International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 8, ISSN No. 2455-2143, Pages 180-185
Published Online December 2020 in IJEAST (http://www.ijeast.com)

SURVEY ON PEER-TO-PEER RIDE SHARING FOR "POOL" A RIDE SHARING APP

Piyush Agrawal
Dept. of Computer Engineering,
NBN Sinhgad School of Engineering,
Ambegaon Pune 411041, India

Harsh Agrawal
Dept. of Computer Engineering,
NBN Sinhgad School of Engineering,
Ambegaon Pune 411041, India

Avinash Bagul
Dept. of Computer Engineering,
NBN Sinhgad School of Engineering,
Ambegaon Pune 411041, India

Apurva Joshi
Dept. of Computer Engineering,
NBN Sinhgad School of Engineering,
Ambegaon Pune 411041, India

Ajinkya Ghorpade
Dept. of Computer Engineering,
NBN Sinhgad School of Engineering,
Ambegaon Pune 411041, India

Abstract-- **Many college students travel in public transports or walk a long distance to reach college. This is problematic because public transports can be slow and not available everywhere as they have a specific time of arrival in their stops and they have to halt at multiple places in the city which can make it quite time consuming for passengers to reach their destinations. The goal of our project is to reduce this problem by providing a ride sharing application for institutes. This will be mutually beneficial for the students providing a ride and the students wanting to reach their destination quickly and cheaply as those who bring their own vehicles anyhow have to go to their homes without anyone sharing the ride with them. This will help them to earn money to at least cover their transportation or fuel cost and in-turn help provide a cheap ride to the ones in need. In this paper, we survey the work that deals with various paradigms of ride sharing and coincides with our idea for the application.**

Keywords: **Peer-to-Peer, Pooling, Ridesharing, Routing, Shortest path**

I. INTRODUCTION

Peer pooling is a way to share transportation services. This concept has been around for quite some time now and it is used by many multinational companies like Uber, lyft etc. It is widely adopted because it not only saves fuel costs but provides a cheap alternative to getting personal rides.

Consider a scenario where one person has to go from a source to a destination. Many times another person has to travel to the same location or a second set of source and destination which is on the way to the source and destination of the first person. Likewise, many subsets of source and destination can be present. Pooling allows the driver to efficiently pick-up the riders and drop them according to the route set by the routing algorithm. There are many challenges and techniques to create the routing algorithms.

According to our goal of providing ride sharing options for university students many problems are to be tackled starting from pinpointing the source and destination, peer to peer communication, predicting fare for the ride by calculating total distance and fuel consumption and the most important optimal routing. The literature survey below addresses issues which are related to our project.

II. BUS POOLING: A LARGE-SCALE BUS RIDESHARING SERVICE

(Liu, K. et *al*, 2019). This paper deals with the problems which arise due to car sharing such as the low capacity of transport, higher costs and not being able to satisfy the demand for getting a ride. The authors have developed a bus sharing service where users can create their accounts and list demands to get a ride. They have proposed algorithms to maximize the success rate of getting rides and increase ride-matching optimization. The direct effect of this paper is reduction in use of vehicles and hence, fossil fuels. They have addressed constraints such as Departure Constraint, Time-Window constraint, Capacity constraint and Cost constraint.

Their solution approach for ride-sharing issues are:

Phase 1:

- Matching key: Where an agency reviews the requests from both drivers and riders to provide a suitable match.
- Primary search criteria: where the origin and destination is treated as two points and a linear path is drafted as a line segment.
- Routing and Time: Finding similar routes which can provide options as well as the shortest time.
- Dynamic Time Warping: This algorithm can calculate the divergence between two sequences with different phases and length.
- Keyword/List: Can search for specific keywords on a predefined list. This predefined list consists of landmarks and their distance from each other.
- Capacitated clustering: Some capacity constraints are employed according to which the demands are

clustered.

- Contraction based method, Insertion based method and Dynamic grid based methods are proposed to solve the capacitated clustering problem.

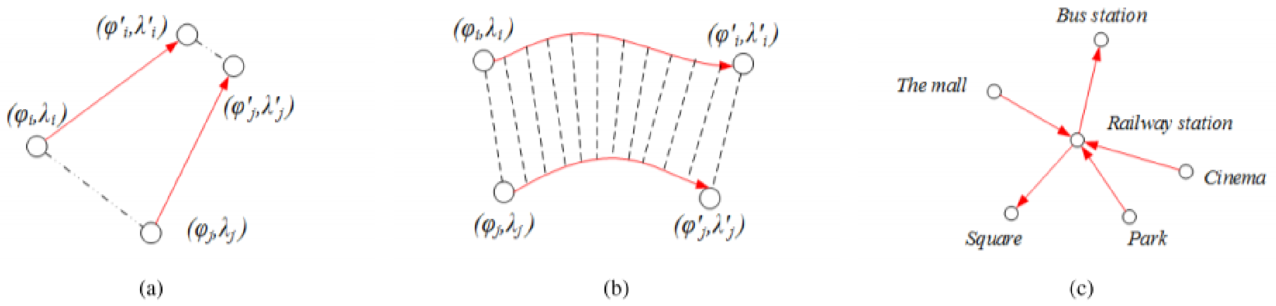

Fig 1: Primary Search Criteria. (a) OD-pair and time. (b) Routing and time. (c) Keyword/list [1]

Phase2: Deals with the pickup and delivery process and the algorithms used to solve location allocation problems.

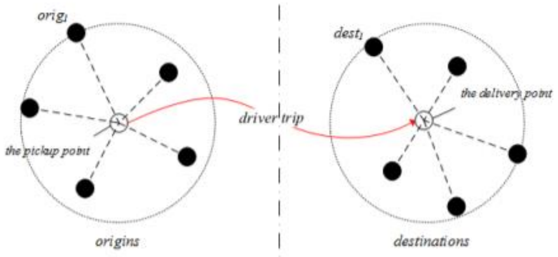

Fig 2. The Pickup/Delivery Point [1]

The whole process needs to be streamlined with minimal cost in both pickup and drop-off.

Phase 3 Pruning: In this phase, pruning is achieved using some constraints after completion of the above phases.

The results of the experiments show an economical and efficient ride sharing option with higher performance taxi service and other sharing options.

III. SMART PEER CAR POOLING SYSTEM

(Hasan et *al*, 2016). Nowadays there is rapid growth of staff and the students in Colleges. Rapid growth results in an increase of college population which results in problems like insufficient transport facilities. So the paper mainly focuses on smart peer pooling systems used by the students and staff members to travel from nearby places to college. Smart Peer Pooling system means sharing of journey among passengers travelling from the same route to the same destination. The smart peer pooling system reduces problems like traffic congestion, waiting time, fuel consumption and reduction of environmental pollution. It also helps in saving a lot of money and at the same time it also helps to earn some money. The application features include a location based station covering a specific area, same pickup and drop stations, charges based on time and distance, communication and hardware based approaches. The model uses Global Positioning System coordinates of starting and ending points and then the nodes will be calculated based on the GPS coordinates and distances in km.

When a driver will update information about a trip it will be stored in the database. Riders will be able to see this information in real time and both driver and user can communicate with each other using an application. Initially our application will be available for motorcycles where only one seat is available to ride with a driver. For future implementation of this application on cars it is important to calculate car occupancy and updating remaining seats after picking every rider.

The model uses a greedy algorithm to find the maximum passenger occupancy for the accommodation of the passengers. A greedy algorithm is an algorithm used to solve optimization problems. The algorithm uses an optimal path to find the overall optimal way to solve an entire problem. Greedy algorithms find many solutions to a problem but it chooses the best solution from all the solutions.

Steps for achieving Greedy Algorithm:

Step 1: Feasibility Study: All the constraints are checked if they are satisfying or not to obtain at least one solution to the problem.

Step 2: Local Optimal Choice: An optimum solution is selected from all the available solutions.

Step 3: Unalterable Property: After taking a decision on a particular problem the solution cannot be altered.

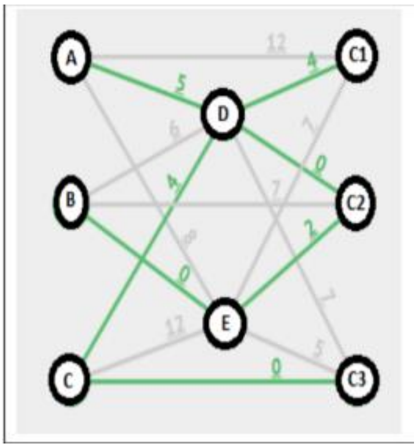

Fig 3: Maximum Possibilities to accommodate the passengers based on greedy algorithm [2].

IV. Research of Shortest Path Algorithm Based on the Data Structure

(Jinhao & Chi, 2012). When it comes to finding distance between two points or in this case two locations having multiple paths it is always better to find the shortest distance. This is where we encounter shortest path algorithms. Djikstra is one of the algorithms used to solve this prominent problem of shortest distance. Main drawback of Djikstra algorithm is when applied on a road network due to a variety of possibilities great computational power will be required. We can overcome this and optimize the algorithm by data storage and organization of data nodes. This optimized Djikstra will improve time complexity, space complexity, efficiency and data redundancy greatly compared to traditional Djikstra algorithm. It will also reduce the number of paths in analyzing the shortest distance.

Dijkstra's Shortest Path Algorithm uses following steps: -

Step 1: First an initial node is selected having current distance 0.

Step 2: Then the non-visited nodes are set with the smallest current distance using a current node.

Step 3: For every neighbor, the current distance is added along with the weight of the edge connecting the neighbors and the current node.

Step 4: Then the current node is visited.

Step 5: If there are non-visited nodes, then repeat step 2

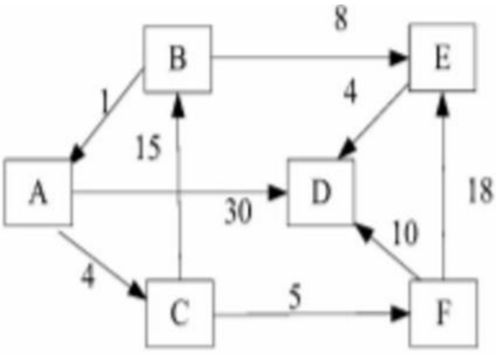

Fig 4: A directed weight graph [3].

V. Minimum Cost Spanning Tree Using Prim's Algorithm

(Abhilasha, 2013). The spanning tree of a graph is a graph that includes all the vertices in a tree. A graph may contain many spanning trees. Different spanning trees have different weights; the weight of the tree is just the sum of weights of its edges.

There are different methods to find the minimum cost of spanning tree, for example prim's algorithm, rascal's algorithm, Dijkstra's algorithm. This paper explains prim's algorithm working and its application.

Prim's Algorithm is an algorithm which helps to obtain minimum spanning tree with a greedy approach. Working principle of prim's algorithm is simple, a spanning tree means that all the vertices must be connected. So the two vertices should be connected to make a spanning tree and the weight of these edges must be minimum to make it a minimum spanning tree.

Algorithm

Step 1 - In the weighted graph remove all the parallel edges and loops

Step 2 - Select any node as starting node

Step 3 - Check for outgoing nodes from current node and select the edge with least weight

Step 4 - Repeat step 3 until all the vertices have been visited

The minimum cost spanning tree has wide real world applications like Travelling salesman problem (minimum distance to travel all cities at most once), in electronics using least wire to connect n pins by using n-1 wires. By using prim's algorithm, we find the optimal route by calculating the minimum distance between source and destination.

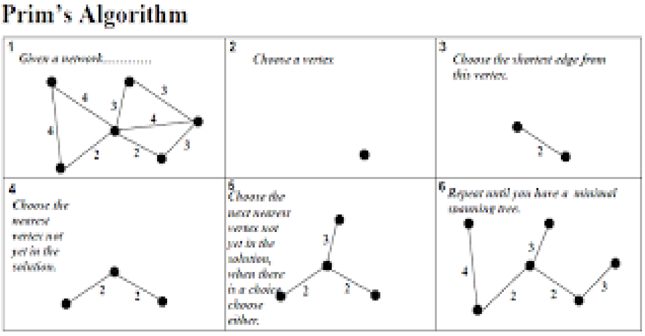
Prim's Algorithm
1 Given a network............
2 Choose a vertex
3 Choose the shortest edge from this vertex.
4 Choose the nearest vertex not yet in the solution.
5 Choose the next nearest vertex not yet in the solution, when there is a choice choose either.
6 Repeat until you have a minimal spanning tree.
4
3
3
4
4
2
2
3

Fig 5: Prim's Algorithm [4]

VI. Real-Time Carpooling and Ride-Sharing: Position Paper on Design Concepts, Distribution and Cloud Computing Strategies

(Dimitrijevic et *al*, 2013). Main aspect for any application to grow its user base is using experience of application, simple and attractive graphical interface and scalability of application. Due to different operating systems available for devices like android, IOS, windows, developers have to build indistinguishable interfaces for simplicity of the user. Although client side application can be natively written for each platform, there are always changing user interface technologies for almost every device which a developer has to overcome through an update every now and then.

A dynamic bike or carpooling application can subsequently become difficult to handle due to increased number of functions to handle and complex technology when the user base grows. Eventually there might be a need for data distribution and cloud computing strategies.
For an application like carpooling, seamless real time communication is very important for all drivers and riders to communicate and plan trips. It is important for application to update every time from the client side without consuming a lot of network resources. Any change made from any user should get registered on the database simultaneously.
To achieve this there are many technologies available and using cloud computing is one of the methods developed and is being used in taxi dispatching solutions in Serbia. This prototype was based on .NET and was cloud hosted on Microsoft azure. For this prototype Signal, an open source library for ASP.NET was used to add real time web functionality.

Following library was responsible for connecting server side and client side by updating data simultaneously in real time. This prototype was web based and incorporated on HTML5/CSS3 compared to our proposed application which is completely based on mobile application. Building a mobile application overcomes many drawbacks of web applications such as security and integrity of application. As a mobile application is verified by the app store it is more reliable and secure. As well as many features are available in mobile applications compared to web applications.

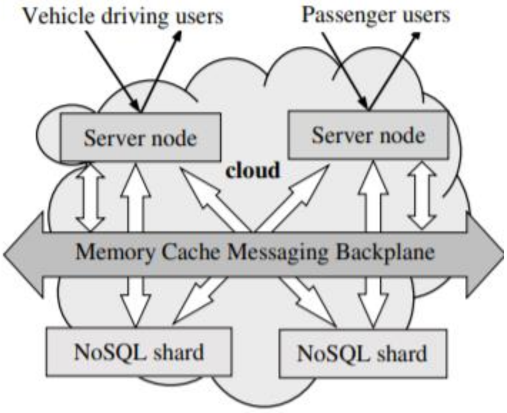

Fig 6: Basic cloud design [5]

VII. ANDROID BASED APPLICATION FOR EFFICIENT CARPOOLING WITH USER TRACKING FACILITY

(Binu & Viswaraj, 2016). Nowadays there are a lot of cars which travel over long distances with empty seats. So by bringing people in a single car who are travelling to the same destination will decrease the number of vehicles on road which will reduce pollution to a greater extent and at the same time it will also reduce stress on drivers. There are a lot of web and mobile applications for carpooling which have features like sharing journeys. But none of them provide efficiency in location tracking. So the project particularly focuses on an efficient carpooling application having some unique features like location tracking and traffic anomaly detection. The anomaly detection includes getting reports about road works, accidents and traffic in a particular area. But the anomaly can be false sometimes. So to overcome these difficulties the model uses a truth estimation technique using recursive expectation maximization algorithm. Recursive Expectation Maximization algorithm is a streaming algorithm which is usually used in social sensing applications to solve estimation problems.

The location tracking feature is used for sharing current location. With this feature driver and passenger can track each other’s location. This features also helps the user to share the location to friends and family members for security reasons. The model uses google map apis for the location tracking. The application continuously interacts with the server using HTTP connection for data management. The data like user details, trip information, location information and anomaly details are stored in a database using PostgreSQL. The model also uses OTP, email verification and Facebook integration for authentication and safety.

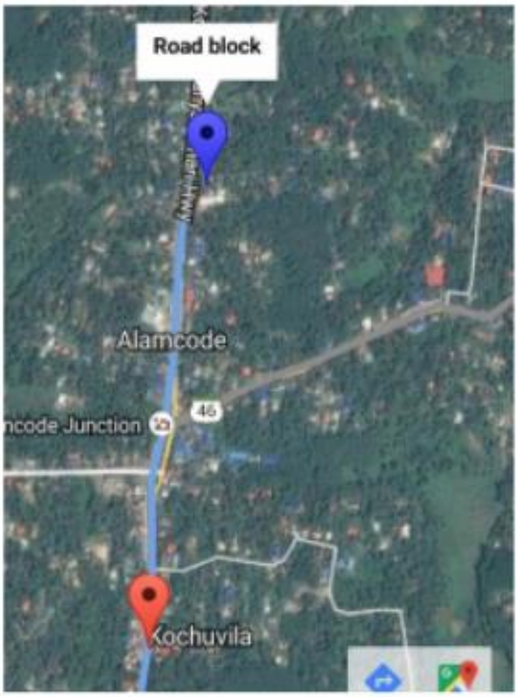

Fig 7: Anomaly Report Status [6]

VIII. ACTIVE PEER TO PEER

(Lin et *al*, 2007). Communication is key to express one's desires or to advise someone for their benefit, as such an important way of file sharing or communication is peer to peer. Peer to peer, commonly referred as p2p, is an act of sending data from sender to receiver without intermediary. Currently in a proper p2p system, knowledge about data name, hash values is a must which is burdensome as the internet is very vast and mutable. To solve this problem this paper presents a p2p system named Active P2P which converts naturally passive files into logically active ones.

The paper references several p2p systems like Pure unstructured p2p, centralized unstructured p2p, structured p2p, hierarchical p2p and semantic p2p. The concept model of this paper proposes an Active p2p system. In this system files stored in peer space are called Active files (mp3,mp4,papers) in which metadata (properties) are assigned. In the next step an agent is assigned to active files which explains the metadata to simulate competition, movement, and provide virtual lives to the data files. The properties of active files are implicit, genetic and explicit properties. In the next phase the author describes how a file will be replicated, there will be a competition in which the winner will be duplicated and the loser will be eliminated.

Active P2P is used to communicate between rider and driver. Rider will directly contact the driver for instructions like where to pick up and drop off, here active p2p is used. The communication is important because then only the parties will come to an understanding about their travel.

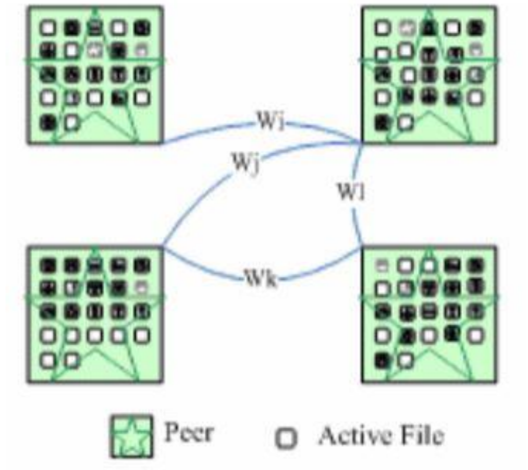

Fig 8: Cooperation network [7]

IX. Shortest Path Distance Approximation using Deep Learning Techniques

(Rizi et *al*, 2018). This paper deals with the scalability of traditional shortest path algorithms and proposes a solution of using feed forward neural netowrks to approximate the shortest path with low distortion error. The author's approach this issue by proposing the following method.

- Distance Approximation: In this method the goal is to approximate the shortest path function (d) using feed forward neural network. Here, the shortest path function is used to map a pair of vector embeddings to real-valued shortest path distance.

$$d : \Phi(u) \times \Phi(v) \rightarrow R^{+}$$

The training data pair is generated by calulating the actual shortest distance using breadth-first-search (BFS) from selected landmarks or start nodes to every other node in a graph < φ(v), φ(u) >. The extracted pair is sent as input to the neural network after performing some binary operation. The whole network consists of an input layer, a hidden layer and an output layer. Stochastic Gradient Descent (SGD) is used as the optimizer for the neural network while the quality of the model is assessed using Mean Squared Error (MSE).

- Computational Complexity: The proposed model achieves a time complexity of O(n). To learn the vector embeddings the time required is O(n) with n being the number of nodes. By choosing a constant number of landmarks, the shortest path computation is minimized and ground truth for training data is achieved. Using graph embeddings, the feed forward neural network can quickly answer a distance query between two nodes in O(1) time independent of the graph size. If a starting node *u* is present, the time taken to calculate the distance from *u* to all the other nodes is O(n).

The proposed model is tested on publicly available graph datasets like Facebook, BlogCatalog, Youtube, Flickr and achieved excellent results. It was able to find the shortest path for majority of node pairs while matching the accurate ground truth. All of the results are achieved quickly in linear time.

X. Conclusion

Hence, we study about peer-to-peer ride sharing and their paradigms. We learn about various issues related to ridesharing such as finding the shortest path, p2p communication, optimal routing and the methods proposed to solve them. In addition, we discuss about a distance approximation method using deep learning. We hope to encompass and significantly improve all of the techniques discussed in this paper in our upcoming project "Pool" which is a peer-to-peer ride sharing application for university students and teachers.

XI. References

[1] Liu, K., Zhang, J., & Yang, Q. (2019). Bus Pooling: A Large-Scale Bus Ridesharing Service. IEEE Access, 7, 74248–74262. doi:10.1109/access.2019.2920756

[2] Hasan, R., Bhatti, A. H., Hayat, M. S., Gebreyohannes, H. M., Ali, S. I., & Syed, A. J. (2016). Smart peer car pooling system. 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC). doi:10.1109/icbdsc.2016.7460384

[3] Jinhao Lu, & Chi Dong. (2012). *Research of shortest path algorithm based on the data structure. 2012 IEEE*

International Conference on Computer Science and Automation Engineering. doi:10.1109/icsess.2012.6269416

[4] R Abhilasha, Minimum cost spanning tree using prim's algorithm. Volume 1, Issue 1, June 2013 International Journal of Advanced Research in computer.

[5] Dejan Dimitrijevic, Vladimir Dimitrieski, N. Nedic (2013), Real-time carpooling and ride-sharing: Position paper on design concepts, distribution and cloud computing strategies Computer Science 2013 Federated Conference on Computer Science and Information Systems.

[6] Binu, P. K., & Viswaraj, V. S. (2016). Android based application for efficient carpooling with user tracking facility. 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC). doi:10.1109/iccic.2016.7919536

[7] Lin, X., Huai, J., Liu, X., & Han, J. (2007). Active Peer to Peer. Sixth International Conference on Networking (ICN'07). doi:10.1109/icn.2007.17

[8] Rizi, F. S., Schloetterer, J., & Granitzer, M. (2018). *Shortest Path Distance Approximation Using Deep Learning Techniques. 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).* doi:10.1109/asonam.2018.8508763

[9] Craig Standing, Susan Standing & Sharon Biermann (2018): The implications
of the sharing economy for transport, Transport Reviews, DOI: 10.1080/01441647.2018.1450307

[10] Wang, X., Yang, H., & Zhu, D. (2018). Driver-Rider Cost-Sharing Strategies and Equilibria in a Ridesharing Program. Transportation Science. doi:10.1287/trsc.2017.0801

[11] Carmody, P., & Fortuin, A. (2019). "Ride-sharing", virtual capital and impacts on labor in Cape Town, South Africa. African Geographical Review, 1–13. doi:10.1080/19376812.2019.1607149

[12] Hartl, B., Kamleitner, B., & Holub, S. (2020). Take me on a ride: The role of environmentalist identity for carpooling. Psychology & Marketing. doi:10.1002/mar.21340

[13] Zhao, T., Yang, Y., & Wang, E. (2020). Minimizing the average arriving distance in carpooling. International Journal of Distributed Sensor Networks, 16(1), 155014771989936. doi:10.1177/1550147719899369

[14] Gedam, C., Sahare, M., Sachdeo, R., & Kulkarni, N. (2020). Smart Transportation Based Car Pooling System. E3S Web of Conferences, 170, 03004. doi:10.1051/e3sconf/202017003004