Physica A 540 (2020) 122890

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Methods for calculating walking distances

Søren Wichmann [a,b,c,*], Harald Hammarström [d]

[a] Leiden University Centre for Linguistics, Leiden University, Leiden, Netherlands
[b] Laboratory for Quantitative Linguistics, Kazan Federal University, Kazan, Russia
[c] Beijing Language University, Beijing, China
[d] Department of Linguistics and Philology, Uppsala University, Sweden

ARTICLE INFO

Article history:
Received 1 March 2019
Received in revised form 31 July 2019
Available online 15 October 2019

ABSTRACT

In many scientific disciplines it is often necessary to refer to geographical travel distances. While online services can provide such distances, they fail for larger distances or for distances between points not connected by roads, and they do not allow for the calculation of many distances. Here we describe two novel methods of measuring travel distances which overcome these problems. Both use waypoints of populated places from the geonames.org database. The more efficient and accurate of the two uses the Dijkstra algorithm to find the shortest path through a Delaunay graph of neighbouring populated places.

1. Introduction

Across many disciplines it is often useful to measure a travel distances between two geographical points. For instance, correlational studies in disciplines such as economics [1], linguistics [2,3], genetics [4,5] or archaeology [6,7] often need to take into account geographical distances. They typically do so using the great circle distance (GCD), perhaps combined with a few waypoints constraining the route at geographical bottlenecks. The GCD, however, is not a very accurate measure of distance when it comes to human interaction, and it is expected that a correlation involving human interaction will decrease if the distances used are inaccurate. Tellingly, a linguistic dialectological study [8] obtained an $r = 0.54$ correlation between linguistic distances and geographical distances when using (the logarithm of) travel time estimates from the late 19th century and an $r = 0.41$ correlation when using the GCD.

In general, a correlation between X and Y, where Y in this case is a true geographical distance, changes within an upper and a lower bound when Y is replaced by Z, in this case a less accurate distance, according to (1), cf. [9]. In (1), R symbolizes the correlation coefficient (Pearson's r) for two sets of variables.

$$\begin{aligned} lower &= R(X,Y) \times R(Y,Z) - \sqrt{\left(1 - R(X,Y)^2\right) \times \left(1 - R(Y,Z)^2\right)} \\ upper &= R(X,Y) \times R(Y,Z) + \sqrt{\left(1 - R(X,Y)^2\right) \times \left(1 - R(Y,Z)^2\right)} \end{aligned} \tag{1}$$

For instance, if X and Y have an $r = 0.500$ correlation and Z has an $r = 0.990$ correlation with Y, the correlation between X and Z will lie within the bounds $0.373 \leq r \leq 0.617$. Thus, even if a distance Z has a very good correlation with the true distance Y, using the former instead of the latter can upset the true correlation with geography considerably. Precision matters: if X and Y again correlate as $r = 0.500$ and Y and Z now as $r = 0.999$, we get $0.461 < r < 0.538$ for the correlation

* Corresponding author at: Leiden University Centre for Linguistics, Leiden University, Leiden, Netherlands.
E-mail address: wichmannsoeren@gmail.com (S. Wichmann).

https://doi.org/10.1016/j.physa.2019.122890
0378-4371/

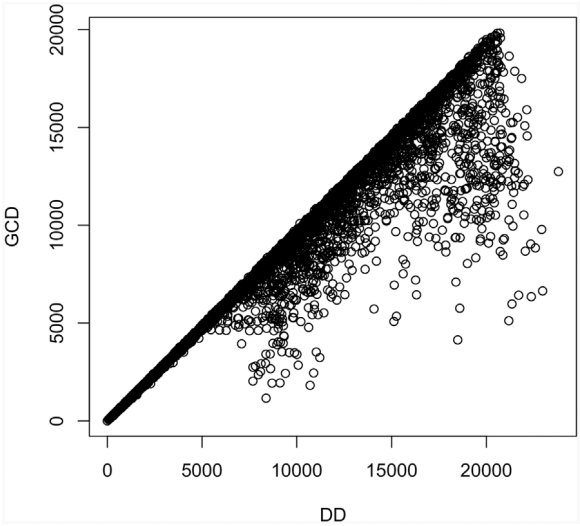

Fig. 1. Scatterplot of distances between 10,000 randomly selected pairs of locations on Earth as computed by a method sensitive to geography (DD-1) and the great circle distance (GCD).

between Y and Z. If the precision is lowered such that the correlation between Y and Z now is $r = 0.998$ we get $0.444 < r < 0.554$ for the correlation between X and Z. Thus, the precision at the third decimal of a substitute variable can affect the precision of the estimate of the correlation with the original variable at the second decimal.

Two alternatives to the GCD are tested here, both of which draw upon large numbers of waypoints from the geonames.org database as a way of taking into account geography. Briefly, the 'inexperienced traveller' (IT) algorithm chooses, at each step, the nearest populated place that diminishes the GCD to the goal. The Delaunay–Dijkstra (DD) method draws upon a Delaunay graph [10] of nearest neighbours built from geonames.org data and computes the shortest path through the graph using Dijkstra's [11] algorithm. This method applies to cases where the traveller, while being constrained to move between habitable parts of the landscape, knows which path to take. Both methods are described in more detail in the Materials and Methods section below.

It is an empirical question to what extent the GCD differs from a more naturalistic travel distance, but it is at least clear that routes spanning large, irregular land masses are expected to sometimes deviate drastically because the GCD ignores bodies of water and other natural impediments. Fig. 1 is a scatterplot of distances between 10,000 randomly selected pairs of locations on Earth as computed by a measure sensitive to geography – more precisely the DD-1 measure to be described below – and the GCD. The really dramatic differences start cropping up when the GCD is greater than around 2000 km and the travel distance measure is greater than around 5000 km (the differences are always such that the GCD is shortest, which is because the GCD is the shortest distance between two points on the Earth's surface, considered a sphere, whereas the travel distances plotted are sums of a set of GCDs joining a series of points). This plot merely serves as a preview of what to expect from a more detailed study of differences between the GCD and a more realistic travel distance. Possibly the GCD will suffice for smaller distances, but it certainly would be problematical to use it for larger ones if the goal of a study is to look at human movements and interactions. It should be clear, then, that there is a strong motivation for the development of an alternative to the GCD.

In 'Materials and methods' we describe the methods in detail, and in 'Results and discussion' we evaluate them using the walking distances of Google Maps (https://maps.google.com/), henceforth GM, as yardsticks.

2. Materials and methods

2.1. Distance measures for comparison

The Great Circle Distance (GCD) is computed using the standard formula in [12]. This is used for comparison with our own travel distance measures. Additionally, we use as a yardstick the shortest walking distance obtained by Google Maps (GM) at https://maps.google.com/ (accessed 2017-02-22). A survey of services providing walking distances, including Bing Maps (https://www.bing.com/maps/) and Yandex Maps (https://yandex.com/maps/) showed GM to have the best coverage. Nevertheless GM has its limitations. When there are no roads connecting two locations, such as any two locations separated by the Darién Gap straddling the Panama-Columbia border, GM fails to provide a walking distance. It is also limited in access to information on sailing routes, so travelling distances involving most of the world's islands are excluded. This imposes limits on the utility of GM and also reduces the number of cases for which it can be used

as a yardstick. More seriously, perhaps, the algorithm used by GM is not publicly accessible. Trusting GM to be the best available yardstick, then, requires a leap of faith, so if one method is seen to perform better in relation to GM than another, this alone cannot form the basis of which to choose—the choice should also be motivated by an inherent reason to expect why the apparently best performing method is, in fact, better. In the present case we will see that the version of the DD method whose results correlate best with GM is also the one having the greatest resolution in terms of waypoints, so here the results of the comparison lines up with expectations, which gives stronger support to a choice of the method than blind faith in GM.

2.2. Waypoints and elevation data

The novel idea behind both of the methods to be described is to take into account geography using populated places as waypoints. If a place is populated today (or has been in recent past) this place should, in the vast majority of cases, at least be traversable throughout much of the Holocene. In cases where the landscape is known to have changed, adjustments can easily be made by modifying the database of populated places. The implementation of a distance measure using such waypoint alone, as opposed to using voluminous data on topography and road networks, is straightforward and the computation of distances is quick.

It deserves explicit mentioning that constraining a route to waypoints on land does not exclude a scenario where a body of water is traversed. An algorithm that takes a traveller to a waypoint defined as being nearest will choose an island if an island happens to be nearest by the chosen definition, and will then take the traveller across water to this island.

Conveniently, the database hosted at http://www.geonames.org/ offers open access to ∼4.4 million populated places. The GeoNames database includes in its definition of a populated place anything from an area with "a small group of dwellings or other buildings" to a country capital and even places that were only populated in historical times, according to http://www.geonames.org/export/codes.html. The code for populated places is 'P'. The download page is http://download.geonames.org/export/dump/.

We also carried out experiments using elevations. Since GeoNames only offers elevation data for selected locations (mainly from the USA), elevation data were obtained from Version 1 of the Shuttle Radar Topography Mission (SRTM) data, which have a 3 arc-second resolution (https://dds.cr.usgs.gov/srtm/version1/) (the 1 arc-second resolution of Version 2 is unnecessarily high for our purposes). GeoNames coordinates were matched to those of the SRTM for the purpose of assigning elevations to populated places.

2.3. The inexperienced traveller (IT) algorithm

Using a large set of locations (waypoints) and GCDs for individual steps, the IT algorithm first finds the location closest to A, the origin, and checks whether going there would diminish the GCD to B, the goal. If so, it goes there; if not, it checks whether the next-closest location works and continues like that until the closest location which reduces the distance to B has been found. This procedure is repeated until B has been reached. This will produce a walking route the length of which can be approximated through summing up GCDs between stations—the more waypoints the better the approximation.

The method offered here is perhaps particularly apt for studies of human prehistory. The routes produced can be conceived of as similar to that of a prehistorical migrant who knows in which direction to go by gazing at the stars but not how to get to the destination in the quickest possible way.

2.4. The Delaunay–Dijkstra (DD) algorithm

A Delaunay triangulation for a set P of vertices in a plane (populated places in our case) is a triangulation DT(P) such that no vertex in P is inside the circumcircle of any triangle in DT(P). Since Delaunay triangulation maximizes the minimum angle of all the triangles in the triangulation, it tends to avoid sharp-angled triangles [10]. Fig. 2 provides some help to develop an intuitive understanding of the sharp-angle avoidance property. The two figures show the two different ways of triangulating 4 vertices. The one to the left satisfies the requirement that no vertex is inside any of the two circumcircles whereas the one to the right, which exhibits sharper angles, violates this requirement (twice). We follow [13] in finding a shortest path through a set of vertices by first connecting them in a Delaunay graph and then identifying the shortest path from an origin (A) to a destination (B) using Dijkstra's algorithm [11]. Dijkstra's algorithm can be roughly described as follows. Initially it assigns a distance of infinity from A to each other vertex. Starting at A it then looks at each neighbour and notes the distances, after which A is assigned to 'done' set, members of which are ignored in the next steps. Subsequently the neighbours are visited in ascending order of distance to A. For each such neighbour the cumulative distance to *its* neighbours is noted. This operation is iterated. Distances are only updated when they are smaller than a distance to the given vertex that was previously recorded. After each operation vertices are ranked according to their total distance to A and once all neighbours of a given vertex have been visited it is assigned to the 'done' set. Once B is reached, the algorithm finishes. Many textbooks and videos explain this algorithm better than the previous sentences, but at least they should give some idea of how it works.

As said, in our case P is constituted by populated places from GeoNames. When A and B pertain to a limited area, such as a single country, all members of P can be used. We label this method DD-all. In our implementation of this method a

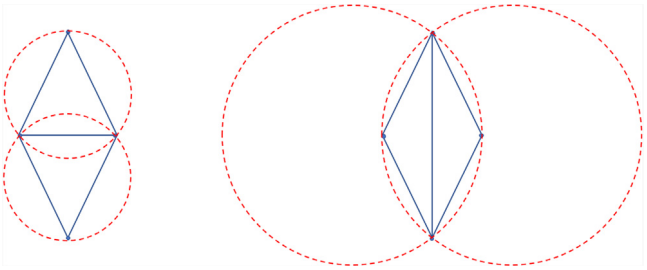

Fig. 2. Two different ways of triangulating 4 vertices. Left figure: Delaunay triangulation; right figure: its alternative.

Delaunay graph is built from all members of P also including A and B, if they are not already members of P. Building a Delaunay graph for each distance measurement is feasible when the size of P is small.

When A and B are unrestricted, a Delaunay graph for the whole world must be built. Since it is not computationally feasible to use all ∼4.4 million populated places in the database we make selections before building the Delaunay graph. In one selection we take one random populated place per cell in a 1 degree world grid, i.e., in a grid where cells are 1 degree latitude high and 1 degree of longitude wide. The method resulting from that is labelled DD-1. In another selection we take one random populated place per cell in a 0.25 degree world grid, labelling this higher-resolution method DD-0.25. Since each side of a 1 degree cell is 111.12 km, DD-1 has a very coarse resolution. The side of a 0.25 degree cell is 27.78 km, so this a more adequate resolution for most purposes although not for very short distances.

For both DD-1 and DD-0.25 it is computationally intensive to build a Delaunay graph. The DD-1 graph contains 14,226 waypoints and the DD-0.25 graph 133,769. Thus, such a graph must be built once and for all before distances are computed. This means that when A and/or B are not themselves members of P it is necessary to add to the total distance through the graph the distance from A to the appropriate entry point, p, in the graph and/or the distance from the appropriate exit point, q, to B. Because of the grid organization, A can have eight neighbours vying for being the entry point, p, that minimizes the distance from A through p to B. To identify p, the eight nearest neighbours of A in P are found and subsequently p is selected among these candidates. By a similar procedure q is identified.

All the computational work was carried out in R. The deldir function of the deldir library [14] was used for the Delaunay graphs and the shortest.paths function of the igraph library [15] was used to carry out Dijkstra's algorithm.

3. Elevations

We conducted an experiment where the GCD distances between waypoints in the DD-0.25 model were modified to take into account the elevation of stations along the route using the formula in (2), where D' is the modified distance (in km), $GCD(a, b)$ is the Great Circle Distance between the two stations a and b, E is the mean of the elevations of a and b (in km), and C is a constant varied from 0 to 1.

$$D' = GCD\,(a, b) \times (E \times C + 1) \tag{2}$$

The constant C in Eq. (2) allows to adjust the effect of elevations on distances from a zero effect (when $C = 0$) to larger effects, meant to take into account the extra distances involved in travelling up- and downhill.

4. Results and discussion

A test set of GM distances was produced getting minimal walking distances, when available, between (1) the largest and the next-largest city in each of the world's 252 countries carrying distinct ISO 3166-1 alpha-2 codes, (2) the 10th and 11th largest city in each country with enough cities, and (3) the largest cities in each of two randomly picked countries. Sizes were judged by the population figures in GeoNames. This selection resulted in 384 distances, 236 of which are within-country distances and 148 between-country distances. IT and DD-all, both of which use all populated places in GeoNames as waypoints, were only tested for the 236 within-country distances, whereas DD-1 and DD-0.25 were tested for all distances.

The experiment taking into account elevations using the formula in (2) and varying the correction factor C in steps of 0.01 from 0 to 1 proved unsuccessful: the best correlation with GM was found for $C = 0$, i.e., the value corresponding to not taking elevation into account at all. We assume that this negative result is obtained because the selection of waypoints already indirectly take elevation into account by excluding uninhabited cells in the grid—in this case cells that are uninhabited because of inaccessibility. Moreover, the GM distances may also not take elevation into account directly. In order to gauge the usefulness of adding some cost function for elevations we would need a different kind of dataset for evaluation purposes. These issues are deferred to future studies.

Table 1
Correlations between six different methods of computing geographical distances.

	GM	GCD	IT	DD-all	DD-1
GCD	0.98552				
IT	0.98495	0.99787			
DD-all	0.98557	0.99998	0.99785		
DD-1	0.97959	0.99321	0.99156	0.99323	
DD-0.25	0.98516	0.99964	0.99760	0.99965	0.99365

Values of r are rounded off to 5 digits; results are based on 236 within-country distances.

Table 2
Correlations between four different methods of computing geographical distances.

	GM	GCD	DD-1
GCD	0.98548		
DD-1	0.98598	0.99979	
DD-0.25	0.98648	0.99989	0.99986

Values of r are rounded off to 5 digits; results are based on all 384 distances.

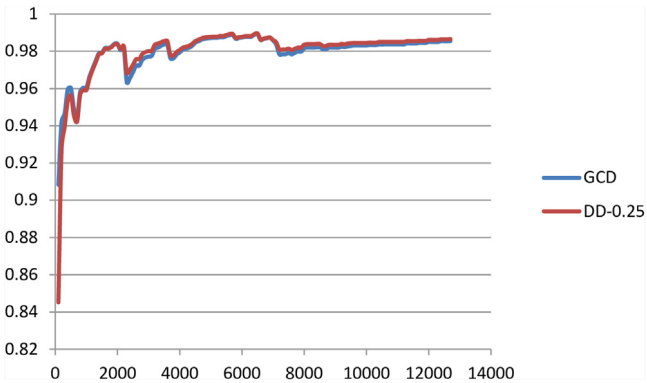

Fig. 3. A comparison of correlations between Google Map (GM) distances and great circle distances (GCD) with correlations between GM and DD-0.25 (Delaunay–Dijkstra using one random populated place per cell in a 0.25 degree world grid) in GCD ranges increased successively by 100 km.

The table of locations, coordinates, and distances are provided in S1. Pearson's r for linear correlations among the within-country distances are shown in Table 1 and those for all distances (both within-country and between-country) are in Table 2. DD-all and IT are excluded from the latter.

In terms of correlations with GM in the first column of Table 1, the best performing method is DD-all, but it is only slightly better than the GCD. DD-0.25 comes in third. DD-all, however is restricted because it is not computationally viable to build a Delaunay graph of all waypoints in GeoNames for the whole world. Turning to Table 2, which includes correlations for large, between-country distances, we see that DD-0.25 performs best. Now the GCD has the poorest performance.

As shown in Fig. 1, the GCD generally seems to perform acceptably at relatively small distances. Fig. 3 compares the correlation between GM and GCD with that of GM and DD-0.25 in GCD ranges increased successively by 100 km. Although somewhat difficult to make out from the curves, at each step in the increase of the range from 100 km to 2000 km the GCD has a better correlation with GM, and at each successive step DD-0.25 has the better correlation. This indicates that for a correlational study limited to distances up to 2000 km the GCD will do, unless, perhaps, some very good special alternative is available, such as the 19th century travel times used in [8]. DD-0.25, however, works better for larger distances.

The computing time of DD-0.25 in its current non-optimized implementation is a few seconds per distance on a professional laptop. While finding the shortest path through the Delaunay graph is fast, finding appropriate entry and exit points slows down the method. R programs and data necessary for calculating IT, DD-all, DD-1, and DD-0.25 are provided at https://github.com/Sokiwi/Distances.

5. Conclusions

This paper has discussed different geographical distance measures, looking for an alternative to the Great Circle Distance for the purpose of correlational and other studies necessitating the computation of a large number of distances. The best alternative, evaluated through the fit with Google Maps minimal walking distances and efficiency, is the shortest path, using Dijkstra's algorithm, through a Delaunay graph connecting neighbours of populated places in a 0.25 degree

world grid. We would normally recommend this alternative, especially for a study involving Great Circle Distances exceeding 2000 km. The other major alternative, that of the 'inexperienced traveller' algorithm, may also be useful, especially when modelling a traveller who is entering completely unknown regions.

Our own immediate motivation for this study is to provide a measure of travel distance that may be used when combining previous research on determining origins [16] and dates [17] of language group in order to infer language migration rates. But we imagine many other applications, for instance in fields such as economics, genetics, and archeology. We also imagine possible improvements, for instance with respect to the problem of how to take into account elevations when measuring travelling distances.

Acknowledgements

We would like to thank Eric W. Holman for comments. S.W.'s research was carried out under the auspices of the project "The Dictionary/Grammar Reading Machine: Computational Tools for Accessing the World's Linguistic Heritage" (NWO Proj. No. 335-54-102) within the European JPI Cultural Heritage and Global Change programme, under the call Digital Heritage. It was additionally funded by a subsidy of the Russian Government to support the Programme of Competitive Development of Kazan Federal University, Russia and a grant (KYR17018) from Beijing Language Innovation Center, China in support of a subproject directed by Qibin Ran.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.physa.2019.122890. It consists of the S1 File with distance data. This is an Excel sheet with distances between 384 locations as measured using the methods discussed in this paper. Place names are in ascii. All distances are rounded off to integers.

R implementations. See https://github.com/Sokiwi/Distances.

References

[1] S. Eckel, G. Löffler, A. Maurer, V. Schmidt, Measuring the effects of geographical distance on stock market correlations, J. Empir. Financ. 18 (2011) 237–247.
[2] J. Nerbonne, How much does geography influence language variation? in: P. Auer, M. Hilpert, A. Stukenbrock, B. Szmrecsanyi (Eds.), Space in Language and Linguistics. Geographical, Interactional, and Cognitive Perspectives, De Gruyter, Berlin, 2013, pp. 220–236.
[3] Q.D. Atkinson, Phonemic diversity supports a serial founder effect model of language expansion from Africa, Science 332 (2011) 346–349.
[4] O. Lao, T.T. Lu, M. Nothnagel, O. Junge, S. Freitag-Wolf, A. Caliebe, et al., Correlation between genetic and geographic structure in Europe, Curr. Biol. 18 (2008) 1241–1248, http://dx.doi.org/10.1016/j.cub.2008.07.049.
[5] S. Ramachandran, O. Deshpande, C.C. Roseman, N.A. Rosenberg, M. Feldman, L.L. Cavalli-Sforza, Support from the relationship of genetic and geographic distance in human populations for a serial founder effect originating in Africa, Proc. Natl. Acad. Sci. USA 102 (44) (2005) 15942–15947.
[6] R. Pinhasi, J. Fort, A.J. Ammerman, Tracing the origin and spread of agriculture in Europe, PLoS. Biol. 3 (12) (2005) 2220–2228.
[7] J. Fort, M.M. Pareta, L. Sørensen, Estimating the relative importance of demic and cultural diffusion in the spread of the Neolithic in Scandinavia, J. R. Soc. Interface 15 (2018) 201805, http://dx.doi.org/10.1098/rsif.2018.0597.
[8] C. Gooskens, Norwegian dialect distances geographically explained, in: B.-L. Gunnarson, L. Bergström, G Eklund, S. Fridella, L.H. Hansen, A. others Karstadt (Eds.), Language variation in Europe: Papers from ICLaVE 2, Uppsala University, Uppsala, 2004, pp. 195–206.
[9] I. Olkin, Range restrictions for product-moment correlation matrices, Psychometrika 46 (1981) 469–472, http://dx.doi.org/10.1007/BF02293804.
[10] B.N. Delaunay, Delaunay BN Sur la sphère vide, B. Acad. Sci. USSR 6 (1934) 793–800.
[11] E.W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1959) 269–271.
[12] D. Zwillinger, CRC Standard Mathematical Tables and Formulae, Chapman & Hall/CRC, Boca Raton, 2003.
[13] G.E. Jan, W.C. Tsai, C.-C. Sun, B.-S. Lin, Delaunay triangulation-based shortest path algorithm with $O(n \log n)$ time in the Euclidean plane, in: Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2010, pp. 186–189, https://www.researchgate.net/publication/261272473_A_Delaunay_triangulation-based_shortest_path_algorithm_with_On_log_n_time_in_the_Euclidean_plane, 11–14 July 2012, Kaohsiung, Taiwan. (Accessed 2019-07-30).
[14] R. Turner, Deldir: delaunay triangulation and dirichlet (voronoi) tessellation, 2016, R package version 01-12. https://CRAN.R-project.org/package=deldir.
[15] G. Csardi, T. Nepusz, The igraph software package for complex network research, Inter J. Complex Syst. (2006) 1695, http://igraph.org.
[16] S. Wichmann, A. Müller, V. Velupillai, Homelands of the world's language families: A quantitative approach, Diachronica 27 (2010) 247–276.
[17] E.W. Holman, C.H. Brown, S. Wichmann, A. Müller, V. Velupillai, H. Hammarström, S. Sauppe, H. Jung, D. Bakker, P. Brown, O. Belyaev, M. Urban, R. Mailhammer, J.-M. List, D. Egorov, Automated dating of the world's language families based on lexical similarity, Curr. Anthr. 52 (2011) 841–875.