

# Decentralized Publication and Consumption of Transfer Footpaths

Harm Delva, Julián Andrés Rojas Meléndez, Pieter Colpaert, Ruben Verborgh

IDLab, Department of Electronics and Information Systems, Ghent University – imec

**Abstract.** Users expect route planners that combine all modes of transportation to propose good journeys to their destination. These route planners use data from several sources such as road networks and schedule-based public transit. We focus on the link between the two; specifically, the walking distances between stops. Research in this field so far has found that computing these paths dynamically is too slow, but that computing all of them results in a quadratically scaling number of edges which is prohibitively expensive in practice. The common solution is to cluster the stops into small unconnected graphs, but this restricts the amount of walking and has a significant impact on the travel times. Moreover, clustering operates on a closed-world assumption, which makes it impractical to add additional public transit services. A decentralized publishing strategy that fixes these issues should thus (i) scale gracefully with the number of stops; (ii) support unrestricted walking; (iii) make it easy to add new services and (iv) support splitting the work among several actors. We introduce a publishing strategy that is based on the Delaunay triangulation of public transit stops, where every triangle edge corresponds to a single footpath that is precomputed. This guarantees that all stops are reachable from one another, while the number of precomputed paths increases linearly with the number of stops. Each public transit service can be processed separately, and combining several operators' can be done with a minimal amount of work. Approximating the walking distance with a path along the triangle edges overestimates the actual distance by 20% on average. Our results show that our approach is a middle-ground between completeness and practicality. It consistently overestimates the walking distances, but this seems workable since overestimating the time needed to catch a connection is arguably better than recommending an impossible journey. The estimates could still be improved by combining the great-circle distance with our approximations. Alternatively, different triangulations could be combined to create a more complete graph.

## 1. Introduction

Fueled by the need for more sustainable transportation, users can pick between more forms of transport than ever before – and they expect that route planners can combine these to find the ideal journey. Those route planners integrate various kinds of data

---

Copyright © 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

such as road networks and schedule-based public transit. In this article, we focus on the link between the two, specifically on how to walk between stops. The walking time from one stop to another is commonly referred to as a footpath [1, 2].

Computing these paths dynamically is often too slow in practice [3, 4], yet computing them beforehand quickly becomes prohibitively expensive in practice [5, 1, 2]. A small country such as Belgium already contains 100,000 public transit stops, yielding 10 billion pairs of stops. These scaling issues are often avoided by only evaluating algorithms on small unconnected footpath graphs. However, this restricts the amount of walking in a journey and has a significant impact on travel times [4]. On top of that, determining which stops belong to the same graph ultimately relies on a closed-world assumption; which makes it impractical to add additional public transit services.

Our goal is to find a method to compute and publish these footpaths in a scalable and decentralized manner. This implies that we need not publish all the paths, but ideally the ones that are published should allow for unrestricted walking. In practice this means that every stop in the footpath graph must be reachable from every other stop if there is a path between those stops in the real world. Various actors should be able to add information to this footpath graph, and adding a new service should not require changes in existing information. These actors would need a common frame of reference to combine their efforts, which can be provided by Semantic Web technologies.

## 2. Related work

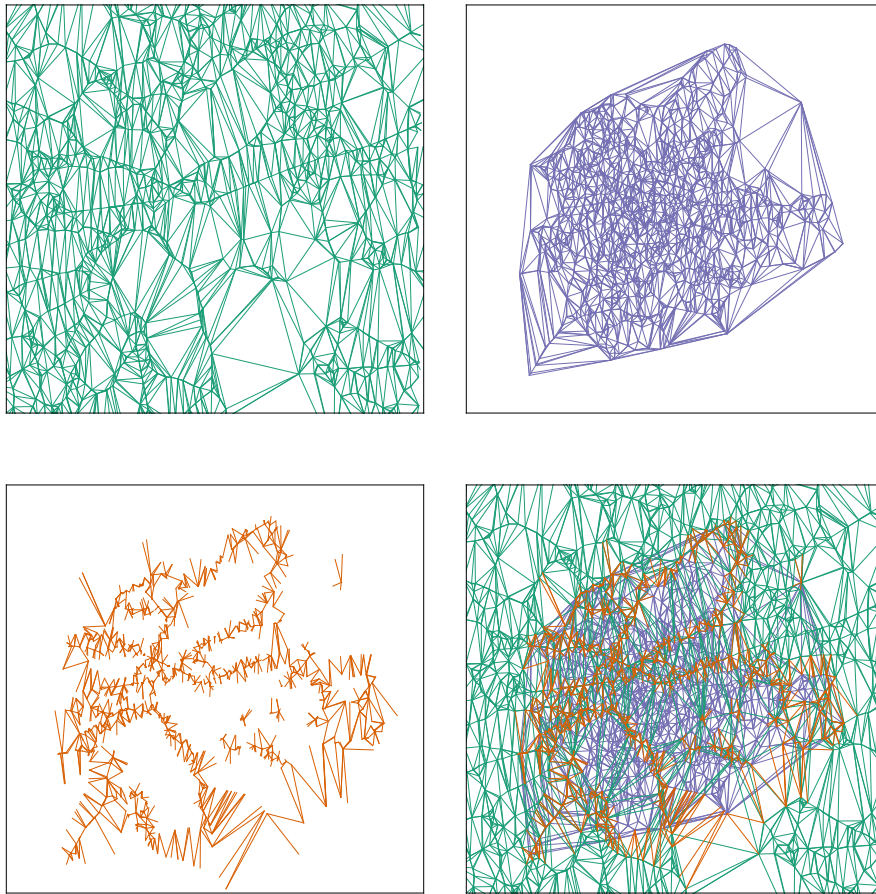
The publication of transfer footpaths is closely related to the publication of road networks. The Routable Tiles [6] dataset was recently proposed as a scalable and cost-efficient way to republish the OpenStreetMap roads using Linked Data Fragments [7]. The data is fragmented into *hierarchical tiles*, which are interconnected with hypermedia controls for *automatic data discovery*. This allows clients to find and download the data they need without having to ingest an entire data dump, while being significantly cheaper than a full-fledged querying API.

Determining which stops have to be connected with each other can be seen as a problem in the field of computational geometry. One of the classical problems in this field is the creation of a *point set triangulation*. A point set triangulation of a set of points  $P$  can be defined as a maximal set of non-crossing edges between the points of  $P$ . Any triangulation of  $n$  points has exactly  $3n - h - 3$  edges where  $h$  is the number of points on the convex hull. Triangulations are not unique however, and certain triangulations are NP-hard to compute [8].

One particularly interesting triangulation, which can be computed in just  $\mathcal{O}(n \log n)$  time, is the *Delaunay triangulation* [9]. This triangulation tries to maximize the minimum angle of all the triangles, effectively avoiding long and narrow triangles.

The *Delaunay graph*, which is simply the graph representation of the triangulation, contains several other interesting graphs such as the nearest neighbor graph and the minimal spanning tree of  $P$  [10, 11]. They are also known to be good approximations of complete graphs: the length of the shortest path between two points along Delaunay edges is at most  $4\pi/(3\sqrt{3}) \approx 2.418$  times the Euclidean distance between them [12].

### 3. Method



**Fig. 1:** Illustration of how two footpath graphs are merged. The subfigures from top to bottom, left to right: (i) an existing triangulation of a larger network; (ii) the

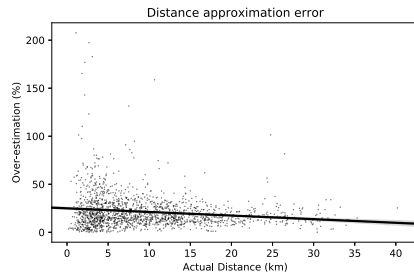
triangulation of an additional regional operator; (iii) the edges that are needed to attach the new triangulation to the existing one and (iv) the end result.

We use a Delaunay triangulation of the public transit stop locations to approximate the full footpath graph, based on the insights from the related work section. The triangle edges represent which stops will be directly connected through a footpath. Our representation of a footpath differs slightly from the conventional one; we use walking distances instead of walking times to avoid making assumptions about walking speeds. A road network route planner is used to compute the walking distances between the stops.

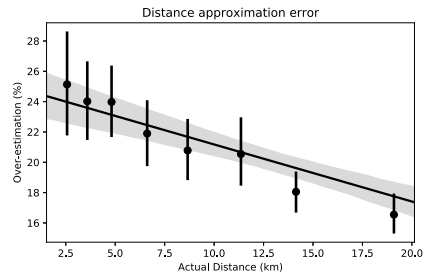
Delaunay triangulations can be generalized to any metric space, but the walking distance is not a mathematical metric because it could, in theory, lack the required symmetry property. Even on foot, you cannot always retrace your steps – either due to legal restrictions or due to physical limitations. And, equally importantly, this would bring us back to the case where an impractical number of path lengths have to be precomputed. This is why the triangulation uses the Euclidean distance between the stops' WGS84 coordinates instead. Other metrics such as the *great-circle distance* can be used, but none of them have any relation to the walking distance so we opted for the most simple metric to implement. This also means that the theoretical guarantees of the Delaunay triangulation only hold for the Euclidean distance because that's the metric was used for its construction.

The paths between stops of a single operator can be calculated independently from the rest. Combining the graphs of two operators is done by comparing their individual triangulations to the triangulation of their combined stops – and computing the missing footpaths. This process is illustrated in Fig. 1 where the graph of a city's public transit network is merged with the one from the region around it. The combined graph is the union of each operators' graph with the additional edges that connect the two graphs. Operators typically have their own service area, which means that the vast majority of paths will be between stops of the same operator, thus making it possible for anyone to combine several operators' graphs with minimal work. The results can be shared and reused, which stands in stark contrast to the conventional approaches. The paths can be published similar to the Routable Tiles dataset so that sharing them does not come at a significant extra cost.

## 4. Results



**2a:** Linear regression plot comparing the actual distances to the approximations. The relative approximation error decreases as the distance increases. Note that the approximations are strictly overestimations due to the triangle inequality.



**2b:** The same linear regression plot but with 8 bins. The mean and standard deviation is shown for each bin, along with the 95% CI in the grey shaded area. The approximation error does not exceed 25% in 95% of cases.

**Fig. 2:** The relative difference between the actual and the approximated distance has a higher variance for shorter paths. Every data point in these plots is a path in between railway stations where the distance is approximated using a graph of bus stops and the footpaths between bus stops and railway stations.

We aim to verify two claims about our method: (i) the graph we construct is almost as good as the complete graph for route planning and (ii) footpath graphs can be merged with minimal effort. To verify the first claim we have used a graph of paths between bus stops to approximate the distances between railway stations. Edges between train stations and bus stops have also been added to make the railway stations reachable. Fig 2 shows that the approximation overestimates the actual distance by about 20% on average. This seems reasonable for route planners – it is better to underestimate the time needed to catch a connection than to recommend an impossible route.

**Table 1:** The number of paths that need to be computed to connect two operators.

Case	Paths of first operator	Paths of second operator	Additional required paths
Flanders + Brussels	107,171	7,969	2,508
Flanders + Wallonia	107,171	94,730	4,020

To verify that combining the graphs from two operators does not require excessive work, we consider both the usual and the worst case scenario, respectively being mostly disjoint service areas and entirely overlapping service areas. We have taken the public transit operators from the Belgian regions of Flanders and Wallonia for the usual case and we revisit the case from Fig. 1 for the worst case, where the Brussels public transit network is added to the one from Flanders. The results in Table 1 show that in the usual scenario merging two operators is relatively straight-forward as most work has already been done. The worst case scenario shows that it is possible that merging two operators can still require a significant amount of work, although the vast majority has already been done. Our implementation contains some additional experiments and is available on <https://github.com/hdelva/planner.js/tree/footpaths> as a frozen branch of an experimental serverless route planner.

## 5. Conclusion and future work

This paper introduced a decentralized publishing method for transit footpaths based on Delaunay triangulations. We have shown that different actors can add information to the footpath graph, splitting the work between them. The amount of paths scales linearly with the amount of public transit stops, and every stop is reachable from every other. This last property could prove useful in its own right, as it allows for unrestricted walking unlike conventional methods. However, this comes at the cost of precision; the walking distance between stops is consistently slightly overestimated which could cause route planners to discard the most optimal journeys. Future work could focus on comparing the results from a route planner with a complete footpath graph to the one as we describe in this paper. If the difference in quality is negligible our methods could prove useful even for centralized services.

The approximations may still be improved with different kinds of point set triangulations and several triangulations could even be combined to create a more complete graph. Alternatively, combining the haversine distance and the currently approximated distance seems promising as well – although this could lead to underestimations. Another open question is how scalable this approach is when more modes of transportation are required. It seems undesirable to create separate graphs for bicycles, electric bikes, wheel chairs, etc. but they might be able to share information.

## References

1. Delling, D., Pajor, T., Werneck, R.F.: Round-based public transit routing. *Transportation Science*. 49, 591–604 (2014).
2. Dibbelt, J., Pajor, T., Strasser, B., Wagner, D.: Connection scan algorithm. *Journal of Experimental Algorithmics (JEA)*. 23, 1–7 (2018).

3. Delling, D., Dibbelt, J., Pajor, T., Wagner, D., Werneck, R.F.: Computing multimodal journeys in practice. In: International Symposium on Experimental Algorithms. pp. 260–271. Springer (2013).
4. Wagner, D., Zündorf, T.: Public transit routing with unrestricted walking. In: 17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017).
5. Bast, H., Carlsson, E., Eigenwillig, A., Geisberger, R., Harrelson, C., Raychev, V., Viger, F.: Fast Routing in Very Large Public Transportation Networks Using Transfer Patterns. In: Algorithms - ESA 2010, 18th Annual European Symposium. Proceedings, Part I. pp. 290–301 (2010).
6. Colpaert, P., Abelshausen, B., Rojas Meléndez, J., Delva, H., Verborgh, R.: Republishing Open Street Map’s roads as Linked Routable Tiles. In: Proceedings of the 16th ESWC: Posters and Demos (2019).
7. Verborgh, R., Vander Sande, M., Hartig, O., Van Herwegen, J., De Vocht, L., De Meester, B., Haesendonck, G., Colpaert, P.: Triple Pattern Fragments: a Low-cost Knowledge Graph Interface for the Web. *Journal of Web Semantics*. 37–38, 184–206 (2016).
8. Mulzer, W., Rote, G.: Minimum-weight triangulation is NP-hard. *Journal of the ACM (JACM)*. 55, 11 (2008).
9. Leach, G.: Improving worst-case optimal Delaunay triangulation algorithms. In: 4th Canadian Conference on Computational Geometry. p. 15. Citeseer (1992).
10. Jaromczyk, J., Toussaint, G.T.: Relative neighborhood graphs and their relatives. *Proceedings of the IEEE*. 80, 1502–1517 (1992).
11. Matula, D.W., Sokal, R.R.: Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane. *Geographical analysis*. 12, 205–222 (1980).
12. Keil, J.M., Gutwin, C.A.: Classes of graphs which approximate the complete Euclidean graph. *Discrete & Computational Geometry*. 7, 13–28 (1992).