2022

TEAM INSIDERS

A SOLUTION FOR EV LOCATION PROBLEMS WITH UNIFIED MANAGEMENT SYSTEM SOFTWARE WHICH HANDLES THE PROBLEM OF DEMAND AND SUPPLY

TABLE OF CONTENTS

ABBREVIATION AND TERMINOLOGY:

ABBREVIATION	FULL FORM
QGIS	Quantum Geographic Information System
AOI	Area of Interest
GHMC	Greater Hyderabad Municipal Corporation
OSM	Open Street Map
GDP	Gross Domestic Product
OF	Occupancy Factor
RIF	Route Increment Factor
PIF	Proximity Increment Factor
Charge Anxiety	The fear of driving an EV and running out of power
Celestial Admin	Government personnel who approves the requests for infrastructural upgrade
Zonal Admin	A government employee who watches over EVs in that Zone
EV traffic index	RIF is added to the grids attribute which is EV traffic index

DETAILED INFORMATION ABOUT THE SOLUTION

Our solution revolves around the breaking down of the Target city, with a population of more than 4 million into an array of small squares, the basic unit of which is a fixed length. This is done by using Quantum Geographic Information System or QGIS.

What is QGIS? QGIS is a free and open-source geographic information system that supports most geospatial networks and raster file types and database formats. We can also automate analysis using the python inbuilt console.

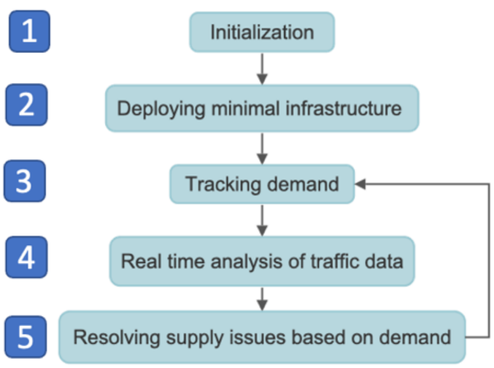
It provides a Feature called Grid Network, which generates a network of squares over the Area of Interest. Every Grid in this network acts as an object and a unique id is generated while creating, these Grids can have user-defined attributes which make the Grid network a powerful feature in our project for analysis. We can also generate heatmaps using an attribute in the grid on our AOI.

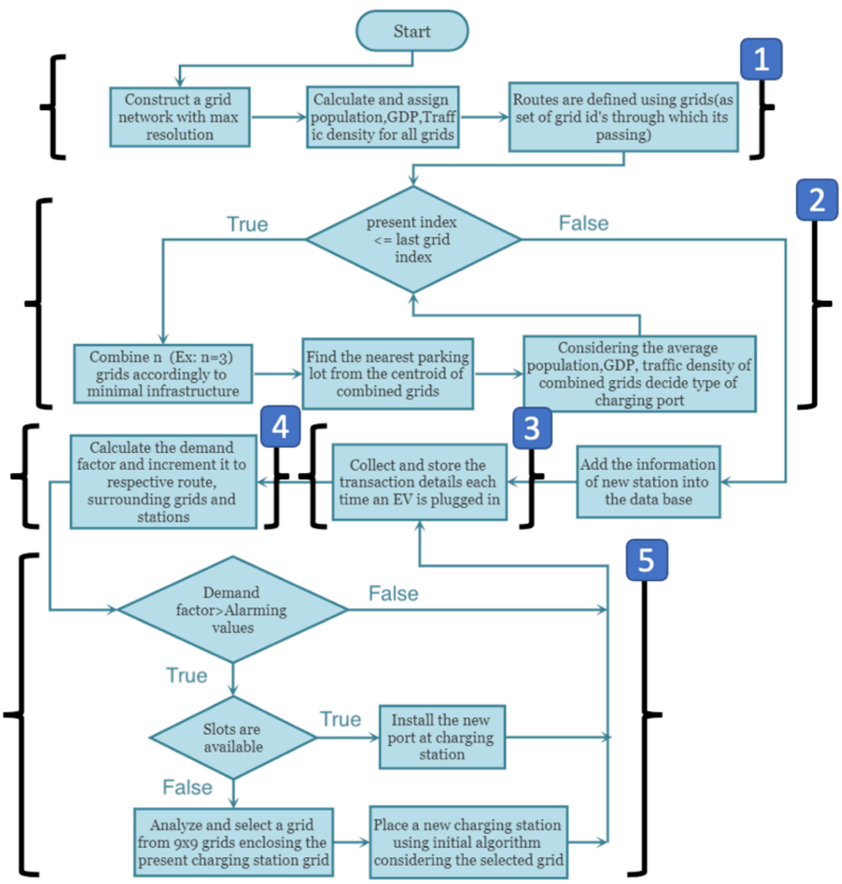
SOLUTION

Our main aim is to reduce the Charge Anxiety which is caused by the lack of availability of EV charge stations in key areas and to solve the Demand and Supply issue which is the core of the problem.

It's not possible to allocate new space and create infrastructure for EV charging for all locations as it can be costly. Govt and the Ministry of Housing and Urban affairs should make it mandatory to reserve slots at parking for EV charging stations.

Steps Involved in Our Solution-





INITIALIZATION:

Initially, the data required is procured from the government, this includes the data like population density, parking areas, availability of land, vehicle data etcetera. At every grid, we define attributes as parameters that affect the EV charging station locations, convert raw data to analyzable data. This is Initialization.

We chose Hyderabad as our test city, a metropolitan city, the administrative boundaries are obtained from Greater Hyderabad Municipal Corporation or GHMC's website. Following the interpretation of Hyderabad's administrative boundaries, we've broken the city down using a Grid network of Squares with a side length of 666.67m initially. Which is the max resolution for our analysis of AOI.

We would be combining n x n grids Initially according to the minimal infrastructure defined at the convenience of the Celestial Admin (Government Authorized Personnel). For example, say 1 station per every 4 km^2 we combine 3x3 square grids and virtually generate a Network of combined grids using the algorithm. In this Solution part, we will be considering the minimal infrastructure as 1 station per every 4 km^2 for simplicity.

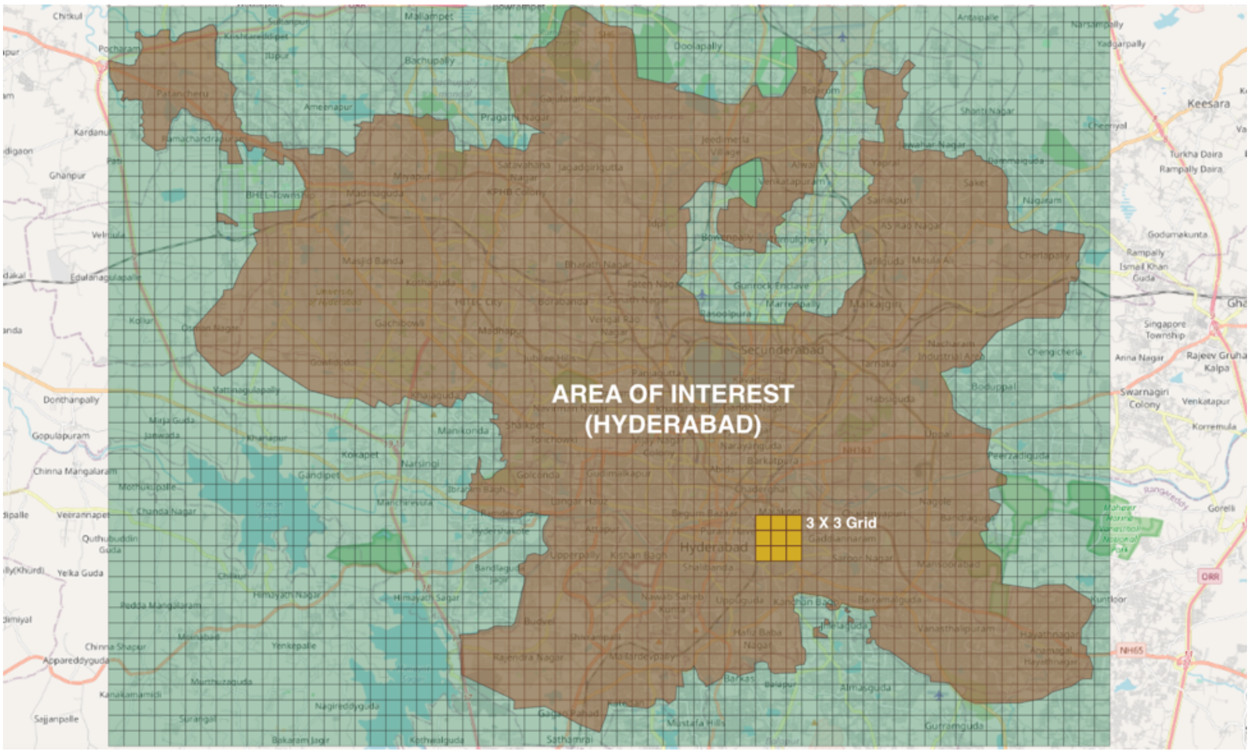


Figure 1. Grid Network of 666mX666m laid over Test City(Hyderabad)

Now we will be analyzing the raw data of roads to extract traffic density from them. The roads shapefile was downloaded from bbbike.org which extracts roads data from OSM which is not directly accessible. Upon our research, we have concluded that traffic density and the number of intersections are directly proportional in a grid. We used a tool in QGIS to get the intersections shapefile.

Intersections of roads in that grid ∝ Traffic density

Similarly, population density has been analyzed using the building polygon shapefile and concluded that population density and building density are directly proportional. We considered the number of building centroids lying in that grid.

Number of buildings in that grid ∝ Population density

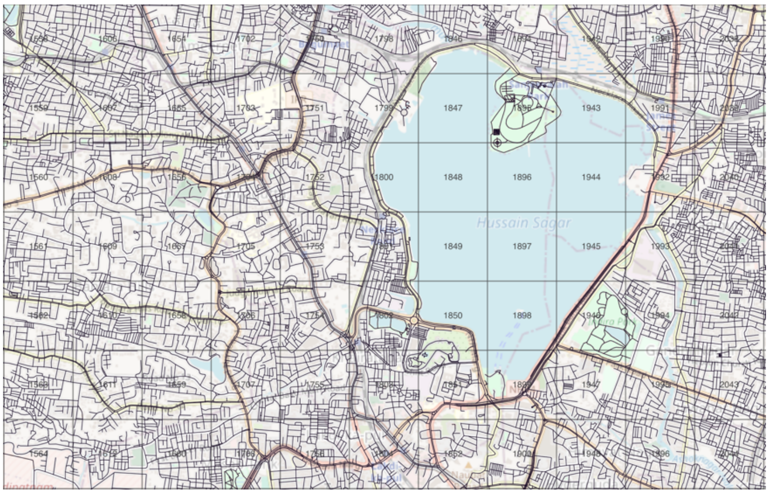


Figure 2.1. Road shapefile layer over AOI

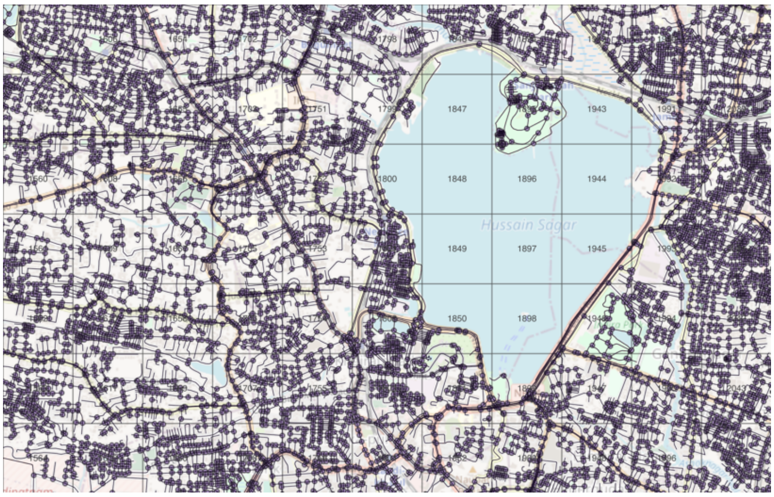


Figure 2.2. Cleaned and Reprojected Road intersections layer over AOI

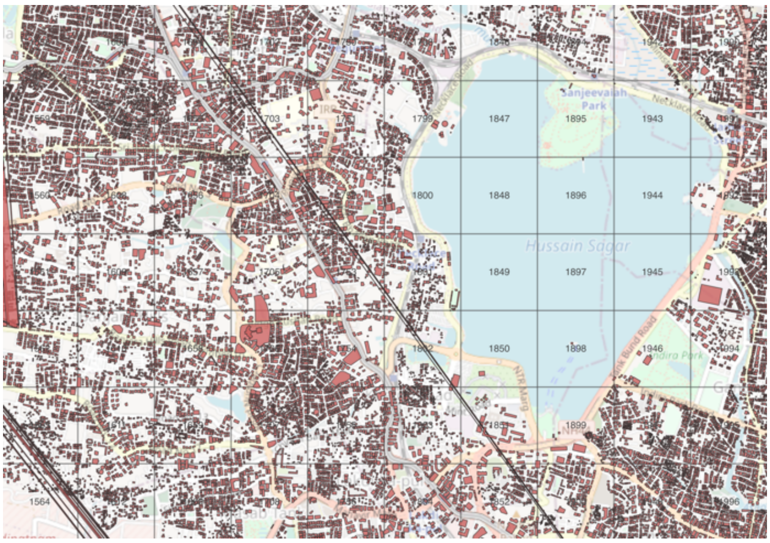


Figure 3. Building polygon shapefile layer over AOI

	fid	id	left	top	right	bottom	EvTraficIndex	trafic_density	opulation_densit
1	2229	2229	8740159.656...	1972874.784...	8740826.32...	1972208.117...	NULL	170	1919
2	1587	1587	8731492.990...	1984874.784...	8732159.656...	1984208.117...	NULL	241	1652
3	1376	1376	8728159.656...	1965541.450...	8728826.32...	1964874.784...	NULL	158	1637
4	2234	2234	8740159.656...	1969541.450...	8740826.32...	1968874.784...	NULL	183	1568
5	2463	2463	8743492.99...	1976874.784...	8744159.656...	1976208.117...	NULL	171	1525
6	2277	2277	8740826.32...	1972874.784...	8741492.990...	1972208.117...	NULL	182	1508
7	2634	2634	8745492.99...	1958874.784...	8746159.656...	1958208.117...	NULL	166	1487
8	2286	2286	8740826.32...	1966874.784...	8741492.990...	1966208.117...	NULL	183	1462
9	734	734	8719492.990...	1977541.450...	8720159.65...	1976874.784...	NULL	137	1417
10	1261	1261	8726826.32...	1978208.117...	8727492.990...	1977541.450...	NULL	144	1411
11	2186	2186	8739492.99...	1969541.450...	8740159.656...	1968874.784...	NULL	279	1401
12	2373	2373	8742159.656...	1972874.784...	8742826.32...	1972208.117...	NULL	165	1380

Figure 4. The final attribute table of the grid after adding traffic density and population density

Routes are defined as a set of Grid ids through which it is passing as shown below. This is the most effective and advanced way of defining routes. This allows us to analyze parts of routes effectively using grids.

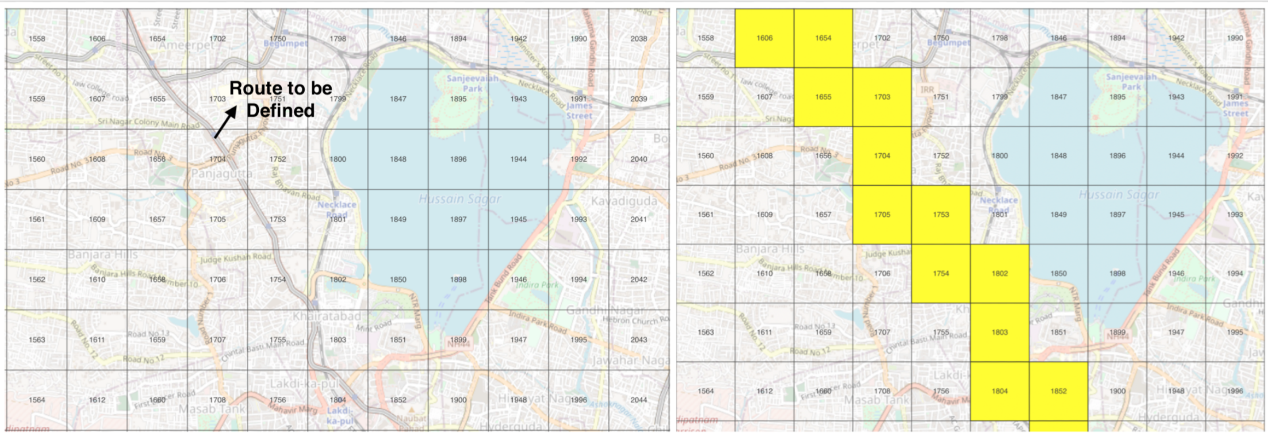


Figure 5. The final attribute table of the grid after adding traffic density and population density

DEPLOYING MINIMAL INFRASTRUCTURE:

When we're given the locations of the existing EV charge stations we use the given data to find an optimal location for the placement of new stations for meeting the basic needs to Increase the demand, decreasing the charge anxiety. This step can be categorized as Deploying Minimal Infrastructure.

For minimal infrastructure, we must make sure that there is at least one charging station in every combined virtual grid. Consider the parking lots nearest to the centroid first, if it is not available in that grid then search for free spaces nearest to the centroid. The shortest distance and GDP parameters are used to determine the location of the new station Now for deciding what type of charger ports are used in the parking slots we consider the average population, traffic, GDP densities of that virtual grid.

NOTE: If a charging station already exists in that virtual grid then skip that grid.

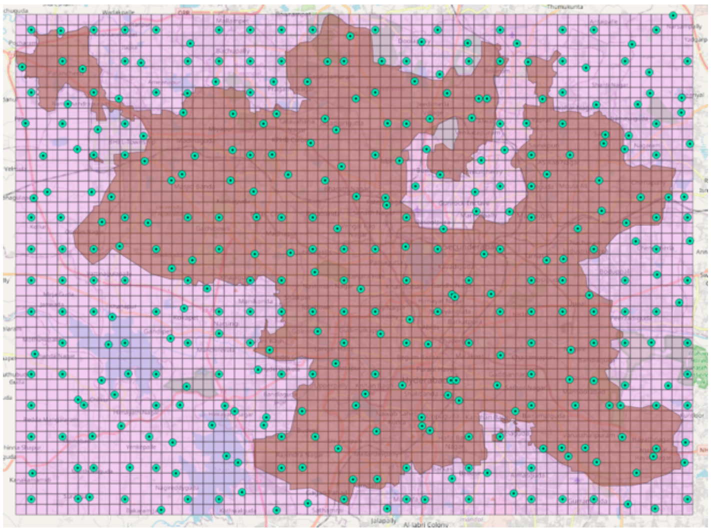


Figure 5. Charge station locations are automated using random parking and free lots

TRACKING DEMAND:

Now, the new stations are deployed. In addition to power transfer, data related to battery status, vehicle ID code, registration number or emergency messages data can be simultaneously transferred when a port is connected. This is used to adjust the charging current depending on the battery status, To display the charging status at the user level app.

car_number	stationid	start_charge	end_charge	charge_kwh	amount	transaction_id	model_id	date	time
TS01AB1234	2	20	70	20	100	2	901	NULL	NULL
TS22BH9999	1	30	90	25	150	102	901	NULL	NULL
TS01AB1234	2	20	70	20	100	103	901	NULL	NULL
AP15AD2500	2	25	30	100	500	104	1	NULL	NULL
MH20BH2002	4	10	90	20	100	105	1	NULL	NULL
TS22GH2003	1	15	87	150	750	106	1	NULL	NULL
TS20GH2001	1	20	80	100	500	107	1	NULL	NULL
TS01AB1234	2	20	70	20	100	110	901	NULL	NULL
TS20BH2020	1	10	90	100	500	111	1	NULL	NULL
AP20JJ9999	1	15	80	120	600	112	1	NULL	NULL
TS09AG3519	2001	10	80	100	500	113	1	NULL	NULL
ts20bh2012	2001	10	50	30	100	114	1	NULL	NULL

transactions 1

Figure 6.1. Database format of charge transaction details stored in real-time

We would be collecting vehicle charge status in real-time in a format as shown below, in the database

{Station ID, Transaction ID, Vehicle number, Charge time, Energy charged, Bill details}

Database to store information of charge stations, EV models for real-time analysis, Grids database is maintained for incrementation of demand factors(like EV traffic index)

model_id	model_name	maximum_range	charge_time_50	charge_time_100
901	Mahindra SUV	270	20	60
NULL	NULL	NULL	NULL	NULL

Figure 6.2. Database of Information about EV models procured from govt

grid_id	range_left	range_right	range_top	range_bottom	ev_traffic_index	traffic_density	population_density	routes_covered_id
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 6.3. Database of Grid's attributes

name	latitude	longitude	stationid	location	gridid
ABC Charging Station	71.5	78.2	101	NULL	1
CDE Charging station	NULL	NULL	102	NULL	4
BCD Charging station	71.5	78.2	103	NULL	3
DEF Charging station	NULL	NULL	104	NULL	4
EFG Charging station	NULL	NULL	105	NULL	1
FGH Charging station	NULL	NULL	106	NULL	2
GHI Charging station	NULL	NULL	107	NULL	2
HIJ Charging station	NULL	NULL	108	NULL	3
NULL	NULL	NULL	NULL	NULL	NULL

Figure 6.2. Database of all charge stations

```
class Grid:
    grid_id = None
    range = {}
    ev_traffic_index = None
    traffic_density = None
    population_density = None
    building_density = None
    routes_covered = []
    stations_covered = []
    demand_factor = []
```

```
class Station:
    station_id = None
    grid_id = None
    location = {'latitude': None, 'longitude': None}
    charge_list = []
    installed_capacity = None
    energy_per_day = None
    ports_count: Dict[str, int] = {'BHARATDC': None, 'BHARATAC': None, 'CHADEMO': None, 'CCS': None}
    cumulative_ports_charge_time = {'BHARATDC': None, 'BHARATAC': None, 'CHADEMO': None, 'CSS': None}
    demand_factor = None
```

```
class Route:
    route_id = None
    covered_grids = []
    included_stations = []
    source_grid = None
    destination_grid = None
    ev_traffic_index = None
```

```
class VehicleModel:
    model_id = None
    model_name = None
    mileage = None
    charge_time_50 = None
    charge_time_full = None
```

```
class Vehicle:
    vehicle_number = None
    model_number = None
    model_name = None
    avg_mileage = None
    expected_mileage = None
    frequent_grids = []
    frequent_route = None
    expected_charge_time = None
    avg_charge_time = None
```

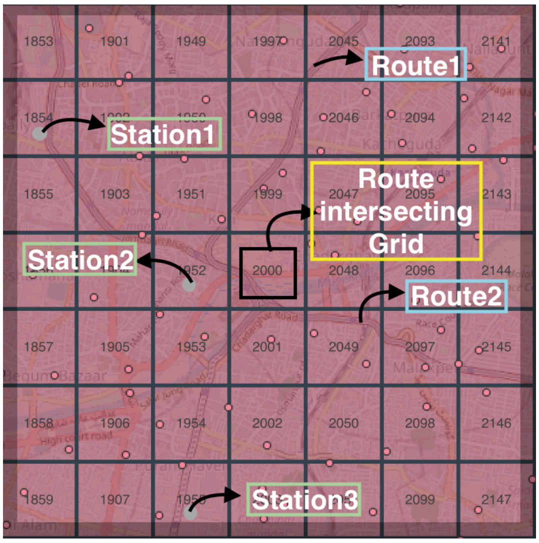
```
class Charge:
    charge_id = None
    vehicle_number = None
    charge_time = None
    start_charge = None
    end_charge = None
    energy = None
```

Figure 6.3. Objects created in Python Backend

REALTIME ANALYSIS OF DATA

Now we have real-time vehicle data using which we analyze and find out different types of Demand factors:

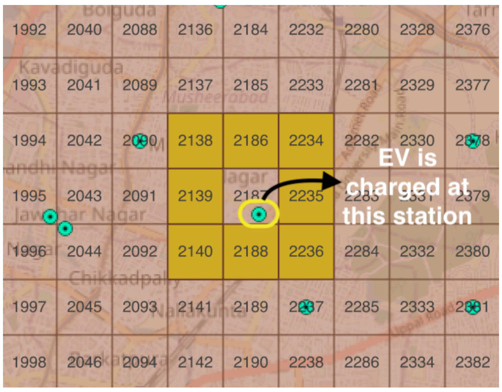
1. OF or Occupancy factor: It is the measure to calculate the occupancy of a particular port at a given station at the end of each day, which also helps us find the occupancy of a particular station considering the sum of individual Occupancy factors of all ports in the given station



2. RIF or Route increment factor (EV traffic index): When a vehicle is charged at a particular station, by accessing its grid_id we increment the corresponding routes which pass through that particular grid by a factor called route increment factor

In the given picture, consider three existing stations namely station 1, station 2 and station 3 and the routes passing through those particular grids namely route 1 and route 2, say 100 vehicles are charged at one of these three stations the corresponding routes passing through that grid in which the station (route 1 or route 2) EV traffic index in grid gets incremented, say by a factor of 100x where x is the route increment factor of that station, when all the grids passing through the routes increase by a factor of 100x, grid with grid_id 2000 will increase by a factor of 200x as it is common to both the routes, which is the highest priority grid when we do grid level analysis to place a new station. Hence, grid with grid_id 2000 is an optimal location to install a new EV charge station as it is most frequently travelled by EVs to charge at the particular station. This way Grid analysis is more efficient than normal analysis.

Figure 7.1. Intersection analysis using Grid



3. PIF or Proximity increment factor: Surrounding grids are incremented by a factor every time a vehicle is charged at that station, this is because the vehicle may come from the surrounding locality instead of the public transport routes through the grid, this factor is called the Proximity Increment Factor.

Figure 7.2. Intersection analysis using Grid

RESOLVING THE SUPPLY ISSUES:

Now, we can track the demand for EVs using demand factors. Thereafter, we need to address the supply issue. This can be resolved in two ways, one by increasing slots at the Station level and the other by placing a new station in that locality.

If the demand factor at the station level crosses the alarming value then an automated prompt will be sent to the zonal admin for upgrading the present infrastructure, by increasing the number of ports at that station and can send a request to the celestial admin at the ease of a click in prompt. This can be done until there is space left in allocated car slots for EV charging. Even then if the demand factor doesn't decrease then the zonal admin can send a request for a new charging station.

If the celestial admin approves the request for a New charging station then the demand factor(EV traffic index) gets fetched to attribute tables in QGIS from the database. Then Considering the demand factors of stations in the locality of the target station, optimal locations for placing a new charge station would be sent to the celestial admin, from the UI celestial admin can select a location for a brand new station from at most top 3 optimized locations.

For placing a new charging station we would be considering an $n^2 \times n^2$ grid that encloses the present target grid at its Centre, it is bound to encounter at least a few stations that also might have demand factors closest to alarming values and then we would find which factors that are to be preferred for optimal location, in this case, proximity to all the high demand stations followed by others, and is sent to a Machine Learning based optimization algorithm to find the best grid where the new station can be placed. Now using the initialization algorithm we find the 3 best locations to place a charging station using the grids which are selected by the machine learning model.

UI FOR OUR SOLUTION

User Type: Celestial Admin (Government authorized official):

- Can define the minimal infrastructure (say 1 station per 4km^2)
- Can view the real-time traffic and energy usage insights at a Station/ Area / Grid / Route.
- Can approve the request from zonal admin for infrastructural changes

User Type: Zonal Admin (Authorized personnel of charge station)

- Can view the Usage and occupancy statistics, energy insights of each slot in that station.
- Will get prompted, to send a request for infrastructural development when daily demand crosses a threshold value.

User Type: Vehicle Owner- Common People (Future Scope):

- Can view the availability of slots and schedule a charge at any station in his/her way.

In UI users can also view the real-time traffic and energy usage statics, traffic density, population density as a heatmap at any of the Station/Area/Grid/Route levels, can also get statistical tables.

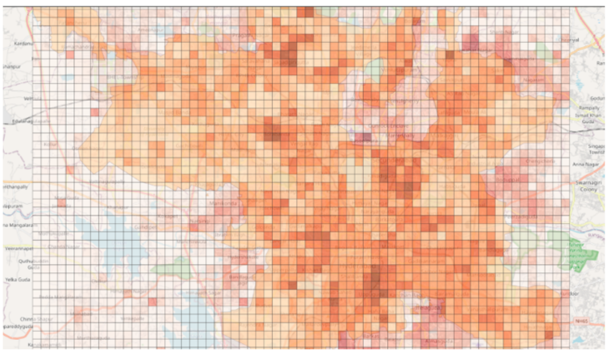


Figure 8.1. Heat map of Traffic density(Road Intersection).

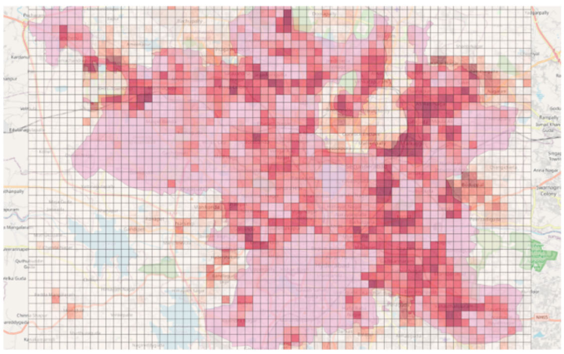


Figure 8.2. Heat map of Population density(Buildings)

CONCLUSION

In conclusion, this is the detailed approach to the solution, which we proceeded through which would solve the issues of Charge Anxiety, Demand and Supply issues the government would face if it wanted to increase the EV charging infrastructure in a metropolitan city.