EPiC Series in Engineering

Volume 2, 2018, Pages 206–217

SUMO 2018- Simulating Autonomous
and Intermodal Transport Systems

Dynamic Route Optimization for Heterogeneous Agent Envisaging Topographic of Maps

Muhammad Shafiq[1,2*]

[1] Bio-inspired Simulation & Modeling Lab

[2] Information Technology University, Lahore, Pakistan.

cmshafiq@gmail.com

Abstract

Dynamic Route Optimization is a generic problem for the commuter traveling diagonally in the smart cities; complex road network poses challenges for the heterogeneous agents to opt for route from source to destination. In smart grid of road network where intersections, roundabouts, footpaths, pedestrian bridges and tunnels having variant topographic features as a result route optimization create diversity. In various part of the cities where gridlock observed, consequently routing application recommend a route where a grid of intersecting streets completely congested where no vehicular movement is possible. In the paper we explored that how to enhance the utility of an existing application, prevent the gridlock affects and non-deterministic delay by considering topographic features of road networks using optimal shortest path routing algorithm Dijkstra. For this purpose, instituting a profile of Agents and feature recording of road network, e.g. height, width, speed and capacity is a prerequisite. Augmentation of Dijkstra algorithm according to the topographies of a Heterogeneous Agent, road network and agent simulation using SUMO (Simulation of Urban Mobility) in real time environment.

Commuter described the intricate parameter of an agent, e.g. height, width and capacity at the time profile creation, recall the parameter while devising route and its calculation of alternative preferences. Functional utility authenticated and compared with the existing application, e.g. Google Maps, OSM (Open Street Maps)

Keywords: Dynamic Route Optimization, Dijkstra's Algorithm, SUMO (Simulation of Urban Mobility), topographic, Road Network, Heterogeneous Agent

[*] MS Information Technology University Lahore

1 Introduction

Dynamic Route optimization for heterogeneous-agent using shortest path (Dijkstra) algorithm and augmentation of algorithm by consuming existing road network and envisaging the topographies of maps. Classify the limitation and test the real time utility of routing algorithms by simulating the behavior of an agent in real time environment.

Navigation genus applications are quite prevalent for route planning especially where the commuter are less familiar with the road network. Applications illustrated a path with alternative preferences from source s to destination t without envisaging the structures and the capacity of a road, resultantly various agents trapped inside the tunnel, underpasses and made gridlock for the entire region, causing delay in school, office and in case of emergency suffering lives. In order to prevent the loss of precious lives and avoid undetermined delay causing due to the gridlock affects and ignoring of topographic specifics of road networks, we developed a profiling base agent system, commuter should describe the intricate parameter of an agent, e.g. height, width and capacity at the time profile creation. Application will recall the parameter while devising route and its calculation of alternative preferences.

Dijikstra algorithm were cautiously picked for the augmentation and solving the cited problem due to its optimal performance. OSM (Open Street Map) were extracted and recycled for road network, apiece junctions were denoted as vertex and boulevard as edges while building a graph of network. Correspondingly, the parameters of each edge, e.g. height, width and capacity in addition to the length as weight defined for traversing the road network to building a route to avoid the gridlock. Various tools are available for the simulation and modeling of Multi Agent, e.g. SUMO (Simulation of Urban Mobility), MATSIM, both platform provides the simulation of transportation and road network, SUMO were used due to its freeware availability.

2 Background

Finding an optimal and feasible path from source to destination is the core function of the proposed algorithm, currently available application discovered a route regardless the agent specifications, serious accidents were being reporting across the town that number of lives were lost in such accidents where navigating application failed to detect the topographies of the landmark. To preclude such calamities and humanoid fault we augmented the Dijkstra’s shortest path finding algorithm based on agent and road network profiling.

3 Related Work

Route planning gains significant importance in the researcher’s realm since shortest path algorithm conceived, research work contributed in line to make profitable and time dependent route planning. The (Lee & Yang, 2012) purpose of the research was to design and develop crossbreed methodology for rerouting that can cope with dynamic traffic situation using huge network based on Dijkstra and genetic algorithm, installed in a car navigation system for re-routing with the transmitted of traffic information.

Anticipated a centralized algorithm, (Sun, Xie, Chen, Lu, & Chen, 2014) based on *Minimum-Cost Maximum-Flow Routing* (MCMF-R), for the route planning, MCMF-R premeditated heavy traffic centered in rush hours. With a central control unit, the routing of MCMF-R can make full use of global traffic information. Central (MCMF-R) and distributed traffic Split (TS) methodologies developed to mitigate the probability of communication and computational overhead. By coalescing complementary MCMF-R and TS developed a hybrid framework for efficient route planning.

Shortest path Algorithm Progression (Bast, Delling, & Goldberg, 2016) for route planning in transportation and road networks analyzed, compute driving directions in milliseconds or less even at continental scale, techniques variation deliver dissimilar trade-offs between preprocessing effort, space requirements, and query time. Algorithms response queries in a fraction of a microsecond, while some deal efficiently with real-time traffic.

Dynamic route guidance system (Dong, 2011) contained centralized and decentralized decision system. The precision of information exchanged between vehicle and centralized system information center guaranteed whereas the second one is based on estimated travel time to make route decisions operated within the vehicle unit. Centralized route guidance system architecture studied, e.g. (Cameron, Regnier, Galloy, & Savoie) discussed Northern Telecom's dynamic alternative routing system, which was a centralized routing system and selected routes based on probabilities. In another study by (MAHMASSANI, 1996), examined a method to provide real time route guidance in congested road networks. Distributed route guidance algorithm operated to insignificant congestion effect and tiny network, was anticipated by (JenniferFarver & IsmailChabini, 2003), however, the most prevalent systems was centralized was deployed in vehicles.

Search engine used for route planning and finding an optimal means of travelling between two or more given locations, searches may be optimized on different criteria, for example fastest, shortest, perhaps restricted for example to leave or arrive at a certain time. Route and Journey planning are distinctive where route planning is more specific for mode of transportation such as driving, walking, or cycling would make use of at least one public transport mode, which functions according to available plans.

4 Empirical Model

This section illustrated the empirical model of Dijikstra's algorithm that computes the shortest path from source *s* to destination *t* in an edge-weighted graph with non-negative weights *u*. Let *V* be the set of vertices of Graph, Primarily vertices marked as unvisited and creates a set of *unvisited vertices,* set distance to zero for source vertex and to infinity for all other vertices. Set the source vertex *s* as current and consider of all its unvisited neighbors for computing the *tentative* distance, set the lesser value whereas calculating the latest one. Mark the in-progress vertex as visited and eliminate it from the *unvisited set* after considering all its neighbors. Set as current from unvisited vertices with smallest tentative distances after recursive iteration marked as visited. Destination *t* marked as visited if path exist otherwise *Unvisited set* as infinity.

```
function Dijkstra(Graph, source):
dist[source] ← 0
Q←V
for all v ∈ V
        dist[v] ← INFINITY
        prev[v] ← UNDEFINED
           S←∅
While Q≠∅
        do u ← vmindistance(Q,dist)
                                Remove u from Q
        S←S∪{u}
                  for all v ∈ neighbors[u]
   alt ← dist[u] + length(u, v)
         if alt < dist[v]:
                dist[v] ← alt
                prev[v] ← u
    return dist[], prev[]
```

Algorithm 1: Dijkstra Algorithm 1

- Initially set source distance, dist[source]=0, Visited Node S = ∅, and dist[v]=INFINITY for all other vertexes v. One after another-select vertexes from V to add to S.
- Dijkstra's algorithm utilities by retaining a subset of vertices, $S \subseteq V$, for which the distance is deterministic that is dist[u] = length (u,v)
- The length of a path p=⟨v0, v1,….,vk⟩ is the sum of the weights of its constituent edges:

$$\mathbf{length}(P)=\sum_{i=1}^{K}(V i\text{-}1, V i)$$

- Length(⟨a,d,e,f,h ⟩)=50
- G distance from a to h 50

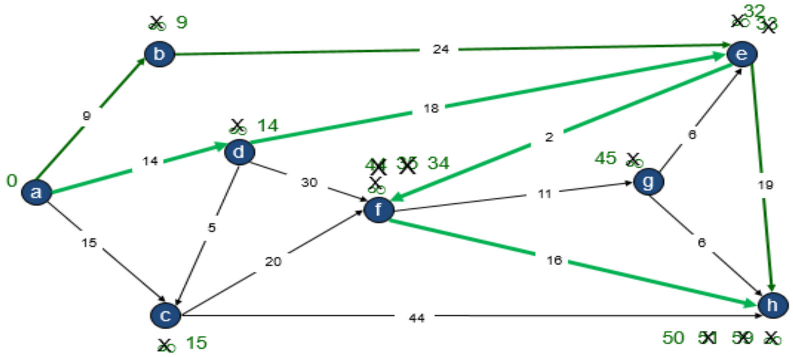

Figure 1: Mathematical Representation 1

An adjacency matrix used to represent a digraph shown in table1, the elements of the matrix indicate whether pairs of vertices adjacent or not in the graph. Infinity symbol means that there is no connection between the vertices.

	a	b	c	d	E	f	g	h
a	0	9	15	14	∞	∞	∞	∞
b	∞	0	∞	∞	24	∞	∞	∞
c	∞	∞	0	∞	∞	20	∞	44
d	∞	∞	5	0	∞	30	∞	∞
e	∞	∞	∞	∞	0	2	∞	19
f	∞	∞	∞	∞	∞	0	11	16
g	∞	∞	∞	∞	6	∞	0	6

Table 1: Adjacency Matrices Digraph Table for Figure 1

5 Augmented Model

This section demonstrated the augmented model of Dijkstra's algorithm that computes the shortest path from source s to destination t in an edge-weighted graph with non-negative weights for heterogeneous agents keeping topographic features in memory. The compatibility and feasibility of road network and of heterogeneous agents should be determined while before predicting a route in order to ensure interruption free drive. Dijkstra's algorithm augmented to crisscross the constraints of an edge for dynamic routing with respect to agent specification. Initially the profile of an agent defined with some basic parameter, e.g. height and width, speed, capacity, secondly the road network topographic features were also encapsulated, e.g., length, height, width, capacity and speed. Algorithm compute the optimum path envisaging the topographic featured of network and heterogeneous agents dynamically. Edge compatibility and constraints of heterogeneous agents anticipated and validated on the bases of topographic feature extraction. An intricate set of the parameter predefined as height and width. Primarily define for the known places but, INFINITY in case of unknown.

Functional constraints dynamically compare the agent specification and feasibility of route while before computing a path. In case, detect incompatibility of the topographic feature with respect to agent specification. The *edge* length set as INFINITY and vertex marked as the visited node. Agent profile basic parameter defined in Figure 2.

Agent	
Height	∞/ or in meter
Width	∞ or in meter
…	

Figure 2: Agent Profile Parameter 1

```
 function Dijkstra(Graph, source, Agent):
dist[source] ← 0
Q←V
for all v ∈ V
          dist[v] ← INFINITY
     prev[v] ← UNDEFINED
     S←∅
While Q≠∅
  do u ← vmindistance(Q,dist)
S←S∪{u}
            for all v ∈ neighbors[u]
   alt ← dist[u] + length(u, v)
          if alt < dist[v] & checkconstraint(agent, edge(u,v)):
           dist[v] ← alt
           prev[v] ← u
    return dist[], prev[]
```

Algorithm 2: Augmented Dijkstra Algorithm 1

The graph consists of vertices and edges representing intersections and roads correspondingly. The parameter of edge defined as height and width in addition to the weight. Initially the height and width of an edge (tunnel, pedestrian bridges and underpasses) defined whereas INFINITY incase unknown as described in figure [3].

The same optimal path computed for an agent, having the similar specifications or matched the topographic feature of a tunnel or underpass. The vertices representing intersection and the edges could bridge them.

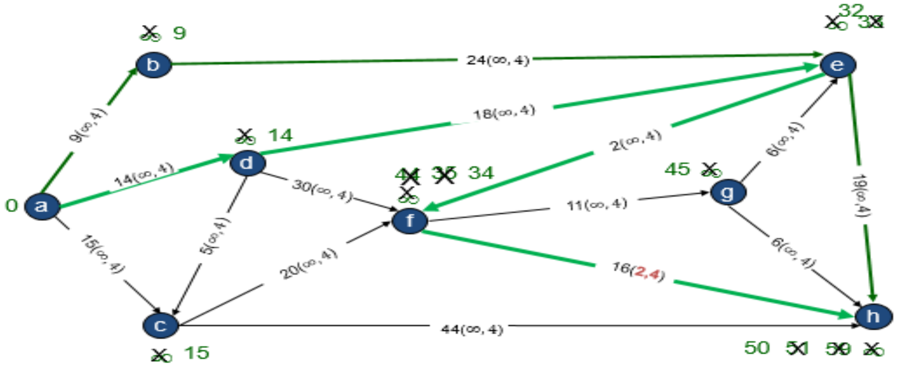

Figure 3: Augmented Method Agent Compatibility

The computation algorithm computed a different path for an agent, which have dissimilar specification, or the capacity does not matched the topographic feature of an edge whereas the performance of algorithm enhanced and the route planning ensured in accordance to the route feasibility. Illogicality sensed at the time of traversing of an edge where the length highlighted in RED, the length set to INFINITY and marked as visited vertex and computed the path on alternative connected edge.

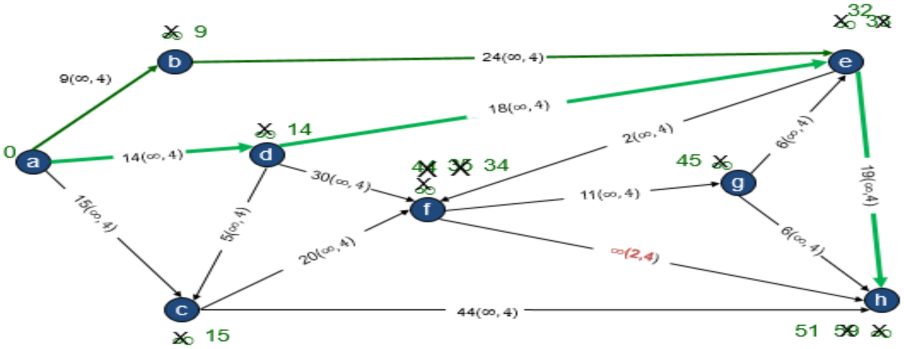

Figure 4: Augmented Method Agent non-Compatibility

Adjacency matrix defined in table [2] for graph given in figure 3 and 4 comprising V vertexes. The coma separated defined edges height, width and length; first ∞ indicate no direct link between the vertexes while the subsequent parameter defined the height and width of an edge, in case not known set as∞.

- Transform Roadway network into a weighted “graph”.
- Each edge in this directed graph denotes a direct connection between two vertexes carried a weight representing travel distance in KM.
- Graph characterize all possible connection between Source *s* to Destination *t*.
- Each edge in this directed graph denotes a direct connection between two vertexes carried a weight representing travel distance in KM along with height and width.

	a	b	c	d	e	f	g	h
a	0,∞,4	9,∞,4	15,∞,4	14,∞,4	∞	∞	∞	∞
b	∞	0,∞,4	∞	∞	24,∞,4	∞	∞	∞
c	∞	∞	0,∞,4	∞	∞	20,∞,4	∞	44,∞,4
d	∞	∞	5,∞,4	0,∞,4	18,∞,4	30,∞,4	∞	∞
e	∞	∞	∞	∞	0,∞,4	2,∞,4	∞	19,∞,4
f	∞	∞	∞	∞	∞	0,∞,4	11,∞,4	16,2,4
g	∞	∞	∞	∞	6,∞,4	∞	0,∞,4	6,∞,4

Table2: Digraph Adjacency Matrices for Figure 3 and 4

6 Key Contribution

The essence of progression to compute dynamic optimal path for heterogeneous agents envisaging the topographic feature of landmark and the augmentation of the existing algorithm of Dijkstra. The algorithm encompasses the agent specification by shaping the profile of an agent and encapsulating the topographic feature of road networks. The augmented algorithm computed the path envisaging the problematic edges for an agent, the most of advancement in algorithm based on topographic feature, time and congestions. The primary aim of this research was addressed the problem being evolved and coupled with rapid growth of infrastructure.

7 Simulation And Modeling

To analyze and evaluate the performance of augmented model the simulation of heterogeneous agents and modeling of road networks were the essential component. For simulation, several software are available such as SUMO, MATSim, MITSIMlab, AIMSUN, CORSIM, Paramics, SimTraffic, VISSIM and TRANSIMS. MATSim (Andreas Horni, 2016) is an activity-based extendable, multi-agent simulation framework implemented in Java. MATSim designed to model a single day activity of an agent, the common unit of analysis for activity-based models. An agent repeatedly optimizes its daily activity scheduled while in competition for space-time slots with all other agents on the transportation infrastructure. This is somewhat similar to the route assignment iterative cycle, but goes beyond route assignment by incorporating other choice dimensions like time choice, mode choice, or destination choice into the iterative loop. SUMO (Simulation of Urban Mobility) analyses the approximate working conditions by applying the simulation software and evaluated the performance of algorithm and its failure. The Modelling of road network, pedestrians and the simulation of Heterogeneous Agents were required the Spatial data, for the citing task the data downloaded from OSM (Open street Map), by using command line utility NETCONVERT and generates road networks as given Figure[5]. SUMO (Krajzewicz, Erdmann, Behrisch, & Bieker, 2012) is a suite of applications, which helps to prepare and perform the simulation of traffic scenario; SUMO uses its own formats for road networks and traffic demand. Road network can import using digital road map using "netconvert" or can be generated using application "netgenerate". SUMO road network symbolize as graphs contained nodes and roads representing intersection and edges where edges are unidirectional connections between two nodes and contain a fix number of lane. A lane contains geometry, the information about vehicle classes allowed on it, and the maximum allowed speed.

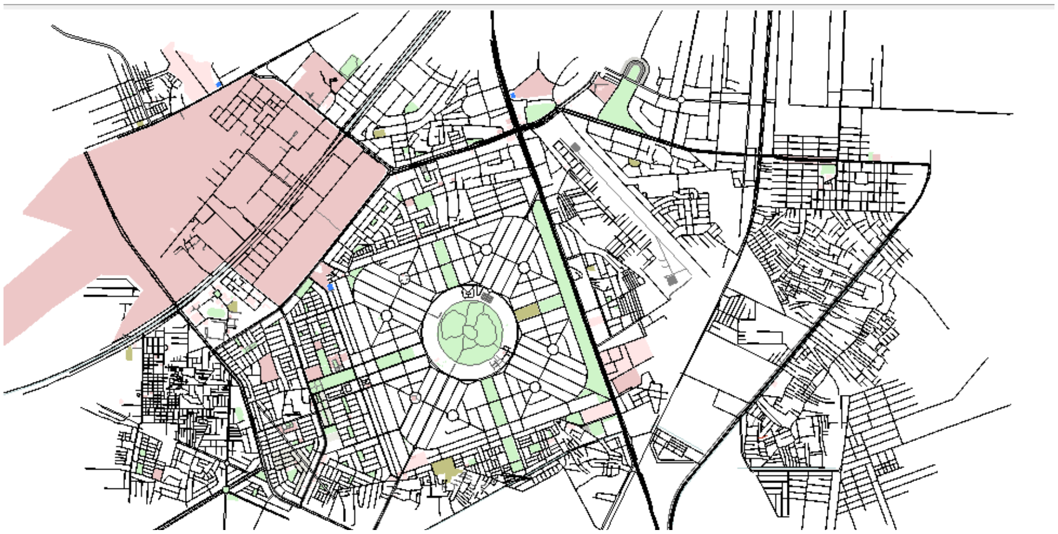

Figure 5: Spatial Data Converted for Desired Area

Vertexes and edges properties given in the configuration files based on their geographic location information. As shown in the Figure 5 and Figure 6. Heterogeneous agents launched with diverse specification given at the time of profile creation, e.g. height, width and speed, simultaneously the topographic features and parameter of road network personalized, e.g. the height of pedestrian bridges and underpasses determined. On the bases of customized road network, the origin and destination supplied for the computation of route for divergent agents with the changed specifications. Algorithm dynamically computed the path for heterogeneous agents from the same origin and destination in different ways illustrated in the figure 6 and figure 7 respectively.

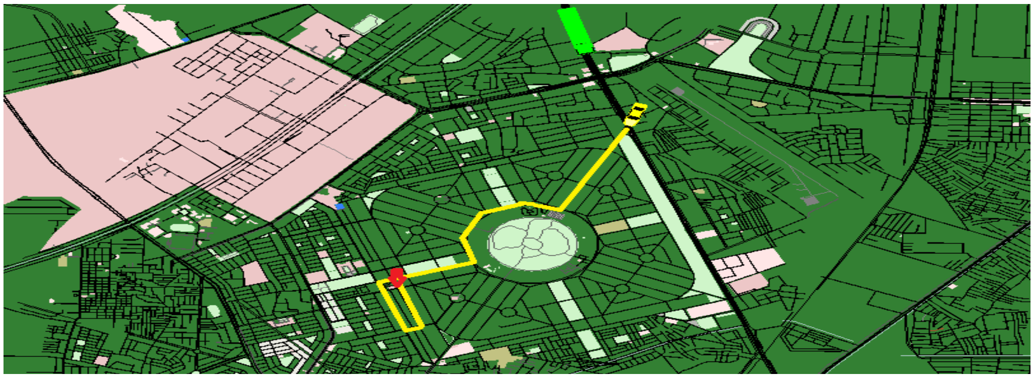

Figure 6: Augmented Method Route Computation

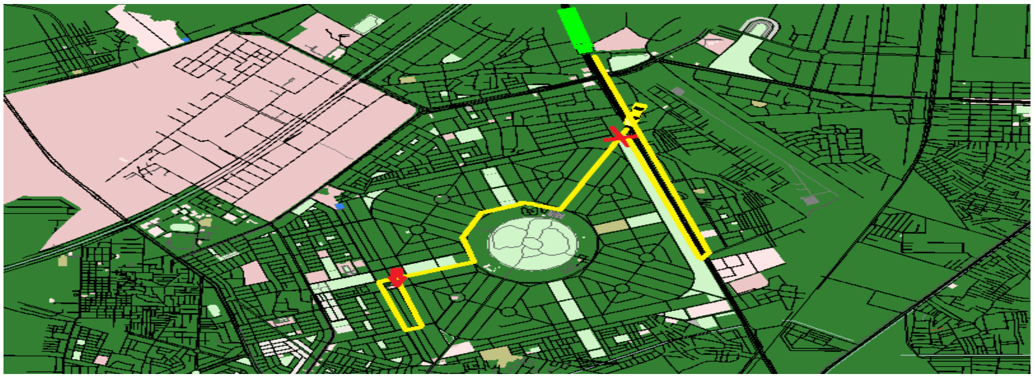

Figure 7: Augmented Method Topographic feature Compatibility

The figure [8] illustrates the speed of agents traveling in a curved line as a function of time.

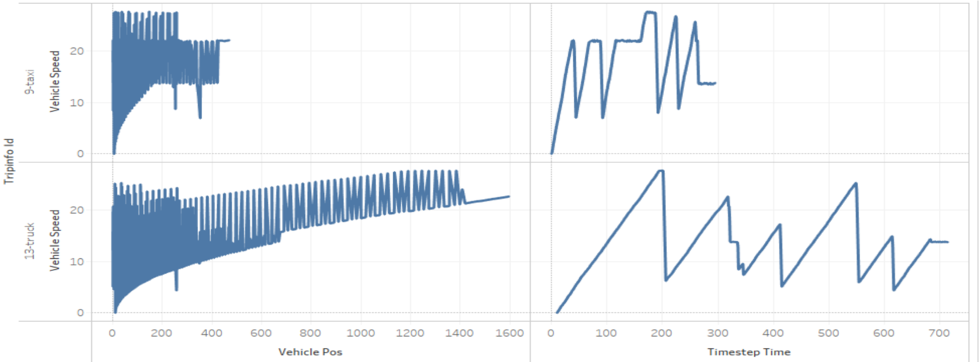

Figure 8: Augmented Model Speed Simulation Results

The figure [9] demonstrate the trip duration and time loss for heterogeneous agents.

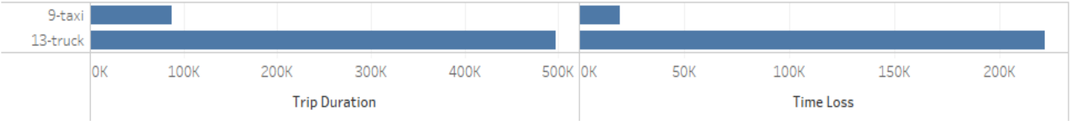

Figure 9: Augmented Model Trip Duration and Time Loss

The figure [10] validate the fuel consumption, vehicle CO2 omission, noise and speed.

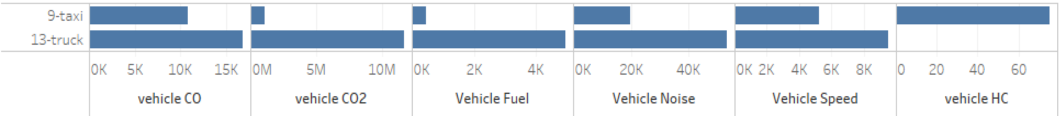

Figure 10: Augmented Model Fuel and Noise

8 Conclusion

This paper presents the augmented model of Dijkstra's algorithm based on dynamic route optimization for heterogeneous agents envisaging the topographic feature of road networks. Vertexes and edges contain variant features like height and width, capacity, congestion and speed. Computing the route dynamically based on the topographic feature was the key element while computing the path, algorithm considers the facts and feature of agents with respect to road networks. Dynamic re-routing based on obstacles detection using image processing for topographic feature extraction and intercepting any problem at the time of computation of route or at run time foreseen as a future work. The augmented model will assist the commuter to choose the optimal and feasible path with respect to agent's specification. This paper provides the basic framework for dynamic route optimization envisaging the topographic features.

9 Acknowledgement

This work would not have been possible without the support and guidance of Dr. Suleman Mazhar Assistant Professor Information Technology University Lahore, Director Bio-inspired Simulation & Modeling Lab.

References

Andreas Horni, K. N. (2016). *The multi-agent transport simulation MATSim.* London: Ubiquity Press .

Bast, H., Delling, D., & Goldberg, A. (2016). Route planning in transportation networks. *Springer*, 19-80.

Cameron, W., Regnier, J., Galloy, P., & Savoie, A. (n.d.). Dynamic routing for intercity telephone networks. *10th International Teletraffic Congress.* Montreal, Canada: 1983.

Dong, W. (2011). An overview of in-vehicle route guidance system.

JenniferFarver, & IsmailChabini. (2003). Hybrid Vehicle-Centric Route Guidance. *IFAC Proceedings Volumes*, 149-154.

Krajzewicz, D., Erdmann, J., Behrisch, M., & Bieker, L. (2012). Recent development and applications of SUMO-Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*.

Lee, J., & Yang, J. (2012). A fast and scalable re-routing algorithm based on shortest path and genetic algorithms. *With Emphasis on the Integration of Three Technologies*, 459.

MAHMASSANI, Y. E. (1996). Comparative analysis of robustness of centralized and distributed network route control systems in incident situations. *Transportation Research Record: Journal of the Transportation Research Board*, 83-90.

Sun, Y. S., Xie, L., Chen, Q. A., Lu, S., & Chen, D. (2014). Efficient route guidance in vehicular wireless networks. *Wireless Communications and Networking Conference (WCNC), 2014 IEEE* (pp. 2694-2699). Istanbul: IEEE.