Adaptable Metro Maps

Hannah Bast
University of Freiburg
Freiburg, Germany
bast@cs.uni-freiburg.de

Patrick Brosi
University of Freiburg
Freiburg, Germany
brosi@cs.uni-freiburg.de

Sabine Storandt
University of Konstanz
Konstanz, Germany
sabine.storandt@uni-konstanz.de

ABSTRACT

We investigate the problem of finding the most likely geographic path a public transit vehicle (road-, rail- or water-bound) will take between a sorted list of given station coordinates. This sparse map-matching task frequently arises when real-world schedule data (where the geographic information often only consists of imprecise station locations) should be prepared for visualization, for example in route-planners or in transit-map drawing. It can also be a valuable pre-processing step for the closely related problem of on-line matching live passenger GPS data (e.g. from a smartphone) to a specific public transit vehicle. Our approach is implemented in a publicly-available tool called *pfaedle* which matches arbitrary transit schedules (given as GTFS files) against geographic data from OpenStreetMap with high precision. We test our approach by comparing vehicle paths published by the transit agencies of several cities against the paths found by our tool. Previous work either primarily considered non-sparse private transport scenarios, did not handle inter-hop turn-restrictions, did not find a globally optimal path and/or lacked an extensive evaluation against real-world schedule data.

CCS CONCEPTS

• **Information systems → Geographic information systems**; • **Theory of computation → Routing and network design problems**;

KEYWORDS

Public Transit, Map Matching, Schedule Data, GTFS

ACM Reference Format:
Hannah Bast, Patrick Brosi, and Sabine Storandt. 2020. Adaptable Metro Maps. In *Proceedings of SIGSPATIAL '20.* ACM, New York, NY, USA, **??** pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Maps of public transit networks usually depict the lines in a schematized way to ensure readability. In 1931, Harry Beck presented his idea to draw the London subway lines as alternating sequences of horizontal, vertical and diagonal line segments [?]. This octilinear design has since become the de facto standard and its usage goes beyond the cartographic representation of public transit networks.

SIGSPATIAL '20, November 3-6 2020, Seattle, Washington, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

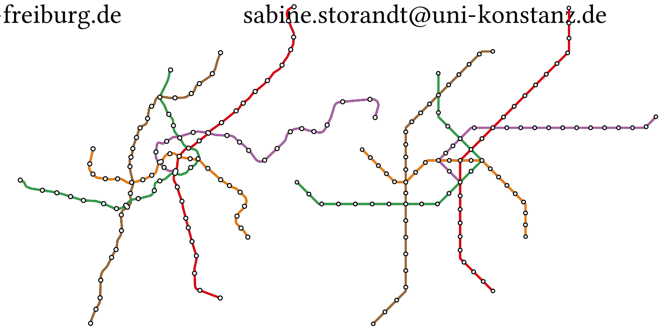

Figure 1: Left: The Vienna subway network, drawn with real-world geographical station positions and line courses. Right: Octilinear drawing by our approx. approach. Octilinearization took 202 ms.

The high practical relevance of these maps has lead to numerous approaches to render them automatically. We give an overview of existing work in Section **??**. However, existing methods usually do not guarantee octilinear results, require impractically long solution times and/or only allow a small fixed number of bends (or none at all) along edges in the final drawing. This leads to several restrictions in their practical applicability. In particular, previous work which guaranteed octilinear results often did not have solution times fast enough to be used interactively in a map editor. Additionally, we are not aware of any previous work which allows octilinear drawings to approximate the real geographical courses of a line between stations, which is a requirement if the final maps should be combined with either existing maps or satellite imagery. In this work, our goal is to overcome these restrictions.

1.1 Contributions

We consider the following as our main contributions:

- A
- B

1.2 Problem definition

1.3 Related Work

2 NODE SPLITTING

3 OCTILINEAR HANAN GRID

4 QUAD-TREE GRID

5 ORTHO-RADIAL GRID

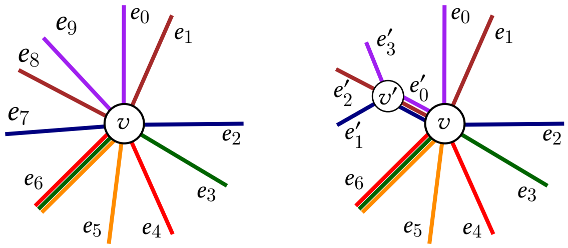

Figure 2: Left: Node v in an input line graph has a degree of 10, making it impossible to render the graph in an octilinear fashion. Right: We keep the first (in clockwise order) 7 adjacent edges of v, combine the lines of the remaining edges e_7, e_8 and e_9 into a single new edge e'_0 and connect it to a new non-station node v'. In reality, v' is given the exact same location as v to not distort node move penalties later on.

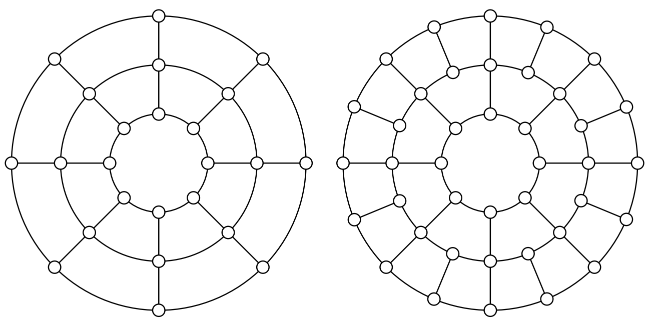

Figure 3: Two kinds of ortho-radial grid graphs. Left: Ortho-radial grid graph with $b = 8$ and a central node. Right: Ortho-radial grid graph where b is doubled each time the radius doubles.