

# Optimal Path Calculation for Route Leveraging APIs Using Location Based Service

Zahoora Abid<sup>1</sup>, Tahera Abid<sup>2</sup>, Khutaija Abid<sup>3</sup>

<sup>1,2</sup>Assistant Professor, Nawab Shah Alam Khan College of Engineering and Technology, Hyderabad, Telangana, India

<sup>3</sup>Assistant Professor, Al-Habeeb College of Engineering and Technology, Hyderabad, Telangana, India

## ABSTRACT

A location based service (LBS) is an application for user's mobile device that requires necessary information about where the mobile device is located. Location-based services shall be query-based and provide the end user with valuable information, for e.g LBS enable mobile users to query points-of-interest (e.g., ATM, restaurants) on various features (e.g., category, price, quality, and variety). They manage points-of-interest (POIs) specific to their applications, and enable mobile users to query for POIs that match with their preferences and time constraints and also it secures a user identity and locality within basic mobile communication services. For this approach this paper provides the survey about various techniques for providing the accurate and efficient query for the mobile users.

**Keywords:** Location Based Services (LSB), POIs, Query Processing

## I. INTRODUCTION

There is growing need for useful approaches involving use of global positioning systems (GPS) information from GPS trackers for traffic analysis. In recent years, activity-based evaluation using GPS equipment's as information collectors has been a principal challenge. All these forms of research focus on data from wearable GPS recorders considering of effortless specified undertaking logging and interactive validation with users. As data have improved, more sophisticated approaches of data assortment have been developed, represented at first by the shift from travel to activity diaries, and continuing on to the progress of GPS enabled activity surveying. Traffic evaluation is a component of key points in developing international locations that require better and efficient monitoring.

In order to compensate for uncertainties in journey time due to accidents, bad weather, traffic congestion, and so on, vehicles hauling time-sensitive freight build "buffer time" into their routes in order to help ensure that deliveries will probably be made on time. Building buffer time into routes tends to increase the likelihood of on-time delivery, an important measure of service. However, buffer time also tends to reduce measures of productivity

associated with cost, such as driver and equipment idle time and the number of miles travelled per hour. An LBS requires 5 basic components: the service provider's software applications, a mobile network to transmit information and requests for service, a content provider to supply the end user with geo-specific information, a positioning component (see GPS) and the end user's cellular device. There is an escalating demand to accurately and quickly determine the location of a mobile at low cost. Location-based applications are flattering popular and available and provide the user with information based on their location. Computing the shortest path is foremost challenge within the spatial databases. The path computed by making use of the pre-stored information is not accurate. Thus, there is necessity for the live traffic data. There are a number of online service traffic providers like NAVTEQ [4], TomTom [5], Google maps [6]. However these traffic vendors do not furnish.

They always were growing popularity of online map applications and their wide deployment in mobile devices and car navigation systems, increase number of client search for point to point fastest paths and the corresponding travel times. On static road networks where edge costs are constant, this problem has been

extensively studied and many efficient speedup techniques have been developed to compute the fastest path in a matter of milliseconds. At the present time, a few online services provide live traffic information (by using examining accrued information from road sensors, traffic cameras, and crowd sourcing approaches). These systems can calculate the image shortest path queries based on present traffic information; nevertheless, they don't record routes to drivers constantly because of high operating costs. Answering the shortest paths on the live traffic data can be considered as a continuous monitoring difficulty in spatial databases, which is termed as online shortest paths computation in this work. Short-time period traffic forecasting objectives to furnish more reliable travel information service, to be able to support people in making more reasonable traffic decisions. With the growing availability of traffic data along with the progress of conversational science, both the capacity and accuracy of travel time forecasting were enormously enhanced in real-time conditions and a great quantity of forecasting methods have been applied throughout recent years. Nonetheless, they're insufficient when confronted with the actual world traffic issues, considering the real-time traffic condition can be affected conveniently and changed continuously.

In most cases, the shortest path is computed by using offline information pre-stored within the navigation techniques and the weight (travel time) of the road map is estimated by way of the average distance or old data. Unluckily, road traffic circumstances change over time. Without live traffic instances, the route returned through the navigation system is now not guaranteed a correct result. Those historical navigation techniques would endorse a route based on the pre-stored distance information. Note that this route passes through four street upkeep operations (Indicated by way of preservation icons) and one traffic congested street (indicated by means of a red line). These days, several on-line offerings furnish reside visitors knowledge (through analysing collected information from road sensors, traffic cameras, and crowd sourcing strategies). These systems can calculate the photograph shortest path queries based on current reside visitors information; nonetheless, they don't record routes to users regularly due to high working charges. Answering the shortest paths on the live traffic data may

also be considered as a steady monitoring problem in spatial databases.

## II. EXISTING SYSTEM

The shortest direction influence is computed based on the live traffic circumstances [1]. They carefully analysed the prevailing work and speak about their inapplicability to the problem. To deal with the difficulty, they suggested a promising architecture that proclaims the index on the air. They first determined a predominant characteristic of the hierarchical index structure which permits us to compute shortest path on a small component of index. This major feature is utilized in their resolution, Live Traffic Index (LTI). Their experiments verified that LTI is a Pareto optimal answer in terms of four performance factors for online shortest path computation.

Shortest path computation is an principal function in present day auto navigation methods [2]. This function helps a driver to figure out the excellent route from his current place to destination place. In most cases, the shortest path is computed by using offline information pre-stored within the navigation techniques and the weight (travel time) of the road map is estimated by way of the average distance or old data. The online shortest route computation; the shortest route result is computed based on the traffic situations. Analyse the prevailing work and speak about their inapplicability to the situation (because of their prohibitive protection time and massive transmission overhead). To address the concern, advocate a promising structure that publicizes the index on the air. First determine a foremost characteristic of the hierarchical index constitution which enables us to compute shortest path on a small section of index.

## III. PROPOSED SYSTEM:

When person desire to know destination information based on consumer's requirement say for illustration user needs to reach nearest ATM or hospital. He can get ATM or hospital information using internet service provider. However he wishes effective result with respect to travel time and fee (i.e. nearest route). Consequently person needs application that supplies all of the expertise he desires. Figure 1 shows the architecture of proposed

process. The proposed procedure entails almost always threepredominant modules, user module, LBS module and Route-Saver module. In user module user receives a location mapincludes locations, user location and route map from user place (source) and possible destination. In our proposed work, the users require accurate results that are computed with appreciate to live traffic information. The entire works require the LBS to know the weights (travel times) of all road segments. Considering that the LBS lackthe infrastructure for monitoring road traffic, the above works are inapplicable to our problem. Some works try andmodel the travel occasions of street segments as time-various features, which may also be extracted from historicaltraffic patterns. These services may just capture the consequences of periodic events(e.g. rush hours, weekdays).Nevertheless; they nonetheless cannot reflect traffic information, which can be effected by sudden events, e.g. Congestions, accidents and road maintenance. The LBS module is responsible for accumulating the specified data from consumer and LBS generate optimizedinformation which includes consumer's present area and route log to the destinations. Then this information istransferred to the Route-saver. Route-saver utilizes the contemporary traffic understanding bought from traffic providerand calculates the journey time and most beneficial path to source and destinations by using Nearest Neighbor queries.

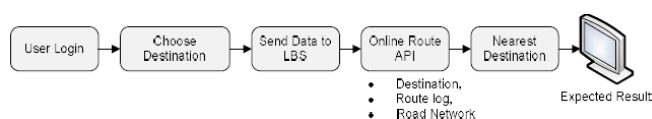


Figure 1: Framework for Proposed Work

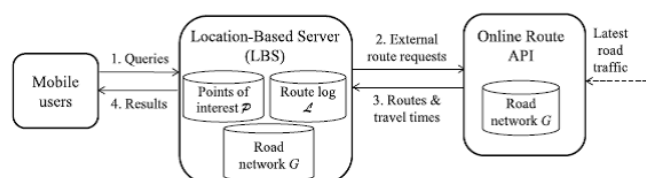


Figure 2. Maintenance of Structures at LBS

Conservative travel time bounds. Given an edge  $e(v, v-)$ , we define  $cw^-(e)$  and  $cw^+(e)$  as conservative lower boundand upper-bound of travel time one, observe that the lower bound  $cw^-(e)$  is limited by the Euclidian distance of eand maximum driving speed  $V_{max}$ ,  $cw^-(e) = \text{dist}(e) / V_{max}$

To reduce the number of route requests while providing efficient results, we combine information throughout a couple of routes within the log to derive tight lessen/higher bounding journey times. We also propose effective strategies.

to compute such bounds efficiently. Additionally, we compare the influence of exclusive orderings for issuing routerequests on saving route requests. And we learn the best way to parallelize route requests in order to reduce the queryresponse time additional.

Above figures show the screen shots our proposed work, the user has an access to router via an internet. Based on hispresent location he has to choose the destination point, and then LBS will communicate with server and shows you thenearest places of his choice.

Figure (a)and(b) shown window where you can choose the destination you need to go, onceyou selected point server will give you the nearest requested destination with shortest path. We can see the nearest destination point in the figure 3, here user wants to visit to a hospital, and using an applicationhe will enter the hospital as destination point. For the requested information server will revert back with all hospitalswhich is located around the user's current location. As we see in the figure 3, all the hospitals are marked in red colorand the nearest hospital is marked in a purple color, here user location is marked as source and is in green color.

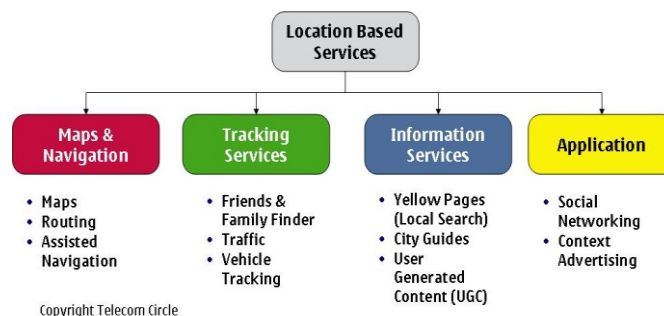
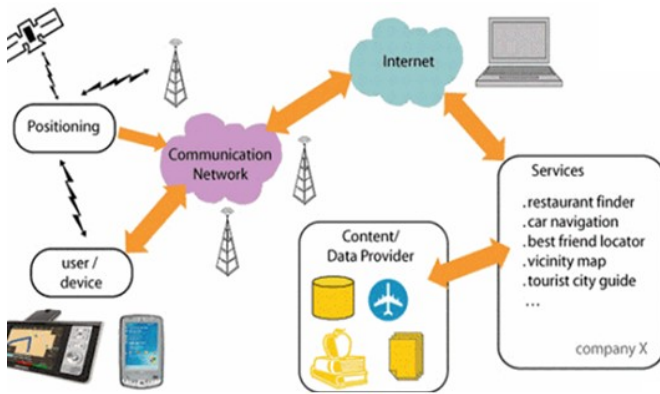
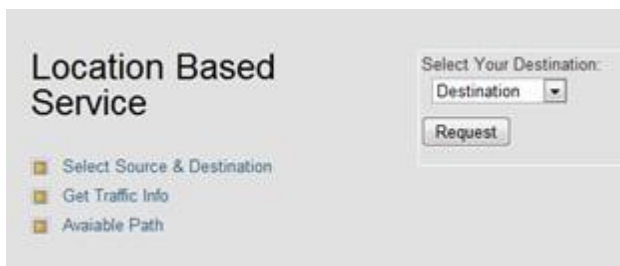


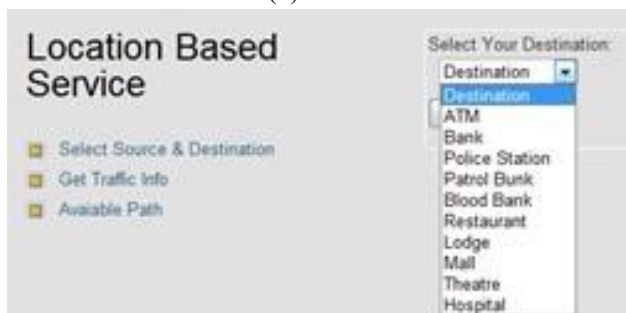
Figure 3. Types of location based services



**Figure 4:**Location based services network architecture

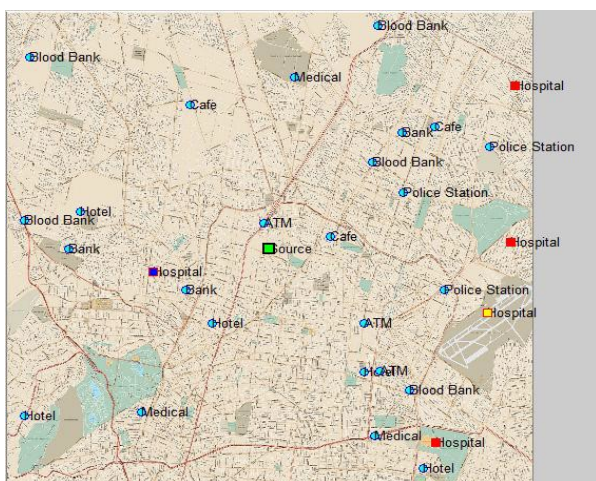


(a)



(b)

Destination point request

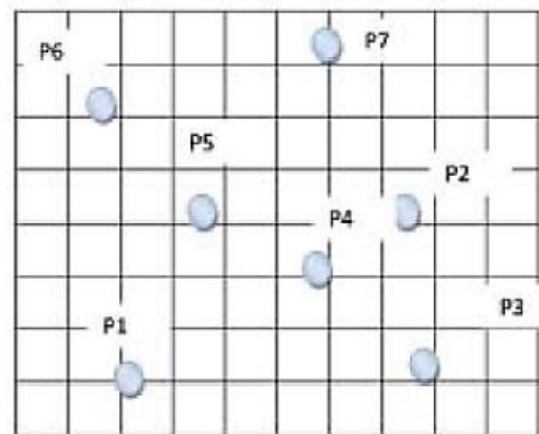


Nearest Destination Point

## Query processing

### Algorithm 1: Location Point based Tree (LPT with Inverted Index):

The location point tree saves the location points along with the service details. The service identifies the service by using the spatial POI which will be stored on the LPT\_II. To speed up itinerary process in this indexes all the objects' subspace spatial scopes by an LPT tree where the subspace spatial scopes are stored in the leaf nodes as data entries. Additionally, to support service recommendation, this follows mapping module and KNN search [7]. In the diagram p1, p2, p3, p4, p5...p7 are the service points. In the proposed system every service point will be stored with an mapping key for fast service search. The proposed system stores the above service point in the tree style.



### Algorithm 2: LPT\_II maintenance tree Steps

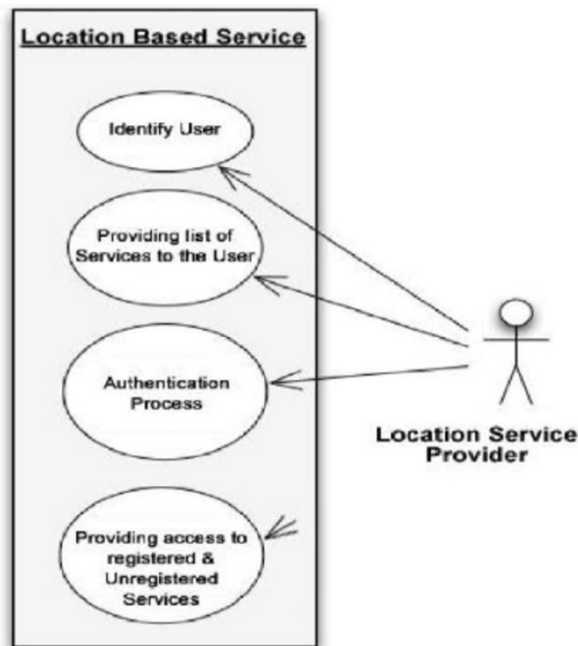
The following table represents the location point and available service details in the location. The service can able provide appropriate services for the user interest based on the location point based service list. This helps in two ways. One is the retrieval time has reduced. And the next is fast search in the LPT helps to avoid the unavailable services [8].

POI	Services (S)
P1	S1,S3
P2	S4,S6
P3	S5,S4
P4	S4,S7
P5	S2,S8

**Figure 3.** LPT\_II maintenance tree Steps

1. Initial location source with respect point  $p_1, p_2, \dots, p_n$
2. For every point in the source P do
  - a. Initialize service  $S_1 \dots S_n$  for respective points.
  - b. Set service and code for every service
3. Get rank for each service and store into the ascending order.
4. Store the node in the top level based on the rank
5. Prune the other items from the LPT.

#### USE CASE DIAGRAM:



**Figure 5 :** Use case diagram for LBS

#### IV. CONCLUSION

In this paper, we propose a solution for the LBS to process range/KNN queries such that the query results have accurate travel times and the LBS incurs few number of route requests. Our solution Route-Saver collects recent routes obtained from an online route API (within  $d$  minutes). During query processing, it exploits those routes to derive effective lower-upper bounds for saving route requests, and examines the candidates for queries in an effective order. We have also studied the parallelization of route requests to further reduce query response time.

#### REFERENCES

- [1] M. Argon and K. Sirisha, "Live Activity List Based Shortest Path Calculation", Issue 09, 2015.
- [2] R.Subashini and A.Jeya Christy, "Online Shortest Path based on Live Traffic Circumstances" Volume 3, Issue. 11, 2014.
- [3] Dekonda Sindhuja, R Vasavi and A Kousar Nikhath, "Online Shortest Path Computation using Live Traffic Index", Volume 25, Issue 3, 2015.
- [4] "NAVTEQ Maps and Traffic," <http://www.navteq.com>, 2014.
- [5] "TomTom NV," <http://www.tomtom.com>, 2015.
- [6] "Google Maps," <http://maps.google.com>, 2015.
- [7] D. Zhang, CY. Chow, Q. Li, X. Zhang, Y. Xu, —Efficient Evaluation of k-NN Queries Using Spatial Mashups, Advances in Spatial and Temporal Databases (Lecture Notes in Computer Science), Vol. 6849, pp.348-366, 2011.
- [8] D. Zhang, CY. Chow, Q Li, X Zhang, Y Xu, SMashQ: Spatial mashup framework for k-NN queries in timedependent road networks", Distributed. Parallel Databases, Vol. 31, pp. 259–287, 2012.