An instant matching algorithm in the context of ride-hailing applications, using isochrones and social scoring

Malte Eilers[1] Simon Hempen[2] Sebastian Frers[3] Benjamin Stahr[4] Natalia Hölscher[5]
Paula Bürmann[6] Christian Janßen[7] Prof. Dr.-Ing. habil. Jorge Marx-Gómez[8]

Abstract: Ride-sharing nowadays is a term most people have heard about. It is an efficient way to reduce both climate detrimental emissions and the overall traffic within large cities. A key aspect of ride-sharing is to match drivers and passengers according to a set of attributes. By matching people based on static and dynamic settings such as social preferences, this paper shows a fast and efficient approach for solving this problem. As a key difference to other solutions the core of the developed instant matching algorithm is built up on isochrones. These are used to match trips on specific geographical constraints, like the maximum detour and time until a passenger's pickup. Furthermore, social metrics allow optimization based on individual preferences in the users trip selection. The foregoing steps result in a comparative score to allow an optimal set of trips as an output.

Keywords: ride-sharing; ride-hailing; ride-pooling; dial-a-ride-problem; instant matching algorithm; isochrones; person matching; social scoring; sustainability

1 Introduction

Through the rise of mobility, metropolitan areas face traffic congestion problems. Especially in highly populated cities, this can lead to increased air pollution [CLC17].Seen globally, traffic-related emissions increased by 68 percent from 1990 to 2015, accounting for 24 percent of the total amount of released greenhouse gases in the year 2015 [Sc18]. Though a multidisciplinary approach is needed to tackle this problem, one way to reduce the number of cars on the road and subsequently the overall pollution is ride-sharing [Ja17]. Since 2010, ride-sharing services have been expanding the possibilities of passenger transportation more and more [Ç17]. Established ride-sharing solutions like BlaBlaCar focus on arranging shared long-distance trips for individuals [CC]. Uber on the other hand is more like a cab company by matching individuals with licensed drivers [BP14]. This excludes the everyday short-distance travel by individuals within major cities from the potentially

[1] University of Oldenburg, 26129 Oldenburg, Germany malte.eilers1@uol.de
[2] University of Oldenburg, 26129 Oldenburg, Germany simon.hempen@uol.de
[3] University of Oldenburg, 26129 Oldenburg, Germany sebastian.frers@uol.de
[4] University of Oldenburg, 26129 Oldenburg, Germany benjamin.stahr@uol.de
[5] University of Oldenburg, 26129 Oldenburg, Germany natalia.hoelscher@uol.de
[6] University of Oldenburg, 26129 Oldenburg, Germany paula.buermann@uol.de
[7] University of Oldenburg, 26129 Oldenburg, Germany christian.janssen@uol.de
[8] University of Oldenburg, 26129 Oldenburg, Germany jorge.marx.gomez@uol.de

pollution-reducing benefits of ride-sharing. Challenges in the realization of a system for short-distance travel range from matching participant's preferences to providing a reliable system that optimizes matches in real-time [Ag10].At this point the algorithm comes into the foreground. The aim of this paper is to present an algorithm for matching participants dynamically in real-time. By determining the accessibility using isochrones, an efficient matching between driver and passenger in short-distance travel range can be achieved. In addition to spatial factors, the matching is also influenced by social preferences. To create an understanding of the topic, current matching algorithms for dynamic ride-sharing are presented in Chapter 2. Chapter 3 explains how the matching algorithm works. This is followed by an evaluation using a simulation to validate various scenarios in Chapter 4 and finally a conclusion in Chapter 5.

2 State of the art

The matching process in ride-sharing can be subordinated to the Dial-a-Ride Problem (DARP). A matching process for short-distance destinations is dynamic and thus increases the difficulty of solving the problem. The solutions for the dynamic ride-sharing problem are various. In literature many solutions for matching algorithms use heuristics [Wo04, Ja86, AGK20]. This helps to reduce complexity as DARP is considered NP-hard [CAF16]. One approach includes a time window the matching algorithm. Time windows contain the maximum time of a customer's trip and the time of pickup, in order to ensure the waiting time may only be within in a certain time window. The algorithm minimizes waiting time as far as possible [Ja86]. To match the users, Schreieck et al. developed an algorithm which checks whether trip offers are close to the departure and destination of trip requests. A trip request is matched with a trip offer if they are close to each other. This approach has low calculation time, but does not consider obstacles like parks and rivers into the matching [Sc16]. Some solutions to the Dial-a-Ride-Problem also include social aspects such as gender, employment status or sociality. Aydin et al. assign different weights to the social aspects, depending on the interests of the customer. When a passenger sends a request, the Needleman-Wunsch algorithm is used to determine which offered routes are similar to the passenger's route. For remaining drivers an average of the social aspects is calculated. The users are matched with the highest corresponding average of the social score. The algorithm matches participants with similar routes and social aspects [AGK20]. Other objectives when solving the ride-matching problem include minimizing total system travel cost [SX13], maximizing number of matches [SX13, NJ16] and maximizing distance savings [NJ16].

The algorithm described in Chapter 3 is carried out according to the methodology of Design Science Research with the model of Österle and Becker. This four-step model - Analysis, Design, Evaluation, Diffusion - was followed to create a better performing information system, in this case the matching algorithm for ride-hailing [Ö11]. In contrast to other research this paper extends the matching process by using isochrones around the pick-up and drop-off points to pay more attention to the environmental conditions and further

explores the application of using social metrics in the objective function by updating driver preferences on the decisions made.

3 Functionality of the matching algorithm

The main task of the matching algorithm is to select the best trips for a passenger from all available trips previously calculated within the system using a routing engine. Firstly, after the passenger requested a ride, the whole set of trips gets preselected. This is done by fixed filter parameters, which allow to sort out all the trips that violate these general constraints. To reduce the set of trips even more, isochrone matching is used next to match the geographical conditions. Once the pre-filtering of the trips is completed, secondly a score is calculated to compare the resulting trips. In addition to the detour for the driver and the waiting time for the passenger, the score also includes social preferences. The matching algorithm follows a holistic concept including several steps which are pictured in figure 1.

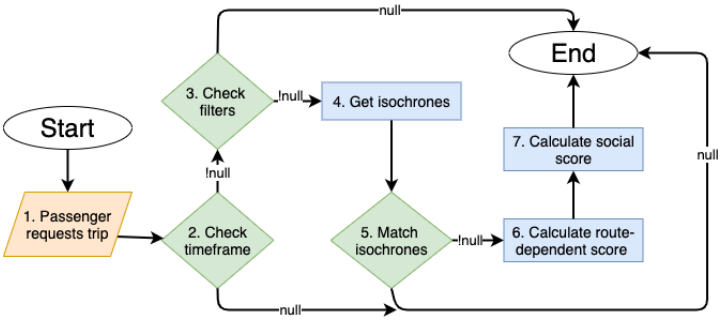

Fig. 1: Flow diagram of the matching algorithm process (Source: own representation)

3.1 Preselection of the trips

In the first step of the matching algorithm in figure 1 the passenger requests a trip. Afterwards fitting trips are filtered based on the fixed parameters shown in step two and three. The time of departure (step two), extended by six hours[9], is used to create a valid timeframe and remove trips too far in the future. Then in step three the filter options (contact or friend, user rating and number of seats) are applied to exclude unwanted trips.

3.2 Isochrone Matching

A further reduction of possible trips is achieved in the fourth and fifth step from figure 1, based on isochrones. Isochrones are polygons indicating an area reachable from the central

[9] The six hour time frame can be adjusted as needed and will be used within this implementation to maintain a reasonable number of possible trips.

point in a certain time or distance [Ga11]. The isochrones are calculated with the help of a routing engine in step four from figure 1 and result in a high reduction of trips for the matching process. Thus, the calculation of isochrones represents a decisive step in the matching algorithm. Trips involving a large detour for the driver are no longer taken into account in matching in step five. By spatially limiting the pick-up and drop-off points of the passenger, the driver is prevented from driving past the destination of the planned route in order to deliver the passenger at the drop-off point.

3.2.1 Route reduction through isochrones

For the consideration of the driver's detour, a representation for a time interval is chosen. It means that as soon as a node of the route is contained within this polygon, the driver can reach the center of the polygon with a maximum detour of twice the value of the isochrone, turn around and continue the planned route at the same nodal point. However, this is an idealized value, since in reality the driver would still have to search for a turning point. This estimation can only determine whether the points can be reached at all, with a maximum detour. An exact determination of the specific detour of the driver is not possible, because the point's within the polygon can not be determined or a possibly optimized route with a shorter detour can occur due to a changed routing. Figure 2 shows the possible changes compared to the initial route.

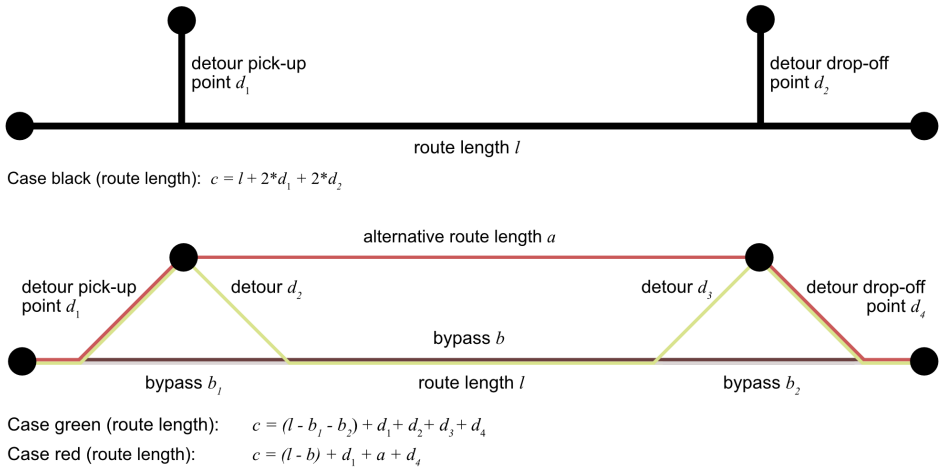

Fig. 2: Options for changing the route (Source: own representation)

For the red and green case, the length of sections *b1* and *b2*, as well as *a* and *b*, are not presented for the determination of the absolute route length. However, these changes are always shorter than the route variant shown in the black case, whose partial routes are all

estimated to the maximum or, in the case of route length l, have already been calculated. The route selection is based on the principle of whether the system can guarantee the driver that the detour is always smaller than the driver prefers. In addition, in this step the maximum detour can be compared to a specified maximum system value or a dynamic value based on the route length.

3.2.2 Estimation of the maximum detour

To estimate the detour more precisely, several isochrones with an increasing range are calculated around the start and end points of the passenger. Four isochrones with a range of 1.25, 2.5, 3.75 and 5 minutes are used. The individual polygons reflect the doubled detour of 2.5, 5, 7.5 and 10 minutes. For each route, the smallest polygon around the start and end point is now determined by finding a nodal point of the route. By adding the intervals of the two determined isochrones, the maximum deviation from the driver's optimal route can be estimated. An example calculation of this deviation can be seen in Figure 3.
The first route, Route I, intersects the 3.75 minute isochrone formed around the passenger's

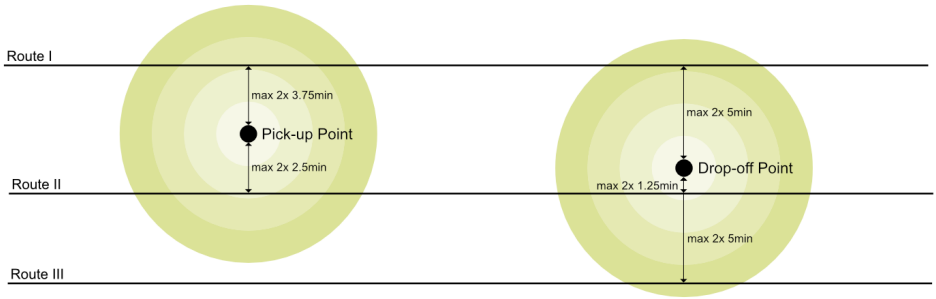

Fig. 3: Estimation of the detour (Source: own representation)

starting point. At the end point, this route intersects the isochrone describing a maximum detour of 10 minutes. As a result, the maximum detour of the trip is:

$$7.5min + 10min = 17.5min > 15min$$

If the driver of Route I specifies, for example, a maximum detour of 15 minutes, based on the calculation of the isochrones, it cannot be guaranteed that this driver requirement will be met. The route is removed from the further matching procedure. For Route II, a maximum detour of 7.5 minutes can be calculated by the isochrones:

$$5min + 2.5min = 7.5min < 15min$$

In this case, the estimation can ensure that the identical requirement of the driver is maintained during the matching procedure. The route is therefore still considered during the further matching procedure. For the case of Route III, if a route does not intersect any of

the isochrones at either start or end point, the route will also not be considered in the further steps due to the lack of maximum estimation. Another problem may arise if the direction of the driver's movement is opposite to the direction of the passenger. For example, the driver may have a small detour from certain route points to both points of the passenger, but the driver's route passes the passenger's drop-off point first. In this case, the maximum estimated detour must be increased due to the turnaround during the trip and the trip may need to be removed from the matching process. To solve this problem, only the isochrones around the pick-up point of the driver are initially considered. If the route does not intersect any of these isochrones, the process can be terminated at this point since no estimation is possible. If the route intersects an isochrone of the pick-up point, the first route point seen from the route's starting point is cached. For the determination of the drop-off point isochrone, the route is limited to the subsection between this stored route point and the end point of the route. Despite this consideration, the method has weaknesses in determining points that are close to each other and therefore can be reached via the identical route point. The solution to this problem is challenging without a concrete reachability determination of those points. However, this would increase the complexity and computation time to the extent that these cases remain in the matching process.

3.2.3 Spatial limits of pick-up and drop-off points

In addition to the calculation, the location of the pick-up and drop-off point of the trip is considered in relation to the location of the start and end points of the rides. This extra step is intended to prevent the driver from driving past his actual destination to drop off a passenger. Furthermore, this ensures that the driver always drives in the intended direction and does not have to drive in the opposite direction first. The method used is based on pure vector calculations due to their high-performance implementation. Figure 4 shows the steps of the method. To obtain the area in which both the pick-up and the drop-off point can be located, the air line between the start and end point of the trip is calculated first.

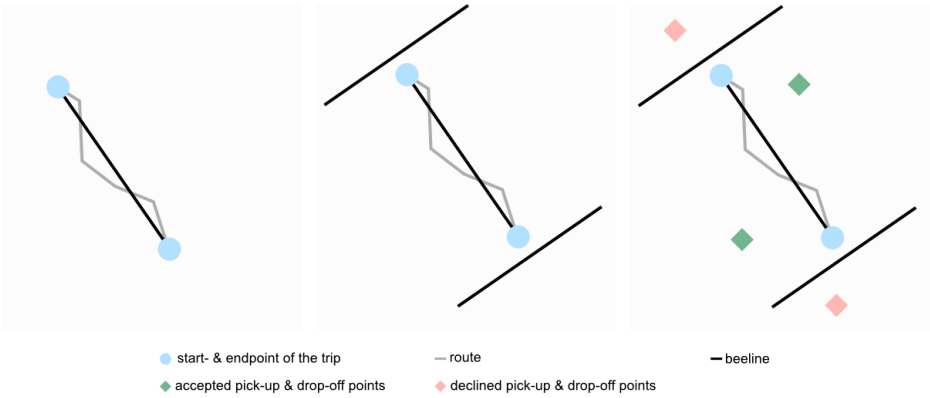

Fig. 4: Spatial limits of pick-up and drop-off points (Source: own representation)

The air line is rotated 90 degrees at both points to create a square area between the start and end point. Due to the simpler calculations compared to more complex shapes such as convex contours, a square shape is used. To include points in the close vicinity of the start and end point, the line is shifted away from the start and end point by 20 percent of the air lines length respectively. With a five kilometer air line between the routes start and end point, the line shift would integrate points with a one kilometer air line into the search area. In the context of short-distance trips, this range can be considered reasonable. For trips that initially move away from the end point of the trip, the route point with the greatest distance to the end point is selected. If after these steps either the passenger's pick-up or drop-off point is located outside these defined boundaries, the trip is excluded from the matching process. Otherwise the trip is considered in the final step of the process.

3.3 Calculation of a route score

After applying the matching in Paragraph 3, the best routes are selected in the sixth and seventh step from figure 1 based on a set of metrics. The metrics can be divided into route-dependent (step six) and individualized social metrics (step seven). Route-dependent metrics include absolute detour time, the proportional increase in route length compared to the original route and time to passenger's pickup. Since the route reduction using isochrones only estimates a maximum time detour and not the mentioned route-dependent metrics, a routing engine is used in an additional step to accurately recalculate the routes including the passenger's pick-up and drop-off point. Subsequently, the metrics are determined based on the difference between the original route and this newly recalculated route. Although the routing engine has a disadvantage in the computing time of the new routes compared to other methods such as approximations, the accuracy of the passenger's pick-up time in particular is a key argument for the integration into this algorithm. In order to compare the different metric types, the various metrics get mapped on a range of 0 to 1, where 1 is the best result. Optimizing these metrics benefits both the driver and passenger. Route-dependent metrics take up 67.5 percent of the score, where each metric has a 22.5 percent share of it. These values have been chosen for the first version of this algorithm, since the route-dependent metrics have a higher relevance for a well functioning match. Social metrics add a drift of suggested routes toward passenger's preferences regarding their driver to enhance the user experience. These include smoking, type of relationship, gender, user rating and willingness to either talk or hear music during the trip. Social metrics take up 32.5 percent of the score, whereby each metric has a 5.4 percent share of it. Except for user rating and type of relationship, these metrics get dynamically optimized based on the passenger's previously selected trips using weights. Every category of a metric has a weight $\omega \in [0, 0.05416]$ attached to it, all weights of one metric sum up to 0.05416. For first-time passengers, the weights of a metric are evenly distributed, which means exemplary for three categories, that at the beginning ω is equal to 0.01805. For adjusting the weights to the preferences of a passenger, the softmax function is used to scale the current weights of a metric to the interval $[0, 1]$, but these maintain their ratios to each other [Nw18]. Then, for each selection

that matches the currently selected category, a small value $\varepsilon \in \mathcal{R}$ is added to ω or subtracted for any mismatch. To compensate that a passenger's choice is sometimes the consequence of choices offered, ε is multiplied by $(1/f)$ where f is the frequency of the category's occurrence in the options offered. If the relative frequency of a category in options offered was 1, this compensation is not performed, because the passenger would have no choice. To adjust the resulting weights to the score's calculation, the softmax function is used again. Then the weight of the selected category is multiplied by the proportion of its metric, which is 5.4 percent. In this way, a score is calculated for each route, taking into account route-dependent and social metrics. Based on this, the best matches are returned to potential passengers.

4 Evaluation

An evaluation of the algorithm is carried out, according to the methodology of Design Science as mentioned in paragraph 2. It should be ensured that the instant matching algorithm generates appropriate matches for both the driver and the passenger and that its outputs provide optimized solutions to the aspects defined in paragraph 3. To achieve this, a test followed by an evaluation is mandatory [He04]. First, the general solution space of matching will be outlined. The evaluation is based on the score criteria set out in paragraph 3 and is carried out using different test scenarios. Those test scenarios are intended to check whether the desired behavior of the algorithm occurs and thus verify its functionalities. Due to the short duration of the project and the COVID-19 pandemic, there is no opportunity of in-person rides as well as to evaluate the influence of social metrics. Furthermore, there are no suitable data sets to allow independent evaluation, so randomly generated trips are used to validate the algorithm. The algorithm has been developed in a first stable version and is based on a database of randomly generated routes. A total of 5292 routes have been created, which constitute the basis for the evaluation. By using various filters, the pool of all available trips is significantly reduced, which means not all routes are used for the matching process.

Table 1 and Table 2 show which routes are put out after a request from the passenger. For this purpose, the desired start and end position of the passenger, as well as the preferred start time, are transferred and then the calculated scores are evaluated. The first test scenario was requested on 04/14/2021 at 06:34:47. Results of the request are shown in table 1. Table 2 shows the output for a request on 04/02/2021 at 16:00:00. The delay of the driver results

Route ID	Score	Detour time (h:m:s)	Detour distance (%)	Time until pickup (h:m:s)
2464	0.79587	00:05:35	35	00:19:12
2363	0.58698	00:08:23	38	03:40:48
2594	0.13314	00:10:58	46	04:48:04

Tab. 1: First test scenario

from the detour time and the distance the driver has to travel to pick up the passenger. Both detour time and distance have a 22.5 percent influence on the matching score. In table 2 it is

Route ID	Score	Detour time (h:m:s)	Detour distance (%)	Time until pickup (h:m:s)
432	0.77405	00:08:44	25	01:07:12
444	0.59168	00:09:32	23	00:37:12
286	0.46281	00:09:15	35	00:35:38
366	0.25393	00:09:41	37	02:30:00
510	0.16085	00:09:34	35	06:00:36

Tab. 2: Second test scenario

shown that the routes with IDs 286, 366 as well as 510 get a score below 0.5. On the other hand, the score for the routes with IDs 432 and 444 is above 0.5. The detour distance is smaller for the drivers in relation to the other three routes and is therefore rated better in the score. Especially the comparison between routes 286 and 444 clearly shows the influence of the detour distance on the score, as the other metrics show similar values. In table 1, the influence of detour time is illustrated, as the percentage detour distance of routes 2464 and 2363 differs by only three percent. Since the detour time for route 2363 is significantly higher, the score is proportionally lower. By comparing the two routes with IDs 2464 and 2363 in the first test scenario shown in table 1, it demonstrates the calculation of scores is influenced by the waiting time of the passenger. Despite similar values in terms of detour, route 2363 receives a lower score due to its long waiting time for the passenger. This is equally clear in the comparison of routes 286 and 366 from table 2. Both the interests of the driver and the passenger are considered in the calculation of the score. Since a large number of available drivers is essential for a dynamic ride-sharing system, the interests of the driver are weighted with a total of 45 percent. The time to pick up is included in the calculation with 22.5 percent. As a result, trips whose values are lower in terms of distance and duration of detour are displayed preferentially in the matching. If the comparison between two possible routes is similar in terms of the detour, as is the case for routes 286 and 366 in the second test scenario, it will become clear that the route with low waiting time receives a significantly better score. Considering the performance of the matching algorithm, it should not take longer than ten seconds to keep the user's attention [Ni94]. The test scenarios in table 1 and 2 take about one second on average, which is below the threshold. The reduction out of the 5292 offered routes by the pre-filtering processes has to be emphasized here. In the first test scenario 17 possible routes are considered and nine of them get filtered out because of their detour. The second test scenario takes 12 possible routes but only needs to recalculate five routes. Only by reducing the routes through the matching process, this acceptable result can be achieved for the user.

5 Conclusion

The paper represents a matching algorithm in the context of ride-hailing to join a passenger with a driver using isochrones and social scoring. The algorithm's internal evaluation points out the success of isochrone matching as an efficient way to represent both, the drivers and

passengers interests while still getting suitable results in terms of the drivers detour. It is a lightweight implementation mostly limited by external sources such as the used routing engine, which handles the isochrone calculation as well as the routing itself done within the matching process. Since this evaluation is based on a locally hosted engine, processing times might increase when running the operations on an external server. It is the only bottleneck in terms of speed though, having the greatest impact on the process.

To optimize the performance of this algorithm in a next version, tweaking the static weights might result in a better output since those are currently equally distributed. Furthermore, the evaluation must be conducted in a public study to obtain meaningful results for both matching itself and the influence of social factors.

Bibliography

[Ag10] Agatz, N. A. H.; Erera, Alan; Savelsbergh, Martin; Wang, X.: Sustainable Passenger Transportation: Dynamic Ride-Sharing. ERIM Report Series Reference No. ERS-2010-010-LIS, 01 2010.

[AGK20] Aydin, Omer Faruk; Gokasar, Ilgin; Kalan, Onur: Matching algorithm for improving ride-sharing by incorporating route splits and social factors. PLOS ONE, 15(3):1–23, 03 2020.

[BP14] Bonazzi, Riccardo; Poli, Michaël: Beyond Uber. Business model considerations for alternatives to traditional taxis. Proceedings of the 11th Conference of the Italian Chapter of the Association of Information Systems (ITAIS) : Digital Innovation and Inclusive Knowledge in Times of Change 2014, 11 2014.

[CAF16] Campbell, I.; Ali, M. M.; Fienberg, M. L.: Solving the dial-a-ride problem using agent-based simulation. South African Journal of Industrial Engineering, 27:143–157, 11 2016.

[CC] Consumers, Health, Agriculture; (Chafea), Food Executive Agency: Exploratory study of consumer issues in online peer-to-peer platform markets: Task 4 – Case study: BlaBlaCar. Technical report, Consumers, Health, Agriculture and Food Executive Agency (Chafea).

[CLC17] Chang, Yu; Lee, Yong; Choi, Sungsup: Is there more traffic congestion in larger cities? -Scaling analysis of the 101 largest U.S. urban centers. Transport Policy, 59:54–63, 10 2017.

[Ga11] Gamper, Johann; Böhlen, Michael; Cometti, Willi; Innerebner, Markus: Defining Isochrones in Multimodal Spatial Networks. Proceedings of the 20th ACM International Conference on Information and Knowledge Management, p. 2381–2384, 2011.

[He04] Hevner, Alan R.; March, Salvatore T.; Park, Jinsoo; Ram, Sudha: Design Science in Information Systems Research. MIS Quarterly, 28(1):75–105, 2004.

[Ja86] Jaw, Jang-Jei; Odoni, Amedeo R.; Psaraftis, Harilaos N.; Wilson, Nigel H. M.: A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. Transportation Research Part B: Methodological, 20(3):243–257, 1986.

[Ja17] Jalali, Roozbeh; Koohi-Fayegh, Seama; El-Khatib, Khalil; Hoornweg, Dan; Li, Heng: Investigating the Potential of Ridesharing to Reduce Vehicle Emissions. Urban Planning, 2(2):26, 6 2017.

[Ni94] Nielsen, Jakob: Usability Engineering. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 11 1994.

[NJ16] Nourinejad, Mehdi; J.Roorda, Matthew: Agent based model for dynamic ridesharing. Transportation Research Part C: Emerging Technologies, 64:117–132, 03 2016.

[Nw18] Nwankpa, Chigozie; Ijomah, W.; Gachagan, A.; Marshall, Stephen: Activation Functions: Comparison of trends in Practice and Research for Deep Learning. CoRR, abs/1811.03378, 2018.

[Sc16] Schreieck, Maximilian; Safetli, Hazem; Siddiqui, Sajjad Ali; Pflügler, Christoph; Wiesche, Manuel; Krcmar, Helmut: A Matching Algorithm for Dynamic Ridesharing. Transportation Research Procedia, 19:272–285, 2016. Transforming Urban Mobility. mobil.TUM 2016. International Scientific Conference on Mobility and Transport. Conference Proceedings.

[Sc18] Scheelhaase, Janina; Grimme, Wolfgang; O'Sullivan, Marlene; Naegler, Tobias; Klötzke, Matthias; Kugler, Ulrike; Scheier, Benedikt; Standfuß, Thomas: Climate Protection in the Transport Sector — Current Examples. Wirtschaftsdienst, 98(9):655–663, 2018.

[SX13] Santos, Douglas O.; Xavier, Eduardo C.: Dynamic Taxi and Ridesharing: A Framework and Heuristics for the Optimization Problem. International Joint Conference on Artificial Intelligence Procedia, 13:2885–2891, 08 2013.

[Wo04] Wolfler Calvo, Roberto; de Luigi, Fabio; Haastrup, Palle; Maniezzo, Vittorio: A distributed geographic information system for the daily car pooling problem. Computers & Operations Research, 31(13):2263–2278, 2004.

[Ç17] Çetin, Tamer: The Rise of Ride Sharing in Urban Transport: Threat or Opportunity? Intech, Hamburg, 1 edition, 01 2017.

[Ö11] Österle, Hubert; Becker, Jörg; Frank, Ulrich; Hess, Thomas; Karagiannis, Dimitris; Krcmar, Helmut; Loos, Peter; Mertens, Peter; Oberweis, Andreas; Sinz, Elmar J.: Memorandum on design-oriented information systems research. European Journal of Information Systems, 20(1):7–10, 2011.