International Journal of Scientific Engineering and Research (IJSER)
ISSN (Online): 2347-3878
Impact Factor (2018): 5.426

Taxi Ride Sharing with the Aid of Social Media

Silpa K M[1], Naveena Treesa Joseph[2]

[1]Post Graduation Student, College of Engineering Kidangoor

[2]Assistant Professor, College of Engineering Kidangoor

Abstract: ***As urban populations grow, cities face many challenges related to transportation, resource consumption, and the environment. Ride sharing has been proposed as an effective approach to reduce traffic congestion, gasoline consumption, and pollution. With the deep penetration of smartphones and geo-locating devices, ridesharing is envisioned as a promising solution to transportation-related problems in metropolitan cities, such as traffic congestion and air pollution. Despite the potential to provide significant societal and environmental benefits, ridesharing has not so far been as popular as expected. Notable barriers include social discomfort and safety concerns when traveling with strangers. To overcome these barriers, in this paper, we propose a new type of Social-aware Ridesharing Group (SaRG) queries which retrieves a group of riders by taking into account their social connections and spatial proximities. While SaRG queries are of practical usefulness, we prove that, however, the SaRG query problem is NP-hard. Thus, we design an efficient algorithm with a set of powerful pruning techniques to tackle this problem. Experimental results on real datasets show that our proposed algorithms achieve desirable performance.***

Keywords: Group Queries**,** Location-based Services, Query Processing, Ridesharing, Social Acquaintance

1. Introduction

In recent times, due to population explosion, the affordability and mass production and distribution of automobiles, the possession of an automobile has changed from being a luxury to a necessity [1]. The increase in use of fuel-powered vehicles has resulted in a drastic increase in fuel prices as well as traffic congestion. It has also impacted the environment in the form of global warming and air-pollution. A few methods devised to reduce the impact were public transport, non-conventional fuel resources and walking/cycling to reach ones destination. The merits of the above solutions were the reductions in the amount of pollution as well as lesser road congestion. However, public transport is not a well-developed system in India and apart from the inconvenience with respect to time, it is also usually unreliable. Though non-conventional fuel resources attempt to stem pollution, there has not yet been devised a cost effective manner in which to harness it for automobiles. Physical means of transport are not an option when faced with a transit of long distances [1]. Our intended system aims to remove all of the above discrepancies. We plan to create a carpooling system which gives users the same flexibility that a private car gives and which reduces the number of vehicles used at the same time. Availability and convenience issues can be solved through connectivity to an online social media (Facebook) and a smartphone application (android) for creation of dynamic carpools.

Ride-Sharing Services (RSSs) provide partner discovery services to drivers and riders with similar rides for initializing sharing travel experiences. RSSs allow drivers to share vacant seats of their vehicles on the road, bringing about various benefits to individual users (e.g., improved vehicle occupancy, shared travel costs and extended social circles) and the society (e.g., reduced traffic congestion, fuel consumption and carbon dioxide emissions) [2]. Due to these appealing advantages, many service providers have emerged to offer ride-share partner discovery services, e.g. Flinc, Lyft Line, UberPool, Waze Carpool and Blablacar. Ride-sharing has become increasing popular in metropolis to reduce crowded traffic and expensive transportation costs [3].

Due to the steady growth in urban populations [4], cities now face huge challenges related to transportation, resource consumption, and pollution. Ride sharing has been proposed as a strategy to decrease road traffic and gasoline consumption [5], while at the same time serving the transportation needs of city dwellers. In large cities, there is substantial unused taxi capacity that can be filled by ridesharing services. Despite the potential to provide significant societal and environmental benefits, ridesharing has not so far been as popular as expected. Notable barriers include social discomfort and safety concerns when traveling with strangers [6]. To overcome these barriers, in this paper, we propose a new type of method which retrieves a group of riders by taking into account their social connections and spatial proximities.

Social networks represent certain types of social interaction, such as acquaintance, friendship, or collaboration between people or groups of people that are referred to as actors [7]. In social networks, vertices (nodes, dots) usually stand for actors, and edges (arcs, links, lines) represent the pairwise relations or interactions between the actors.

One of the central concepts in social network analysis is the notion of a cohesive subgroup, which is a tightly knit subgroup of actors in a social network. While the notion of a clique embodies a perfect cohesive group, in which every two entities are connected to each other, this definition is overly conservative in many practical scenarios. Indeed,(i) not need to require every possible link to exist between elements of a cohesive subgroup; (ii) the social network of interest may be built based on empirical data, which are prone to errors, so, even if a completely connected cohesive subgroup is sought for, it may be impossible to detect due to erroneous data [7]. To overcome this impracticality of the clique model, other graph-theoretic formalizations of the cohesive subgroup concept have been proposed in the

Licensed Under Creative Commons Attribution CC BY

literature. Not surprisingly, all these alternative definitions canbe viewed as clique generalizations, each of which relaxes one of the elementary clique properties, such as familiarity, reachability, or robustness. Hence, we use the term clique relaxations in reference to such models [7].

This model aims to find a ridesharing group with a desired level of social acquaintance. To model such social acquaintance, we assume the existence of a social network graph in which users are connected if they have acquaintance relationships (e.g., friends or colleagues). Such a network might be derived from call graphs based on telephone call detail records (CDRs) or online social networks such as Facebook and Twitter. There are a number of social models that can be employed to measure the social cohesiveness of a ridesharing group, such as star (friend) (one central user has direct connections to all other users), star (friend of friend) (one central user has direct or through-a-friend connections to all other users), and k-core.A k-core of a graph is a maximal connected subgraph in which every vertex is connected to at least k vertices in the subgraph. Along with social acquaintance, trip matching of a ridesharing group must also be measured. The primary cost of a rider is the travel cost between the riders origin, destination and the drivers origin, destination. For ease of description, we use Dijkstra's shortest path distance in this paper.

The returned ridesharing group should have the smallest travel cost because naturally only riders whose origin and destination are close to those of the driver are willing to join the drivers ridesharing. Note that the minimum travel cost requirement and the social constraint are equally important in our problem. Our proposed solution does implicitly support a preference over these two factors by setting k to different values (e.g., setting k = 0 makes the minimum travel cost the sole factor to consider). However, weighted social relations could be particularly useful in some applications [6].

The k-core decomposition of a graph maintains, for eachvertex, the max-k value: the maximum k value for which a k-core containing the vertex exists. This decomposition enables one to quickly find the k-core containing a given vertex for a given k. Algorithms for creating k-core decomposition of a graph in time linear to the number of edges in the graph exist [8]. For applications that manage dynamic graphs, applying such algorithms for every edge insertion and removal is prohibitive in terms of performance.

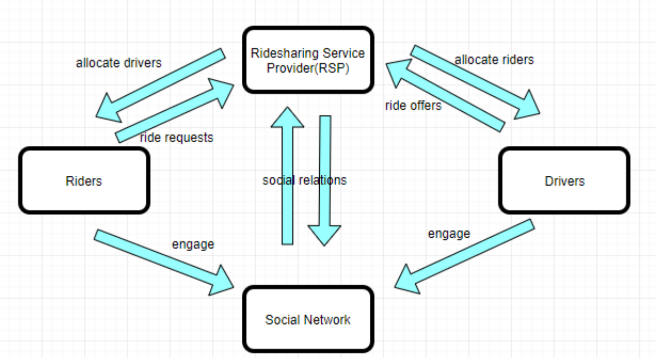

Figure 1: A framework of the social-aware ridesharing System

Our proposed ridesharing system consists of three parties: (i) riders (or passengers who want to participate in ridesharing), (ii) drivers (or private car owners who offer ridesharing), and (iii) ridesharing service provider (RSP) (the server in charge of the arrangement of ridesharing). The riders submit ride requests to the RSP, while the drivers send in ride offers. In other words, a ride offer provided by a driver forms an SaRG query; the riders who submitted ride requests form the data space (or search space); the RSP arranges the best ride matches of ridesharing by jointly considering trip matching, social connections as well as the capacity of a car [27]. Designing efficient matching algorithms for the RSP is the most important task to make the system work effectively. Note that, in our problem, the RSP hosts a set of active ride requests (expired requests might be dropped and re-submitted). Once there comes a ride offer from a driver, the RSP will match the most suitable riders to the driver. A ridesharing group is composed of a driver and the most suitable riders.

Slugging assumes that the driver's trip is fixed. Riders would walk to the origin location of the drivers trip. Board at the departure time, alight at the driver's destination, and then walk to their own destinations. SaRG query aims to find a ridesharing group with a desired level of trip matching and social acquaintance.

2. Scope of this Survey

Papers that proposes social-aware ridesharing are surveyed and classified according to their major methodology. From this several issues such as performance, execution time, and privacy preserving ability for each method are reported. It is inappropriate to explicitly declare which methods actually demonstrate the highest performance. One of the scope of this survey is to find best method in real life which can be used for ridesharing
.

3. Literature Survey

3.1 En-Route Ride-Sharing

This assesses the potential of ride-sharing for reducing traffic in a city based on mobility data extracted from 3G Call Description Records(CDRs).CDRs are generated when a cellphone makes or receives a call or uses a service, e.g. SMS [9]. Information regarding the time/date and the location of the Base Transceiver Station (BTS), used for the communication, are then recorded. More specifically, the main fields of each CDR entry are the following: (1) the originating cellphone number (2) the destination cellphone number (3) a time-stamp (when call started) (4) the duration of the call and (5) the BTS tower used by one, or both if applicable, cellphones. First the recorded cell towers of a user are clustered to produce the list of places that the user visits [10]. Then, regression analysis is applied to determine the features of the clusters that represent important places. The used features are: (1) the number of days that the user appeared on the cluster; (2) the duration of user appearances on the cluster; and (3) the rank of the cluster based on number of days appeared. Once important locations have

Licensed Under Creative Commons Attribution CC BY

been inferred, and the algorithm chooses which of these are home and which are work locations. According to their results, the best features that characterize home and work are: (4) the number of phone calls between 7PM - 7AM, i.e. Home Hour Events, and (5) number of phone calls between1PM - 5PM, i.e. Work Hour Events. [11] Privacy of passengers is not concerned. Passengers may need to travel with strangers. As the home and work locations of passengers are inferring from CDRs, large numbers of CDRs are thus generated everyday which is difficult to handle.

3.2 Simulation Algorithm

STaRS supports the simulation of real-time ride sharing which serves unplanned trips. A wide deployment of ride sharing requires a better understanding of its tradeoffs. This is challenging since there are multiple stakeholders with different, and sometimes conflicting, interests. Governments want less traffic and pollution; taxi companies want to maximize their profits; and passengers would like to reach their destination quickly and cheaply. To design an effective policy, these interests need to be considered [12].

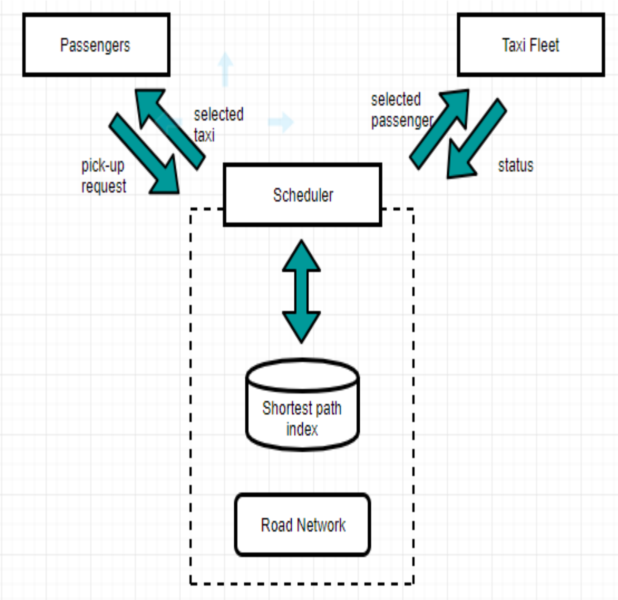

Figure 2: Taxi ride-sharing model

The main components of our simulation model are:

Taxi Fleet: The taxi fleet refers to the set of taxis that are involved in the simulation. In contrast to previous works, where taxis are considered as homogeneous objects, to support a multi-vendor environment and different types of vehicles, we consider each taxi as a distinct object with its own specifications.

Passengers: We assume that passengers ride in groups of size greater than or equal to one. Each group is associated with a drop-off location and a set of ride-sharing constraints.

Scheduler: For each pick-up request, the scheduler finds the most appropriate taxi based on pre-defined metrics. To do so, the scheduler must know all taxi locations along with their current states at all times.

Road Network: The underlying road network of a city is represented as a directed graph. All taxis travel along this road network. Each directed edge represents a road segment, and each node represents the intersection of two or more roads. When a road allows traffic flow in both directions, there are two directed edges for that road.

In the taxi ride-sharing problem, the goal is to minimize the total cost or maximize the total utility of sharing while meeting a set of constraints. A straightforward way to compute the additional cost is to explicitly find an optimal route for the cab that includes the pick-up and drop-off locations of rider and to compare its cost with the cost of the current route for the cab. However, computing the optimal path is known as the Sequential Ordering Problem (SOP) which is a version of the Traveling Salesman Problem and is NP-hard [13].

Privacy of passengers is not concerned. As the social acquaintance between the passengers is not concerned, passengers may need to travel with strangers.

3.3 Clique relaxation

Social networks represent certain types of social interaction, such as acquaintance, friendship, or collaboration between people or groups of people that are referred to as actors. In social networks, vertices usually stand for actors, and edges represent the pairwise relations or interactions between the actors. One of the central concepts in social network analysis is the notion of a cohesive subgroup [14], which is a tightly knit subgroup of actors in a social network. While the notion of a clique embodies a perfect cohesive group, in which every two entities are connected to each other. This definition is overly conservative in many practical scenarios.

To overcome this impracticality of the clique model, other graph-theoretic formalizations of the cohesive subgroup concept have been proposed in the literature. Not surprisingly, all these alternative definitions can be viewed as clique generalizations, each of which relaxes one of the elementary clique properties, such as familiarity, reachability, or robustness [15]. Hence, we use the term clique relaxations in reference to such models.

Relaxation Models

We are considering a simple undirected graph, G=(V,E). A graph is said to be complete if there exists an edge between every pair of vertices in the graph. An induced subgraph is a subset of the vertices of a graph G together with any edges whose endpoints are both in this subset. A clique in a graph is an induced subgraph which is complete.

3.3.1 Relaxation of Reachability

The length of the shortest path between any two vertices u,v in a graph is denoted by d(u,v).A k-clique S is the subgraph of a graph : for all vertices u,v in S, d(u,v)<=k.

3.3.2 Relaxation of Familiarity

Relax the minimum number of neighbors or the maximum number of non-neighbors within the group. Familiarity is another important property one wants to have in a cohesive

Licensed Under Creative Commons Attribution CC BY

subgroup. Every member of the group should be familiar with every other member of the group. The k-core concept imposes a lower bound on the minimum degree within the subgraph [16]. Ensures that each vertex in the group is connected to at least k other vertices in the graph. A k-core S is a subgraph of a graph: for every vertices u, v in S, deg(v)>=k [7].

Drivers and passengers willing to share rides may not have any way of identifying one another. Several studies
have shown that the majority of drivers are unwilling to incur more than a 5-10 minute delay in order to pick up and 250 drop off passengers

3.4 Demand Responsive Transit (DRT)

On-Demand Bus is a Demand Responsive Transit (DRT) service. It allows potential passengers to request service via the Internet or mobile phone. The requests compose of pick-up location, delivery location and desired delivery time (or pick-up time). The computer executes two main algorithms which are vehicle-choosing algorithm and routing algorithm. After calculation, the system will report to the customer whether the request is accepted or not. If it is accepted, the vehicle will pick up and deliver him to his destination within a guaranteed time [17].

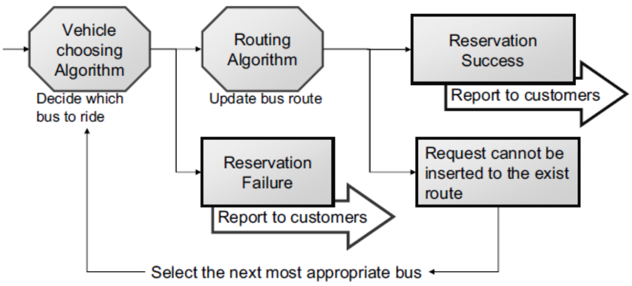

Figure 3: Algorithm Outline

In this problem, N customers have to be transported by maximum V vehicles. Each customer, customer n, has to specify pick-up bus stop, p(+n), and delivery bus stop, p(-n) [18]. He also has to specify either desired pick-up time (DPTn) or a desired delivery time (DDTn).

Vehicle Choosing Algorithm

We try to introduce an effective algorithm with less calculation time, especially when solving big problems. First, we define direction vector of customer. On the other hand, we define bus direction vector. Then we define the direction decision Variable. When a new reservation comes into the system, the vehicle choosing algorithm will be executed for each available bus. The bus with the most value of decision variable is the one with the closest direction to the new demand. That bus will be firstly selected to be executed by the next algorithm, routing algorithm [19]. It is very time consuming as the system has to run both vehicle choosing algorithm and route finding algorithm or each taxi and request.

3.5 T-Finder

Recommender system for both taxi drivers and the riders. Done by using the knowledge of passengers' mobility patterns and drivers' picking up/dropping off behaviours. First the recommender system provides the drivers with some locations and the routes to these locations [20]. These are the locations towards which they are more likely to find passengers quickly. Second it recommends the people with some locations where they can easily find taxis. Taxis report their present location to a data center in a certain frequency. Besides geo-position and time, occupancy is also recorded. A large number of GPS trajectories are thus generated everyday. These trajectories include two aspects of knowledge: Passengers' mobility ie, when and where passengers get on and off a taxi. The other are taxis pick up/drop off behaviours. We consider three states for a working taxi: occupied (O), cruising (C) and parked (P) [20].

A taxi trajectory is a sequence of GPS points logged for a working taxi. Each point p has the fields: time stamp p.t, latitude p.lat, longitude p.lon, located road segment p.r, state p.s. A taxi trip is a sub-trajectory which has a single state, either cruising or occupied. We develop an approach to detect the parking places from GPS trajectories and segment the GPS trajectories [21]. We first keep checking the distance between the current point and the latter point until the distance is smaller than a threshold [22]. A large number of GPS trajectories are thus generated everyday which is difficult to handle. Passengers' privacy and their social acquaintance is not concerned. It is based on trip matching and social acquaintance.

3.6 Privacy-preserving Partner Selection

The ride-sharing system consists of a large number of users and a RS-server. A user is either driver or rider with trip requests. The users use their mobile phones with ridesharing applications to access the RS-server. Rider has to locally specify her/his trip, including source, destination, the earliest departure time and the latest arrive time, and then encrypts it to generate ride request, as well as driver [2].

Additionally, to form a ride offer, driver needs to set and encrypt her/his acceptable spatial region set, including all locations where the driver is willing to pick up or drop off a rider in her/his trip. All ride offers and ride requests are sent to the RS-server. With PRIS, they obtain the information about the partners recommended by the RS-server. The driver and rider can start their travel, if both accept this recommendation [23].

A RS-server is a platform with computational and storage capabilities offering the RSS for particular metropolitans. The RS-server selects feasible matches between ride offers and ride requests. The RS-server further selects the best ride-share partner for user and arranges ride-share with objectives in accordance to benefits, e.g., maximizing system-wide TTS, and returns results to the riders and drivers. After receiving the encrypted ride offers/ requests from users, the RS-server uses the three-step partner selection mechanism to discover feasible partners for ride-sharing. [3]

Licensed Under Creative Commons Attribution CC BY

Dynamic trip matching between riders and drivers is not possible since the RSS has to decrypt the encrypted trip requests submitted by riders which is time consuming.

3.7 Zero-effort carpooling

In current carpooling systems, drivers and passengers offer and search for their trips through available mediums, for example, accessing a carpool Web site by smartphone to find a possible match for a journey. Although efforts have been made to achieve fast matching for known trips, the need for accurate mobile tracking for individual users still remains a bottleneck. For example, drivers may be too impatient to input their routes before driving, or centralized systems may have difficulty tracking a large number of vehicles in real time. In this study, the authors present the idea of Mobility Crowdsourcing (MobiCrowd) [24], which leverages private smartphones to collect individual trips for carpooling without any explicit effort on the part of users. The authors scheme generates daily trips and mobility models for each user and then makes carpooling zero-effort by enabling travel data to be crowd-sourced instead of tracking vehicles or asking users to input their trips. With prior mobility knowledge, one users travel routes and positions for carpooling can be predicted according to the location of the time and other mobility context

On the basis of a realistic travel survey and simulation, the authors show that their scheme can provide efficient and accurate position estimation for individual carpools. Compared to previous carpooling servers, the needs for active mobile tracking are removed. MobiCrowd cuts down a great deal of communication spending between server and each vehicle. [25] Adapting position estimation instead of mobile tracking is not always accurate. Some methods to correct estimation errors should be considered.

3.8 Noah: A dynamic ridesharing system

This demonstration presents Noah, a dynamic ridesharing system. Noah supports large scale, real-time ridesharing with a service guarantee on road networks. Taxis and trip requests are dynamically matched. The system differs from traditional systems in that a taxi can have more than one customer on board provided that all waiting-time and service-time constraints of trips are satisfied. Noahs realtime response relies on three main components: (1) a fast shortest-path algorithm with caching on road networks, (2) fast dynamic matching algorithms to schedule ridesharing on the fly, and (3) a spatial indexing method for fast retrieval of moving taxis. Users are able to submit requests from a smartphone and to choose specific parameters such as number of taxis in the system, service constraints, and matching algorithms to explore the internal functionalities and implementations of Noah. The system analyzer shows the system performance including average waiting time, average detour percentage, average response time, and average level of sharing. Taxis, routes, and requests are animated and visualized through a Google Maps application programming interface. The demonstration is based on the trips of 17,000 Shanghai taxis for 1 day (May 29, 2009); the dataset contains 432,327 trips. Each trip includes the starting and destination coordinates and the start time. An iPhone application is implemented to allow users to submit a trip request to the Noah system during the demonstration [26].

4. Conclusion

We have shown that this model attains a good balance between simplicity and expressiveness. Another important contribution of this work is the novel shortest path indexing scheme where we make use of cache-coherent layout to speed up shortest path queries substantially. An SaRG query aims to find a group of riders in which each riders ridesharing trip is close to that of the query issuer and each member in this group is familiar with at least k other members. The system is planned in a way that makes it easy to implement without compromising user functionality and ease of use. In the future work, we will design a privacy preserving partner selection scheme considering the trust levels of drivers and riders.

References

[1] Talele, T., Pandit, G., and Deshmukh, P., 2012. “Dynamic ridesharing using social media”. InternationalnJournal.
[2] Agatz, N., Erera, A., Savelsbergh, M., and Wang, X., 2012. “Optimization for dynamic ride-sharing: A review”. European Journal of Operational Research, n223 (2), pp. 295–303.
[3] He, Y., Ni, J., Wang, X., Niu, B., Li, F., and Shen, X. S., 2018. “Privacy-preserving partner selection for ride-sharing services”. IEEE Transactions on Vehicular Technology.
[4] Barbosa, L., Pham, K., Silva, C., Vieira, M. R., and Freire, J., 2014. “Structured open urban data: understanding the landscape”. Big data, 2(3), pp. 144–154.
[5] Handke, V., and Jonuschat, H., 2012. Flexible ridesharing: new opportunities and service concepts for sustainable mobility. Springer Science & Business Media.
[6] Li, Y., Chen, R., Chen, L., and Xu, J., 2017. “Towards social-aware ridesharing group query services”. IEEE Transactions on Services Computing, 10(4), pp. 646–659.
[7] Pattillo, J., Youssef, N., and Butenko, S., 2012. “Clique relaxation models in social network analysis”. In Handbook of Optimization in Complex Networks. Springer, pp. 143–162.
[8] Badger, E., 2011. “Slugging the people transit”. Retrieved on September, 18, p. 2017.
[9] Isaacman, S., Becker, R., C´aceres, R., Kobourov, S., Martonosi, M., Rowland, J., and Varshavsky, A., 2011. “Identifying important places in peoples lives from cellular network data”. In International Conference on Pervasive Computing, Springer, pp. 133–151.
[10] Cho, E., Myers, S. A., and Leskovec, J., 2011. “Friendship and mobility: user movement in location-based social networks”. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 1082–1090.

Licensed Under Creative Commons Attribution CC BY

[11] Cici, B., Markopoulou, A., Frias-Martinez, E., and Laoutaris, N., 2014. "Assessing the potential of ridesharing using mobile and social data: a tale of four cities". In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, pp. 201–211.
[12] Yang, J., Jaillet, P., and Mahmassani, H., 2004. "Realtime multivehicle truckload pickup and delivery problems". Transportation Science, 38(2), pp. 135–148.
[13] Ota, M., Vo, H., Silva, C., and Freire, J., 2017. "Stars: Simulating taxi ride sharing at scale". IEEE Transactions on Big Data, 3(3), pp. 349–361.
[14] Sampson, R. J., and Groves, W. B., 1989. "Community structure and crime: Testing social-disorganization theory". American journal of sociology, 94(4), pp. 774–802.
[15] Luce, R. D., 1950. "Connectivity and generalized cliques in sociometric group structure". Psychometrika, 15(2), pp. 169–190.
[16] Seidman, S. B., 1983. "Network structure and minimum degree". Social networks, 5(3), pp. 269–287.
[17] Psaraftis, H. N., 1980. "A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem". Transportation Science, 14(2), pp. 130–154.
[18] Yuan, J., Zheng, Y., Xie, X., and Sun, G., 2011. "Driving with knowledge from the physical world". In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 316–324.
[19] Tsubouchi, K., Hiekata, K., and Yamato, H., 2009. "Scheduling algorithm for on-demand bus system". In Information Technology: New Generations, 2009. ITNG'09. Sixth International Conference on, IEEE, pp. 189–194.
[20] Yuan, J., Zheng, Y., Zhang, C., Xie, X., and Sun, G.Z., 2010. "An interactive-voting based map matching algorithm". In Proceedings of the 2010 Eleventh International Conference on Mobile Data Management, IEEE Computer Society, pp. 43–52.
[21] Yuan, N. J., Zheng, Y., Zhang, L., and Xie, X., 2013."T-finder: A recommender system for finding passengers and vacant taxis". IEEE Transactions on knowledge and data engineering, 25(10), pp. 2390–2403.
[22] Ziegelmann, M., 2001. "Constrained shortest paths and related problems".
[23] Hou, Y., Zhong, W., Su, L., Hulme, K., Sadek, A. W., and Qiao, C., 2016. "Taset: Improving the efficiency of electric taxis with transfer-allowed rideshare". IEEE Transactions on Vehicular Technology, 65(12), pp. 9518–9528.
[24] Burris, M. W., and Winn, J. R., 2006. "Slugging inhoustoncasual carpool passenger characteristics". Journal of Public Transportation, 9(5), p. 2.
[25] Liu, N., Feng, Y., Wang, F., Liu, B., and Tang, J., 2013."Mobility crowdsourcing: Toward zero-effort carpooling on individual smartphone". International Journal of Distributed Sensor Networks, 9(2), p. 615282.
[26] Tian, C., Huang, Y., Liu, Z., Bastani, F., and Jin, R.,2013. "Noah: a dynamic ridesharing system". In Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, ACM, pp. 985988.
[27] Cordeau, J.-F., 2006. "A branch-and-cut algorithm for the dial-a-ride problem". Operations Research, 54(3), pp. 573–586.

Author Profile

Silpa K M, She is a post graduation student in college of engineering kidangoor. Specialization in computer and information science. She received the graduation in computer science and engineering from Govt. Engineering College, Palakkad.

Naveena Treesa Joseph, She is an Asst. Professor at College of Engineering, Kidangoor. She received graduation and post graduation in Computer Science & Engg. from St. Joseph's College of Engg. & Technology. Her interested areas are Big data, Image Processing.

Licensed Under Creative Commons Attribution CC BY