AGILE: GIScience Series, 2, 13, 2021. https://doi.org/10.5194/agile-giss-2-13-2021
Proceedings of the 24th AGILE Conference on Geographic Information Science, 2021.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras.
This contribution underwent peer review based on a full paper submission.

An Approach to Assess the Effect of Currentness of Spatial Data on Routing Quality

Martin Schmidl[a], Gerhard Navratil[b] and Ioannis Giannopoulos[b]

martin.schmidl@alumni.tuwien.ac.at, gerhard.navratil@geo.tuwien.ac.at, igiannopoulos@geo.tuwien.ac.at

[a]TU Wien, Austria
[b]Research Group Geoinformation, Department of Geodesy and Geoinformation, TU Wien, Austria

Abstract. During spatial decision making, the quality of the utilized data is of high importance. During navigation these decisions are crucial for being routed to the desired destination (usually going by the shortest or fastest route). Road networks, the main data source for routing, are prone to changes which can have a big impact on the computed route and therefore on travel time. For instance, routes computed using an outdated street network can result in longer travel times, in longer distance, as well in cases where the desired destination might not be anymore reachable via the computed route. Data from OpenStreetMap with different timestamps allows us to download road network snapshots from different years, i.e., from 2014 to 2020. On each of those datasets the fastest route between 500 randomly chosen point pairs in Vienna, Austria, was computed. These routes were also reconstructed on the most recent dataset for evaluation reasons. The resulting travel times, travel length as well as feasibility of the route were compared with the most recent dataset. The results provide a first assessment of temporal quality based on the currentness of a dataset.

Keywords. data quality, navigation, routing, VGI, OpenStreetMap

1 Introduction

The usage and provision of the most up-to-date street network data is important, especially in the vehicular navigation. This is not limited to the private sector, it may also apply to business cases (e.g., adoption of the travelling salesman problem on delivery services). Originally, such services were using commercial data with high costs for acquisition and updates of the data. In the last decade, Volunteered Geographic Information (VGI) (Goodchild, 2007) caused a paradigm shift. Services like OpenStreetMap (OSM) started in 2004 and provide a further valuable data source. VGI enables citizens to share spatial information with everyone. OSM data is published under the Open Database License (OdbL) and is well suited for navigation purposes, amongst other.

In this work we focus on comparing routes computed based on historical OSM street network data. An outdated dataset might lead to wrong instructions that would require unnecessary manoeuvres and therefore frustrate the user. As OSM data also contains road-networks, the information required for routing services can be extracted and used. Technical issues, such as importing the data in an application or merging them with other data sources can be easily resolved. An important question is, next to data integration, *how often the data should be updated in order to provide the best possible routes*. In order to answer this question, it is necessary to know the degradation of routing results over time due to missing data updates. The approach presented in this work uses the comparison of data snapshots at different years to assess the effect. Geofabrik[1] offers annual snapshots of OSM data, some of which are limited to continents or even countries, starting with the year 2014. Therefore, data for seven different years between 2014 and 2020 were available for the analysis.

The fastest route computation between an origin-destination pair might show some variation between years since the underlying data (such as turn restrictions or speed limits) change over the years. The routes

[1]http://www.geofabrik.de/en/index.html/

derived from the 2014 to 2019 datasets were mapped onto the 2020 dataset, simulating a user being routed based on outdated data. In the best case, travel time and route geometry will not change at all. In other cases, the optimal route might change and the solution produced based on the old dataset might be longer (e.g., when new streets would allow for a shorter route), slower (e.g., when speed limits have been changed), or even not feasible (e.g., when the direction of a one-way has been changed).

The presented approach is based on the computation of a large number of different routes. 500 routes with random start and end points were computed for Vienna, Austria for all the years between 2014 and 2020 (in total 3500 routes). This should provide a first quantitative assessment of the deterioration of the decision based on the age of the dataset and answer the question: *Is it possible to quantify the effect of outdated road network data on routing quality?* However, even minimal changes of the decision or even the location may lead to significantly different results and thus the derived numbers may not be representative.

It is inevitable to raise concerns about the data quality whenever spatial data is used. Especially in the field of VGI this quality is difficult to assess, as many devices and methods can be used to record the data. Also different viewpoints of the contributing users could give potential for tag wars between them (Goodchild and Li, 2012). Previous work on spatial data quality and especially VGI is described in section 2, while section 3 mentions the problems regarding navigation in general. Section 4 lists the used datasets and software packages, explaining also the processing pipeline, which is based on Python scripts accessing a server running the Open Source Routing Machine (OSRM). The analysis of the computed routes is described in detail, as routes have to be recomputed as a result of incomplete or unsuccessful matching (i.e., not feasible routes). After successfully computing all routes, the resulting data was used for statistical analysis in order to determine how outdated information can affect the travel time and distance as well as if a route is still feasible to follow. A first assessment of the impact on the data and examples on why a route might show those changes are described in section 5.

Due to the freely available data and software tools, our proposed quality assessment of road-network data from the OpenStreetMap can be easily reproduced. The goal of this work is to provide a first quality assessment of how an outdated data set can impact routing decisions.

2 Data Quality and Navigation Problems

Modern information and communication technologies provide easy methods for the exchange of data. Spatial data is no exception, thanks to open-data initiatives and VGI platforms. Today, a vast amount of data is available. However, the quality of these data is very heterogeneous. Quality information is connected to the usage of spatial information in several ways: First, the quality needs to be described. This requires parameters describing specific aspects of the quality. Secondly, the parameters need to be quantified for a given data set. Thirdly, the effect on the use needs to be addressed. Since the OSM data are used, some remarks on the quality of OSM are necessary. The same is true for problems that may occur with the matching between OSM data from different years.

2.1 Description of Spatial Data Quality

The ISO (International Organization for Standardization) created a standard for the quality of spatial data (ISO 19157) based on work initiated by the International Cartographic Association (Guptill and Morrison, 1995). The standard aims to define parameters for the assessment of the quality of a dataset. In addition, a recommended workflow to quantify the parameters is described. These quality parameters are part of the metadata, defined in a further ISO standard (ISO 19115). Each data quality element in ISO 19157 describes a different aspect of the quality of the underlying data:

- *Completeness* refers to the amount of missing (omission) or excess (commission) data.
- *Logical consistency* describes contradictions in the model. The standard distinguishes between conceptual, domain, format, and topological consistency.
- *Positional Accuracy* is defined as the accuracy with which the position of the features contained in the dataset is described.
- *Thematic accuracy* describes the accuracy of quantitative and correctness of non-quantitative attributes.
- *Temporal quality* refers to the quality of attributes and relationships, which have one or more temporal characteristics. Time in this case can be defined as a single point in time, but also as a time period.

ISO 19157 also defines optional qualitative aspects, for which the authors propose following elements: *Lineage, usage, credibility, trust worthiness, content quality, vagueness, local knowledge, experience, recognition, and reputation*. However, these concepts are more difficult to assess than statistical parameters. Some of these elements may even have different approaches to assess the parameter. Yu et al. (2014), for example, define the trustworthiness of sensory data by the correspondence with the contributions of other sensors.

Fogliaroni et al. (2018) assess the trustworthiness of information and the reputation of contributors by analyzing their edits geometrically, qualitative, and semantically.

These elements describe quality from the producer's perspective. Since this could be difficult to understand, Chrisman (1984) defined ***fitness-for-use***. Here, the suitability of a data set for a specific application is described to provide a basis for the decision of users with similar applications.

2.2 Quantification of Spatial Data Quality for VGI

While in the professional sector (e.g., national mapping agencies etc.) the quality assessment is done after the criteria defined in the ISO standard, this is not as easy for VGI. Some proposals have been made in the past. Goodchild and Li (2012) proposed three approaches to assess VGI quality: the crowdsourcing approach, the social approach, and the geographic approach. Meek et al. (2014) suggest to use a third quality model, which sits in between the internal and external quality models which is called the *"stakeholder"* model, consisting of the six elements *vagueness, ambiguity, judgement, reliability, validity, and trust*. Senaratne et al. (2017) provide a detailed overview of quality assessment methods, and even distinguish between map-based VGI, image-based VGI and text-based VGI. They also define that **data-mining** consists of recognizing geographical patterns and rules by machine learning. This can be used completely independent of the other mentioned aspects.

2.3 Assessing the Effect on the Application Results

Uncertainty in geographic data can have an effect on decisions made based on assumptions derived from the data. Heuvelink (2002) proposed the use of the Monte Carlo method to assess the uncertainty. While computationally demanding it can deal with a wide range of situations. A suitable system was implemented by Karssenberg and De Jong (2005). However, Heuvelink also mentions a number of problems related to the description of data quality. One of them is the spatial and eventually even temporal variation of quality, a topic also addressed by Tsutsumida and Comber (2015). Krek (2002) used a different aspect of data quality. She investigated the effect of incomplete information on a wayfinding application.

2.4 Quality of OSM data

As this work uses OSM data, quality aspects should be investigated before using it. As stated above, Senaratne et al. (2017) list previous work and the relevant quality aspects for map-based VGI, thus these are also relevant for the OpenStreetMap. Haklay (2010) presented a comparison between OSM and Ordnance Survey data for England and Zielstra and Zipf (2010) between OSM and TeleAtlas data for Germany to assess completeness. In his work, Will (2014) describes some quality assessments especially for road-networks. Temporal quality has not been mentioned in the literature very often, Girres and Touya (2014) however made a comparison between data from the OSM and official data available in France. Of all quality elements described there, the most relevant would be *"currentness"*. To give a first overview of that, existing features between June and October 2009 were counted. In this time, the amount of objects has increased by nearly 30 percent. Apart from this, no further literature dealt with currentness of a data set and how it impacts the spatial decisions made on the data.

2.5 Navigation Problems

In the experiment presented in this paper, the fastest route is determined using snapshots of the road network from different years. The latest dataset is defined as current ground truth. The routes determined as optimal using older datasets are then matched onto the latest dataset and compared to the fastest route determined using the latest dataset (the reference route). Four cases that can occur in this process. The first case is the best possible solution: A route does not change in any dataset used. This is represented in the data by a constant travel time over all seven years. Of course this will not be the case with every route, as the road network is constantly facing changes and updates and the OpenStreetMap data reflect these changes. A typical change is a lower speed limit. This leads to the second case, where the matching is successful (i.e., the routes can be driven), but the the original time estimate is wrong. The third case refers to the situation where the route can be matched but there is a faster route in the reference dataset following a different geometry. The final case occurs, if the matching was not successful at all. The matching might be incomplete up to an incidence point. This is mostly represented by a change in one-way restrictions or access restrictions, as a road might not be able to be used in the most-recent dataset. In this case, the existing matching from the starting point up until the incident point is seen as a first part of a route (i.e., matching part). To simulate the user going to the original destination from the incident point, the fastest route from this incident point to the original destination on the most-recent dataset is computed. This might not include the time the user needs to find this solution, but it provides an optimal solution to estimate the time and distance needed to reach the destination. This part is forming the second part of the route then (i.e., routing part). Both parts are coupled by adding the two geometry parts and summing

up the travel times and distances. This case assumes that the user would have an alternative option to compute a route on current data. Tab. 3 shows the frequency of each situation.

3 Methods

3.1 Data and Software Availability

The data used for this experiment can be retrieved from the Open Science Framework (OSF)[2]. Geofabrik is offering OSM snapshots for the whole world, but the data can also be downloaded for different subregions, in this case for Austria. The data is available under the Open Data Commons Open Database License (ODbL). Osmconvert[3] was then used to crop the data to the area of Vienna. The cropped data as well as the polygon used to crop the data can be found on OSF.

The Open Source Routing Machine (OSRM)[4] was used for routing and matching. It was installed via a Docker image on a Virtual Machine running the Lubuntu Linux distribution. OSRM is available under the 2-Clause-BSD License. The communication with the OSRM server and the analysis itself was done via Python scripts. The Anaconda data science tool-kit and some freely available Python modules were used for the scripts (i.e., NumPy, requests and geojson).

The Anaconda Individual Edition[5] was used as an environment for running the Python scripts and is available under the 3-Clause-BSD License. QGIS[6] was used for the display of the routes. It is available under the GNU General Public License 2. As every route computed will be different, the 500 computed routes are also provided on OSF in accordance with the AGILE Reproducible Paper Guidelines. A more detailed setup process is available in the work of Schmidl (2021).

The workflow underlying this paper was partially reproduced by an independent reviewer during the AGILE reproducibility review and a reproducibility report was published at https://doi.org/10.17605/osf.io/bdu28.

3.2 Workflow

The workflow of creating the routing data follows a linear structure. For each combination of start and end point the following steps are performed:

- Create a set of the fastest routes for the seven different network data sets (*01_get_routes.py*)
- Check the fastest routes for completeness and correctness in QGIS
- Match the routes in the 2014 to 2019 datasets into the 2020 dataset (*02_match_routes.py*)
- Visually check for possible errors in the matched routes in QGIS again

In case of a severe error in the the matching process (e.g., variations of start end end point throughout the years) the set of fastest routes was discarded and also would not be included in the total of 500 sets of routes. This guarantees that all 3000 matched routes consist of complete routes that are able to be followed from the start to the end point.

After the 500 sets of matched routes have been checked manually, the travel times and distances are stored in a CSV file with the third Python script (*03_write_data.py*). This CSV file is then loaded into a spreadsheet tool to further assess the quality change.

3.3 Implementation

The OSRM server setup is realised on a Virtual Machine running the Lubuntu Linux distribution and OSRM over a Docker image. Each dataset is addressed through its own port. Ports 5014 to 5020 were used, where the last two digits specify the year of the dataset. Three services of the OSRM engine were used, *Nearest, Route, and Match.*

As random sets of coordinates within the bounding box of Vienna were computed, several of them were not on the road network. The Nearest service is useful for snapping the random points on the network, as it returns the coordinate pair on the road network with the least distance to a given coordinate pair. The Route service is the main routing service included in OSRM and finds the fastest route between the two points that have been snapped to the road network before. Finally, the Match service takes any number of coordinate pairs and tries to match a route to the road network in the most likely way, with the points being kept in their original order. It supports a threshold around the points, which can be useful in this case, as links of roads might move slightly over the years and the service is able to remove outliers automatically.

For the pre-processing pipeline, the implemented Multi-Level Dijkstra (MLD) approach (Luxen and Vetter, 2011) was used, as it is recommended for default usage over the Contraction Hierarchies (CH) approach. The Multi-Level Dijkstra approach is an extended Dijkstra algorithm that uses precomputed overlay cells to speed up the computation (Möhring et al., 2005; Hamme, 2013). OSRM can derive routes for different modes of transportation. The car profile was used

[2] https://doi.org/10.17605/osf.io/rxcgj or https://geoinfo.geo.tuwien.ac.at/resources/
[3] https://wiki.openstreetmap.org/wiki/Osmconvert
[4] http://project-osrm.org/
[5] https://docs.anaconda.com/anaconda/
[6] https://www.qgis.org

in this study. This has an effect on the rules used by the system, e.g., respecting one-way roads and turn restrictions. The partitioning and customizing process described in the OSRM Quick Start guide was needed to prepare the multi-level partitions and overlay cells for the MLD approach so that the actual routing service can be run in a reasonably short amount of time. The only parameter that has been different to the given instructions was the maximum matching size of up to 100 coordinate pairs. This limit could be reached by a resulting route. Thus, the limit had to be increased. The low default value is an attempt to avoid unnecessary workload for the server due to excess requests. This can be ignored in our case, as the server is used for the experiment only.

When setup correctly, HTTP requests can be realized through a URL. The response shows if the request was successfully computed, and returns a routing information. Two additional parameters were used for the experiment with each routing request: The parameter *overview=full* returns a non-simplified geometry and the parameter *geometries=geojson* provides the decoded, GeoJSON-formatted route. Therefore a request will give back following parameters:

- code - "Ok" when a route could be computed, otherwise it will display an error code.
- waypoints - An array with details about certain waypoints on the route (e.g., street names or segment lengths). This result is ignored.
- routes - An array containing the route itself and useful parameters like travel time and distance driven.

Detailed step-by-step instruction for each decision point could be obtained with the parameter *steps=True*. However, this is not necessary for this experiment.

Each route is calculated by the following process: First two random points are determined in a bounding box over Vienna. Then, these points are snapped to the nearest point on the street network to prevent problems with the different network snapshots. A random point with an almost identical distance to two or more nearest road segments could be mapped to different road segments if edits of the road geometry occurred. For that reason, all randomly created points have been snapped to the road network on the 2020 dataset. The random points are visible in Fig. 1. Green dots represent starting points and red points destination points. The point pair is then used as input for the route calculation for each year between 2014 and 2020.

The fastest routes are computed with individual requests to the OSRM server, thus seven requests per route are performed, one for the network of each year (i.e., from 2014 to 2020). Four parameters were saved for the results: The route itself in two formats, as geometry in the LineString format and completely in

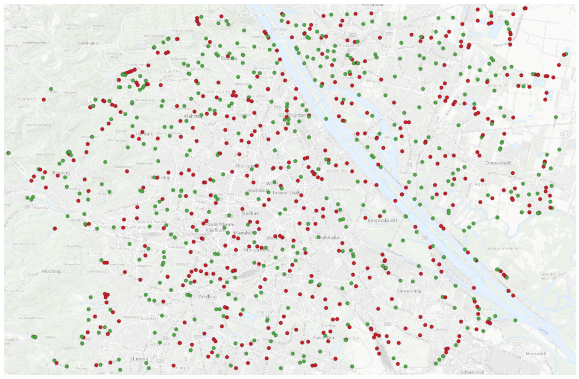


Figure 1. Distribution of the 1000 origin and destination points used in the experiment

a GeoJSON-conform format, the travel time and the travel distance. Each of these parameters can be found in the 'routes' array returned by the server. The fastest routes are saved in a fitting format in order to examine them for consistency and correctness. The GeoJSON file format can be used as routes are just simple LineStrings. GeoJSON is an open standard for the representation of simple geographical features (Butler et al., 2016). Elements also can have non-spatial attributes, which is suited for saving travel time and travel distance along with each route. Thus, in each single GeoJSON file a FeatureCollection is created, which always contains seven LineStrings representing the fastest route for each year, with each of those routes having two non-spatial attributes, namely duration and distance.

The parameters of the routes can be accessed and compared through attribute tables. Tab. 1 shows the attributes for the seven results for a single route.

year	duration [s]	distance [m]
2014	1325.5	18140.1
2015	1327.0	18138.1
2016	1326.3	18132.8
2017	1326.4	18133.0
2018	1322.8	18132.0
2019	1326.7	18136.7
2020	1329.3	18137.6

Table 1. Attribute table of a GeoJSON file containing the fastest routes per year, as viewed in QGIS

Small deviations in travel time and travel distance (columns duration and distance in Tab. 1) may occur even if no significant change in the route geometry is visible. These small deviations can result from small shifts of the road segments or more detailed modelling of reality. For each file of fastest routes certain requirements are examined manually. This includes completeness of the attributes for the route for each year via the attribute table and a visual inspection of the route it-

self. This visual inspection guarantees for consistency in start and end point. Once this examination is successful, the focus can be shifted to recreating the routes from 2014 to 2019 on the most recent (2020) dataset. The 'match' routine of OSRM is used for this step. It matches a given track to a road network in the most plausible way and is mainly used to match a track from GNSS to a road network. In this experiment, the route descriptions derived by the route service are used.

Again, two parameters in request for the matching are changed: *overview=full* forces the routine to return a non-simplified geometry and *geometries=geojson* sets the return format to that of the routing results. A matched route consists of more segments than the computed route because it merges the segmentation of paths with that of the network. The result from a matching request is similar to a routing request:

- code - "Ok" when a matched route could be computed, otherwise it will display an error code.
- tracepoints - Waypoint objects representing all points of the trace in order. This result is ignored.
- matchings - An array containing route objects and useful parameters like travel time and distance driven.

A particularity of the matching algorithm is the confidence parameter. It describes how confident the matching was between 0 and 1. A value of 0 means, that the route is very unlikely while 1 specifies confidence that the matching is correct. This parameter could be used in an automated routine to judge if the matching should be used or it should be discarded. The matching part of this experiment is performed by the second Python script (*02_match_routes.py*). It loads a GeoJSON file created from the first script used before and matches each route in the most recent dataset. The fastest route in the 2020 dataset is taken directly from the corresponding result. The result is the GeoJSON file consist of seven fastest routes from 2014 to 2020 and six recreated routes from 2014 to 2019 on the 2020 dataset. While the six fastest routes from 2014 to 2019 are only included for completeness reasons after this step, the fastest route from 2020 acts as the reference route for the six matched routes from 2014 to 2019, which the reference route is compared against. Again, the GeoJSON file can be loaded in QGIS for a quality examination. A style file defines that each entry gets a different color grouped by the 'year' attribute. Year numbers define the directly computed routes, routes with '.2020' attached to their value in the attribute 'year' are routes that have been mapped onto the 2020 dataset.

The first inspection in QGIS is done via the attribute table, where the travel time and distance for each dataset is listed. The first aspect to check for is that none of the recreated routes can have a travel time that is lower than that of the fastest route in 2020. This can have an array of reasons but would always be immediately visible and therefore that route would have to be either recalculated or withdrawn. Furthermore, if a route could not be fully recreated, it would be visible as being much shorter, concerning both, distance and time. An example can be seen in Tab. 2. The routes 2014 to 2020 have the same values as the ones in Tab. 1 as they are added to the final route file for completeness reasons. As visible in the duration column in A1, all values including the 2020 suffix in the year column and the fastest route in 2020 itself are the same, and therefore suggest to have the same travel time in each year. Another hint for the routes being the same is the visual overlap of all routes on the map view, when each of the six recreated routes and the fastest route from 2020 are displayed together. The small differences of about one meter in the distance column can be neglected, they are a product of the matching algorithm.

year	duration [s]	distance [m]
2014	1325.5	18140.1
2014.2020	1329.3	18138.0
2015	1327.0	18138.1
2015.2020	1329.3	18137.8
2016	1326.3	18132.8
2016.2020	1329.3	18137.3
2017	1326.4	18133.0
2017.2020	1329.3	18137.3
2018	1322.8	18132.0
2018.2020	1329.3	18137.8
2019	1326.7	18136.7
2019.2020	1329.3	18138.4
2020	1329.3	18137.6

Table 2. Attribute table of a GeoJSON file containing the fastest and matched routes, as viewed in QGIS

Tab. 2 shows the best case, where a route (and also the travel time) did not change because of using an older dataset. For this route in particular it means that none of the changes in the OSM data had an effect on the computed route. However, this is not the case for all 500 routes. For several routes, manual adjustment had to be performed, for example, if the matching produces an incomplete route. In such a case a route from the point, where the incident occurred, was computed to the final destination. This case is pictured in Fig. A2, in which a route from the incident point to the destination was computed for the routes from 2014 and 2015. Also this route had to be replaced later, as the travel time in 2020 was shorter, albeit it resembled the same route as in the years before. Both the travel time and the distance were added up and the data was saved in a GeoJSON file similar to a correctly computed route. If severe errors made a matching impossible or unlikely,

a completely new set of fastest routes was calculated and matched.

After all 500 sets of routes have been matched and checked successfully, the Python script (*03_write_data.py*) was used to export the data. In this process, a CSV-file containing the following information for each route was created: Route ID, Travel times in 2020 and then from 2014 to 2019, Distance in 2020 and then from 2014 to 2019. The difference between the route calculated in the 2020 dataset and the route calculated with any other dataset for both, distance and time, are stored in absolute numbers and in percent.

4 Results

In order to understand how the underlying OSM data changed the routing decisions over the years, it was counted how often a route could be matched successfully, whether it changed the travel time or not, and how often it was incomplete and had to be completed by a manual routing from the incident point to the original destination. For a total of 3000 routes that have been matched to the 2020 dataset, Tab. 3 lists the amount of each case described.

year	Identical route to 2020	Longer route than in 2020	Routes with a matching error
2014	252	235	13
2015	274	213	13
2016	322	170	8
2017	350	142	8
2018	362	130	8
2019	403	89	8
total	1963	979	58

Table 3. Comparison of results from the route matching process

After running the last script to automatically extract the data from the 500 GeoJSON files, the CSV file was opened in a spreadsheet software and statistical analysis was performed. The mean extension of travel time implied by using an out-dated dataset for vehicular navigation in Vienna can be described as follows, where the year states which dataset was used to match it to the 2020 dataset:

- 2014: + 1.47%
- 2015: + 1.34%
- 2016: + 1.09%
- 2017: + 0.70%
- 2018: + 0.65%
- 2019: + 0.36%

It is obvious that the use of older datasets leads to an extension in travel times and that there seems to be a continuous, linear deterioration of the quality of the decision. Thus, it can be assumed that the use of a four year old dataset leads to a decision that is approximately 1% worse than the optimum for this kind of application. In order to make the assessment more accessible, a second degree polynomial has been fitted to the values. This yields the following coefficients:

$$f(x) = -0.008x^2 + 0.2911x + 0.0319.$$

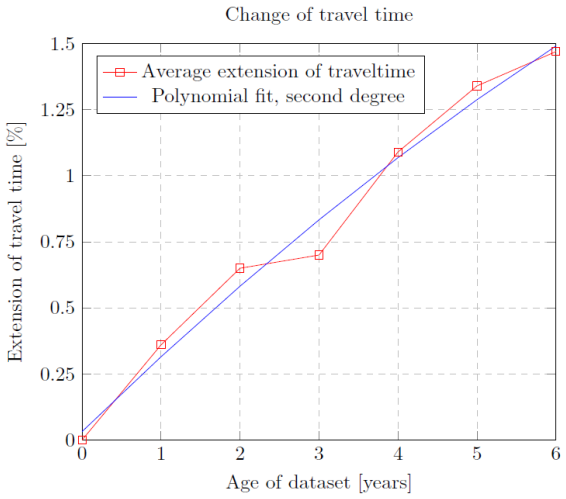


Figure 2. Travel time extension in percent based on the age of the data set in years

The mean extension of travel distance is less straightforward:

- 2014: + 0.04%
- 2015: - 0.15%
- 2016: - 0.23%
- 2017: - 0.25%
- 2018: - 0.11%
- 2019: + 0.14%

The reason for this behaviour is that the costs used to calculate the routes was based on travel time. Changes in the speed limit on a route segment could lead to avoidance of this segment. Travel time might not be heavily impacted by such a change but travel distance will most probably be affected. Thus, any quality assessment will have to consider the strategy of the algorithm used.

Several reasons for route changes that may lead to travel time extensions in the computed routes can occur. These reasons included U-Turns or turn restrictions that had been allowed before, but are prohibited in newer datasets. Two examples of construction works

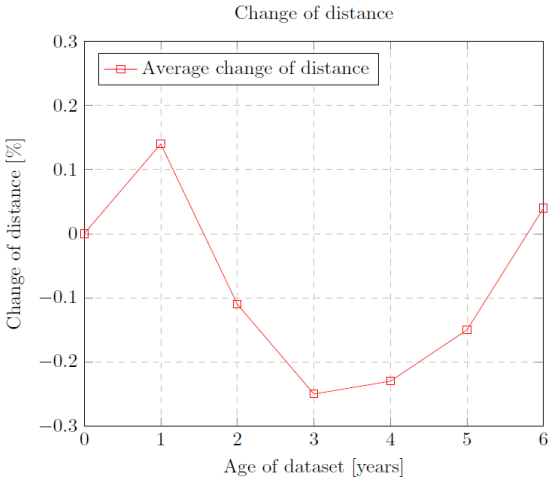


Figure 3. Travel distance extension in percent based on the age of the data set in years

influencing the routing heavily are the St.Marx motorway junction after 2016 or the area around Rennweg in 2020. Speed restrictions or one-way restrictions might also force the routing engine to look for a different route, whether reflecting the changes in reality or those have been added to the OpenStreetMap later on. Especially for changing speed restrictions another directly computed route on the 2020 dataset might have a shorter travel time (but a longer travel distance) because of the higher driving speed.

5 Discussion

In vehicular navigation it is important to always use and provide the most up-to-date representation of the road network to prevent illegal or even dangerous maneuvers. Due to the amount of spatial data that is available for every possible application and most of them being completely free to use, this data can be also used for navigation. As periodical dumps of OpenStreetMap data are no exception to that, the following question was raised: *Is it possible to quantify the extension of travel time by recreating fastest routes from previous years on the most-recent dataset?*.

By choosing a routing engine that is capable of computing fastest routes based on raw OSM data, a nearly linear extension in travel time could be observed. This extension could even be roughly fitted to polynomial of second degree in an attempt to model this extension. However, this should be analysed for different cities or agglomerations to include the factor of how different layouts of road networks have an influence on the extension of travel time. Another point to look at is the routing engine. The usage of different routing engines, e.g., Graphhopper[7] might result in different values for the extension of travel time.

The analysis of the change in travel distance was less straight-forward and no clear assumptions can be made. These changes might even vary more by conducting the same analysis on different cities or agglomerations and therefore again reflect changes based on the layout of the road network used in those.

Limitations clearly have been raised by the mostly manual approach to analyse all routes in QGIS. Due to the sheer amount of factors to consider, a full practical framework would have taken more time to create than to proceed with the whole approach manually. A good start for an automated framework could be to use the confidence parameter used by the matching progress. The manual approach has no influence on the result however. Also the only solution used for all aspects of the analysis was to investigate results based on the optimal solution. Of course in reality many other factors (e.g., traffic jams, temporary detours that have not been mapped) play a role in the process of finding a fastest route. This could be realised by using an agent-based approach, where an agent tries to follow the proposed route, especially when reaching an incident point and the planned routes is not possible to follow anymore. In that case, it uses a specific strategy to continue the trip, e.g., move straight ahead and recalculate the route from the current position to the destination. Different strategies could be implemented and tested with the goal to minimize the effect of using outdated datasets for navigation.

We also based the analysis on solely looking at the travel time and distance. For further work on that subject, it might be worth to split the routes into segments and check how many segments of the different routes are equal. Based on that, a relative value can be determined for every set of routes and compared to others or in general by computing meaningful statistical values.

6 Conclusion

With computing enough random routes across Vienna we could show that the currentness of a dataset indeed results in extended travel times, which increase depending on how out-dated the dataset is.

The following lessons were learned: The age of a dataset has a demonstrable effect on the quality of the decision, which increases over time, where as the effect of the use of older datasets on other aspects of the solution than the optimised one is unpredictable. Furthermore the experiment can be automated with a reasonable amount of work. This makes it easier to reproduce the analysis in other cities or even agglomerations after downloading the datasets for each year. Another qual-

[7] https://www.graphhopper.com/

ity assessment method that is worth having a further look into is based on how many segments of a route are equal to the reference route (fastest route) on the most current dataset.

It might also be interesting to adopt a different approach to assess the effect of the path from the outdated dataset: An agent-based approach could be used, which might provide a better assessment of the effects in real trips and could be used to adapt the agent strategy to minimize the effect of outdated datasets. It may also be interesting to extend the analysis to other locations. Navigation in rural areas may show a different pattern. Different target functions for the routing may also be interesting to include. Examples are the routes with the fewest turns (Duckham and Kulik, 2003), the least risk path (Grum, 2005), or the path with the least amount of segments.

Acknowledgements. We would like to thank the members of the OSRM-talk mailing list for providing helpful advice in using the OSRM routing engine. We also like to thank all OSM contributors for their efforts because they made the experiment possible.

References

Butler, H., Daly, M., Doyle, A., Gillies, S., Hagen, S., Schaub, T., et al.: The GeoJSON Format, Internet Engineering Task Force (IETF), https://www.hjp.at/(de)/doc/rfc/rfc7946.html, 2016.

Chrisman, N. R.: The role of quality information in the long-term functioning of a geographic information system, Cartographica, 21, 79–88, https://doi.org/10.3138/7146-4332-6j78-0671, 1984.

Duckham, M. and Kulik, L.: "Simplest" Paths: Automated Route Selection for Navigation, in: Spatial Information Theory. Foundations of Geographic Information Science, edited by Kuhn, W., Worboys, M. F., and Timpf, S., pp. 169–185, https://doi.org/10.1007/978-3-540-39923-0_12, 2003.

Fogliaroni, P., D'Antonio, F., and Clementini, E.: Data trustworthiness and user reputation as indicators of VGI quality, Geo-Spatial Information Science, 21, 213–233, https://doi.org/10.1080/10095020.2018.1496556, 2018.

Girres, J. and Touya, G.: Quality Assessment of the French OpenStreetMap Dataset, Transactions in GIS, 14, 435–459, https://doi.org/10.1111/j.1467-9671.2010.01203.x, 2014.

Goodchild, M.: Citizens as sensors: the world of volunteered geography, GeoJournal, 69, 211–221, https://doi.org/10.1007/s10708-007-9111-y, 2007.

Goodchild, M. and Li, L.: Assuring the quality of volunteered geographic information, Spatial Statistics, 1, 110–120, https://doi.org/10.1016/j.spasta.2012.03.002, 2012.

Grum, E.: Danger of getting lost: Optimize a path to minimize risk, in: 10th International Conference on Information Communciation Technologies (ICT) in Urban Planning and Spatial Development and Impacts of ICT on Physical Space, edited by Schrenk, M., pp. 709–715, TU Wien, Vienna, Austria, 2005.

Guptill, S. C. and Morrison, J. L.: Elements of spatial data quality, Elements of spatial data quality, 202, 1–12, https://doi.org/10.1016/s0098-3004(97)87525-5, 1995.

Haklay, M.: How good is volunteered geographical information? A comparative study of OpenStreetMap and ordnance survey datasets, Environment and Planning B: Planning and Design, 37, 682–703, https://doi.org/10.1068/b35097, 2010.

Hamme, J.: Customizable Route Planning in External Memory, Bachelor thesis, Institute of Theoretical Computer Science, KIT Karlsruhe, pp. 9–10, https://i11www.iti.kit.edu/_media/teaching/theses/ba-hamme-13.pdf, 2013.

Heuvelink, G. B.: Analysing uncertainty propagation in GIS: Why is it not that simple?, Uncertainty in Remote Sensing and GIS, pp. 155–165, https://doi.org/10.1002/0470035269.ch10, 2002.

Karssenberg, D. and De Jong, K.: Dynamic environmental modelling in GIS: 2. Modelling error propagation, International Journal of Geographical Information Science, 19, 623–637, https://doi.org/10.1080/13658810500104799, 2005.

Krek, A.: An Agent-based Model for Quantifying the Economic Value of Geographic Information, Ph.D. thesis, Institute for Geoinformation, TU Wien, 2002.

Luxen, D. and Vetter, C.: Real-time routing with OpenStreetMap data, in: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11, pp. 513–516, https://doi.org/10.1145/2093973.2094062, 2011.

Meek, S., Jackson, M., and Leibovici, D.: A flexible framework for assessing the quality of crowdsourced data, 17th AGILE conference on Geographic Information Science, http://hdl.handle.net/10234/98927, 2014.

Möhring, R., Schilling, H., Schütz, B., Wagner, D., and Willhalm, T.: Partitioning Graphs to Speed Up Dijkstra's Algorithm, in: International Workshop on Experimental and Efficient Algorithms, vol. 3503, pp. 189–202, https://doi.org/10.1007/11427186_18, 2005.

Schmidl, M.: Assessing the Effect of Currentness of Spatial Data on the Quality of Routing, Thesis, Research Group Geoinformation, TU Wien, https://doi.org/10.34726/hss.2021.33701, 2021.

Senaratne, H., Mobasheri, A., Ali, A., Capineri, C., and Haklay, M.: A review of volunteered geographic information quality assessment methods, International Journal of Geographical Information Science, 31, 139–167, https://doi.org/10.1080/13658816.2016.1189556, 2017.

Tsutsumida, N. and Comber, A. J.: Measures of spatio-temporal accuracy for time series land cover data, International Journal of Applied Earth Observation and Geoinformation, 41, 46–55, https://doi.org/10.1016/j.jag.2015.04.018, 2015.

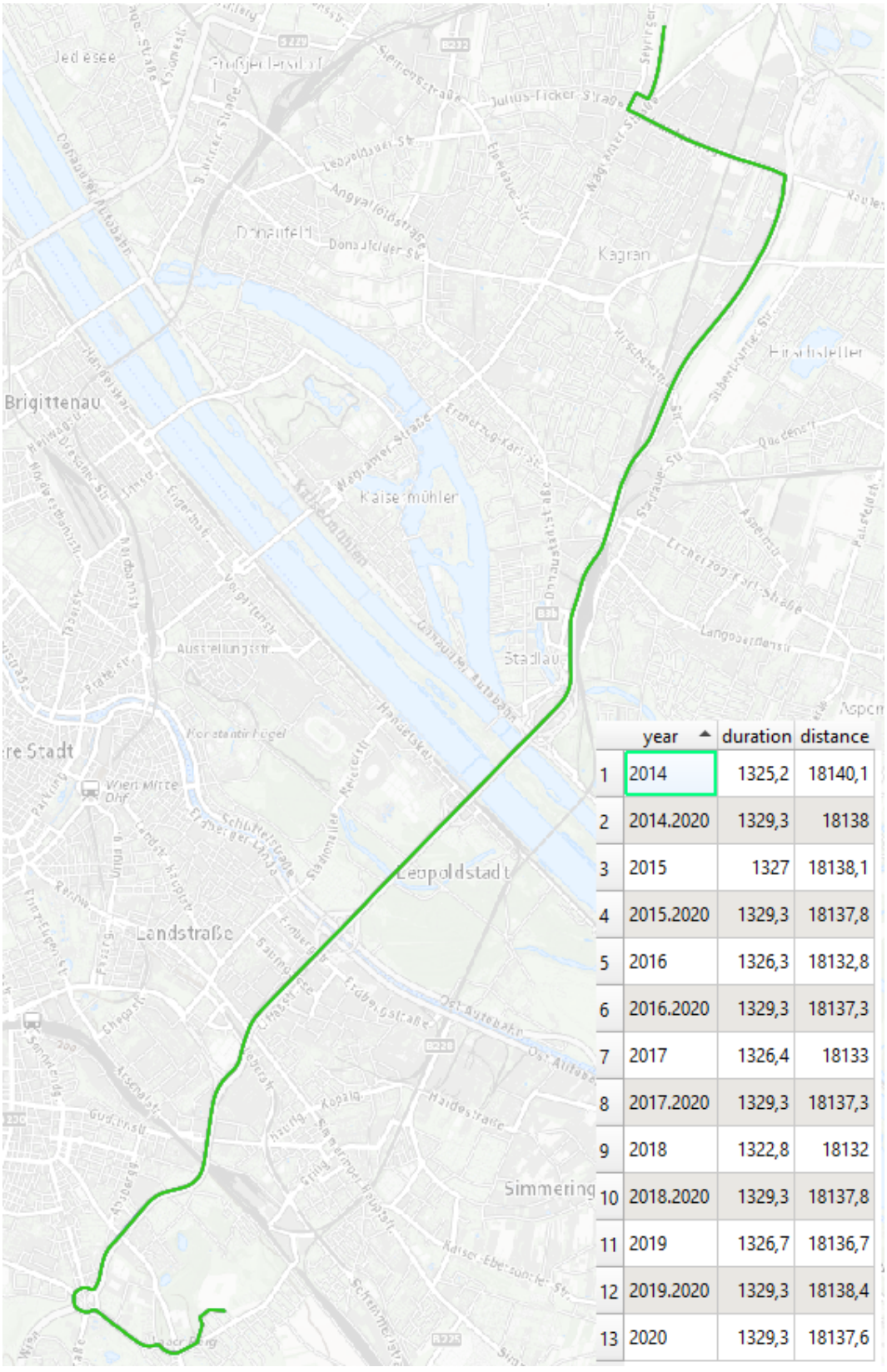
Will, J.: Development of an automated matching algorithm to assess the quality of the OpenStreetMap road network : a case study in Göteborg, Sweden, Student thesis series INES, http://lup.lub.lu.se/student-papers/record/4464336, 2014.

Yu, R., Liu, R., Wang, X., and Cao, J.: Improving data quality with an accumulated reputation model in participa-

tory sensing systems, Sensors (Switzerland), 14, 5573–5594, https://doi.org/10.3390/s140305573, 2014.

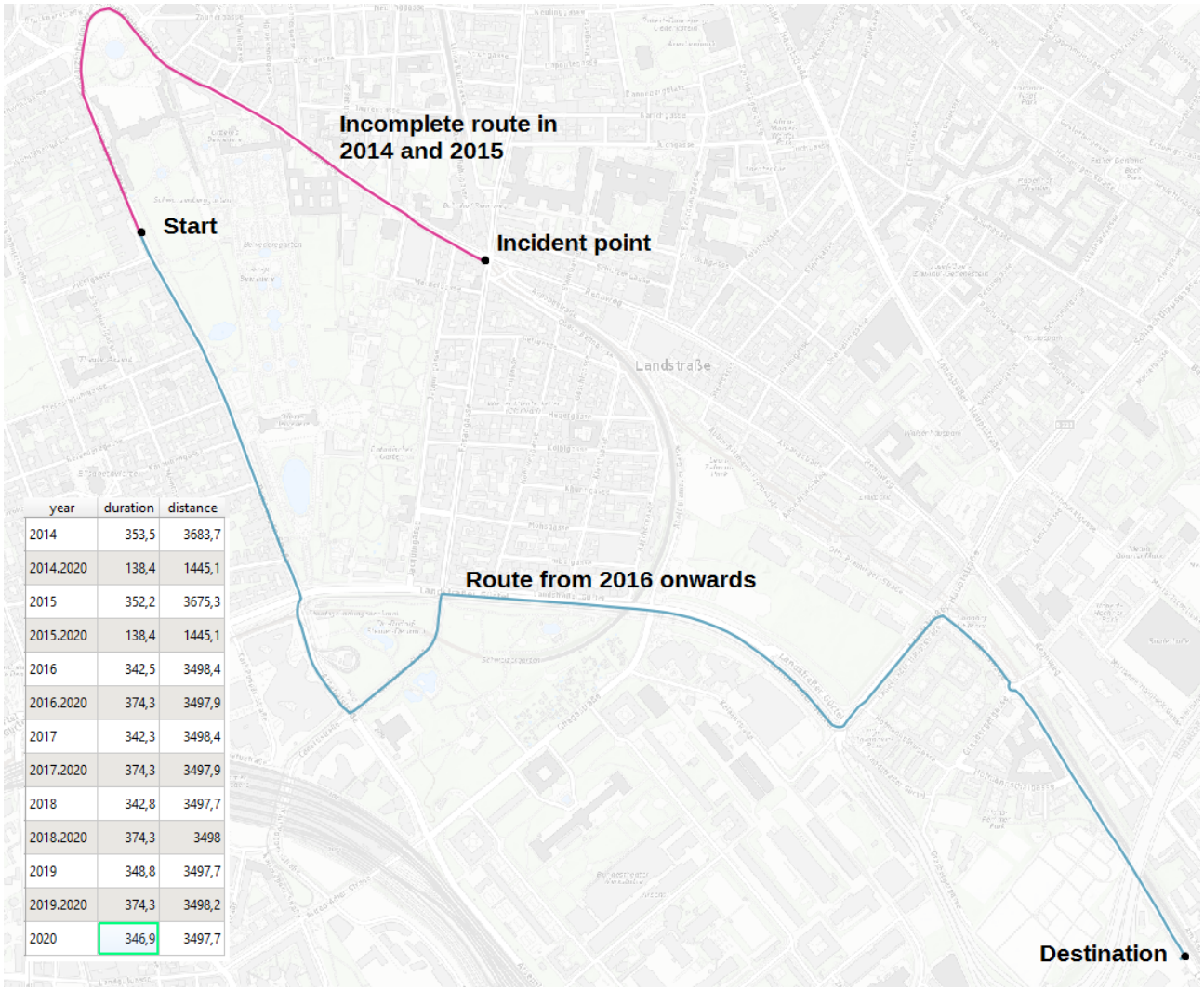
Zielstra, D. and Zipf, A.: A comparative study of proprietary geodata and volunteered geographic information for Germany, in: 13th AGILE International Conference on Geographic Information Science, http://agile2010.dsi.uminho.pt/pen/shortpapers_pdf/142_doc.pdf, 2010.

Appendix A: Full-size figures



	year	duration	distance
1	2014	1325,2	18140,1
2	2014.2020	1329,3	18138
3	2015	1327	18138,1
4	2015.2020	1329,3	18137,8
5	2016	1326,3	18132,8
6	2016.2020	1329,3	18137,3
7	2017	1326,4	18133
8	2017.2020	1329,3	18137,3
9	2018	1322,8	18132
10	2018.2020	1329,3	18137,8
11	2019	1326,7	18136,7
12	2019.2020	1329,3	18138,4
13	2020	1329,3	18137,6

Figure A1. Example of a visual route check in QGIS, with route ID 001 pictured



year	duration	distance
2014	353,5	3683,7
2014.2020	138,4	1445,1
2015	352,2	3675,3
2015.2020	138,4	1445,1
2016	342,5	3498,4
2016.2020	374,3	3497,9
2017	342,3	3498,4
2017.2020	374,3	3497,9
2018	342,8	3497,7
2018.2020	374,3	3498
2019	348,8	3497,7
2019.2020	374,3	3498,2
2020	346,9	3497,7

Figure A2. Example for a route that needed manual adjustment and was removed later on for inconsistency