Marseille, among other cases. This stands in stark contrast to AS2914, another Tier-1 provider, where the largest clusters correspond to local prefixes, i.e., those originated near the tagging router.

Operator validation: To test our approach on communities without ground truth, we applied it to communities from eight ASes that we believed with high confidence signal city-level information. We then reached out the network operators of those ASes for which we obtained contact information through private channels. We asked whether they could validate the city name of five city communities that we selected randomly. We shared all our mappings with those operators. *Five out of eight operators*—4 Tier-1 and one eyeball—replied and generously explained their community schema to us. While our ground truth dataset shows that city communities are used across various network types (see Table 1), we acknowledge our limited operator outreach.

Operators A and B confirmed all 10 communities to be correct, with the note that one community signaling Gardena (60K inhabitants) was misclassified as Los Angeles (3.9M inhabitants). We note that Gardena, as part of the Greater Los Angeles metropolitan area, is closely linked to Los Angeles through shared infrastructure. Operator C confirmed 4 out of 5 communities to be correct. One community signaling Boston MA was misclassified as Worcester MA, ≈60 km away. Operator D reported that we inferred only 2 out of 5 city communities correctly. The remaining 3 communities were misclassified with an error distance of ≈80, ≈190 and ≈240 km, respectively. Overall, we consider 16 out of 20 communities to be inferred correctly. Finally, operator E (Tier-1) responded that the five communities we asked for feedback on actually identified individual customers rather than cities.

8 Conclusions

In this work, we introduced a novel method for clustering geo-cohesive prefixes tagged with the same city community, enabling finer-grained analysis of routing preferences as well as the identification of local prefixes to infer the geographic location signaled by city communities. Using ground truth data, we validated our method and were able to predict locations for 80% of 1,482 ground truth communities within a 70 km margin of error. We further applied our approach to city communities for which no ground truth is available and evaluated the results in collaboration with network operators from five ASes. While our method does not always pinpoint the exact location, it represents an important first step toward a better understanding of routing behavior at the city level. Future work includes the exploration of alternative geolocation databases, as well as improving the reliable identification of city-level communities in the wild.

Acknowledgments

This material is based on research sponsored by the National Science Foundation (NSF) grant OAC-2131987 and CNS-2120399. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF. This research was also supported by the Walloon Region through the CyberExcellence project (No. 2110186) and the *Les Semestres Thématiques* program.

The authors would like to thank the network operators who assisted in validating some of our inferences. We also gratefully acknowledge the providers of the datasets used in this study, in particular RouteViews and RIPE RIS, as well as MaxMind and GeoNames, for making their data publicly available and supporting the research community.

References

[1] E. Aben. 2022. BGP Community propagation. https://observablehq.com/@emileaben/bgp-community-propagation.
[2] Arelion. 2025. BGP Communities. https://www.arelion.com/our-network/bgp-routing/bgp-communities.
[3] B. Augustin, B. Krishnamurthy, and W. Willinger. 2009. IXPs: Mapped?. In *Proc. ACM Internet Measurement Conference (IMC)*.
[4] J. L. Bentley. 1975. Multidimensional Binary Search Trees Used for Associative Searching. *Commun. ACM* 18, 9 (1975), 509–517.
[5] bgp.tools. 2025. AS3356 BGP Communities. https://bgp.tools/communities/3356.
[6] bgp.tools. 2025. bgp.tools. https://bgp.tools/.
[7] R. Chandra, P. Traina, and T. Li. 1996. *BGP Communities Attribute.* RFC 1997. Internet Engineering Task Force.
[8] B. A. da Silva Jr, A. B. de Carvalho, I. Cunha, T. Friedman, E. Katz-Bassett, and R. A. Ferreira. 2024. Uncovering BGP Action Communities and Community Squatters in the Wild. *Proc. ACM on Measurement and Analysis of Computing Systems* (2024).
[9] B. A. Da Silva Junior, P. Mol, O. Fonseca, I. Cunha, R. A. Ferreira, and E. Katz-Bassett. 2022. Automatic Inference of BGP Location Communities. In *Proc. ACM SIGMETRICS*.

[10] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, kc claffy, and G. Riley. 2007. AS relationships: Inference and validation. *ACM SIGCOMM Computer Communication Review (CCR)* 37, 1 (January 2007), 29–40.
[11] B. Donnet and O. Bonaventure. 2008. On BGP Communities. *ACM SIGCOMM Computer Communication Review (CCR)* 38, 2 (April 2008), 55–59.
[12] L. Gao. 2001. On Inferring Autonomous System Relationships in the Internet. *IEEE/ACM Transactions on Networking (ToN)* 9, 6 (December 2001), 733–745.
[13] GeoNames. 2025. GeoNames Cities1000 Dataset. http://download.geonames.org/export/dump/cities1000.zip.
[14] V. Giotsas, C. Dietzel, G. Smaragdakis, A. Feldmann, A. Berger, and E. Aben. 2017. Detecting Peering Infrastructure Outages in the Wild. In *Proc. ACM SIGCOMM*.
[15] V. Giotsas, T. Koch, E. Fazzion, I. Cunha, M. Calder, H. V. Madhyastha, and E. Katz-Bassett. 2020. Reduce, Reuse, Recycle: Repurposing Existing Measurements to Identify Stale Traceroutes. In *Proc. ACM Internet Measurement Conference (IMC)*.
[16] V. Giotsas, M. Luckie, B. Huffaker, and kc claffy. 2014. Inferring complex AS relationships. In *Proc. ACM Internet Measurement Conference (IMC)*.
[17] V. Giotsas, G. Smaragdakis, C. Dietzel, P. Richter, A. Feldmann, and A. Berger. 2017. Inferring BGP Blackholing Activity in the Internet. In *Proc. ACM Internet Measurement Conference (IMC)*.
[18] V. Giotsas, S. Zhou, M. Luckie, and kc claffy. 2013. Inferring multilateral peering. In *Proc. ACM CoNEXT*.
[19] B. Huffaker and V. Giotsas. 2016. AS Relationships – with geographic annotations. http://www.caida.org/data/as-relationships-geo/.
[20] Y. Jin, C. Scott, A. Dhamdhere, V. Giotsas, A. Krishnamurthy, and S. Shenker. 2019. Stable and Practical AS Relationship Inference with ProbLink. In *Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.
[21] M. Jonker, A. Pras, A. Dainotti, and A. Sperotto. 2018. A First Joint Look at DoS Attacks and BGP Blackholing in the Wild. In *Proc. ACM Internet Measurement Conference (IMC)*.
[22] T. King, C. Dietzel, J. Snijders, G. Doering, and G. Hankins. 2016. *BLACKHOLE Community*. RFC 7999. Internet Engineering Task Force.
[23] T. Krenc, R. Beverly, and G. Smaragdakis. 2020. Keep your Communities Clean: Exploring the Routing Message Impact of BGP Communities. In *Proc. ACM CoNEXT*.
[24] T. Krenc, R. Beverly, and G. Smaragdakis. 2021. AS-level BGP Community Usage Classification. In *Proc. ACM Internet Measurement Conference (IMC)*.
[25] T. Krenc, M. Luckie, A. Marder, and kc claffy. 2023. Coarse-grained Inference of BGP Community Intent. In *Proc. ACM Internet Measurement Conference (IMC)*.
[26] Y. Liu, T. Wu, J. H. Wang, J. Wang, and S. Zhuang. 2024. Collecting Self-reported Semantics of BGP Communities and Investigating Their Consistency with Real-world Usage. In *Proc. ACM Internet Measurement Conference (IMC)*.
[27] M. Luckie, B. Huffaker, A. Dhamdhere, V. Giotsas, and kc claffy. 2013. AS relationships, customer cones, and validation. In *Proc. ACM Internet Measurement Conference (IMC)*.
[28] MaxMind. 2022. Using MaxMind's accuracy radius. https://blog.maxmind.com/2022/06/using-maxminds-accuracy-radius/.
[29] MaxMind. 2025. GeoLite2 Free Geolocation Data. https://dev.maxmind.com/geoip/geolite2-free-geolocation-data.
[30] F. Mazzola, P. Marcos, and M. Barcellos. 2022. Light, Camera, Actions: characterizing the usage of IXPs' action BGP communities. In *Proc. ACM CoNEXT*.
[31] Merit Network, Inc. 2025. RADb Routing Assets Database. https://www.radb.net/. Accessed October 2025.
[32] B. Quoitin and O. Bonaventure. 2002. *A Survey of the Utilization of the BGP Community Attribute*. Internet Draft (Work in Progress) draft-quoitin-bgp-comm-survey-00. Internet Engineering Task Force.
[33] R. Raszuk, J. Haas, E. Aman, B. Decraene, J. Uttaro, and R. Bush. 2017. *BGP Large Communities Attribute*. RFC 8092. Internet Engineering Task Force.
[34] RIPE. 2025. RIS - RIPE Network Coordination Centre. http://ris.ripe.net/.
[35] RouteViews. 2025. University of Oregon RouteViews project. http://www.routeviews.org/.
[36] J. Scudder and E. Chen. 2006. *Experience with the BGP-4 Protocol*. RFC 4277. Internet Engineering Task Force.
[37] Stichting NLNOG. 2025. NLNOG Looking Glass. https://github.com/NLNOG/lg.ring.nlnog.net.
[38] F. Streibelt, F. Lichtblau, R. Beverly, A. Feldmann, C. Pelsser, G. Smaragdakis, and R. Bush. 2018. BGP Communities: Even more Worms in the Routing Can. In *Proc. ACM Internet Measurement Conference (IMC)*.
[39] D. Tappan, Y. Rekhter, and S. R. Sanglli. 2009. *4-Octet AS Specific BGP Extended Community*. RFC 5668. Internet Engineering Task Force.
[40] Q. Vohra and E. Chen. 2012. *BGP Support for Four-Octet Autonomous System (AS) Number Space*. RFC 6793. Internet Engineering Task Force.

A Historical MaxMind Geo-Databases

Year	Prefixes	1	5	10	20	50	100	200	500	1000
2017	2,949,311	297,335	267,346	274,273	340,639	413,535	207,505	308,705	193,941	644,516
2018	3,073,411	150,475	237,124	278,544	379,868	444,609	291,326	343,094	205,566	740,822
2019	3,209,492	182,092	245,099	311,912	478,900	531,299	414,125	391,918	227,448	424,794
2020	3,153,099	193,575	257,286	333,893	492,789	516,774	445,279	374,954	197,513	339,457
2021	3,446,025	204,877	272,368	340,993	519,912	560,158	527,061	449,167	207,044	362,651
2022	3,552,723	–	381,054	286,932	1,183,956	402,237	409,804	357,644	218,664	310,315
2023	3,734,440	–	393,814	279,390	1,301,957	418,320	433,953	363,472	213,671	329,747
2024	2,409,296	–	378,609	218,220	818,217	203,358	204,453	142,913	106,025	337,044
2025	3,363,963	–	327,880	226,760	1,435,531	276,561	277,990	336,023	149,360	333,465

Table 2. Prefix total and accuracy radius counts per year (2017–2025).

Received June 2025; accepted September 2025