CityChrone: an Interactive Platform for Transport Network Analysis and Planning in Urban Systems

CityChrone: an interactive platform for transport network analysis and planning in urban systems

Indaco Biazzo[1]

Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy,
indaco.biazzo@polito.it,
WWW home page: http://indacobiazzo.me

Abstract. Urban systems studies in the last decades have greatly benefited from the digital revolution and the accumulation of a massive amount of data. Extracting useful information from these data calls for new and innovative theoretical and computational approaches. This work presents an open-source, modular, and scalable platform for urban planning and transports network analysis, the CityChrone [citychrone.org]. The platform shows, on interactive maps, measures of performances of public transport in cities. The measures are based on the computation of the travel time distance between a large set of points. Thanks to the high efficiency of the routing algorithm developed, the platform allows users to create new public transports networks and showing the effect on mobility in a small amount of time. A preliminary analysis of the user-generated scenarios is presented. All the source code of the CityChrone platform is open-source, and we employ only open data to ensure the reproducibility of results.

Keywords: public transport networks, accessibility measures, urban systems, urban planning, temporal networks

1 Introduction

In 2007, for the first time in human history, the population living in urban areas exceeded the global rural population[26]. Moreover, the urbanization process goes on, and it is expected that in 2050 two-third of the world population will live in cities[26]. People in cities are connected thanks to a large number of processes and different interactive networks. The city is a highly non-linear and out of equilibrium process[19], and problems that institutions, stakeholders, and private citizens will have to face will be more and more related to this high level of complexity. Developing sustainable and efficient citizen mobility and commuting systems is one of the most widespread challenge. Nowadays, thanks to the ICT (Information and Communication Technologies)[40], and the subsequent massive quantities of data accumulated gathered the attention of the scientific community and fostered the emergence of many quantitative studies aiming at identifying statistical patterns behind the dynamics of humans mobility within or between cities [37, 31, 50, 15], as well as on their infrastructures and services [38, 27, 29, 47, 32, 48, 49]. Data about cities and their inhabitants' habits are nowadays collected and available for research and commercial purposes. The information extracted from the statistical analysis of the properties coming from mobility-related data can significantly impact everyday life, helping citizens perform better choices in terms of more environment-friendly mobility solutions, more efficient movements in general, and optimal choice of the place to live. Generally speaking, more precise and easy-to-understand information about the criticality or efficiency of transport services in urban environments is essential at each level of modern society (private citizens and companies, public administrations, and research institutes). This work presents the CityChrone project [www.citychrone.org], an interactive public transport network analysis and planning platform. The analysis is based on accessibility measures. The scientific community defined the concept of *accessibility* several decades ago[33, 21, 22], in order to give a precise quantification of the performance of transportation systems per se and with other aspects of people's lives. Despite its importance, there is not a unique possible definition of accessibility: this could depend on the availability of data or the aim of the researchers performing the analysis [46, 21, 41, 30]. On the other hand, such dispersion increases the difficulty for a straightforward interpretation of accessibility metrics, preventing their operational use by policymakers. Recently, new metrics have been proposed [20] in a general framework aiming to provide a unified point of view in which the *temporal dimension* is at the core. These new metrics are robust and general enough to be applied to different urban systems and different means of transportation. Thanks to a crucial modification to an existing very efficient routing algorithm for public transports, the CityChrone platform can compute those quantities in a fast way for public transports. Users can explore different scenarios of public transports networks in nearly real-time (the time needed to re-compute all the accessibility quantities is less than two minutes for medium-sized cities). The CityChrone platform is an open-source project, and it is published on GitHub[1]. The aim is to involve institutions, companies, and private citizens interested in developing an interactive platform for city knowledge, awareness, and planning. The CityChrone project aims to facilitate the way scientific results are presented, enhanced usability and comprehension through an interactive platform on the web. Moreover, particular care has been devoted to using open data or data freely downloadable to ensure the reproducibility of the results. The work is organized in the following way: In the first section, we describe related works, and in the second section, the routing algorithm. Then in the third section, we present the data and preprocessing procedures. In the fourth section the citychrone platform is presented. In the last section, before the conclusion, a preliminary analysis of the public transport network created by users is shown.

2 Related works

The literature about accessibility measures is vast and started several decades ago, as stated in the introduction. Despite this vast production of theoretical tools, the computation of these quantities in a real case scenario and in more than one particular case are very few. Moreover, even less interactive platforms have been created to visualize these quantities. One of the first web-based platforms was mapnificien [2]. In this platform, the user can choose a city and visualize isochrones, computed considering displacements with public transports, selecting a starting point on the map. Mapnificien does not show accessibility maps. The isochrones are computed only between stops, and the walking area is only roughly estimated. A platform for computing and visualize accessibility measures is described in [42]. The authors define an accessibility measure and an interactive platform to show several informative layers. The proposed measure is not based on actual travel time but instead on a public transport network's "centrality" measure. Each city stop has a score based on several factors, such as the velocity of transportation, the distance from train stations, and the capacity. This measure has more than five free parameters with no constraints, reducing the measure's universality and transparency. The platform should be accessible for one case study, the Baltimore-Washington DC region, but the URL of the demo is not reachable [3]. No source code is available. Another example of accessibility measure is described in [22]. The authors introduced an accessibility measure that gives four possible values to the public transport stops, from low to very high, based on frequency and means of transportation of public transports routes passing through the considered stop. No source-code or web-based platform was released. Closer to our approach is the "The Metropolitan Chicago Accessibility Explorer" [51]. This platform shows accessibility measures based on isochrones. In the platform[4] the user can choose several different layers showing, for instance, the number of jobs or other services reachable within a given time for each census block of the city. The travel time is computed thanks to the OpenTripPlanner library [cite], an open-source routing library. The travel times are precomputed, and the platform shows the results. The source code was not released. The CityChrone platform, described in this work, has several key differences from all the above platforms. CityChrone uses only open data and standard processes that can be easily applied to every urban system where public transports data are available. Moreover, as far the author knows, it is the only available platform, that thanks to the efficiency of the routing algorithm used, users can build new scenarios of public transports and visualize the effect on accessibility measures after a small amount of time (less than 2 minutes for medium-size cities like Rome and Boston).

3 Accessibility quantities

The accessibility quantities considered are presented in [20].Here we give just a quick review. We want to measure the performance of public transports to

explore the space and connect people in a city. The starting idea is that the isochrones, i.e., the surface at equal time distance t from a starting point p_0 at time t_0, see Fig.1, could be used to measure the performance of public transports in cities. More extensive is the area of isochrones larger is the portion of the city that is possible to explore given a time t from the starting point. Based on that, we define two measures: the *velocity score* and the *sociality score*.
The *velocity score*, given a starting point p_0 at time t_0, measure the velocity of exploring the space around the point considered. This measure can be interpret as the average velocity taken a random directions of displacement given a typical travel time. The precise definition is the following: consider the *covered area*, $A(t,(p_0,t_0))$ of a isochrone at time t starting from p_0 at time t_0. We define an effective ray $\bar{r}(t,(p_0,t_0))$ as: $\bar{r}(t,(p_0,t_0)) = \sqrt{A(t,(p_0,t_0))/\pi}$. Dividing it by the time t we obtain an effective average velocity: $\bar{v}(t,(P_0,T_0)) = \bar{r}(t,(p_0,t_0))/t$. The effective average velocity is defined for every point p_0 in the map and starting time t_0. The *velocity score* is obtained averaging over the journey time distribution probability $f(t)$:

$$v(p_0,t_0) = \int_0^\infty v(t,(p_0,t_0))f(t)dt, \tag{1}$$

The journey time distribution probability is the probability distribution of travel time on public transport in the city [35]. Observing the *velocity score* of the cities on CityChrone [citychrone.org] it is clear that the center of the city has a very high-velocity score compared to the suburbs. People living in the center of cities are well served by public transports, having all the directions of displacement allowed, usually also with good and fast public transports, like trains and subways. Instead, in the suburbs, only directions towards the center are well served by public transports, and all the others have poor or no public transports services. The *sociality score* measure instead the possibility of meet people starting from the point p_0 at time t_0. It measures the number of people reachable from p_0 at time t_0 in a typical daily working trip. The definition is similar to the 1, where instead of the effective velocity $v(t,(p_0,t_0))$, we take the average of the amount of people $pop(t,(p_0,t_0))$ living inside the isochrone at time t starting from the point p_0 at time t_0. This measure considers, at the same time, the public transports services and the density and distribution of population in the city. For the exact definitions, robustness as well as statistical analysis of these accessibility measures on a large set of cities see[20].

4 Routing Algorithm

Calculating the accessibility measures require to compute travel time distances by public transports between each point in a city. In general, if we consider a grid of points covering a medium-sized city, with reasonable steps, e.g., less than 500m, the number of points is of order $\propto 10^3$ and the number of journeys to compute is of the order of $10^6 - 10^7$ for each city. The availability of efficient routing algorithms is indeed mandatory. Moreover, we want a flexible algorithm that allows for fast computation of all travel time distances between points when the schedules change, meaning that data preprocessing time should be reduced to the minimum. For road networks, one can compute the driving directions in a millisecond or less at the continental scale, but it is not the case for public transport networks [18]. The approaches and speedup techniques used for road network routing algorithms fail [18], or they are not so effective on public transport networks. In the last years, different approaches, not based on the graph structure of the problem, have emerged in literature where the most promising ones are the RAPTOR algorithm [23] and the CSA Algorithm [24]. Between them, the CSA algorithm seems to be the fastest for the computation of the earliest arrival time [25]. Both algorithms are efficient and fast, with short preprocessing time. Both algorithms have some limitations when considering footpaths to change stops and means of transports, reducing the performance and applicability of those algorithms in urban contexts. The algorithm we devised, the ICSA, is based on the CSA algorithm, but we introduced a crucial modification that allows us to use it in urban systems considering realistic footpaths between stops and means of transport. In the Supplementary Material [5], we describe the CSA algorithm and then the ICSA.

5 Urban tessellation and data preprocessing

Cities have no unique and accepted way to define their area and border. Moreover, large urban systems are composed of several different public transport operators, some of which span their operation well beyond the city's limit, up to national scale. In order to limit the area of analysis, we adopt the definition of cities made by OECD/EU as 'functional economic units'[43]. It uses population density to identify urban cores (city core) and travel-to-work flows to identify the hinterlands whose labor market is highly integrated with the cores (commuting zone). We consider in our analysis only the stops inside both regions and only connections that connect them. In order to have uniform measures of the performance of the public transports over the area, we tassellate the city area by a hexagonal grid with distance center to center of nearby hexagons of 0.4km. Then we retain only hexagons that have at least one stop reachable in 15 minutes by walk. For each hexagon in the city, we compute the resident population inside. The data has been gathered through the Eurostat Population Grid [14] for the European cities and the Gridded Population of the world made by the Center for International Earth Science Information Network [34]. This spatial population dataset divides the population into squares with a surface of $1\,\text{km}^2$, while our tessellation uses hexagons of smaller surfaces ($\sim 0,1\text{km}^2$). Hence, we assigned the population in each square to the hexagons overlapping them, proportionally to the fraction of overlapping surfaces. The accessibility measures, based on isochrones, use public transport schedules, streets networks and spatial population distributions in cities. These data are nowadays easily downloadable from the web. Public transport companies usually release their schedules in a uniform way, using the GTFS format[6]. From this file it is possible to extract the locations of the stops and the connections. The variation of public transports connections does not change significantly in the working days [16], [36], so for each city, we choose a Wednesday in the period of validity of the GTFS file that is not a holiday. We downloaded GTFS files from a repository [7], or directly from the public transports company website. For each hexagon and stop present in a city, we compute the time walking distance between all the stops and hexagons reachable in 15 minutes by walk. The walking path are computed by an OSRM backend[39] with street graphs taken from OpenStreetMap[44]. The walking speed was set to 5km/h.

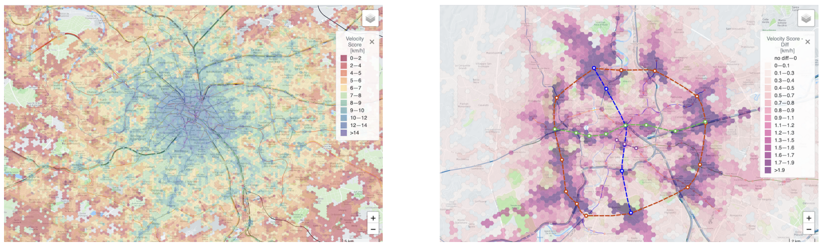
(a) Home page (b) Isochrone layer

Fig. 1: **a:** home page, where it is possible to select the city to explore accessibility quantity. **b:** Example of isochrone compute in the city of Paris

6 The CityChrone platform

CityChrone is an open-source web app, developed within the open-source meteor framework[8]. The CityChrone platform is able, given the city's tesselletion, the population distribution, the stops, and the connections, to compute isochrones and several accessibility quantities based on public transports and shows them on interactive maps for every city. Moreover, the platform allows users to modify the connections adding new metro lines. After recomputing all accessibility quantities, the user can check how the new scenario changes the accessibility measures.

The user experience The platform has three principal sections: the starting page, the visualization page and, the scenario page. On the starting page, fig.1a, the user selects the city. The user, clicking on a city, is redirected to the visualization page. In this section, the user can explore different layers that describe distinct aspects of public transport performances. In the left sidebar, the user can select the layers (velocity score, sociality score, population, isochrone and sociality and velocity score difference). The *isochrone layer* shows the isochrones in the city, fig.1b. If the user clicks on the map, an isochrone is shown starting from the clicked point. If present, on the left sidebar, it is possible to choose a new scenario with new metro lines added to the city and check the difference in the isochrones and the accessibility quantities respect the default scenario. In some cities, there is the button "new scenario" by which the user goes to the *scenario page*.



(a) Velocity score layer

(b) Scenario

Fig. 2: **a:** Velocity score layer of Paris. **b:** Scenario "rer+circle" create by the user "mat" for the city of Rome with the layer of the improvements in the velocity score.

In the *scenario page* users add new metro lines to the city. There is a limited amount of budget available to users to build new metro lines. The cost of the metro lines is computed given a fixed price for each station and a price per kilometer for the subway tube. The user can add new metro lines by clicking on the "add metro" button. The metro lines added can be dragged, deleted, or expanded (making, for instance, bifurcation). When the constructions of new subways are terminated, the user can click on the "compute" button to compute the new accessibility quantities. In the meanwhile, the user can insert his name and the title for the scenario made. Then, ended the computation, the *visualizator page* is loaded, with the scenario created selected. The page shows the "rank", highlighting the position of the scenario created. The rank of the scenarios is computed according to the average value of the *velocity score* per person in the city.

Backend Particular care has been paid to the optimization of some aspect of the platform, the visualization of large amount of data, the minimization of the transfer of data between server and clients, and the scalability of the platform to large audience. The server side of the CityChrone platform is able to, given the data about the public transport of a city, to compute the accessibility quantities and store them in a database. The principal components of the back-end are: the data, the routing functions and the accessibility quantities, and the functions related to the visualizations of the information on maps. Working with standard JavaScript objects and HTML code could reduce the performance and the number of objects visualized on the map. Usually, there are about 10^3 of hexagons in a city reaching for Paris the value of $6 * 10^4$. The hexagons are too many to be displayed in a standard browser, so we merge contiguous hexagons with the same color. Empirically we found that their number is reduced by a factor of ten. The most computational demanding operation is the calculation of accessibility quantities. When a new scenario is created, the computation is made client-side to not overload the server-side. All the routing functions are written to be used both by client and by server-side. On the client-side, the computation exploits the parallel computing that modern browsers allow through Web Workers [9]. The client side computation allows the CityChrone platforms to scale to many users without the need for ample server resources. The current version of City-Chrone runs on a virtual server with eight dedicated X86 64bit cores with 16GB of RAM. The client-side computation of the new scenario required: i) the computation of new connections added to the existing ones, given the list of new metro lines created by users. We assume a fast metro line (see for example [10]), with acceleration of 1.3 m/s^2, a maximum velocity of 30 m/s and frequency every 2 minutes. These parameters can easily changed to reflect local project constraints. ii) For each new stop, the walking times with stops and hexagons reachable in 15 minutes by walk is computed. A OSRM server [39, 11] with the street network taken from OpenStreetMap [44] of the city is active on the server. To save server-side resources, each new stop can take the walking paths equal to those of the nearest hexagon or stop (this could reduce the precision of the calculation, but it avoids having an OSRM server always up). We implement both solutions. iii) The computation of the new travel times between all points and the relative accessibility quantities. The running time for the complete computation of a new scenario is about 1-2 minutes for a medium-size city (Rome, Boston, etc.) on a computer with a 4-core CPU. For a larger city, the time can reach 30 minutes, and for these cities, we do not allow, so far, users to create a new scenario.

7 Preliminary analysis of new public transport network scenarios: the case of Rome

In this section, we analysed the scenarios created for the city of Rome. In two events, in 2017 and 2018, the platform was presented inside two public scientific events organized in Rome [12]. In the period 2017-2021, the web page of Rome has had about 3000 unique views, and users created more than 350 new public transport scenarios. The users have a budget of 5000M€ of euros to construct new metro lines. We assume that a single stops cost 100M€ (see for instance the average costs of a Paris metro station [13]) and the tube 30M€ per km (average cost of a tunnel boring machine tube [17]). These costs are assumed valid on average for a medium size city, but locally can vary depending on several different factor, ranging from architectural projects to soil composition. More specific cost functions can be easily integrated in the platform. The goal is to maximize the gains in velocity scores or sociality scores of the whole city. In order to analyze the user solutions so far created, we cluster them according to the impact on the accessibility measures considered, the sociality score. We associate to each scenario an array containing the sociality score computed for

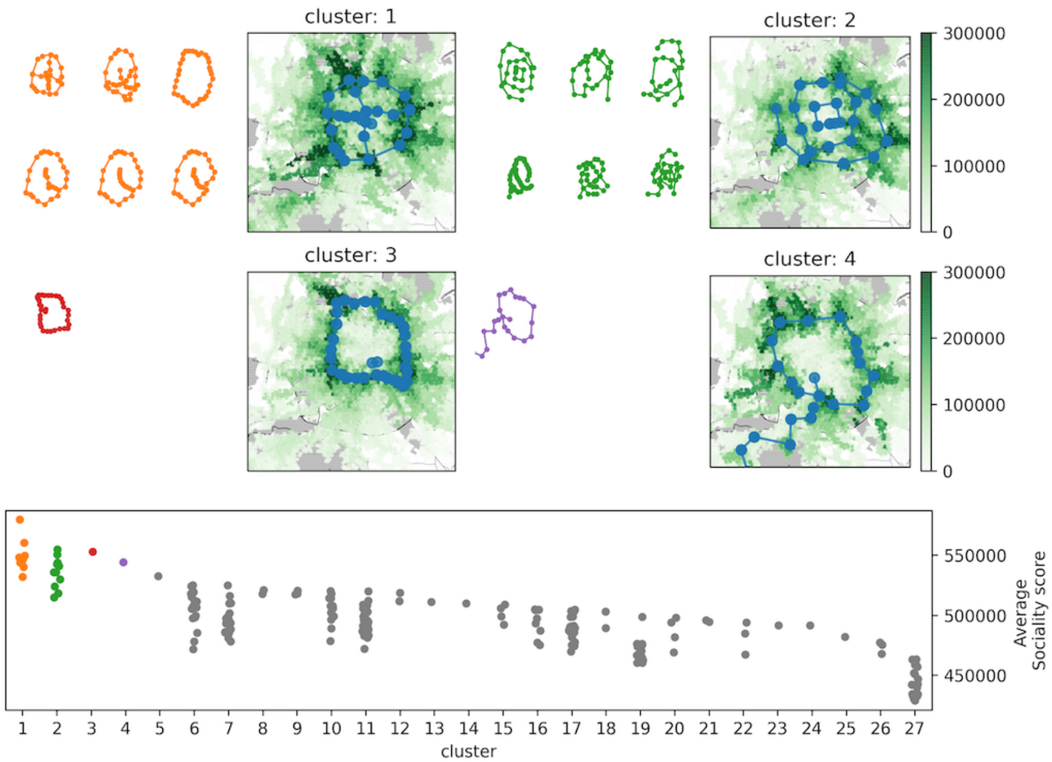


Fig. 3: The figure shows the scenarios of new subways generated by users in the case of Rome. The solutions are clustered according to the distribution on the city of the sociality score. The average value of the sociality score of the scenarios in each cluster is shown in the bottom plot. The clusters are ordered according to the best solution they contain. The four plots above show up to the six best scenarios for the four best clusters. For each cluster, the improvements in the sociality score of the best scenario are shown in the right plots.

each point of the hexagonal grid covering Rome, and we cluster the scenarios according to them. We use the affinity propagation algorithm [28] implemented in scikit-learn[45] because it does not need to specify the number of clusters beforehand. We rank each cluster of scenarios according to the best scenario it contains. Figure 3 shows the best four clusters, together with the six best metro lines proposed by users (if present). All the four clusters of solutions share a future, the need for a circular line in Rome. The best solution so far seems to merge a circular line with high-speed connections that irradiate from Rome's city center. Cluster 1 shows solutions with a circular line and, usually, some small metro lines enter the city's center. The scenarios belonging to cluster 2 have a more complex shape, with lots of metro stations serving the city center and partial circular lines. Cluster 3 has only one scenario, which is a circular line with very dense stops. The solution of Cluster 4 has a circular line with a fast connection to a west part of Rome that overlooks the sea. Closer inspections of the scenarios are possible on citychrone.org.

8 Conclusion

In this work, we presented CityChrone: an interactive platform for public transport analysis and planning. The platform shows several accessibility measures about the performance of public transports in more than 30 cities around the world. The accessible measures shown are based on the computations of a vast number of travel time distances between a grid of points in the city. The platform exploits a very efficient routing algorithm that allows a fast computation of the accessible quantities. Users can explore new public transports scenarios, build new metro lines, and check quickly (in minutes) how the proposed solutions change the accessibility measures. The primary computational requests are designed to run client-side, allowing the CityChrone platform to scale to a large number of users with small server-side resources. New cities can be easily added to the platform having just the schedules of the public transports. A script process the data, and then, after loaded in the CityChron platform, it is possible to visualize isochrones, accessibility quantities and test new transports scenarios. The CityChrone platform is the first step of an open-source project aiming to create a community of companies, public institutions, stakeholders, developers, and private citizens interested in developing interactive platforms to analyze the transports in cities and search for innovative solutions to mobility problems in cities.

9 Acknowledge

This work has been supported by the SmartData@PoliTO center on Big Data and Data Science.

References

1. https://github.com/CityChrone
2. https://www.mapnificent.net/
3. http://www.geosimulation.org/transit.html
4. http://urbanaccessibility.com/
5. https://bit.ly/3hkG613
6. https://developers.google.com/transit/gtfs
7. www.transitfeeds.com
8. https://www.meteor.com
9. https://html.spec.whatwg.org/multipage/workers.html
10. https://tinyurl.com/yycnsau6
11. https://github.com/Project-OSRM/osrm-backend

12. http://www.kreyon.net/kreyonDays/
13. https://marroninstitute.nyu.edu/events/how-much-does-it-cost-to-build-a-subway
14. Eurostat population grid, http://ec.europa.eu/eurostat/web/gisco/geodata/reference-data/population-distribution-demography/geostat
15. Alessandretti, L., Aslak, U., Lehmann, S.: The scales of human mobility. Nature 587, 402–407 (11 2020)
16. Alessandretti, L., Karsai, M., Gauvin, L.: User-based representation of time-resolved multimodal public transportation networks. Royal Society Open Science 3(7) (2016), http://rsos.royalsocietypublishing.org/content/3/7/160156
17. Attinà, M., Basilico, A., Botta, M., Brancatello, I., Gargani, F., Gori, V., Wilhelm, F., Menting, M., et al.: Annex 13 – case study on tunnels of "assessment of unit costs (standard prices) of rail projects (capital expenditure)" (2018), https://tinyurl.com/48chu6r8
18. Bast, H., Delling, D., Goldberg, A., Müller-Hannemann, M., Pajor, T., Sanders, P., Wagner, D., Werneck, R.F.: Route planning in transportation networks. In: Algorithm engineering, pp. 19–80. Springer (2016)
19. Batty, M.: The size, scale, and shape of cities. Science 319(5864), 769–771 (2008)
20. Biazzo, I., Monechi, B., Loreto, V.: General scores for accessibility and inequality measures in urban areas. Royal Society Open Science 6(8), 190979 (2019)
21. Black, J., Conroy, M.: Accessibility measures and the social evaluation of urban structure. Environment and Planning A 9(9), 1013–1031 (1977)
22. Bok, J., Kwon, Y.: Comparable measures of accessibility to public transport using the general transit feed specification. Sustainability 8(3), 224 (2016)
23. Delling, D., Pajor, T., Werneck, R.F.: Round-based public transit routing. Transportation Science 49(3), 591–604 (2014)
24. Dibbelt, J., Pajor, T., Strasser, B., Wagner, D.: Intriguingly simple and fast transit routing. In: International Symposium on Experimental Algorithms. pp. 43–54. Springer (2013)
25. Dibbelt, J., Pajor, T., Strasser, B., Wagner, D.: Connection scan algorithm. CoRR abs/1703.05997 (2017), http://arxiv.org/abs/1703.05997
26. of Economic, D., Social Affairs, P.D..: World urbanization prospects: The 2014 revision, highlights (st/esa/ser.a/352). Population Division, United Nations (2014)
27. Fleurquin, P., Ramasco, J.J., Eguíluz, V.M.: Characterization of delay propagation in the US air-transportation network. Transportation journal 53(3), 330–344 (2014)
28. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. science 315(5814), 972–976 (2007)
29. Gallotti, R., Bazzani, A., Rambaldi, S.: Understanding the variability of daily travel-time expenditures using GPS trajectory data. EPJ Data Science 4(1), 18 (2015)
30. Geurs, K.T., Van Wee, B.: Accessibility evaluation of land-use and transport strategies: review and research directions. Journal of Transport geography 12(2), 127–140 (2004)
31. Gonzalez, M.C., Hidalgo, C.A., Barabasi, A.L.: Understanding individual human mobility patterns. nature 453(7196), 779–782 (2008)
32. Guimera, R., Mossa, S., Turtschi, A., Amaral, L.N.: The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. Proceedings of the National Academy of Sciences 102(22), 7794–7799 (2005)
33. Hansen, W.G.: How accessibility shapes land use. Journal of the American Institute of planners 25(2), 73–76 (1959)

34. for International Earth Science Information Network CIESIN Columbia University, C.: Gridded Population of the World, Version 4 (GPWv4): Population Count (2016), http://dx.doi.org/10.7927/H4X63JVC
35. Kölbl, R., Helbing, D.: Energy laws in human travel behaviour. New Journal of Physics 5(1), 48 (2003)
36. Kujala, R., Weckström, C., Darst, R.K., Mladenović, M.N., Saramäki, J.: A collection of public transport network data sets for 25 cities. Scientific Data 5, 180089 (2018)
37. Kölbl, R., Helbing, D.: Energy laws in human travel behaviour. New Journal of Physics 5(1), 48 (2003)
38. Li, R., Dong, L., Zhang, J., Wang, X., Wang, W.X., Di, Z., Stanley, H.E.: Simple spatial scaling rules behind complex cities. Nature communications 8(1), 1–7 (2017)
39. Luxen, D., Vetter, C.: Real-time routing with openstreetmap data. In: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. pp. 513–516. GIS '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/2093973.2094062
40. Mayer-Schönberger, V., Cukier, K.: Big data: A revolution that will transform how we live, work, and think. Houghton Mifflin Harcourt (2013)
41. Miller, H.J.: Measuring space-time accessibility benefits within transportation networks: basic theory and computational procedures. Geographical analysis 31(1), 1–26 (1999)
42. Mishra, S., Welch, T.F., Torrens, P.M., Fu, C., Zhu, H., Knaap, E.: A tool for measuring and visualizing connectivity of transit stop, route and transfer center in a multimodal transportation network. Public Transport 7(1), 77–99 (mar 2014)
43. OECD: Redefining "Urban". OECD Publishing (2012), https://www.oecd-ilibrary.org/content/publication/9789264174108-en
44. OpenStreetMap contributors: Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org (2017)
45. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
46. Schmöcker, J.D., Quddus, M.A., Noland, R.B., Bell, M.G.: Mode choice of older and disabled people: a case study of shopping trips in London. Journal of Transport Geography 16(4), 257–267 (2008)
47. Sen, P., Dasgupta, S., Chatterjee, A., Sreeram, P., Mukherjee, G., Manna, S.: Small-world properties of the Indian railway network. Physical Review E 67(3), 036106 (2003)
48. Sun, X., Wandelt, S., Hansen, M.: Airport road access at planet scale using population grid and openstreetmap. Networks and Spatial Economics 20(1), 273–299 (2020)
49. Sun, X., Wandelt, S., Zhang, A.: Comparative accessibility of chinese airports and high-speed railway stations: A high-resolution, yet scalable framework based on open data. Journal of Air Transport Management 92, 102014 (2021)
50. Yan, X.Y., Zhao, C., Fan, Y., Di, Z., Wang, W.X.: Universal predictability of mobility patterns in cities. Journal of The Royal Society Interface 11(100), 20140834 (2014)
51. Yin, S., Li, M., Tilahun, N., Forbes, A., Johnson, A.: Understanding transportation accessibility of metropolitan chicago through interactive visualization. In: Proceedings of the 1st International ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics. pp. 77–84. ACM (2015)