Graph Sampling for Map Comparison

JORDI AGUILAR, Universitat Politècnica de Catalunya, Spain
KEVIN BUCHIN, TU Dortmund, Germany
MAIKE BUCHIN, Ruhr University Bochum, Germany
ERFAN HOSSEINI SERESHGI, Tulane University, USA
RODRIGO I. SILVEIRA, Universitat Politècnica de Catalunya, Spain
CAROLA WENK, Tulane University, USA

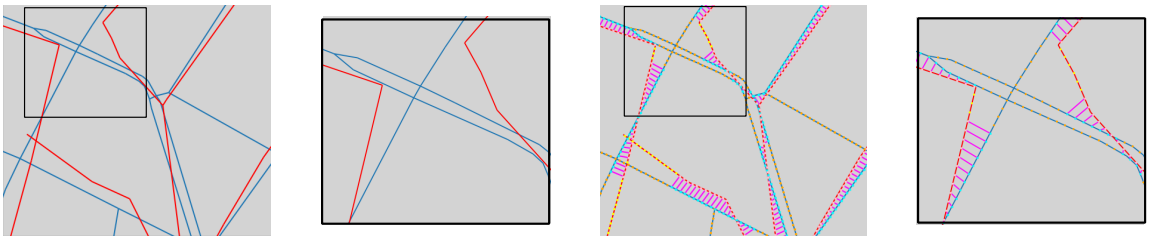


Fig. 1. Two road maps (red, blue) and graph sampling results: matched points are connected by pink lines, unmatched points are yellow or orange.

Comparing two road maps is a basic operation that arises in a variety of situations. A map comparison method that is commonly used, mainly in the context of comparing reconstructed maps to ground truth maps, is based on *graph sampling*. The essential idea is to first compute a set of point samples on each map, and then to match pairs of samples—one from each map—in a one-to-one fashion. For deciding whether two samples can be matched, different criteria, e.g., based on distance or orientation, can be used. The total number of matched pairs gives a measure of how similar the maps are.

Since the work of Biagioni and Eriksson [11, 12], graph sampling methods have become widely used. However, there are different ways to implement each of the steps, which can lead to significant differences in the results. This means that conclusions drawn from different studies that seemingly use the same comparison method, cannot necessarily be compared.

In this work we present a unified approach to graph sampling for map comparison. We present the method in full generality, discussing the main decisions involved in its implementation. In particular, we point out the importance of the sampling method (GEO vs. TOPO) and that of the matching definition, discussing the main options used, and proposing alternatives for both key steps. We experimentally evaluate the different sampling and matching options considered on map datasets and reconstructed maps. Furthermore, we provide a code base and an interactive visualization tool to set a standard for future evaluations in the field of map construction and map comparison.

Authors' Contact Information: Jordi Aguilar, Universitat Politècnica de Catalunya, Spain, jordi.aguilar.larruy@estudiantat.upc.edu; Kevin Buchin, TU Dortmund, Germany, k.a.buchin@tue.nl; Maike Buchin, Ruhr University Bochum, Germany, maike.buchin@rub.de; Erfan Hosseini Sereshgi, Tulane University, USA, shosseinisereshgi@tulane.edu; Rodrigo I. Silveira, Universitat Politècnica de Catalunya, Spain, rodrigo.silveira@upc.edu; Carola Wenk, Tulane University, USA, cwenk@tulane.edu.

Manuscript submitted to ACM

CCS Concepts: • **Information systems** → **Geographic information systems**; • **Theory of computation** → **Computational geometry**.

Additional Key Words and Phrases: Map comparison, map construction, graph sampling

ACM Reference Format:
Jordi Aguilar, Kevin Buchin, Maike Buchin, Erfan Hosseini Sereshgi, Rodrigo I. Silveira, and Carola Wenk. 2018. Graph Sampling for Map Comparison. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 25 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Many situations ask to compare different roadmaps, e.g., roadmaps reconstructed with different algorithms from the same data, or simplifications or generalizations of a given map. When comparing two roadmaps, one wants to take into account both the geometry and topology. Graph sampling was first introduced by Biagioni and Eriksson [11, 12] and Liu et al. [30] for comparing a reconstructed roadmap with a ground truth map. The basic idea is to first *sample* both roadmaps with points at a fixed distance, then *match* points on the two maps within a given distance threshold using a 1-to-1 matching, and finally use the number of matched and unmatched points to compute precision, recall, and F-scores.

These graph sampling scores have been used in many papers to evaluate map construction results [4, 5, 10, 13, 17, 23, 27, 28, 32, 33]. The method has proven useful, as it makes little assumptions on the roadmaps, and thus allows to compare a variety of immersed graphs, and is efficient to compute. However, the two key steps, sampling and matching, allow much freedom in their implementation, and the resulting scores vary greatly based on these. Indeed, Table 1 shows how two implementations of the graph sampling method, which we ran with the same settings, produce different values for precision, recall, and F-score. Sat2Graph's [28] TOPO code is available on Github and Biagioni's code [11, 12] was made available to us by James Biagioni. In the literature, the presented F-scores vary widely, as can be seen in Table 4 in Section 2.1. Hence we revisit the graph sampling method here. The preliminary version of this paper was presented at 2021 ACM SIGSPATIAL Spatial Gems Workshop and was published as a chapter of Spatial Gems Vol. 2 [1].

Chicago	prec.	recall	F
Sat2Graph's TOPO [28]	0.947	0.353	0.514
Biagioni's [11, 12]	0.971	0.523	0.679

Table 1. Graph sampling scores computed by different implementations, with local sampling, 370 seeds, $r = 300m$, $d_{\max} = 15m$ and sampling interval $5m$ on *Biagioni*'s reconstructed map vs. *cropped Chicago (OSM)*.

1.1 Contributions

We present a unified approach to the graph sampling method for map comparison.

- In Section 2 we discuss the different decisions to be made in the method, available choices for these, and how these affect the final score. In particular, we compare local and global sampling, and for the first time visualize the matchings and compare different matching rules.
- In order to enable reproducibility, we make a graph sampling toolkit publicly available, which contains a stable implementation of the graph sampling method as well as an interactive visualization tool. See Section 3.

- In Section 4 we experimentally evaluate the different choices for implementing the graph sampling method, and we provide a thorough comparison of the roadmaps constructed by ten different algorithms on standard datasets such as *Chicago* with ground truth maps from OpenStreetMap (OSM) as well as cropped OSM maps generated by map-matched trajectories. We also compare roadmaps from another data source to OSM maps.

1.2 Related work

There are several methods for comparing roadmaps. Many of them have been developed for determining the quality of map construction algorithms that construct maps from trajectory data or satellite imagery. Since roadmaps are immersed graphs, i.e., all vertices have associated locations and edges have associated curves in 2D or 3D, methods for comparing shapes and graphs are also available for comparing maps. See [4, 5, 15, 19] for surveys.

The path-based [2], shortest path-based [29], and traversal [8] distances represent each graph with paths and compare the paths, and thus measure connectivity to some extent. The Hausdorff distance [9] considers nearest neighbor assignments of points only, hence not taking the connectivity into account, while the Fréchet distance requires establishing a homeomorphism between the graphs [24], however roadmaps are generally not homeomorphic. Less strict requirements on a roadmap between the two graphs are imposed by the weak and strong graph distances [7] and the contour tree distance [14, 15], but many variants are NP-complete. The local persistent homology-based distance [3] compares the topological features in local neighborhoods by comparing locally computed persistence diagrams of the distance filtrations of the graphs but falls short in considering various geometric features such as length. Edit distances, see e.g. [18], can also be defined, but are usually NP-complete. Methodology for locally evaluating map construction algorithms for hiking data trajectories has been provided in [21]. Graph sampling [11, 12, 30], the method we study here, is–arguably–the most popular method for comparing two roadmaps because of its simplicity, effectiveness, and speed; we discuss related work on it in Section 2.1 and Section 2.2.

2 GRAPH SAMPLING METHODS

Graph sampling methods for map comparison typically have a simple structure. First, point samples are computed from each map, using some *sampling method*. Second, a matching between the point samples of each map is computed, according to a *matching rule*. Intuitively, the rule determines when two points should be identified as the same in both maps. Finally, the number of matched points is used to calculate one or more *scores*, typically precision and recall, which measure the proportion of points matched.

Hence the implementation of a graph sampling method involves two key decisions: a sampling method and a matching rule. Since there are multiple options for each, and they can have an important effect on the final scores, we discuss each of them in detail. In the following, the two graphs to be compared are always denoted G and H. Note that in this work we use the terms *map* and *graph* interchangeably. Furthermore, a cropped map is a road map that was cropped using a map-matching algorithm based on GPS trajectories.

2.1 Sampling Method

The sampling method determines which points are sampled from each map. It is important that the sampling is dense enough to include all roads in the map, and that the number of samples along a road segment is proportional to its length. A simple way to achieve this is by sampling along each edge of the graph at a fixed distance between consecutive samples (as long as this distance is smaller than the minimum edge length). Some care must be taken at intersections, to ensure that the distance between consecutive samples is maintained across them as much as possible. Typically, the

sampling is implemented using a graph traversal. This ensures that consecutive samples on paths from the root to the leaves are spaced at a fixed distance. There are two major approaches to graph sampling:

(1) In *global* sampling, the roadmap G is sampled in its entirety with points at a fixed distance (typically 5m), resulting in a point set P_G sampled from G such that $|P_G|$ is proportional to $\text{len}(G)$. Here, $\text{len}(G)$ denotes the total length of all edges in G. The set P_G is a deterministic discretization of G. For the second graph H, the point set P_H is computed analogously.

(2) In *local* sampling, one proceeds in two phases. First, a set $S \subseteq \mathbb{R}^2$ of *seeds* is computed. Typically, S is chosen at random on G. Second, for each $s \in S$, the graphs $G \cap U_s$ and $H \cap U_s$ are sampled deterministically. Here, U_s is a neighborhood of s, usually a disk centered at s with a fixed radius r. Typically the sampling is performed using a graph traversal in $G \cap U_s$ starting at $s \in G$, and a graph traversal in $H \cap U_s$ starting at the nearest neighbor $s_H \in H$ to s, sampling points at a fixed distance.

Another important aspect of sampling is the *graph traversal*. The graph G can be interpreted as an undirected graph, or as a directed graph with edge directions and/or turn restrictions at vertices. (Not all roadmaps, in particular reconstructed ones, come equipped with edge directions or turn restrictions.) In addition, a traversal may traverse only a single connected component, or it may traverse every connected component.[1] Actual roadmaps are of course (strongly) connected, but some reconstructed maps may not be connected. And in particular, local sampling may result in multiple connected components in smaller neighborhoods.

Global vs. Local Sampling. Global sampling is a deterministic sampling method, and for a fixed sampling distance and fixed graph traversal algorithm (in particular one that traverses all connected components), the sets P_G and P_H are uniquely determined. For a fixed matching rule (see Section 2.2), precision is $k/|P_G|$ and recall is $k/|P_H|$, where k is the number of matched samples; the F-score is the harmonic mean of precision and recall. The resulting graph comparison method, based on global sampling, has previously been termed GEO [12, 30].

Local sampling, on the other hand, introduces much more variability into the sampling process, and therefore the sample sets and the resulting scores are not well-defined. The choice and the number of the seeds pose the first problem.

We adapted the graph sampling code that James Biagioni made available to us to implement local sampling using undirected graph traversal, and used it for Tables 2–4. Table 2 shows an example where precision, recall, and F-scores vary widely for different numbers of random seeds. The precision values for the cropped ground truth, for example, vary between 0.702 and 0.938. If seeds are randomly chosen, some areas of the map may be oversampled, some undersampled; and it is not clear how many random seeds to choose. One way to alleviate this problem may be to choose seeds in a systematic way such that G or H or both are covered in a well-defined way; He et al. [27] for example compute seeds by sampling the ground truth map at a fixed distance of $50m$.

One more caveat is how to tackle seeds in G that don't have a close enough sample $s_H \in H$. In this situation, seeds have been omitted from score calculation [12] or have been used for computing recall only [27].

Another source of variability in local sampling is the aggregation of the scores, see Section 2.3.

Local sampling was initially introduced [11] with the intent to measure topological differences between two roadmaps; Biagioni and Eriksson [12] called this method TOPO. For each seed $s \in S$, this graph comparison method only traverses one connected component in $G \cap U_s$ starting from s and one connected component in $H \cap U_s$ starting from s_H, and it uses edge directions and turn restrictions in G and H (as well as bearings and a greedy matching, see Section 2.2). So the only topological feature this method captures is local connectivity. It is, however, extremely sensitive to the

[1] A connected component is a connected subgraph that is not part of any larger connected subgraph.

Biagioni [12]	Chicago			cropped Chicago		
# seeds	prec.	recall	F	prec.	recall	F
10,000	0.859	0.183	0.301	0.894	0.543	0.676
2,000	0.821	0.196	0.316	0.917	0.534	0.675
1,000	0.780	0.185	0.299	0.938	0.551	0.694
200	0.661	0.154	0.250	0.702	0.479	0.569
100	0.879	0.171	0.287	0.931	0.618	0.743

Table 2. Local evaluation with different number of seeds with $r = 300m$ and $d_{\max} = 15m$ on *Biagioni*'s reconstructed map vs. OSM ground truth on *Chicago* data.

Biagioni [12]	Chicago			cropped Chicago		
r	prec.	recall	F	prec.	recall	F
900	0.884	0.111	0.197	0.881	0.456	0.600
600	0.817	0.126	0.218	0.836	0.478	0.608
300	0.661	0.154	0.250	0.702	0.479	0.569
150	0.576	0.238	0.337	0.716	0.495	0.585
100	0.556	0.347	0.427	0.757	0.492	0.597
50	0.558	0.554	0.556	0.813	0.462	0.589

Table 3. Local evaluation with different radii r (in m), $d_{\max} = 15m$, and using 200 seeds on *Biagioni* vs. *Chicago (OSM)*.

definition of locality, i.e., the choice of the radius defining the local neighborhood U_s. See Table 3 for an example where precision, recall, and F-scores vary widely for different choices of radii. The precision numbers for the cropped ground truth, for example, vary between 0.702 and 0.881. It is not clear how this radius should be chosen in order to provide a useful comparison of local connectivity information. Intuitively the local neighborhood would need to be very small to even contain more than one connected component. In the literature, the choice of radii includes $100m$ [17], $300m$ [4, 5, 11–13, 23, 27][2], and a quite large value of 2000m [32] which is 1/4 of the map diameter (for Chicago).

Due to the variability introduced by local sampling, and the limited (and not well-specified) benefit of comparing local connectivity, global sampling may be more beneficial to use in practice, since it is well-specified and reproducible.

	Global Sampling					Local Sampling				
	OSM		cropped OSM			OSM			cropped OSM	
Chicago	[32]	Ours	[33]	[12]	Ours	[32]	[4, 5]	Ours	[12]	Ours
Ahmed [6]		0.09	0.61		0.61		0.27	0.29		0.61
Biagioni [12]	0.24	0.07	0.78	0.78	0.64	0.58	0.35	0.25	0.78	0.57
Cao [16]	0.29	0.10		0.68	0.49	0.53	0.24	0.27	0.68	0.41
Edelkamp [22]	0.36	0.12		0.53	0.60	0.47	0.32	0.31	0.64	0.50
Karagiorgou [29]		0.08	0.82		0.70		0.27	0.28	0.27	0.71

Table 4. Varying F-scores comparing the same reconstructed maps in different papers for $d_{\max} = 15$; most values were visually transcribed from plots. All used $r = 300m$, except [32] used $r = 2000m$. The number of seeds is 200 for [32] and ours, it is 100 for [12], and 1000 for [4, 5].

[2]This assumes [11, 12] used $r = 300m$ as in the code provided to us by James Biagioni.

Graph Sampling Used in the Literature. Graph sampling scores have been used widely to evaluate map construction results [4, 5, 10, 17, 23, 27, 28, 32, 33]. Most use a $5m$ sampling interval and variants of local sampling. However, often not all parameters (e.g., r, number of seeds) or other choices (e.g., traversal, matching rule, score aggregation, map cropping method) are specified, affecting reproducibility, in particular for local sampling. Biagioni and Eriksson [12] use both global sampling (GEO [30]) and local sampling (TOPO [11] with directed road traversal), and they use a cropped ground truth. While the locality radius r and the number of seeds are not specified, in the code that James Biagioni made available to us the default values were $r = 300m$ and 100 random seeds, so we assume these parameter choices were made. Stanojevic et al. [32] also use both global sampling and local sampling (with $r = 2000m$ and 200 seeds). Ahmed et al. [4, 5] use local sampling based on the code provided by James Biagioni (using $r = 300$) and do not crop the ground truth. They introduce the use of a fixed set of seeds for all comparisons in order to increase reproducibility; they use 1000 seeds. He et al. [27, 28] and Van Etten [23] use local sampling with $r = 300m$. Bastani et al. [10] also use local sampling; they present F-scores averaged over multiple datasets, and they introduce a new score based on matching intersections. Chen et al. [17] use local sampling and take 1% of the GPS points of the input trajectories as seeds and $r = 100m$. Tang et al. [33] use a global approach to compute F-scores and manually cropped ground truth maps.

It follows that, even though graph sampling has been widely adopted as a method for comparing roadmaps, there is a large variability in the precision, recall, and F-scores in the literature. In Table 4 we show F-scores from different papers [4, 5, 12, 32, 33], including ours, that were computed on the same reconstructed maps[3] and OpenStreetMap (OSM) ground truth for Chicago. Most values were visually transcribed from plots in the papers, and may therefore contain some noise. The table includes F-scores for local and global sampling methods using (full) and cropped OSM ground truths and $d_{\max} = 15$. Our F-scores were computed using local sampling parameters $r = 300m$ and 200 seeds. While all use OSM ground truth maps, only [4, 5, 33] and our paper use the OSM maps from mapconstruction.org. The locality radius is $r = 300m$ for all, except for [32] it is $r = 2000m$. The number of seeds is 200 for [32] and our paper, it is 100 for [12], 1000 for [4, 5]. It can be seen that the F-scores vary widely in each row. For example, for Biagioni's reconstructed map the local sampling scores vary between 0.25 and 0.58 for OSM, and between 0.57 and 0.78 for cropped OSM. The values for global sampling on cropped OSM are a bit more consistent – note that two approaches agree on 0.78 for Biagioni's map and two agree on 0.61 for Ahmed's map. (Note that our OSM maps cover a rather large area, see the figures in the appendix. The difference of F-scores for uncropped OSM is likely due to the use of an OSM map with different coverage.)

2.2 Matching Rule

The matching rule defines when a pair of points, one from each map, should be considered the same. Recall that a matching is a 1-to-1 correspondence (i.e., no point can be matched to two points). All matching rules include a distance condition, establishing that only points that are closer than some maximum distance threshold $d_{\max}$ can be considered to match; this is the simplest possible rule. In principle, the more points that can be matched, the more similar the two maps will be considered.

Maximum Matching (MM). If the matching rule is only based on $d_{\max}$, the simplest approach is to match as many pairs of points as possible, as long as they are within distance $d_{\max}$. This is equivalent to finding a *maximum matching*

[3]The trajectory data and reconstruction code are publicly available, e.g., at mapconstruction.org. However, reconstructed maps may still differ if parameters were set differently.

in the bipartite graph whose vertices are the sampled points on each map, and whose edges are all pairs of points (from different maps) at distance at most $d_{\max}$.

Greedy Matching. While a maximum matching guarantees to match as many points as possible, it involves finding a global solution, which may be costly in large graphs. Also, all pairs within distance $d_{\max}$ are considered equivalent. Instead, one can find a locally maximal matching that is as large as possible, albeit possibly suboptimal, and gives priority to matching pairs of points that are close to each other. A *greedy matching* can be computed by choosing one point from one map, and matching it to the nearest point in the other map, if possible. If not, the second nearest point is tried, and so on, until the kth one (for a parameter k).

Unfortunately, the greedy matching is not clearly defined: there are multiple ways to implement it, leading to different methods. In particular, the order in which points are matched can result in very different matchings.

We present a greedy matching algorithm here that follows the ideas in Biagioni's implementation of graph sampling as used [11, 12]. It consists of two steps: First assign a nearest neighbor to each point. This produces an assignment that is not 1-to-1. In a second step, a 1-to-1 matching is greedily computed from this initial matching. Pseudocode is shown in Algorithm 1 (an implementation is included in our graph sampling toolkit). Note that a point is only matched to one of its k nearest neighbors (typically, $k = 10$).

The greedy matching has two interesting properties: (i) it gives priority to matching points that are close to each other, as it tries to match closest pairs first. Moreover, (ii) it is more selective than the maximum matching: if none of the k nearest neighbors are available to match a point, the point is not matched. Thus one can expect fewer matched pairs with this method, but possibly *better* matched pairs.

Algorithm 1: Greedy Matching

Input : Set of samples $S_G \subseteq G$, and $S_H \subseteq H$, parameter k
Output : A 1-to-1 matching $M \subseteq S_G \times S_H$

$M_{init} = \emptyset$ *// Priority queue, sorted by matched distance*
// Create initial 1-to-many "matching"
for all $s_G \in S_G$ **:**
    s_H = closest among k-nearest neighbors of s_G that are within distance (and bearing) threshold
    Add (s_G, s_H) to M_{init}
// Convert to 1-to-1 matching, prioritizing shortest distances
while $M_{init} \neq \emptyset$ **:**
    $(s_G, s_H) = M_{init}.pop()$ *// Pop closest pair*
    if s_H *not used*
        Add (s_G, s_H) to M; mark s_H as used
    else
        new_s_H = closest unused sample among k-nearest neighbors of s_G that are within distance (and bearing) threshold
        if new_s_H *found // If not found,* s_G *is discarded*
            Add pair (s_G, new_s_H) to M_{init}
return M

Weighted Maximum Matching (WMM). We propose a new matching rule that combines the strongest points of the maximum and greedy matching. The idea is not only to try to match as many pairs as possible, but also to take the

distance of each matched pair into account. We can formalize this as follows. We consider the same graph as in the maximum matching, but now each edge pq has a weight, defined as $d_{\max} - ||p - q||$, where $||p - q||$ is the Euclidean distance between p and q. The goal is now to compute a matching of maximum total weight, where the total weight of a matching is the sum of the weights of all edges in the matching. In this way, we give priority to pairs that are nearby over those further apart.

The matching obtained may contain fewer edges than a maximum matching, but is expected to contain shorter edges. Note that, while the motivation for the weighted maximum matching is similar to that of the greedy matching, an important advantage of the weighted maximum matching is that it is unambiguously well-defined. Moreover, if no additional constraints are used, it produces matchings that are crossing-free (see Figure 2), a property that seems to be beneficial.

A disadvantage is that it requires a globally optimal solution, thus it can be computationally more expensive. Indeed, the best known methods to compute a weighted maximal matching have complexity $O(nm + n^2 \log n)$ [25], for n and m the number of vertices and edges, respectively. In our context, if $d_{\max}$ is small, one can expect m to be $o(n)$, or even constant. However, implementations in open source software libraries often include simpler but less efficient methods, like those in Boost (C++) and NetworkX (Python), which have complexity $O(n^3)$.

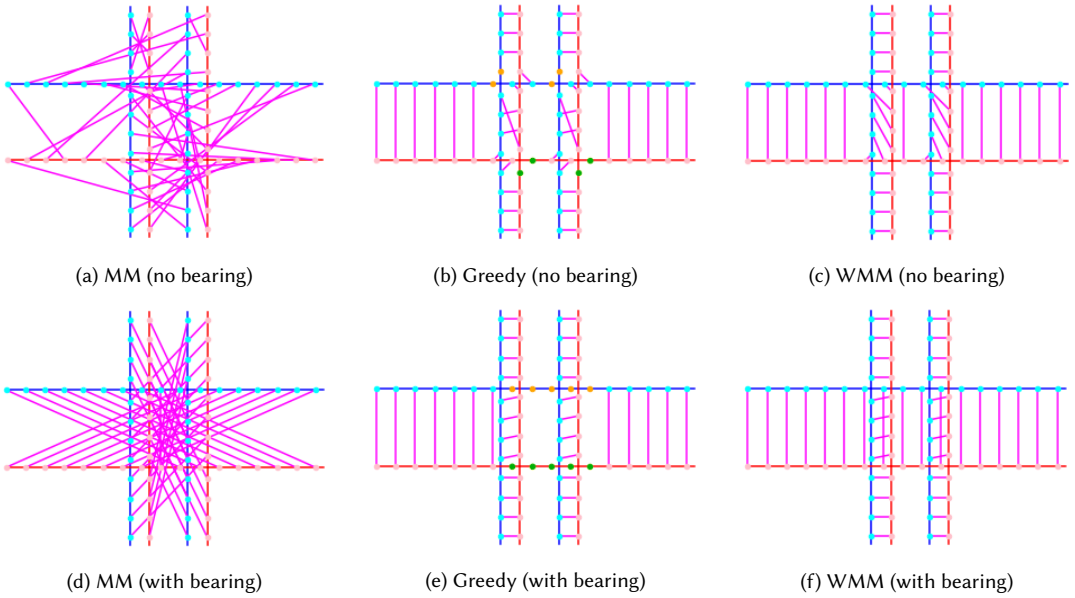
Bearing Conditions. Matching rules can include other aspects in addition to distance. The most important one used in the literature is bearing. The idea is that two points should be matched only when they belong to edges with a similar orientation. The most common way to take it into account is to require that the angle between the two edges is at most 45°. A canonical example to motivate including bearing is to avoid matching two points that are very close to each other, but belong to edges that are perpendicular; in such a case, it is reasonable to argue that the points should not be considered the same, since their edges have opposite orientations.

Matching Rules Used in the Literature. All sampling based methods use some type of matching, but very few papers specify exactly how the matching is computed. In most cases, the description of the matching part only states that two points are matched whenever they are within the distance threshold (see, e.g., [17, 32, 33]), without explaining what is done when the nearest neighbor is already taken, which is often the case.

The exceptions that we are aware of are RoadRunner [27], that uses a maximum matching, and Biagioni and Eriksson [11, 12]—together with a few other papers that reused their code [4, 5, 13]—that implement greedy matching rules. The weighted maximum matching is proposed in this work for the first time. As for bearing, it is included in several papers [11, 12, 27, 32], although the exact bearing threshold used is not always mentioned (RoadRunner [27] uses 30° degrees). All our experiments in Section 4 use bearing (with a 45° threshold).

Comparison of Matching Rules. In order to get some insight into the consequences of choosing one matching rule or another, it is useful to consider a concrete example. Figure 2 presents a simple situation where each map has only three edges, shown in blue and red, respectively. Both maps are sampled in the same way (globally, using sampling distance $5m$). The resulting matchings are shown for the three matching rules (maximum matching MM, weighted maximum matching WMM, and Greedy) with two variations: with and without bearing.

Already in the first row, we can observe striking differences between the three matching rules. Maximum matching, as expected, matches at least as many points as the other rules, but at the cost of including pairs that visually do not seem to correspond to each other. In contrast, the two rules that give priority to shorter edges (WMM, Greedy) produce correspondences that are much more aligned with intuition. It is interesting to note that the greedy matching fails to



(a) MM (no bearing) (b) Greedy (no bearing) (c) WMM (no bearing)

(d) MM (with bearing) (e) Greedy (with bearing) (f) WMM (with bearing)

Fig. 2. Example illustrating three different matching rules, without and with bearing. Two maps are compared, *map 1* with blue edges and *map 2* with red edges. A pair of matched samples is shown with a magenta segment between a sample in *map 1* (cyan) and a sample in *map 2* (pink). Unmatched samples in *map 1* are represented in orange, unmatched samples in *map 2* are represented in green. Sampling distance has been set to $5m$, and $d_{\max} = 50m$.

match some points around the intersections of the map edges. This can be explained by the fact that it is limited to matching among the 10-nearest neighbors of each point. Using such a hard constraint can lead to being too selective in situations like the one shown.

The first row also shows that only taking distances into account can result in matching points that belong to clearly different edges. That is the case in the figure with matchings between horizontal and vertical edges. The second row, that restricts matching pairs to those with bearing difference of less than 45°, solves this issue. This makes the greedy matching avoid diagonal edges, although it still fails to match a few pairs. In contrast, the weighted maximum matching matches all points while respecting orientations. This justifies the inclusion of bearing restrictions in the matching rules.

2.3 Score Calculation

Precision and recall are the two scores typically used to quantify the results of graph sampling methods. In this context, precision is the number of matched samples divided by the total number of samples on H (typically the reconstructed map). Recall is the number of matched samples divided by the total number of samples on G (typically the ground truth map). They are useful to measure the ratio of correct predictions and the ratio of covered ground truth, respectively. These two scores are often combined using the F-score, defined as the harmonic mean of precision and recall (i.e., $F = 2 * (\text{precision} * \text{recall})/(\text{precision} + \text{recall})$).

As mentioned in Section 2.1, when global sampling is used, the matched samples used to compute precision, recall, and F-score are taken over the entire graphs. However, when local sampling is used, there are different options for

aggregation. The number of matched samples and total samples can be aggregated (summed) over all seeds, and precision, recall, and F-score computed using those total number of samples. Or, precision, recall, and F-score can be computed for each seed individually, and then aggregated in some way, e.g., by taking the mean. While it is reasonable to use such local aggregation in combination with local sampling, it does add extra variability to the computation, which should be clearly specified when presenting results. Moreover, unless exactly the same aggregation is used, results will not be comparable across different works.

Cropping the Ground Truth Map. In the context of map reconstruction, the recall values can easily become distorted if the ground truth map used is not appropriate for the reconstructed map. Often, the ground truth map used is significantly larger than the reconstructed map, including roads that are not covered in the GPS dataset. This causes a dilution in the recall value, which also affects F-scores. One way to overcome this situation is to crop the ground truth such that it only contains the roads traversed by trajectories. This can be done manually (see, e.g., [33]) or using map-matching algorithms (e.g., as in [12]). As it can be seen in Tables 2 and 3, the difference in recall between cropping the ground truth or not is significant. However, the use of a cropped ground truth adds an extra level of variability to the experiments, since there are various methods and settings to choose from, making the experiments unlikely to be reproducible if the method used is not specified in full detail (something that seldom occurs in the literature). It is also possible to overcome this problem by obtaining the number of matched samples. When working without a reliable ground truth, using the number of matched samples instead of the recall and the F-score avoids having to compare near-zero and unrealistic recalls such as those in Table 5.

3 GRAPH SAMPLING TOOLKIT

Our graph sampling toolkit consists of three components: the core is the graph sampling evaluation program. Additionally, we provide tools for cropping maps and an interactive visualization program. The Graph Sampling Toolkit is available on Github: https://github.com/Erfanh1995/GraphSamplingToolkit.

3.1 Graph Sampling Evaluation Program

Our main software contribution is the graph sampling evaluation program. This program can be used to compare two input graphs globally, using any one of the matching algorithms that we described in Section 2.2. The starting point of our implementation was Biagioni's code that he made available to us. Our code consists of 3 steps: generating a first point for the traversal on each connected component of the input graphs and returning a file, sampling all identified connected components and producing an evaluation file containing matched and unmatched samples and calculating precision, recall and F-score based on the evaluation file. Some options to set the matching algorithm, bearing threshold, sampling interval and $d_{\max}$ are also available.

3.2 Map Cropper

As mentioned in Section 2.3, in the context of map reconstruction from GPS traces, finding a suitable ground truth map to obtain sensible recall values is a challenge. One way to tackle the issue is by cropping the ground truth map based on the input trajectories, to ensure that the ground truth covers the same areas as the reconstruction, thus making the comparison fair. Our toolkit includes a map matching algorithm, which takes as input a map and a set of GPS traces, and produces a cropped map containing only those map edges that could be matched to some input trajectory. To that

end, we use the Hidden Markov map matching algorithm [31]. This is arguably one of the most popular map matching algorithms, it is conceptually simple, and it is rather efficient.

3.3 Visualization Tool

To facilitate the use of graph sampling we introduce an interactive visualization tool to examine all evaluations from a closer perspective. Our graph sampling program returns a file with all matched and unmatched points which can be used as input for the visualizer (see Figure 3). The visualizer provides options to select the desired sets of samples or inputs to be visualized. It also has the option to display precision and recall values of the selected reconstructed map based on its evaluation file. Furthermore, it is possible to view reconstructed maps, their corresponding ground truth and to overlay a trajectory dataset (that is often the input to map reconstruction programs).

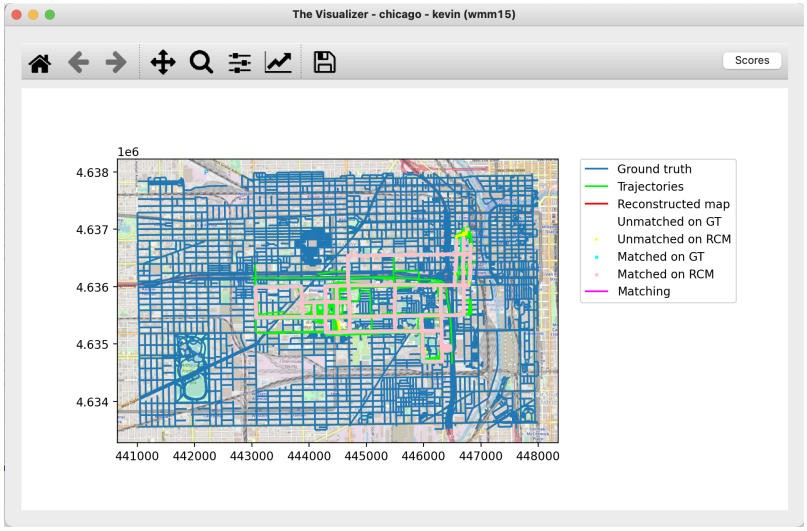


Fig. 3. The Visualizer

4 EXPERIMENTS

4.1 Data

In the experiments, and throughout this paper, we used reconstructed maps and ground-truth roadmaps from OpenStreetMap (OSM) for the *Chicago*, *Athens-small*, and *Berlin* data sets that are available on mapconstruction.org. The maps reconstructed from GPS trajectories by algorithms [6, 12, 16, 20, 22, 26, 29] are the same as in [4, 5]. For [6, 29] the reconstruction code is available on mapconstruction.org and for [12, 16, 20, 22] on https://www.cs.uic.edu/bin/view/Bits/Software. The *Chicago* map produced by [32] is available from the author's repository, and the *Chicago* map by [27] was computed and sent to us by one of its authors. We computed the *Chicago* and *Athens-small* reconstructed maps by [13] using the code available at http://www.cs.tulane.edu/~carola/tmp/JMC.zip. Trajectories for generating cropped OSM maps and all the OSM ground truth maps that are used in this paper are from mapconstruction.org.

We also compare roadmaps from two different sources. For *Berlin*, we have small ($16km^2$) road maps from TeleAtlas (TA) from 2007 and OpenStreetMap (OSM) from April 2013. Similarly, for *Athens*, we have TA maps from 2007 and OSM

Generated map	CCs	samples	MM				WMM				Greedy			
Chicago	45	121,025	matched	prec.	recall	F	matched	prec.	recall	F	matched	prec.	recall	F
Ahmed [6]	1	6,989	5,540	0.793	0.046	0.087	5,506	0.788	0.045	0.086	5,494	0.786	0.045	0.086
Biagioni [12]	6	4,823	4,524	0.938	0.037	0.071	4,499	0.933	0.037	0.071	4,492	0.931	0.037	0.071
Buchin [13]	1	4,705	4,478	0.951	0.037	0.071	4,465	0.949	0.037	0.071	4,461	0.948	0.037	0.071
Cao [16]	16	15,811	6,960	0.440	0.058	0.102	6,665	0.422	0.055	0.097	6,542	0.414	0.054	0.096
Davies [20]	1	2,837	2,772	0.977	0.023	0.045	2,748	0.969	0.023	0.044	2,736	0.964	0.023	0.044
Edelkamp [22]	1	16,635	8,733	0.525	0.072	0.127	8,521	0.512	0.070	0.124	8,443	0.508	0.07	0.123
Ge [26]	1	7,600	5,053	0.665	0.042	0.079	4,939	0.65	0.040	0.077	4,901	0.645	0.040	0.076
Karagiorgou [29]	1	5,498	5,179	0.942	0.043	0.082	5,151	0.937	0.043	0.081	5,144	0.936	0.043	0.081
Kharita [32]	1	17,217	8,483	0.493	0.070	0.123	8,353	0.485	0.069	0.121	8,260	0.48	0.068	0.12
RoadRunner [27]	1	3,749	3,362	0.897	0.028	0.054	3,325	0.887	0.027	0.053	3,317	0.885	0.027	0.053

Table 5. Graph sampling evaluation comparing reconstructed maps to Chicago OSM, using global sampling and $d_{\max} = 15m$. (CCs stands for connected components.) In each column, the highest three scores are shaded; darker colors are higher.

Generated map	CCs	samples	MM				WMM				Greedy			
Chicago	45	121,025	matched	prec.	recall	F	matched	prec.	recall	F	matched	prec.	recall	F
Ahmed [6]	1	6,989	6,835	0.978	0.056	0.107	6,782	0.970	0.056	0.106	6,569	0.94	0.054	0.103
Biagioni [12]	6	4,823	4,808	0.997	0.04	0.076	4,757	0.986	0.039	0.076	4,691	0.973	0.039	0.075
Buchin [13]	1	4,705	4,700	0.999	0.039	0.075	4,676	0.994	0.039	0.074	4,629	0.984	0.038	0.074
Cao [16]	16	15,811	9,937	0.628	0.082	0.145	9,036	0.572	0.075	0.132	7,608	0.481	0.063	0.111
Davies [20]	1	2,837	2,830	0.998	0.023	0.046	2,819	0.994	0.023	0.046	2,792	0.984	0.023	0.045
Edelkamp [22]	1	16,635	11,450	0.688	0.095	0.166	10,606	0.638	0.088	0.154	9,608	0.578	0.079	0.14
Ge [26]	1	7,600	7,262	0.956	0.060	0.113	7,051	0.928	0.058	0.11	6,472	0.852	0.053	0.101
Karagiorgou [26]	1	5,498	5,498	1	0.045	0.087	5,448	0.991	0.045	0.086	5,414	0.985	0.045	0.086
Kharita [32]	1	17,217	11,630	0.675	0.096	0.168	11,047	0.642	0.091	0.16	10,031	0.583	0.083	0.145
RoadRunner [27]	1	3,749	3,626	0.967	0.03	0.058	3,530	0.942	0.029	0.057	3,473	0.926	0.029	0.056

Table 6. Graph sampling evaluation for various algorithms on Chicago ground truth, using global sampling and $d_{\max} = 60m$.

maps from 2010. The TA maps are not publicly available. Figures of reconstructed maps and roadmaps are available in the appendix.

Generated map	CCs	samples	WMM			
Athens-small	1	38,690	matched	prec.	recall	F
Ahmed [6]	8	7,351	5,009	0.681	0.129	0.218
Biagioni [12]	8	4,644	3,510	0.756	0.091	0.162
Buchin [13]	3	5,406	4,042	0.748	0.104	0.183
Cao [16]	6	616	398	0.646	0.010	0.020
Davies [20]	3	535	496	0.927	0.013	0.025
Edelkamp [22]	2	39,467	6,707	0.17	0.173	0.172
Ge [26]	42	4,765	3,318	0.696	0.086	0.153
Karagiorgou [29]	1	7,498	5,539	0.739	0.143	0.24

Table 7. Graph sampling evaluation comparing reconstructed maps to Athens-small OSM, using global sampling and $d_{\max} = 15m$.

4.2 Comparing Reconstructed Maps

We begin by comparing maps reconstructed from the same set of GPS traces with ten different algorithms. This is the first time so many reconstructed maps are compared in a consistent and uniform way. We apply global sampling with a distance of $5m$ for the three matching rules discussed, with a bearing threshold of 45°. Table 5 presents results for $d_{\max} = 15m$ and Table 6 for $d_{\max} = 60m$.

We can observe that the different matching rules used produce different numbers of matched samples, with (small) impact on the precision and recall values. The variation between the three rules is not very significant, but for larger $d_{\max}$ it can be enough to affect the relative order of the scores, for example for $d_{\max} = 60m$, Karagiorgou ranks first in precision when using MM and Greedy, but third if WMM is used. Cao's precision using MM and Greedy changes from 0.628 to 0.481 while WMM yields a precision of 0.572. Ge, Edelkamp, and Kharita also see 10% difference in their precision values between MM and Greedy. It is clear that the rule applied makes a difference, thus this confirms the importance of being very precise about the matching conditions employed. In Table 7 we also compare precision and recall values for *Athens-small* using weighted maximum matching.

Our experiments also show the importance of using a consistent evaluation method and datasets. For example, as illustrated in Table 5 and Table 6, the precision of Kharita stands out as one of the lowest, which contradicts the findings of the original experiments conducted in [32] on the *Chicago* dataset. Additionally, Roadrunner was exclusively tested against alternative methods using TOPO on private GPS trajectory datasets, which appear to be less noisy and more densely populated. However, map construction algorithms are often tested on noisy and sparse datasets. As a result, Roadrunner's recall values rank among the lowest on the *Chicago* dataset.

Despite being the most proper and accurate matching, as we mentioned in Section 2.2, weighted maximum matching can have large runtimes based on the size of the input maps (see Table 8). Therefore, for a small $d_{\max}$, Greedy matching can be a good alternative when evaluating a single reconstructed map or as we explain later in Section 4.3, when working with large roadmaps.

Chicago	MM	WMM	Greedy
Ahmed [6]	2,160	3,792	105
Biagioni [12]	1,566	3,220	96
Buchin [13]	1,660	2,740	92
Cao [16]	4,508	9,188	113
Davies [20]	869	1,767	95
Edelkamp [22]	5,372	13,483	107
Ge [26]	2,084	4,616	130
Karagiorgou [29]	1,881	3,626	90
Kharita [32]	5,964	13,076	118
RoadRunner [27]	1,205	2,033	83

Table 8. Runtime (in seconds) for evaluations with different matching algorithms comparing reconstructed maps to Chicago OSM, using global sampling and $d_{\max} = 15m$.

With respect to recall values, the comparison with the original ground truth maps gives very small recall scores. This is not surprising, since the ground truth map is significantly larger than the area covered by the GPS traces used to reconstruct the maps (see the figures in the appendix). For this reason, we also present results using cropped ground truths (see Figure 4), obtained with the Hidden Markov map-matching as described in Section 3.2. The parameters used

for the map matching algorithms were as follows: *max_dist*=100, *max_dist_init*=50, *obs_noise*=50, *obs_noise_ne*=75, *dist_noise*=50. As can be seen in Tables 9 and 10, the use of the cropped ground truth maps effectively re-scales the recall values so that they use most of the $[0, 1]$ interval.

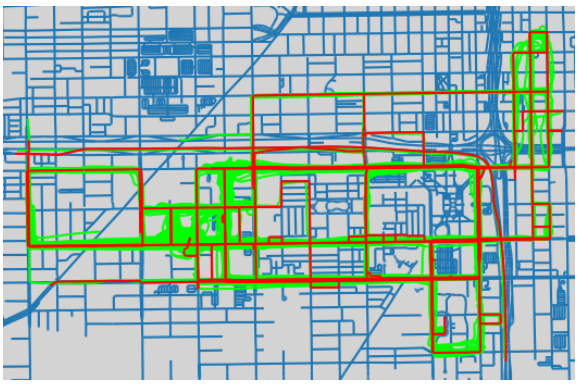


Fig. 4. Chicago OSM (blue; zoomed-in) and cropped Chicago (red). The GPS trajectories are shown in lime.

Generated map	CCs	samples	WMM			
cropped Chicago	1	9,188	matched	prec.	recall	F
Ahmed [6]	1	6,989	4,957	0.709	0.54	0.613
Biagioni [12]	6	4,823	4,493	0.932	0.489	0.641
Buchin [13]	1	4,705	4,428	0.941	0.482	0.637
Cao [16]	16	15,811	6,169	0.390	0.671	0.494
Davies [20]	1	2,837	2,745	0.968	0.299	0.457
Edelkamp [22]	1	16,635	7,726	0.464	0.841	0.598
Ge [26]	1	7,600	4,176	0.549	0.455	0.497
Karagiorgou [29]	1	5,498	5,126	0.932	0.558	0.698
Kharita [32]	1	17,217	7,483	0.435	0.814	0.567
RoadRunner [27]	1	3,749	3,327	0.887	0.362	0.514

Table 9. Graph sampling evaluation comparing reconstructed maps to cropped Chicago OSM, using global sampling and $d_{\max} = 15m$.

4.3 Comparing Roadmaps

In this section, we compare roadmaps from two different sources (TeleAtlas and OSM). In this case precision and recall values can be used interchangeably since both maps cover the same area. One obstacle is that the sizes of these datasets are usually large, which can greatly increase the runtime of maximum matching and maximum weight matching. However, since these are roadmaps, they rarely contain odd artifacts. Thus given the high number of samples on both maps, different matching algorithms yield almost identical results. Hence using the greedy matching with a small $d_{\max}$ in such experiments is preferable. In Table 11 and Table 12, we compare roadmaps from OpenStreetMap and TeleAtlas on *Athens-large* and *Berlin-small* respectively. In Figure 5 we can see that even in dense areas near highway intersections we still get a correct matching.

Generated map	CCs	samples	WMM			
Athens-small	6	10,016	matched	prec.	recall	F
Ahmed [6]	8	7,351	4,835	0.658	0.483	0.557
Biagioni [12]	8	4,644	3,426	0.738	0.342	0.467
Buchin [13]	3	5,406	3,971	0.735	0.396	0.515
Cao [16]	6	616	392	0.636	0.039	0.074
Davies [20]	3	535	466	0.871	0.047	0.088
Edelkamp [22]	2	39,467	6,154	0.156	0.614	0.249
Ge [26]	42	4,765	3,208	0.673	0.320	0.434
Karagiorgou [29]	1	7,498	5,378	0.717	0.537	0.614

Table 10. Graph sampling evaluation comparing reconstructed maps to cropped Athens-small OSM, using global sampling and $d_{\max} = 15m$.

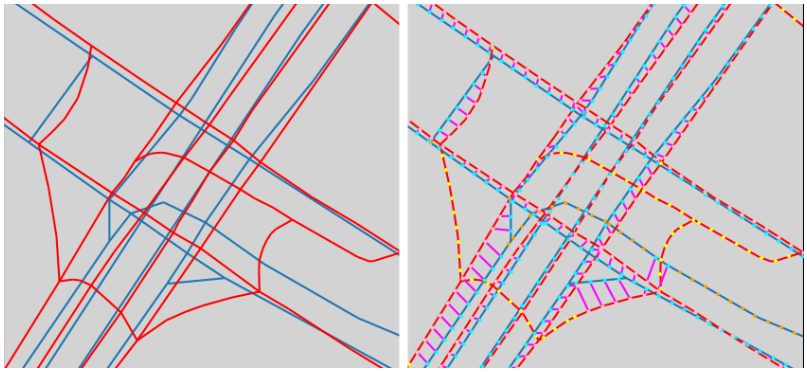


Fig. 5. A highway intersection on *Berlin-small* and its matching. Blue represents OSM and red is TeleAtlas

Athens-large	CCs	samples	matched	prec.	recall	F
OSM	5	399,898	349,725	0.786	0.875	0.828
TeleAtlas	12	445,086				

Table 11. Graph sampling evaluation for Athens-large OSM vs. TeleAtlas, using global sampling and $d_{\max} = 15m$.

Berlin-small	CCs	samples	matched	prec.	recall	F
OSM	6	71,515	67,936	0.824	0.95	0.88
TeleAtlas	7	82,423				

Table 12. Graph sampling evaluation for Berlin-small OSM vs. TeleAtlas, using global sampling and $d_{\max} = 15m$.

5 DISCUSSION / CONCLUSION

The Graph Sampling method is widely recognized for its practicality and utility in evaluating reconstructed maps. However, as demonstrated in this paper, its effectiveness relies on consistent and transparent application.

In our study, we presented a unified approach to graph sampling for map comparison. We highlighted the importance of the choice of the matching algorithm and proposed to take the first step towards this goal by using weighted maximum

matching, a mathematically well-defined, parameter-free, and practical matching algorithm that yields comparable or even superior experimental results compared to other alternatives.

We showed that local sampling does not necessarily preserve topology and instead introduces numerous choices and parameters, resulting in less reproducible scores. (see Table 4) On the other hand, global sampling offers uniformity and reproducibility, making it a suitable choice for future evaluations. Therefore, employing the Graph Sampling method in the global setting, alongside weighted maximum matching and bearing constraints, can serve as a successful strategy for comparing road maps.

Additionally, we provided a graph sampling toolkit to facilitate the use of graph sampling for map comparison, which hopefully can lead to more comparable evaluations in the field of map construction and more accurate results for map comparison.

Fig. 6. Heatmap of Athens small. 5000 seeds were used on the reconstructed map with a radius of 75m. Each one is colored according to its F-score. The brighter the color (yellow), the more accurate the region of the reconstructed map.

Although local sampling may not be suitable for precision/recall evaluation, as has been done in [2, 3], it can still serve a purpose in visualizing local differences by generating heatmaps of all computed scores, see Figure 6. Furthermore, despite the effectiveness of graph sampling as an approach for map comparison, it still remains a discrete method and lacks the capability to accurately assess the topological differences between two road maps. A feasible continuous method might be the key to achieving more comprehensive results.

ACKNOWLEDGMENTS

We thank James Biagioni for making his graph sampling code from [11, 12] available to us. We also thank Songtao He for providing us with Roadrunner's reconstructed map on Chicago [27] and their TOPO evaluation code from

[28]. E.H.S. and C.W. were partially supported by National Science Foundation grant CCF 1637576. R.I.S. was partially supported by project PID2019-104129GB-I00/AEI/10.13039/501100011033.

REFERENCES

[1] Jordi Aguilar, Kevin Buchin, Maike Buchin, Erfan Hosseini Sereshgi, Rodrigo I. Silveira, and Carola Wenk. 2024. *Graph Sampling for Map Comparison* (1 ed.). Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3617291.3617293

[2] M. Ahmed, B.T. Fasy, K.S. Hickmann, and C. Wenk. 2015. Path-based distance for street map comparison. *ACM Trans. Spatial Algorithms and Systems* (2015), 28 pages.

[3] M. Ahmed, B.T. Fasy, and C. Wenk. 2014. Local Persistent Homology Based Distance Between Maps. In *Proc. 22nd ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems*. ACM, 43–52.

[4] M. Ahmed, S. Karagiorgou, D. Pfoser, and C. Wenk. 2015. A Comparison and Evaluation of Map Construction Algorithms. *Geoinformatica* 19, 3 (2015), 601–632.

[5] M. Ahmed, S. Karagiorgou, D. Pfoser, and C. Wenk. 2015. *Map Construction Algorithms* (first ed.). Springer International Publishing.

[6] M. Ahmed and C. Wenk. 2012. Constructing Street Networks from GPS Trajectories. In *Proc. 20th Ann. European Symp. on Algorithms*. 60–71.

[7] H.A. Akitaya, M. Buchin, B. Kilgus, S. Sijben, and C. Wenk. 2021. Distance measures for embedded graphs. *Computational Geometry: Theory and Applications* (2021), 101743.

[8] H. Alt, A. Efrat, G. Rote, and C. Wenk. 2003. Matching planar maps. *Journal of Algorithms* (2003), 262–283.

[9] H. Alt and L.J. Guibas. 1999. Discrete Geometric Shapes: Matching, Interpolation, and Approximation - A Survey. In *Handbook of Computational Geometry*. Elsevier, 121–154.

[10] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, S. Madden, and D.J. DeWitt. 2018. RoadTracer: Automatic Extraction of Road Networks From Aerial Images. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 4720–4728. https://doi.org/10.1109/CVPR.2018.00496

[11] J. Biagioni and J. Eriksson. 2012. Inferring Road Maps from Global Positioning System Traces: Survey and Comparative Evaluation. *Transportation Research Record: Journal of the Transportation Research Board* 2291 (2012), 61–71.

[12] J. Biagioni and J. Eriksson. 2012. Map inference in the face of noise and disparity. In *Proc. 20th ACM SIGSPATIAL GIS* (Redondo Beach, California). 79–88.

[13] K. Buchin, M. Buchin, J. Gudmundsson, J. Hendriks, E. Hosseini Sereshgi, V. Sacristan, R.I. Silveira, F. Staals, and C. Wenk. 2020. Improved map construction using subtrajectory clustering. In *Proc. 4th ACM SIGSPATIAL LocalRec Workshop*. Article 5, 5:1–5:4 pages. https://doi.org/10.1145/3423334.3431451

[14] K. Buchin, T. Ophelders, and B. Speckmann. 2017. Computing the Fréchet Distance Between Real-Valued Surfaces. In *Proc. Twenty-Eighth Annual ACM-SIAM*. ACM, 2443—-2455.

[15] M. Buchin, E. Chambers, P. Fang, B.T. Fasy, E. Gasparovic, E. Munch, and C. Wenk. 2021. Distances Between Immersed Graphs: Metric Properties. In preparation.

[16] L. Cao and J. Krumm. 2009. From GPS traces to a routable road map. In *Proc. 17th ACM SIGSPATIAL GIS* (Seattle, Washington). 3–12.

[17] C. Chen, C. Lu, Q. Huang, Q. Yang, D. Gunopulos, and L. Guibas. 2016. City-Scale Map Creation and Updating Using GPS Collections. In *Proc./ 22nd ACM SIGKDD*. 1465–1474. https://doi.org/10.1145/2939672.2939833

[18] O. Cheong, J. Gudmundsson, H. Kim, D. Schymura, and F. Stehn. 2009. Measuring the similarity of geometric graphs. In *International Symposium on Experimental Algorithms*. Springer, 101–112.

[19] D. Conte, P. Foggia, C. Sansone, and M. Vento. 2004. Thirty Years Of Graph Matching In Pattern Recognition. *Int. Journal of Pattern Recognition and Artificial Intelligence* 18, 3 (2004), 265–298.

[20] J.J̃. Davies, A.R̃. Beresford, and A. Hopper. 2006. Scalable, Distributed, Real-Time Map Generation. *IEEE Pervasive Computing* 5, 4 (2006), 47–54.

[21] D. Duran, V. Sacristán, and R.I. Silveira. 2020. Map construction algorithms: a local evaluation through hiking data. *GeoInformatica* 24 (2020), 633–681.

[22] S. Edelkamp and S. Schrödl. 2003. Route planning and map inference with global positioning traces. In *Computer Science in Perspective*. 128–151.

[23] A. Van Etten. 2020. City-Scale Road Extraction from Satellite Imagery v2: Road Speeds and Travel Times. In *IEEE Winter Conference on Applications of Computer Vision, WACV CO, USA, March 1-5, 2020*. IEEE, 1775–1784.

[24] P. Fang and C. Wenk. 2021. The Fréchet Distance for Plane Graphs. In *European Workshop on Computational Geometry*. 62:1–62:5.

[25] H.N. Gabow. 1990. Data Structures for Weighted Matching and Nearest Common Ancestors with Linking. In *Proceedings of the First Annual ACM-SIAM, CA, USA*. SIAM, 434–443.

[26] X. Ge, I. Safa, M. Belkin, and Y. Wang. 2011. Data Skeletonization via Reeb Graphs. In *Proc. 25th Ann. Conf. on Neural Information Processing Systems*. 837–845.

[27] S. He, F. Bastani, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, and S. Madden. 2018. RoadRunner: improving the precision of road network inference from GPS trajectories. In *Proc. 26th ACM SIGSPATIAL GIS* (Seattle, Washington). 3–12.

[28] S. He, F. Bastani, S. Jagwani, M. Alizadeh, H. Balakrishnan, S. Chawla, M.M̃. Elshrif, S. Madden, and M.A. Sadeghi. 2020. Sat2Graph: Road Graph Extraction Through Graph-Tensor Encoding. In *Proc. 16th European Conference on Computer Vision (Lecture Notes in Computer Science, Vol. 12369)*. Springer, 51–67.

[29] S. Karagiorgou and D. Pfoser. 2012. On vehicle tracking data-based road network generation. In *Proc. 20th ACM SIGSPATIAL GIS*. 89–98.

[30] X. Liu, J. Biagioni, J. Eriksson, Y. Wang, G. Forman, and Y. Zhu. 2012. Mining large-scale, sparse GPS traces for map inference: comparison of approaches. In *Proc. 18th ACM SIGKDD* (Beijing, China). 669–677.

[31] W. Meert and M. Verbeke. 2018. HMM with Non-Emitting States for Map Matching. https://lirias.kuleuven.be/retrieve/512781$$Dmapmatching_ecda18.pdf

[32] R. Stanojevic, S. Abbar, S. Thirumuruganathan, S. Chawla, F. Filali, and A. Aleimat. 2018. Robust Road Map Inference through Network Alignment of Trajectories. In *Proceedings of the 2018 SIAM International Conference on Data Mining, CA, USA*. SIAM, 135–143. https://doi.org/10.1137/1.9781611975321.15

[33] J. Tang, M. Deng, J. Huang, H. Liu, and X. Chen. 2019. An Automatic Method for Detection and Update of Additive Changes in Road Network with GPS Trajectory Data. *ISPRS Int. J. Geo Inf.* 8, 9 (2019), 411. https://doi.org/10.3390/ijgi8090411

A APPENDIX

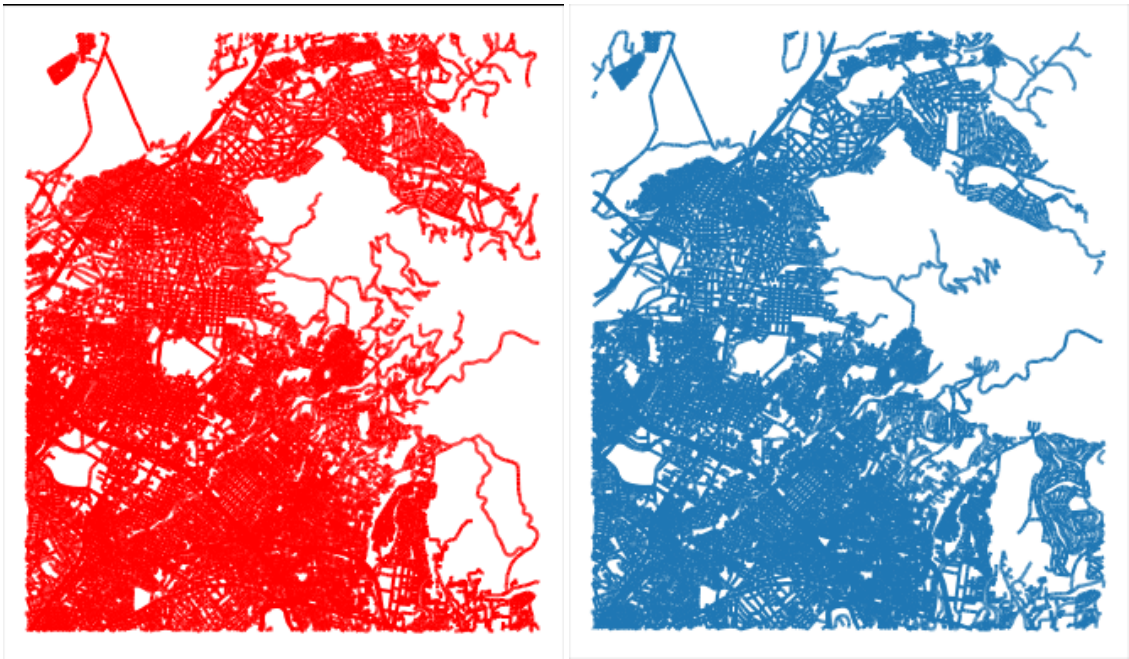


Fig. 7. Athens-large TeleAtlas and OSM maps

Generated map	CCs	samples	$d_{\max}$ = 15m				$d_{\max}$ = 30m				$d_{\max}$ = 60m			
Chicago	45	121,025	matched	prec.	recall	F	matched	prec.	recall	F	matched	prec.	recall	F
Ahmed [6]	1	6,989	5,506	0.788	0.045	0.086	6,399	0.916	0.053	0.1	6,782	0.970	0.056	0.106
Biagioni [12]	6	4,823	4,499	0.933	0.037	0.071	4,695	0.973	0.039	0.075	4,757	0.986	0.039	0.076
Buchin [13]	1	4,705	4,465	0.949	0.037	0.071	4,581	0.974	0.038	0.073	4,676	0.994	0.039	0.074
Cao [16]	16	15,811	6,665	0.422	0.055	0.097	7,726	0.489	0.064	0.113	9,036	0.572	0.075	0.132
Davies [20]	1	2,837	2,748	0.969	0.023	0.044	2,783	0.981	0.023	0.045	2,819	0.994	0.023	0.046
Edelkamp [22]	1	16,635	8,521	0.512	0.070	0.124	9,445	0.568	0.078	0.137	10,606	0.638	0.088	0.154
Ge [26]	1	7,600	4,939	0.65	0.040	0.077	6,177	0.813	0.051	0.096	7,051	0.928	0.058	0.11
Karagiorgou [29]	1	5,498	5,151	0.937	0.043	0.081	5,401	0.982	0.045	0.085	5,448	0.991	0.045	0.086
Kharita [32]	1	17,217	8,353	0.485	0.069	0.121	9,471	0.550	0.078	0.137	11,047	0.642	0.091	0.16
RoadRunner [27]	1	3,749	3,325	0.887	0.027	0.053	3,366	0.898	0.028	0.054	3,530	0.942	0.029	0.057

Table 13. Graph sampling evaluation comparing reconstructed maps to Chicago OSM, using global sampling and weighted maximum matching for various matching distance thresholds $d_{\max}$.

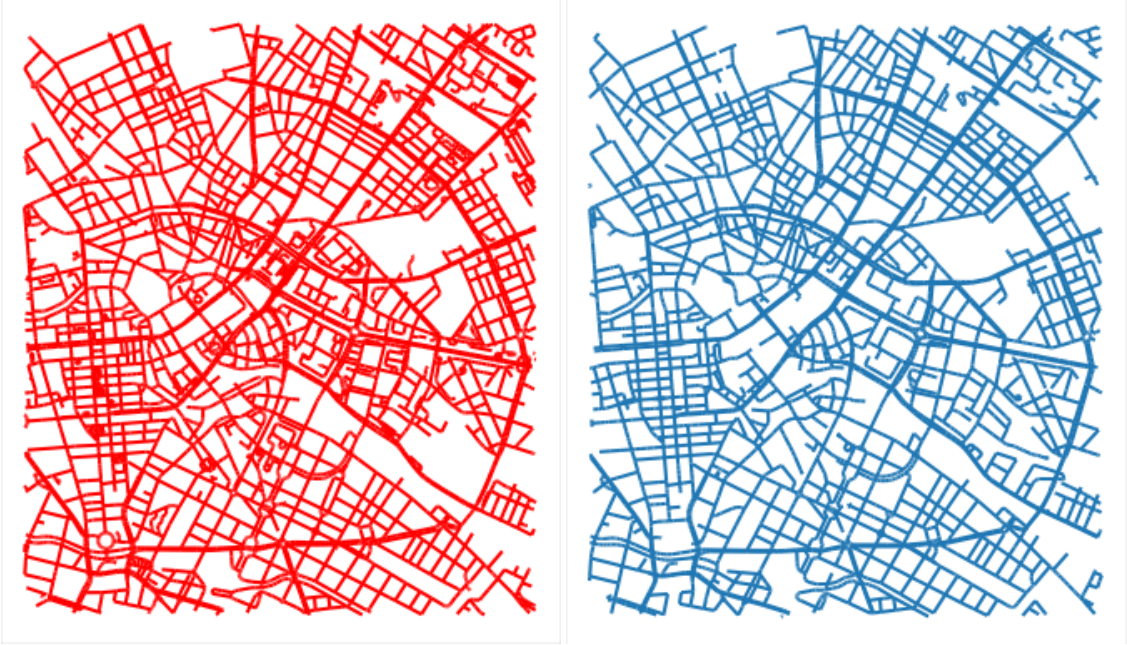


Fig. 8. Berlin-small TeleAtlas and OSM maps

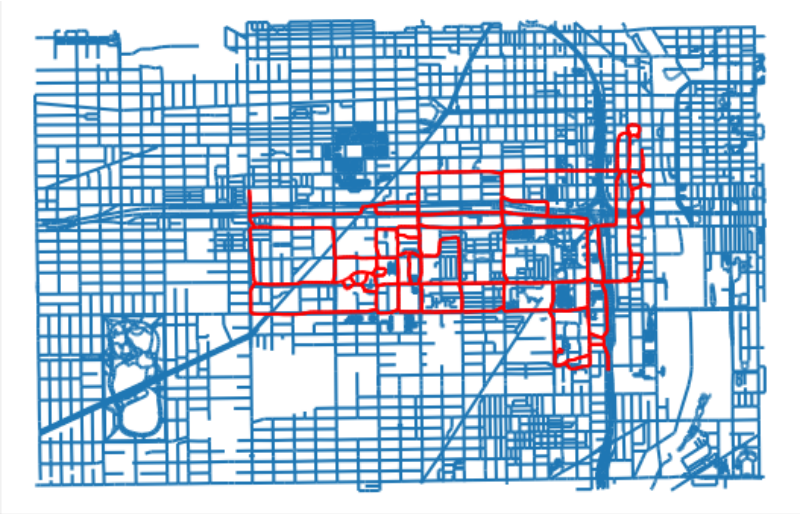


Fig. 9. Ahmed vs. Chicago

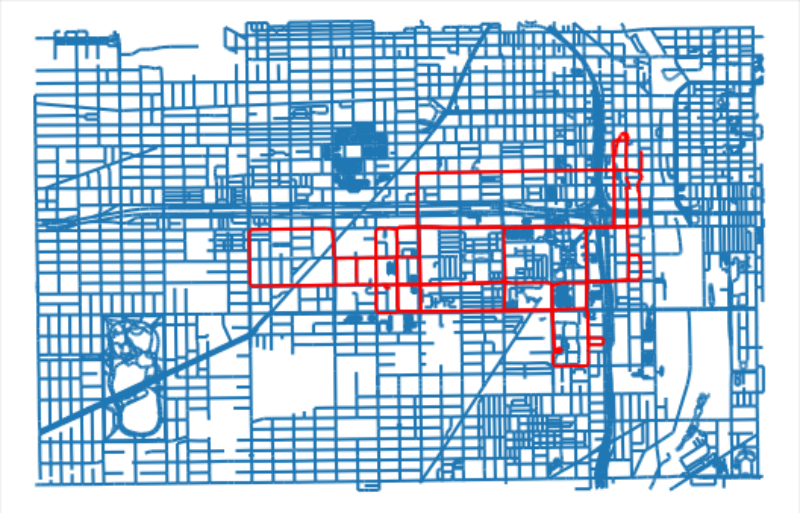


Fig. 10. Biagioni vs. Chicago

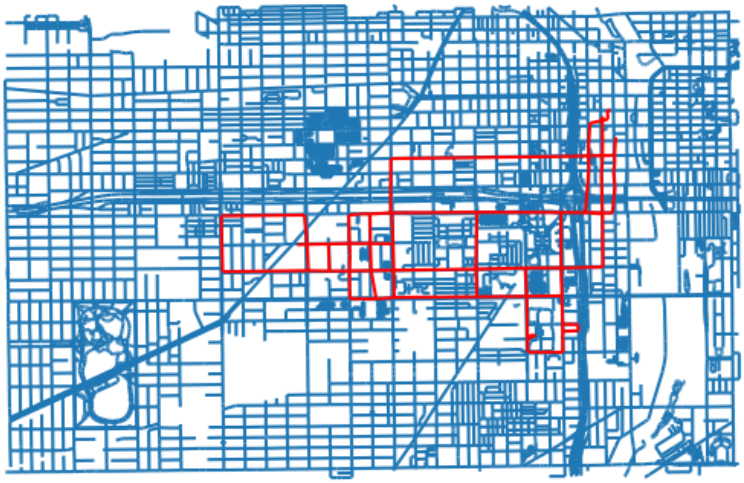


Fig. 11. Buchin vs. Chicago

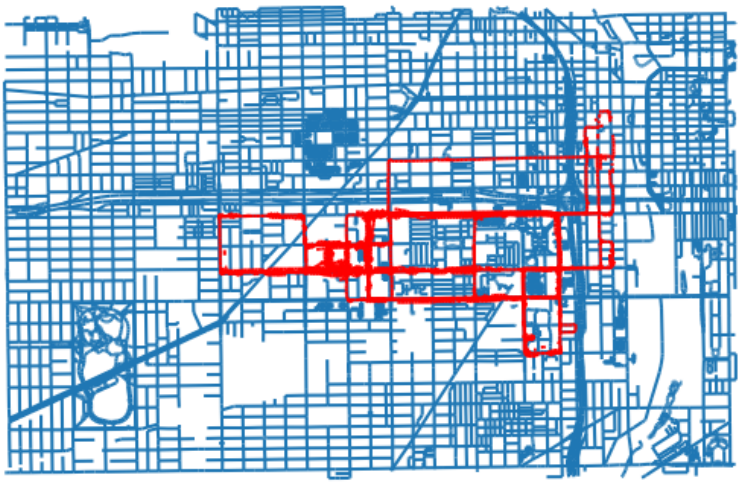


Fig. 12. Cao vs. Chicago

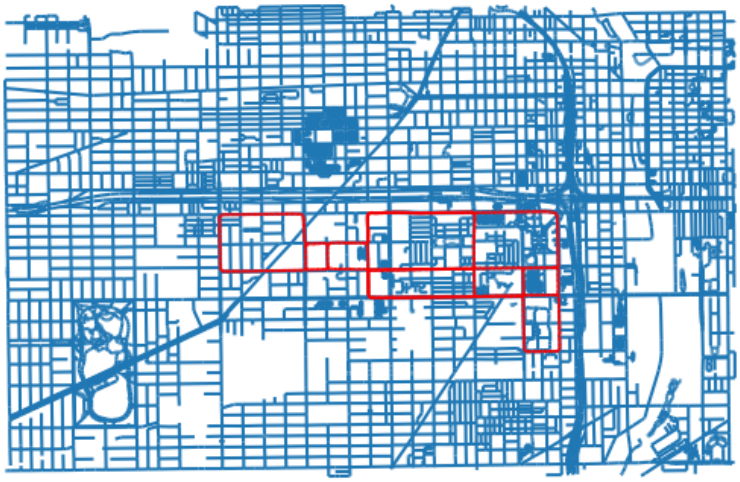


Fig. 13. Davies vs. Chicago

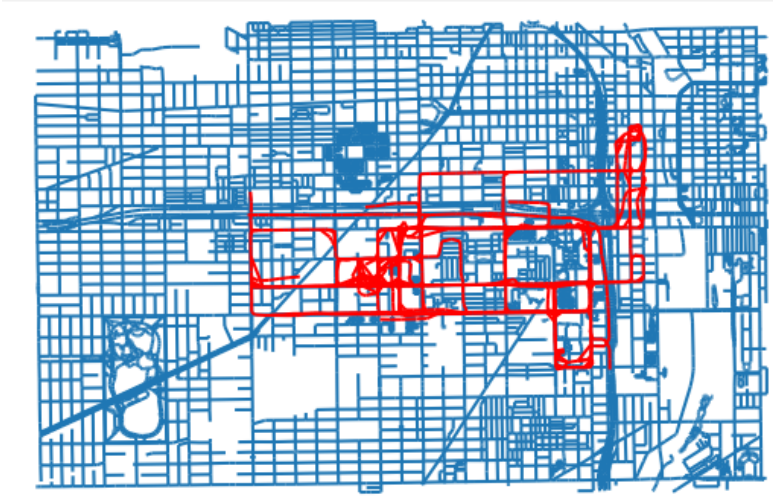


Fig. 14. Edelkamp vs. Chicago

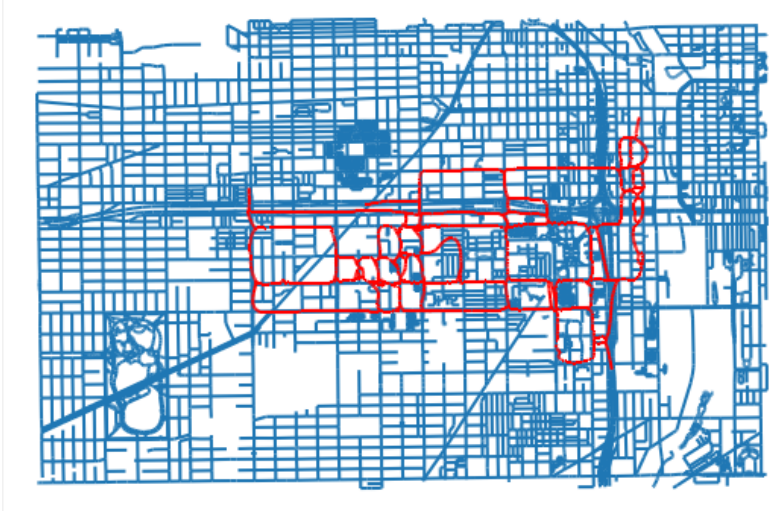


Fig. 15. Ge vs. Chicago

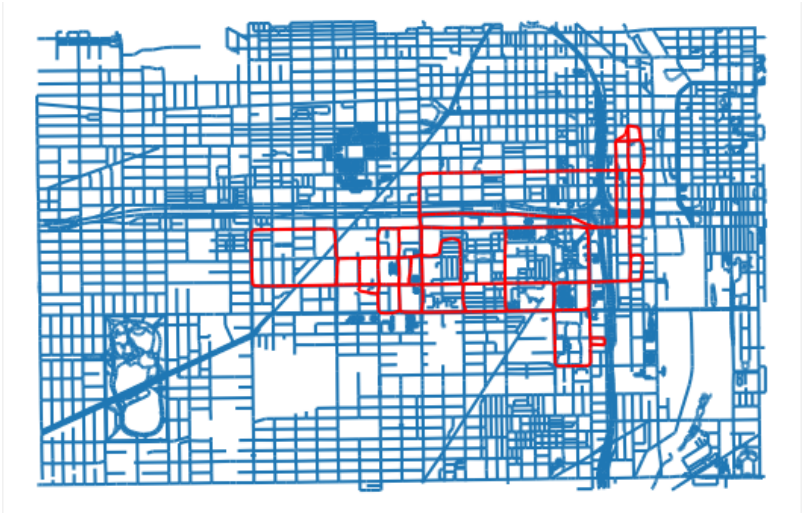


Fig. 16. Karagiorgou vs. Chicago

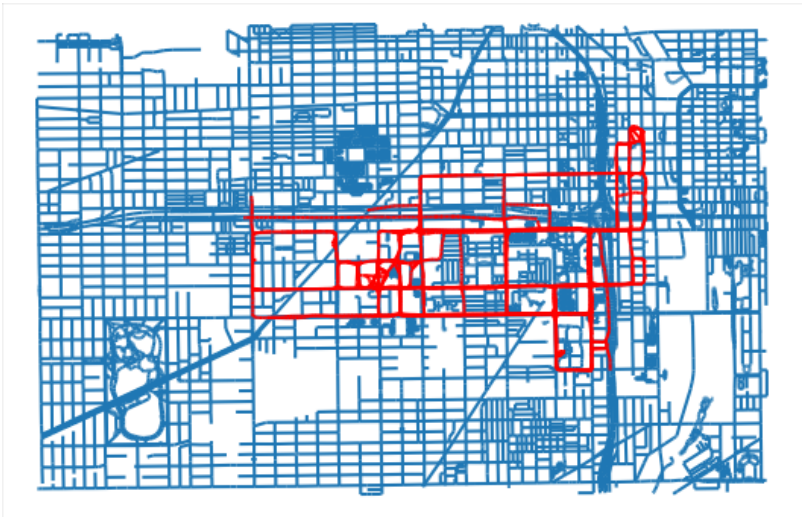


Fig. 17. Kharita vs. Chicago

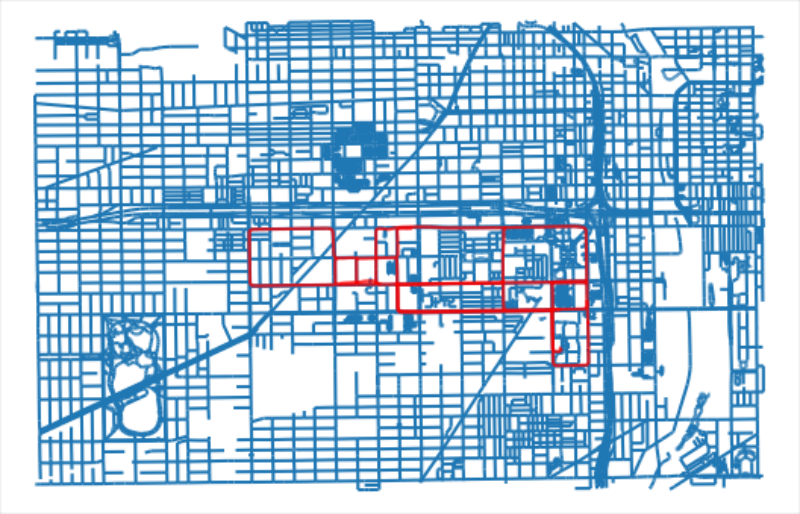


Fig. 18. Roadrunner vs. Chicago