**Optimization Techniques in GPS location based services - Swiggy, Uber, OLA**

Ann Mariya Xaviera, Megha Joshib, Ratnadeep Palc, Bhuvaneswari Anbalagand

aMTech CSE student, School of Computing Science and Engineering, Vellore Institute of Technology, Chennai,

bMTech CSE student, School of Computing Science and Engineering, Vellore Institute of Technology, Chennai,

cMTech CSE student, School of Computing Science and Engineering, Vellore Institute of Technology, Chennai,

dAssociate Professor Senior Grade I, School of Computing Science and Engineering, Vellore Institute of Technology,

Chennai,

## *ABSTRACT: Today most of the people prefer to order food online, to book a cab, online clothes or grocery shopping. There are lots of applications available in market which performs the task of delivering door to door. These Applications uses Global Positioning based services (GPS) to locate the geometry coordinates i.e. Longitude and Latitude so that the path of traveling can be made by using the coordinates of delivery man to the Customers. The existing application uses best first search approach, the closest next customer to deliver. So we are using MongoDB (document-oriented database) as it is useful in locating the GPS location, by using JSON format dataset which contain the coordinates of longitude and latitude and can visualize the location through the coordinates in the schema of MongoDB. The goal is to optimize the response time of the services provided by delivery boys who are near to the restaurant from which the customer orders the food and can deliver it as fast as possible by applying optimization algorithm which will help the delivery boy that which path they need to follow to reach as early as possible to all the customers.*

## *Keywords* – Global Positioning System, MongoDB, Optimization techniques, Folium, Open route service, Ant route service

## INTRODUCTION

## Online restaurant aggregators on demand meal ordering platforms where diners order their favorite cravings from an array of restaurants, also due to COVID-19 people avoid to go out for dinner and online ordering apps are growing at a fast pace. The volume of meal delivery operations is rising quickly worldwide, increasing the potential for new economies of scope, scale, and density. According to Morgan Stanley, “online food delivery could grow by 16% annual compound rate in the next 5 years”. Aiming to capitalize on the market opportunity, emerging providers are investing heavily in the deployment of meal delivery networks that promise restaurants and diners a reliable, fast and cost-effective delivery process. III. LITERATURE SURVEY

## Batching and Matching for Food Delivery in Dynamic Road Networks

## Given a stream of food orders and available delivery vehicles, how orders should be assigned to vehicles so that the delivery time is minimized.

## (1) Assignment of orders to vehicles,

## (2) Grouping orders into batches to cope with limited vehicle availability.

## (3) Adapting to dynamic positions of delivery vehicles.

## Paper contains the development of an algorithm called FOODMATCH, which maps the vehicle assignment problem to that of minimum weight perfect matching on a bipartite graph. To further reduce the quadratic construction cost of the bipartite graph, we deploy best-first search to only compute a sub graph that is highly likely to contain the minimum matching.

## The Travelling Santa

## MongoDB geospatial indexing and queries to select the next city to visit at each stage of the journey. Santa’s strategy finds the closest city to Santa’s current position and sets that city to visit next in Santa’s world tour.

## At each point choose to calculate the distance between each city and write it to MongoDB for use later.

## The Meal Delivery Routing Problem

## Extensive computational experiments using instances with realistic size, geography, urgency and dynamism demonstrate that our algorithmic ideas provide solid foundations for real-world implementations.

## Varying size of order and courier sets.

## Varying travel times.

## Varying structure of courier schedules.

## Varying preparation times.

## A food ordering system with delivery routing optimization using global positioning system (GPS) technology and Google maps

## This research develops an online food ordering system and Performed heuristic algorithm to solve the so called Travelling Sales Problem (TSP) in routing optimization. The system also uses the Global Position Systems (GPS) technology in Android-based mobile phone and takes the advantage of Google Maps for coordinate-to-map solution.

## IV. PROPOSED METHOD

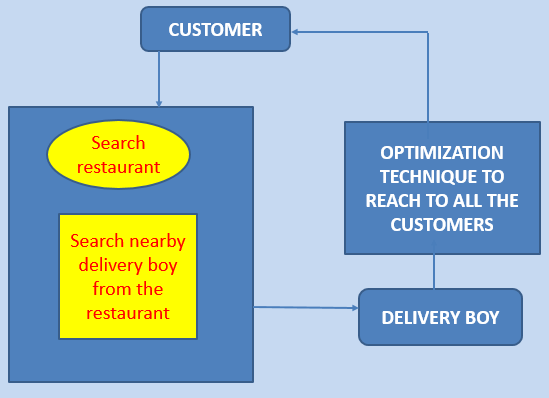
There is huge demand for real time application to locate the latitude and longitude in optimized way. Here the proposed model mainly focusing on two objectives:

1. Finding the nearest delivery boy to the respective restaurant.
2. Find the optimized path between given set of customers.

The detailed architecture of the proposed model is shown in figure 11.2. The model have mainly 3 participants:

1. Delivery Boy
2. Restaurant
3. Customers

After considering the set of customers who order the item from same restaurant we find the current closest delivery boy from the restaurant. After assigning the delivery boy to particular service we find out the best optimized path between set of customer and restaurant so that delivery boy can deliver in most efficient time period.



**Figure 11.2 Architecture of Proposed Model**

## V. IMPLIMENTATION AND RESULT

The implementation of model done through the following steps:

1. Finding the best dataset for proposed model.
2. Finding the best database to solve the problem (here we are using MongoDB )
3. Connection of MongoDB Atlas (server) to MongoDB Compas (client).
4. Connect Python (Jupyter notebook) to MongoDB Atlas.
5. Locate the GPS position of restaurants, delivery boy and all the customers.
6. Find the closest delivery boy with respect to the restaurants.
7. Find the best path to deliver to all the customers.

**Dataset Setting:** Here we had selected the yelp dataset from the online resource called kaggle. Yelp dataset is available in Json format which consist of attributes like ID, name, address, city, latitude and longitude value of location etc. Each entries in dataset where categorized as restaurant, customer and delivery boy.

**MongoDB:** For the implementation of model we had selected the MongoDB. MongoDB is a scalable, flexible NoSQL document database platform designed to overcome the relational databases approach and the limitations of other NoSQL solutions. MongoDB is well known for its horizontal scaling and load balancing capabilities, which has given application developers an unprecedented level of flexibility and scalability.

**Connection of MongoDB Atlas to MongoDB Compass:** we need to create a MongoDB host, while using MongoDB Atlas. Atlas is a cloud-hosted database-as-a-service which requires no installation, offers a free tier to get started, and provides a copy able URI to easily connect Compass to your deployment.

# **Connect Python (Jupyter notebook) to MongoDB Atlas**: To perform the processing using python on the data fetched from the database we had done connection from MongoDB to python jupyter notebook.

# **Locate the GPS position of restaurants, delivery boy and all the customers:** To locate the GPS location of the delivery boy, customer and restaurants we had fetched the latitude and longitude values of corresponding category from the database. After finding the GPS location of each we had done plotting using folium. The plotting of Restaurant on its geospatial location using folium has shown in figure 11.3.

# 

# **Figure 11.3: Locating Restaurant in map**.

# **Find the closest delivery boy with respect to the restaurants**: The closest delivery boy to corresponding restaurant is found using an algorithm called ***indexing***. An index in Mongo DB is a special data structure that holds the data of few fields of documents on which the index is created. Indexes improve the speed of search operations in database because instead of searching the whole document, the search is performed on the indexes that holds only few fields. Here we use indexing to find the location of customers, Delivery boys and the restaurant. By using the indexing instead of searching whole data we are able to reduce the time for searching to O (log^n) complexity. The MongoDB to find the nearby delivery boy corresponding to particular restaurant is given below.

*db.test.find({"geometry":{"$nearSphere":{"$geometry":{"type":"Point","coordinates":[res[0]["longitude"],res[0]["latitude"]]},"$maxDistance":100000}},"categories":"Delivery boy"}, { "name":1, "geometry.coordinates":1,"\_id":0 }).limit (1)*

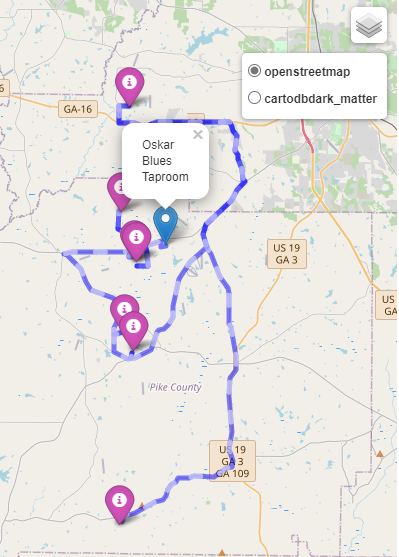
## After finding the nearby delivery boy we depicted the connection from restaurant to delivery boy in the geospatial graph using folium. The spatial connection between the nearest delivery boy and restaurant is shown in figure 11.4.

## 

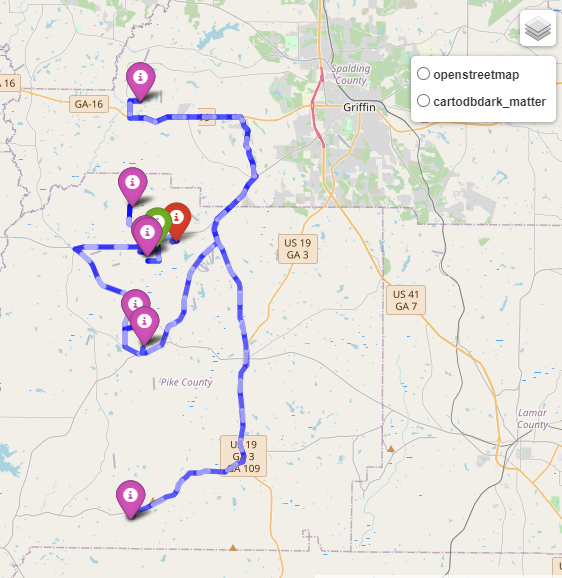
## Figure 11.4, graph denoting spatial connection between delivery boy (green marker) and restaurant (denoted using red marker).

## Find the best path to deliver to all the customers: To optimize the distance travel by the delivery boy to deliver item to all the customers, we had perform shortest path algorithm to find optimized path among the customer and restaurant. To find the optimized path Dijkstra's algorithm. Dijkstra's algorithm is one of the most efficient algorithm to find the most effective path among the set of coordinates. Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks. The algorithm exists in many variants. Dijkstra's original algorithm found the shortest path between two given nodes,[[6]](https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm" \l "cite_note-Dijkstra1959-6) but a more common variant fixes a single node as the "source" node and finds shortest paths from the source to all other nodes in the graph, producing a [shortest-path tree](https://en.wikipedia.org/wiki/Shortest-path_tree).

## After performing the shortest path algorithm to find the optimized path we had plotted the route of delivery boy from his current location to restaurant and to the set of customers using folium. The optimized route obtained is plotted on a geospatial graph which is depicted in figure 11.6.

****

**Figure 11.5, Optimized route between restaurant (red marker) and to the set of customers (violet marker)**

****

**Figure 11.6, Optimized route of delivery boy (green marker) to restaurant (red marker) and to the set of customers (violet marker)**

## VI. CONCLUSION AND FUTURE WORK

The proposed strategy can be utilized to find out the optimized path between the restaurant, delivery boy and customers. But the model will not provide a path which is optimized according to the time series. The future work of the model is to provide optimized path for a particular time series. The future work may include to develop a user defined application.

## VII. REFERENCES

## [1] [Manas Joshi](https://arxiv.org/search/cs?searchtype=author&query=Joshi%2C+M), [Arshdeep Singh](https://arxiv.org/search/cs?searchtype=author&query=Singh%2C+A), [Sayan Ranu](https://arxiv.org/search/cs?searchtype=author&query=Ranu%2C+S), [Amitabha Bagchi](https://arxiv.org/search/cs?searchtype=author&query=Bagchi%2C+A), [Priyank Karia](https://arxiv.org/search/cs?searchtype=author&query=Karia%2C+P), [Puneet Kala](https://arxiv.org/search/cs?searchtype=author&query=Kala%2C+P) “Batching and Matching for Food Delivery in Dynamic Road Networks” August 29,2020

## [2] “The Travelling Santa” blog by Brian Reinero

## [3] The Meal Delivery Routing Problem Damian Reyes1, Alan Erera1, Martin Savelsbergh1 , Sagar Sahasrabudhe2 , and Ryan O’Neil2 1H. Milton Stewart School of Industrial Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 2Decision Engineering Department, Grubhub, Chicago, IL 60602 March 1,2018

## [4] Roy Deddy Hasiholan Tobing “A food ordering system with delivery routing optimization using global positioning system (GPS) technology and Google maps” published by Research Gate in January 2016