Morphing of Schematized Polygons for Animated Travel-Time Maps

Axel Forsch
Institute of Geodesy and Geoinformation
University of Bonn, Germany
forsch@igg.uni-bonn.de

Ruben Kemna
Institute of Computer Science
University of Bonn, Germany
s6rukemn@uni-bonn.de

Elmar Langetepe
Institute of Computer Science
University of Bonn, Germany
elmar.langetepe@informatik.uni-bonn.de

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

***Abstract*—Travel-time maps are an important tool for analyzing the efficacy of public transportation systems. These maps display one or multiple isochrones. An isochrone is a line of equal travel time, which means that reaching any point on it from a given starting point requires the same amount of time. As timetables are usually very irregular over the course of a week, travel-time maps are highly variant with respect to the starting time. Thus, for an extensive analysis of the network, this variance has to be visualized appropriately. One way of doing this is by using a digital map with animated transitions between different isochrones, so called morphs. In this work, we present an algorithm for computing such morphs between isochrones, particularly targeting schematized travel-time maps as they are often used for visualization in the context of public transportation. In our current research, we are optimizing the morphing between two given polygonal lines. In future research we plan to extend the approach to finding the correct polygon correspondences in the given isochrones.**

***Index Terms*—morphing, polyline, schematization, octilinear**

I. Introduction

The travel time is one of the most important factors when analyzing transportation systems. This is especially true for commuters, who typically aim at minimizing the time spent on their way to and from work. For this purpose, travel-time maps designed for public transportation maps have been developed [1]. Given a starting location and a starting time these maps show the area that is reachable within a predefined travel time. The outline of such an area is called *isochrone* and consists of one or multiple polygonal rings. Due to the timetable of the public transportation, the resulting maps are highly variable with respect to the starting time. To analyze travel times, visualizations for different starting times are needed. Hence, the question arises how to visualize this temporal variance in travel times. In our work we present a morph operation to animate changes between different travel-time maps. Another application area is the visualization of user-dependent routing preferences. In this case one may visualize lines of equal travel cost, where the cost function may have been learned from the trajectories of a user [2]. Morphing between lines generated with differed cost functions may help to visualize their differences. To simplify the visualization and to increase the readability of maps, schematization techniques have been developed. One type of schematization is to restrict the orientation of isochrones, e.g. to octilinear edges which are either horizontal, vertical or diagonal [3]. Our approach particularly aims at morphing such schematized isochrones, keeping the schematic property at all stages during the operation. For rectilinear polygons this problem has already been studied in the context of building footprints for continuous integration [4]. For this use-case, the polygons are assumed to be geometrically close to each other, only having differences due to different levels of detail, which is not necessarily the case for isochrones.

DFG Grant Ha 5451/7-1 in Priority Programme 1894 "Volunteered Geographic Information."

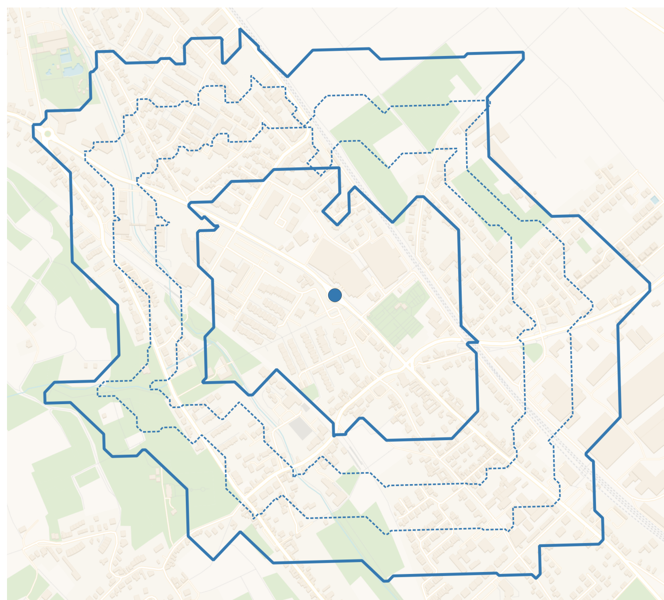

Fig. 1. Morphing operation between two isochrones that consist of a single polygonal ring each. Two intermediate morphing steps are shown by the dotted lines. Map tiles by OpenStreetMap contributors, ©CARTO.

On the poster we concentrate on generating schematized transitions for octilinear input polygons, nevertheless our approach is easily extendable to any set of orientations. Figure 1 shows an exemplary result for octilinear input polygons. Displayed are two intermediate morphing steps when morphing from the inner, smaller isochrone to the outer one.

II. Research Goals

Computing a morphing operation for schematized input polygons imposes additional constraints on the intermediate steps of the morph. In specific, these intermediate steps should have the same schematization properties as the input. We extended a dynamic programming approach as presented by Nöllenburg et al. [5] to be applicable for schematized input polygons under these constraints.

III. Algorithmic Approach

Our algorithm for computing the morphing operation between two polygons extends previous research by Nöllenburg et al. [5] on morphing non-schematic polylines. In a first step, we transform the problem of morphing two polygons into each other to morphing two polylines by taking the polygons' outlines and splitting them at specific points. We continue by finding the best morphing operation between the polylines. In the following we call one polyline the *source* and the other one the *target*. Looking at the problem from the perspective of point movements, the algorithm can be described as a matching process between the vertices of the two given polylines. For the point movement we consider uniform motion, meaning they have constant velocity along their path. We introduce three basic operations that occur during a morph, which are displayed in Figure 2. A *match* is the operation where the two endpoints defining an edge section of the source polyline are matched to two endpoints defining an edge section of the target polyline (Fig. 2, blue). As the schematization of the morph should be consistent with the input polygons, a match operation is only valid between segments of same orientation. The *insert* operation spawns a new line segment in the target from a single point in the source (Fig. 2, green). Further, the *delete* operation is the symmetric operation where one segment in the source gets eliminated by reducing it to a single point in the target (Fig. 2, red).

In some cases, one of the polylines might have a large number of segments in an area where the other polyline only has a single segment. To ensure that the quality of the matching is not influenced by the sampling rate of the polylines we introduce two additional, composite operations namely *multi-insert* and *multi-delete*. These operations allow multiple segments of one polyline to be matched to a single segment in the other polyline. These multi-operations are broken down to a sequence of basic operations by inserting auxiliary points on the single segment, splitting it into several *sections*. One important observation is that no matter where the auxiliary points are inserted, as long as their order is preserved, the resulting morph conforms to the schematization. Figure 2 shows a multi-insert for the segment $s_1 s_2$ of

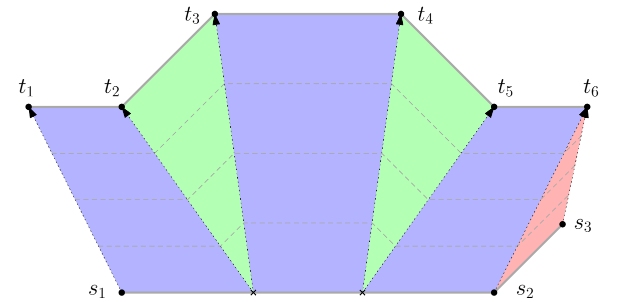

Fig. 2. The lower line s is morphed to the upper line t. To guarantee an octilinear morph auxiliary vertices (crosses) are inserted in s. Matches (blue) are only allowed between segments of same orientation. For all other segments either an insert (green) or delete (red) takes place. All intermediate polylines (dashed gray) during the morph follow the schematization of the input.

the lower polyline. A morph between two sections of non-conforming orientation can be achieved with, depending on the context, either an insert or a delete operation. This way, all intermediate polylines that occur during the morph follow the schematization of the input polylines.

To optimize the morph and to evaluate its quality we propose different scoring functions that consider geometric properties such as the distances between matched points and the similarity of the segments. For computing an optimal morph we can apply the dynamic programming approach by Nöllenburg et al. [5], adding the additional constraint that matches are only allowed between segments with the same orientation. On our poster we will show solutions obtained with different scoring functions.

IV. Conclusion

We sketched an algorithm for computing a morphing operation between two schematized polygons. The resulting morph has the same schematic properties as the input during its execution. In general, isochrones computed for public transportation networks do not consist of a single polygonal ring but of multiple ones. Similarly, two isochrones that are to be morphed not necessarily consist of the same number of rings. Thus, correspondences between the individual rings of each isochrone need to be found. Therefore, in order to apply this approach to travel-time maps future research in the context of polygon matching is needed.

References

[1] J. Gamper, M. Böhlen, W. Cometti, M. Innerebner, "Defining isochrones in multimodal spatial networks," in Proc. 20th ACM CIKM'11, pp. 2381–2384, 2011.

[2] J. Oehrlein, A. Förster, D. Schunck, Y. Dehbi, R. Roscher, J.-H. Haunert, "Inferring Routing Preferences of Bicyclists from Sparse Sets of Trajectories," in Proc. 3rd SDSC, pp. 107–114, 2018.

[3] A. Forsch, Y. Dehbi, B. Niedermann, J. Oehrlein, P. Rottmann, J.-H. Haunert, "Multimodal travel-time maps with formally correct and schematic isochrones." Transactions in GIS, 2021.

[4] J. Li, X. Li, T. Xie, "Morphing of building footprints using a turning angle function." ISPRS Int. J. Geo-Inf., vol. 6, p. 173, 2017.

[5] M. Nöllenburg, D. Merrick, A. Wolff, M. Benkert, "Morphing polylines: a step towards continous generalization." Computers, Environment and Urban Systems, vol. 32, pp. 248–260, 2008.