Anonymization via Clustering of Locations in Road Networks

Jan-Henrik Haunert
Institute of Geodesy and Geoinformation, University of Bonn, Germany
haunert@igg.uni-bonn.de

Daniel Schmidt
Institute of Computer Science, University of Bonn, Germany
daniel.schmidt@uni-bonn.de

Melanie Schmidt
Department of Mathematics and Computer Science, University of Cologne, Germany
mschmidt@cs.uni-koeln.de

Abstract

Data related to households or addresses needs be published in an aggregated form to obfuscate sensitive information about individuals. Usually, the data is aggregated to the level of existing administrative zones, but these often do not correspond to formal models of privacy or a desired level of anonymity. Therefore, automatic privacy-preserving spatial clustering methods are needed. To address this need, we present algorithms to partition a given set of locations into k-anonymous clusters, meaning that each cluster contains at least k locations. We assume that the locations are given as a set $T \subseteq V$ of terminals in a weighted graph $G = (V, E)$ representing a road network. Our approach is to compute a forest in G, i.e., a set of trees, each of which corresponds to a cluster. We ensure the k-anonymity of the clusters by constraining the trees to span at least k terminals each (plus an arbitrary number of non-terminal nodes called Steiner nodes). By minimizing the total edge weight of the forest, we ensure that the clusters reflect the proximity among the locations. Although the problem is NP-hard, we were able to solve instances of several hundreds of terminals using integer linear programming. Moreover, we present an efficient approximation algorithm and show that it can be used to process large and fine-grained data sets.

Funding *Melanie Schmidt*: This work was supported by DFG grant SCHM 2765/1-1.

1 Introduction

When publishing data, a large variety of techniques can be applied to conceal information that can be related to individuals. The existing techniques are usually based on *generalization* or *randomization* [7]. Sometimes these approaches are combined, e.g., by first aggregating geo-referenced data to the level of small areas and, afterwards, swapping records between areas or perturbing the values of attributes [4]. A lot of research has gone into the automatic generation of data zones that can be used for data aggregation [5, 6, 10]. Usually, the task has been approached as a *districting problem* that asks to partition a set of small areal units into subsets, each of which defines a district. Commonly used criteria are population size, compactness of shape, and homogeneity of the population [6]. However, there is a lack of *efficient algorithms* for grouping *micro-level data* (e.g., data on the basis of single addresses, households, or buildings) into sufficiently anonymous units. Moreover, from an algorithmic perspective, there is a lack of efficient algorithms with strong quality guarantees for clustering problems with privacy-related side constraints. Therefore, in this paper, we focus on the development of an efficient *approximation algorithm*, i.e., an algorithm whose solution is guaranteed to be at most a certain factor worse than an optimal solution. In experiments with real data, we compare the approximation algorithm with an exact method based on integer linear programming (ILP) that adopts ideas from existing districting methods [8, 13].

We designed both our ILP-based method and the approximation algorithm to tackle the same task. More precisely, we ask to *cluster* a set of geographical locations in a road network, such as access points to buildings. We do not prescribe the number of output clusters but require that each output cluster contains at least k locations, thus ensuring the well-known k-anonymity property [12]. The output representation for each cluster is a multiline feature that corresponds to a set of road segments spanning all cluster members. Since our methods neither require areal units as input nor return areal districts as output, they are not in a narrower sense districting methods. However, one could generate polygonal representations for the clusters returned by our methods, e.g., by computing a line Voronoi diagram of the multiline features [3]. Generally, when anonymizing data, the aim is to preserve *utility.* We address this aim by trying to keep the geometric cluster representations as concise as possible. More precisely, we aim to minimize the total length of all the road segments selected for the output clusters. The following problem definition states these ideas in a formal way.

▶ **Problem 1** (k-Anonymous Steiner Forest). *Let $G = (V, E)$ be a graph, $T \subseteq V$ a terminal set, $w\colon E \to \mathbb{R}_{\geq 0}$ an edge weighing, and $k \in \mathbb{N}$ the requested level of anonymity. Find a subgraph $G' = (V', E')$ of G spanning T, i.e., $T \subseteq V' \subseteq V$ and $E' \subseteq E$, such that every connected component of G' contains at least k terminals and the total edge weight $\sum_{e \in E'} w(e)$ of G' is minimal.*

Figure 1 illustrates this definition. The graph G represents the road network and the terminals in set T the locations of the data entities that we aim to cluster. The node set V of G contains T as a subset but also nodes representing road junctions. The weighing w reflects Euclidean edge lengths in the simplest case, but other settings can be used as well. For instance, to decrease the chance of selecting road segments crossing administrative borders, their weight could be increased. The term “forest” in the problem’s name reflects that an optimal solution is always a set of trees in G. Each of those trees spans at least k terminals plus an arbitrary number of non-terminal nodes, which are commonly called *Steiner nodes.*

A problem related to our problem is the Steiner Tree Problem [9], but this does not ask for subgraphs of at least k terminals. If we choose $k = |V|$, k-Anonymous Steiner Forest and the Steiner Tree Problem are the same. Hence, k-Anonymous Steiner Forest contains the Steiner Tree Problem as a special case and has at least the same computational complexity. In particular, since the Steiner Tree Problem is NP-hard [9], k-Anonymous Steiner Forest is NP-hard, too. Consequently, the existence of an efficient exact algorithm for our problem is highly unlikely. We focus on developing an ILP-based method, which is exact but has an exponential worst-case running time, and an approximation algorithm, which is efficient but has an approximation error within a certain bound.

To summarize our contribution, we present an approach to data anonymization via the problem k-Anonymous Steiner Forest as well as two methods for tackling that problem, namely (i) an exact ILP-based method that is similar to existing districting methods and (ii) an efficient approximation algorithm that guarantees a solution whose cost is at most twice as high as that of an optimal solution. The latter is derived from a generic framework for constrained forest problems [15]. Based on experiments we show that the approximation algorithm allows us to process large and fine-grained data sets, and we compare the two algorithms based on smaller samples drawn from the test data.

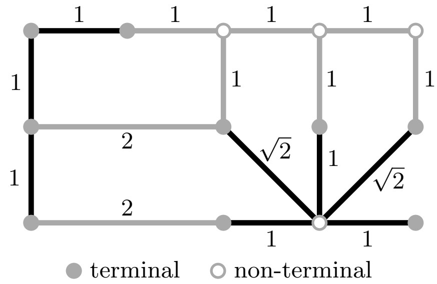


Figure 1 An instance of k-Anonymous Steiner Forest. The edges are labeled with their weights. Black edges correspond to an optimal solution for $k = 4$, whose cost is $6 + 2\sqrt{2}$.

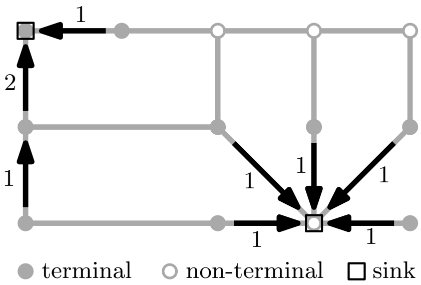


Figure 2 The instance of Fig. 1 with the flow representation of the solution, which is used in our ILP. The black arcs are labeled with the amount of flow they carry.

2 Related Work on Clustering

In the area of discrete optimization, achieving anonymity by enforcing a lower bound on the number of points per cluster has been studied in different papers, going back to the seminal work of Aggarwal et al [1]. Motivated by the concept of k-anonymity in data bases, they define a clustering problem named *r-gather* where every cluster has at least r points and the maximum radius of the clusters is minimized. They give a 2-approximation for this problem. A similar objective is studied by Rösner and Schmidt [11] who show how to add anonymity as a constraint in settings where the clustering has to satisfy multiple constraints of which anonymity is only one. Compared to our objective, radii-based objectives are more prone to outliers. Another approach is to consider clustering problems with anonymity constraints also for sum-based objectives. Svitkina [14] gives an approximation algorithm for facility location with lower bounds which can also be used for k-median (where k refers to a prescribed number of clusters and not to their minimal size). Arutyunova and Schmidt [2] propose *weak lower bounds*: Here, again, every cluster has to have at least a given number of points B, however, it is possible to assign a point to two clusters instead of one (paying for both connections). While the approach makes sense when aggregating clusters into one representative, it makes less sense in our scenario since we need to decide to which group an address belongs to. Compared to the k-median objective, our objective allows points to 'share' connections, i.e., when an edge is bought once for connecting two points in a cluster, further points can use this connection for free, while in the k-median objective, every point has to pay its connection path in full. Another important difference is that we do not prescribe the number of clusters.

3 Integer Linear Program

Our ILP is based on the flow of a single commodity in the directed graph $G' = (V, A)$ whose set A of arcs (i.e., directed edges) contains two arcs uv and vu for each edge $\{u, v\} \in E$. We constrain the flow such that the arcs carrying positive flow correspond to the edges selected for the output forest. More precisely, we ensure that every terminal contributes one unit of flow to the network which can leave the network only at nodes selected as sinks. The sinks serve as counters of the flow to ensure connected components of sufficient size; see Fig. 2.

Our ILP contains three types of variables: For each arc $uv \in A$, there are two variables $x_{uv} \in \{0, 1\}$ and $f_{uv} \in \mathbb{Z}_{\geq 0}$, where $x_{uv} = 1$ means that arc uv is selected and f_{uv} represents the amount of flow on it; and, for each node $u \in V$, there is a variable $s_v \in \{0, 1\}$, where

$s_v = 1$ means that v is a sink. The objective is to minimize $\sum_{uv \in A} w(uv) \cdot x_{uv}$.

All three types of variables are used to express the constraints of the ILP, which we here omit due to the lack of space. However, we refer to the *flow MIP* by Haunert and Wolff [8] for a similar formulation, which has been developed to solve a cartographic aggregation problem. The main differences between the flow MIP and our ILP are (i) that in our ILP the flow variables are integer – hence we have an ILP and not a mixed integer program (MIP) – and (ii) that we need to distinguish between terminals and Steiner nodes.

4 Approximation Algorithm

Our appoximation algorithm is based on a method by Williamson et al. [15, see Sect. 3] for a general class of constrained forest problems. It computes the clusters by simulating a growth process. Initially, each node is a singleton connected component. Then, the connected components grow at the same speed; whenever two components touch, they are joined. Generally, as soon as a component ceases to be a "deficit set", it stops to grow. We can apply the method to k-Anonymous Steiner Forest by defining a deficit set as a set $S \subseteq V$ that contains fewer than k terminal nodes but does not consist entirely of Steiner nodes.

Williamson et al. have shown that under certain conditions their method guarantees a solution whose cost is at most twice the cost of an optimal solution. In particular, the deficit sets must have the following structural property: Any two *crossing* deficit sets – sets cross if their intersection is non-empty, but neither is a subset of the other – may be replaced by non-crossing ones. More precisely, if A and B are crossing deficit sets, it must hold that (i) $A \setminus B$ and $B \setminus A$ *or* (ii) $A \cap B$ and $A \cup B$ are deficit sets as well. In the following, we show that the deficit sets of the k-anonymity problem as defined above satisfy this property.

Let $A, B \subseteq V$ be two deficit sets. Then, suppose (i) does not hold (w.l.o.g. suppose that this is because $A \setminus B$ has no deficit). If $A \setminus B$ has no deficit because it contains at least k terminal nodes, then $A \supseteq A \setminus B$ has no deficit as well, which contradicts the assumption that A is a deficit set. If, on the other hand, $A \setminus B$ has no deficit because it consists entirely of Steiner nodes, $A \cup B = (A \setminus B) \cup B$ has a deficit because B has one. Likewise, $A \cap B$ needs to have a deficit in order for $A = (A \setminus B) \cup (A \cap B)$ to have one. Thus, condition (ii) holds if (i) does not.

To conclude, the method of Williamson et al. is a 2-approximation algorithm for k-Anonymous Steiner Forest. Its running time (essentially) is in $O(|V||E|)$.

5 Experimental Results

We tested our methods for a manually cropped data set of roads and buildings from OpenStreetMap, covering the southwest of the German city Bonn. This area is characterized by a mixture of different types of residential buildings, including blocks of flats, terraced houses, and single-family detached homes. From the road data we generated a geometric graph, including all road types that are accessible for vehicles. We defined the set T of terminals by computing for each building the point in the graph nearest to the building's centroid. This yielded 30801 terminals. Figure 3 (left) shows the extent of the data.

For testing the ILP we generated smaller sub-instances by computing shortest-path trees from randomly selected seed nodes until a prescribed number of terminals was reached. These trees were augmented with shortest paths between the terminals. Choosing ten different seeds, $n \in \{50, 75, 100, \ldots, 400\}$, and $k \in \{4, 5, 6, 7, 8, 9, 10\}$, we obtained 1050 problem instances. Before solving any instance, we iteratively removed non-terminal nodes of degree one until

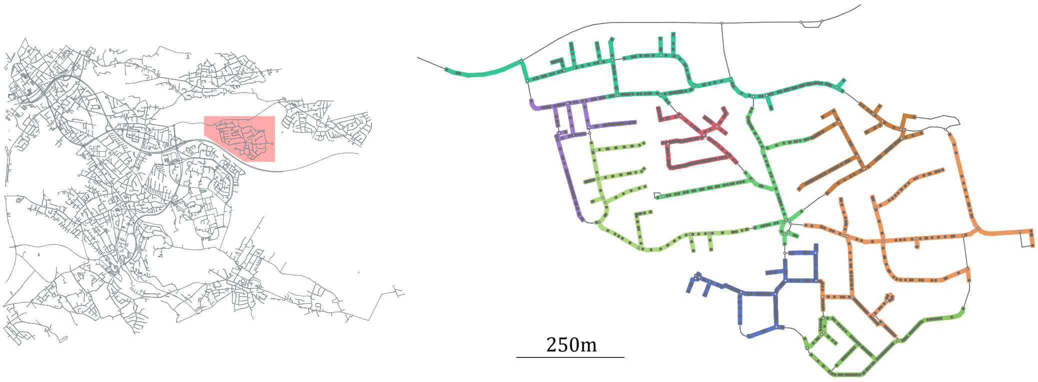
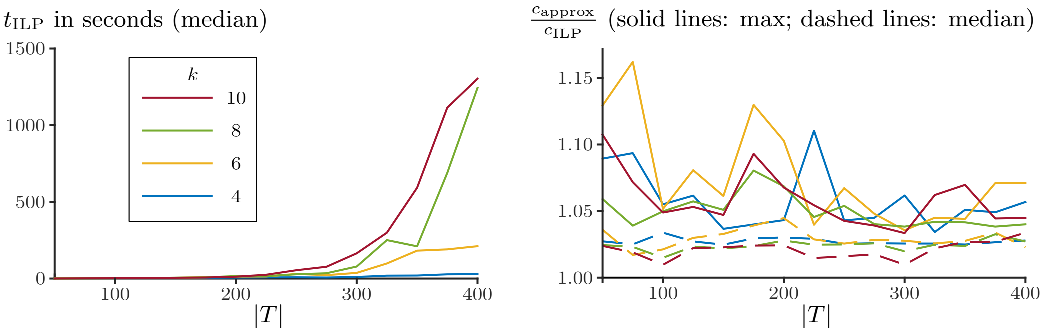


Figure 3 The instance of 30801 locations that we processed with the approximation algorithm with $k = 100$. Left: Whole instance. Right: A part of the instance with nine clusters shown in different colors and locations as gray dots. In the left figure this part is shown as a light red polygon.


t_{ILP} in seconds (median)

$\frac{c_{\mathrm{approx}}}{c_{\mathrm{ILP}}}$ (solid lines: max; dashed lines: median)

Figure 4 Statistics for varying k and number $|T|$ of terminals. Left: Running times of ILP-based method. Right: Objective values of solutions of approximation algorithm relative to optimum.

no such node remained. Afterwards, we iteratively contracted non-terminal nodes v of degree two. All experiments were conducted on a computer with an Intel(R) Xeon(R) W-2125 CPU clocked at 4.00GHz with 128 GiB RAM. The ILPs were solved with the software Gurobi.

For 18 of the 1050 instances, the ILP solver reached a time limit of three hours. In these cases a solution was found but a small optimality gap (at most 0.48%) remained. However, the median running time over ten instances was always below 25 minutes, with relatively long running times occurring for large n and large k; see Fig. 4 (left). For the same 1050 instances we computed solutions with the approximation algorithm. Each of them was at most 17% worse than the corresponding ILP solution; see the solid lines in Fig. 4 (right). In a majority of the cases, the loss of quality was even much less; see the dashed lines in Fig. 4 (right). Therefore, the approximation algorithm is very competitive with respect to quality.

With the approximation algorithm we were able to process the whole instance of 30801 terminals, also for large values of k. For $k = 100$, a solution was returned within 73 seconds. A part of this solution is shown in Fig. 3 (right). It can be seen that the terminals often form densely spaced sequences along a street. The algorithm tends to group the terminals in such sequences. This is generally favorable, e.g., as a row of detached houses can be understood as one semantic unit. On the other hand, some clusters have rather non-compact shapes. Therefore, if high importance is attached to geometrical compactness, radii-based clustering methods may be more appropriate.

6 Conclusion and Outlook

We have presented an ILP-based method and an efficient 2-approximation algorithm for clustering locations in road networks while ensuring the k-anonymity property. The approximation algorithm turned out to be fast and showed an approximation error much below the theoretical bound of 2. As future work we plan to address other concepts of privacy, e.g., ℓ-diversity or t-closeness, and other objectives commonly used in clustering, e.g., minimizing the sum of distances or squared distances between data points and cluster centers.

References

1 Gagan Aggarwal, Rina Panigrahy, Tomás Feder, Dilys Thomas, Krishnaram Kenthapadi, Samir Khuller, and An Zhu. Achieving anonymity via clustering. *ACM Transactions on Algorithms*, 6(3):49:1–49:19, 2010. doi:10.1145/1798596.1798602.

2 Anna Arutyunova and Melanie Schmidt. Achieving anonymity via weak lower bound constraints for k-median and k-means. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 187 of *LIPIcs*, pages 7:1–7:17, 2021. doi:10.4230/LIPIcs.STACS.2021.7.

3 Mark de Berg, Otfried Cheong, Marc J. van Kreveld, and Mark H. Overmars. *Computational Geometry: Algorithms and Applications*. Springer, 2008. doi:10.1007/978-3-540-77974-2.

4 Oliver Duke-Williams and Philip Rees. Can census offices publish statistics for more than one small area geography? An analysis of the differencing problem in statistical disclosure. *International Journal of Geographical Information Science*, 12(6):579–605, 1998. doi:10.1080/136588198241680.

5 Juan C. Duque, Richard L. Church, and Richard S. Middleton. The p-regions problem. *Geographical Analysis*, 43(1):104–126, 2011. doi:10.1111/j.1538-4632.2010.00810.x.

6 Robin Flowerdew, Zhiqiang Feng, and David Manley. Constructing data zones for Scottish neighbourhood statistics. *Computers, Environment and Urban Systems*, 31(1):76–90, 2007. doi:10.1016/j.compenvurbsys.2005.07.008.

7 Kamyar Hasanzadeh, Anna Kajosaari, Dan Häggman, and Marketta Kyttä. A context sensitive approach to anonymizing public participation GIS data: From development to the assessment of anonymization effects on data quality. *Computers, Environment and Urban Systems*, 83:101513, 2020. doi:10.1016/j.compenvurbsys.2020.101513.

8 Jan-Henrik Haunert and Alexander Wolff. Area aggregation in map generalisation by mixed-integer programming. *International Journal of Geographical Information Science*, 24(12):1871–1897, 2010. doi:10.1080/13658810903401008.

9 Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, 1972. doi:10.1007/978-1-4684-2001-2_9.

10 Stan Openshaw and Liang Rao. Algorithms for reengineering 1991 census geography. *Environment and Planning A: Economy and Space*, 27(3):425–446, 1995. doi:10.1068/a270425.

11 Clemens Rösner and Melanie Schmidt. Privacy preserving clustering with constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 107 of *LIPIcs*, pages 96:1–96:14, 2018. doi:10.4230/LIPIcs.ICALP.2018.96.

12 Pierangela Samarati. Protecting respondents identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001. doi:10.1109/69.971193.

13 Takeshi Shirabe. Districting modeling with exact contiguity constraints. *Environment and Planning B: Planning and Design*, 36(6):1053–1066, 2009. doi:10.1068/b34104.

14 Zoya Svitkina. Lower-bounded facility location. *ACM Transactions on Algorithms*, 6(4):69, 2010. doi:10.1145/1824777.1824789.

15 David P. Williamson, Michel X. Goemans, Milena Mihail, and Vijay V. Vazirani. A primal-dual approximation algorithm for generalized steiner network problems. *Combinatorica*, 15(3):435–454, 1995. doi:10.1007/BF01299747.