Distributed Commuting Augmented Shortest Path Finding for Geo Spatial Datasets

Manoj Pandya
BISAG
Gandhinagar, Gujarat
India

Abdul Zummarwala
BISAG
Gandhinagar, Gujarat
India

Prashant Chauhan
BISAG
Gandhinagar, Gujarat
India

ABSTRACT

Geo Spatial information is a large collection of datasets referring to the real world entities. Geo Spatial information has evolved in the last decade which led to produce a vast platform in Government Administration, Scientific Analysis and other various sectors especially in disaster management (DM), site suitability of Check-dams for Irrigation Department etc. It is required to obtain imperative geographically analyzed solutions like finding shortest path between sources (i.e. location having stock of food packets, clinical remedial and therapeutic kits, etc.) to the destination (i.e. location where disaster emerges). Departmental data (i.e. village Maps) covers detailed spatial and attribute information compared to the readily available sources. Hence custom solutions based on Information Technology are required to be constructed, processed efficiently and quickly to outfit seamless performance that facilitates in mission critical incidences.

Keywords

Geo Database, Distributed Computing, Dijkstra, Shortest Path, GIS, Hadoop, Shapefile

1. INTRODUCTION

Accidents are common disasters that cause much damage to people’s lives. Due to population growth and the accompanying development of extensive infrastructures has greatly increased people's financial condition. This has lead to use large amount of vehicles over recent years. In case of accidents, it is a critical situation to get information about the location of the incidence, to reach at that place and supply necessary therapeutic kit and clinical remedies. Crucial need is to find out shortest path between the location of incidence and the nearest location providing clinical remedies or hospitalization. Available sources do not cover complete geo spatial details of road network especially in rural area. Even though detailed information is available with the department, it is difficult to utilize it in the time of disaster due to poor processing performance as the server becomes loaded with handling multiple concurrent requests. Vital issue is to make a timely and effective decision.

2. DISTRIBUTED DATABASES

A distributed database is a database in which storage devices are not all attached to a common processing unit such as the C.P.U. It may be stored in multiple computers located in the same physical location, or may be dispersed over a network of interconnected computers. A database may consist two or more data files located at different sites on a computer network. Different users can access it without interfering with one another. However, the DBMS periodically synchronize the scattered databases to make sure that they all have consistent data.

3. GIS NETWORK MODEL

In GIS systems, networks are modeled as points (For e.g: street intersections, switches, water valves) and lines (For e.g: Streets, transmission lines, pipes). Network topological relationships define how lines connect with each other at nodes. For the purpose of network analysis it is also useful to define rules about how flows can move through a network.

A network is defined as a graph G = (N, A) consisting of a set N of nodes and a set A of arcs with associated numerical values, such as the number of nodes, n=|N|, the number of arcs, m=|A|, and the length of an arc connecting nodes i and j, denoted as l (i, j).

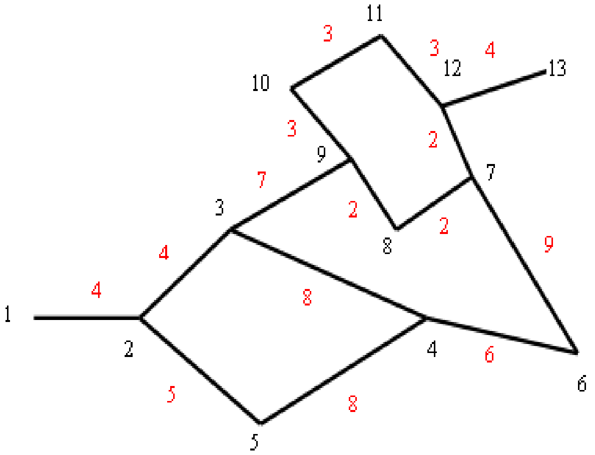


Figure 1: A typical Road Network

As shown in the above figure 1, each vertex is marked with black color. The segment length or weight is marked with red color near the center of the segment.

A road network as shown in above figure 1 can be interpreted in the form of attributes that is compatible to dijkstra as shown in the below table.

Table 1: Input tabular information for dijkstra

Fnode_	Tnode_	Length
1	2	4
2	1	4
2	3	4
3	2	4
2	5	5
5	2	5
5	4	8
4	5	8
4	6	6
6	4	6
3	4	8
4	3	8

It is required to pre-process shortest path from one node to other node with all possible permutation and combinations. The graph is non-directional. Hence both combinations of Fnode_ and Tnode_ are to be considered as shown in the following table.

Table 2: Dijkstra processed path for each Fnode_ and Tnode_

Fnode_	Tnode_	path
1	2	1,2
2	1	2,1
1	9	1,2,3,9
9	1	9,3,2,1
1	6	1,2,5,4,6
6	1	6,4,5,2,1
1	4	1,2,5,4
4	1	4,5,2,1
6	9	6,7,8,9
9	6	9,8,7,6
5	8	5,2,3,9,8
8	5	8,9,3,2,5

4. SHORTEST PATH

In graph theory, the shortest path problem is the problem of finding a path between two vertices (or nodes) in a graph such that the sum of the weights of its constituent edges is minimized. An example is finding the quickest way to get from one location to another on a road map; in this case, the vertices represent locations and the edges represent segments of road and are weighted by the time needed to travel that segment.

For undirected graphs, the shortest path problem can be formally defined as follows. Given a weighted graph (that is, a set V of vertices, a set E of edges, and a real-valued weight function f : E → R), and elements v and v' of V, find a path P (a sequence of edges) from v to a v' of V so that

$$\sum_{p \in P} f(p) \qquad \text{Eq. 1}$$

is minimal among all paths connecting v to v'.

Shortest paths from one (source) node to all other nodes on a network are normally referred as one-to-all shortest paths. Shortest paths from one source node to a subset of the nodes on a network can be defined as one-to-some shortest paths. Shortest paths from every node to every other node on a network are normally called all-to-all shortest paths. When the goal is to obtain a one-to-one shortest path or one-to-some shortest paths, the Dijkstra algorithm offers some advantages because it can be terminated as soon as the shortest path distance to the destination node is obtained. However, there are other algorithms that can also be used for finding shortest path in various scenarios.

Dijkstra's algorithm: solves the single-source shortest path problems.

Bellman–Ford algorithm: solves the single-source problem if edge weights may be negative.

A* search algorithm: solves for single pair shortest path using heuristics to try to speed up the search.

Floyd–Warshall algorithm: solves all pairs shortest paths.

Johnson's algorithm: solves all pairs shortest paths, and may be faster than Floyd–Warshall on sparse graphs.

Dijkstra’s algorithm to find out a shortest path between source and destination is commonly practiced and easy to implement. Dijkstra’s algorithm is used in this paper to demonstrate a prototype model as a code snippet.

Code Snippet:

```
-- Run the algorithm until we decide that we are finished
   WHILE 1 = 1
   BEGIN
 -- Reset the variable, so we can detect getting no records in the
next step.
      SELECT @FromNode = NULL

 -- Select the Id and current estimate for a node not done, with
the lowest estimate.
      SELECT TOP 1 @FromNode = Id, @CurrentEstimate =
Estimate
      FROM #Nodes WHERE Done = 0 AND Estimate <
9999999.999
      ORDER BY Estimate

      UPDATE #Nodes SET Done = 1 WHERE Id =
@FromNode

 -- Update the estimates to all neighbour node of this one (all the
nodes
 --there are edges to from this node). Only update the estimate if
the new
 -- proposal (to go via the current node) is better (lower).
      UPDATE #Nodes
                        SET Estimate = @CurrentEstimate +
e.Weight, Predecessor = @FromNode
      FROM #Nodes n INNER JOIN dbo.Edge e ON n.Id =
e.ToNode
      WHERE Done = 0 AND e.FromNode = @FromNode AND
(@CurrentEstimate + e.Weight) < n.Estimate

   END;
```

The code can be compiled as a dynamic link library (dll) to integrate in a program to solve shortest path for given datasets. The executed algorithm stores each vertex as a path in a defined format.

5. DATA MODEL

Spatial dataset can be stored either in normalized or binary format. Enterprise geo database like Microsoft SQL Server, Oracle etc. can be used to process large amount of data.

The Dijsktra can’t be directly implemented in GIS datasets. The Spatial data format is not known to Dijkstra as it can understand the data in the form of FROM_NODE, TO_NODE and WEIGHT or LENGTH of the segment. It is required to calculate each node and edge (segment) length in GIS dataset and store in the tabular data table.

At higher hierarchical level, there is a great density of roads covering national Highway, State Highway, Village Road, ODR, MDR etc. Dijsktra performs all possible permutations and combinations for finding the shortest path.

For each request, dijkstra algorithm has to process multiple node and edges that can affect the performance of the system when considering the concurrent access of multiple users.

6. PRE-PROCESSING NETWORK ROUTES

The task of finding shortest path for large network is a time consuming process. It is memory and process intensive. In web based interface, multiple concurrent users accessing various combinations of nodes require high end servers. In case of enterprise environment, single server can’t serve concurrent requests at the optimal level. To optimize server, there are several optimization methods available.

Divide and Conquer: In this technique, the task is divided into several fragments. These fragments are processed in different process units and after the completion of task, they are integrated. The information can be either in database or in file format.

Database is fragmented into several parts by taking ratio of number of records to the number of processors in a machine. A file is divided by automated tool (Apache Hadoop) that handles the consistency and integrity of information.[6]

7. VIRTUALIZATION

It is the creation of a virtual (rather than actual) version of a hardware platform, operating system (OS), storage device, or network resources. Virtualization reduces system integration and administration costs by maintaining a common software baseline across multiple computers in an organization. A virtual machine is subjectively a complete machine (or very close), but objectively merely a set of files and running programs on an actual, physical machine. [10]

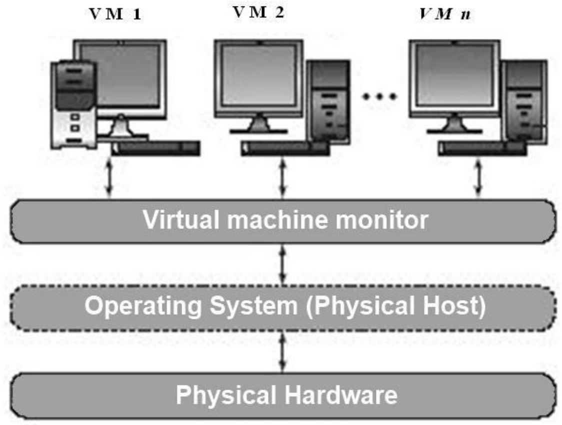


Figure 2: A diagram showing the layout for virtualization.

8. TASK DISTRIBUTION USING HADOOP

The Apache Hadoop is an open source software framework that supports data-intensive distributed applications. [5] It enables applications to work with thousands of computational independent computers and petabytes of data. Hadoop works on file based architecture. Hadoop provides a distributed file system (HDFS) that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both Map/Reduce and the distributed file system are designed so that node failures are automatically handled by the framework.

Map/Reduce is a programming paradigm that was made popular by Google where in a task is divided in to small portions and distributed to a large number of nodes for processing (map), and the results are then summarized in to the final answer (reduce). [1] Google and Yahoo use this for their search engine technology. By the execution of shortest path algorithm using hadoop map/ reduce processing mechanism, the hardware resources (CPU) are fully utilized. That is capable to handle large size of files with HDFS. Redundancy and replication of files are maintained by Hadoop automatically as soon as the task executes. [6]

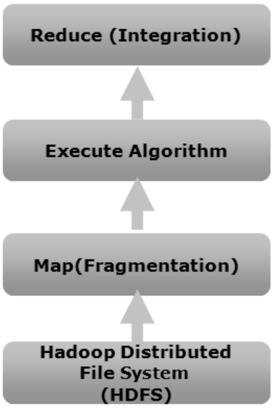
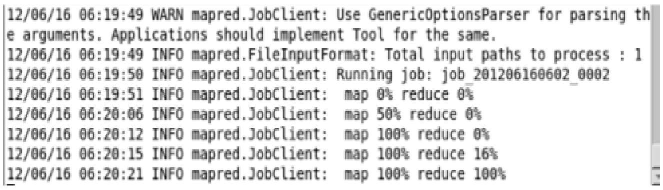


Figure 3: A synoptic view of Hadoop processing mechanism

9. EXPERIMENTATION RESULTS

The edges cross the count of more than 1 lac records for complete dataset. The dataset corresponds to the road network of Gujarat state, India provided by BISAG. Shortest Path algorithm is executed with HDFS.



```
12/06/16 06:19:49 WARN mapred.JobClient: Use GenericOptionsParser for parsing th
e arguments. Applications should implement Tool for the same.
12/06/16 06:19:49 INFO mapred.FileInputFormat: Total input paths to process : 1
12/06/16 06:19:50 INFO mapred.JobClient: Running job: job_201206160602_0002
12/06/16 06:19:51 INFO mapred.JobClient:  map 0% reduce 0%
12/06/16 06:20:06 INFO mapred.JobClient:  map 50% reduce 0%
12/06/16 06:20:12 INFO mapred.JobClient:  map 100% reduce 0%
12/06/16 06:20:15 INFO mapred.JobClient:  map 100% reduce 16%
12/06/16 06:20:21 INFO mapred.JobClient:  map 100% reduce 100%
```

Figure 4: Snapshot of system showing hadoop map-reduce functionalities.

The output is obtained using Distributed processing over Hadoop framework.

Table 3: length, Fnode_, Tnode_ as input and Length, Path as output

Sr	FNODE_	TNODE_	Length (m)	Path
1	21626	1	753286.4	21626,21640,21930, ...,18,22,3,2,1
2	21626	135	736610.7	21626,21640,21930, ...,281,232,135
3	21626	205	727586.5	21626,21640,21930, ...,198,206,205

Table 3 represents the processed nodes and their respective length and shortest path for possible permutation and combinations.

GIS map is the graphical representation of entities and objects. The derived output as shown in the figure 5 is linked with GIS dataset. The red colored segments correspond to the shortest path between two node points as show in the below figure.

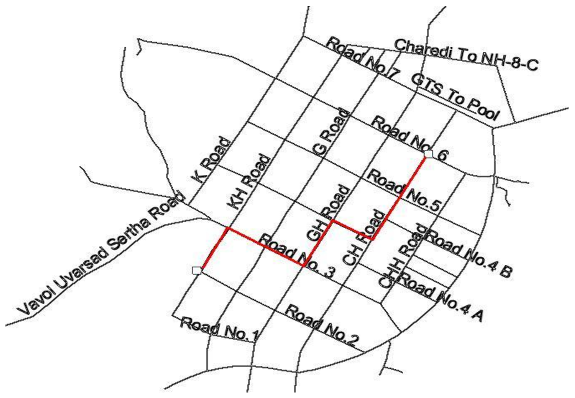


Figure 5: The map highlighted with red colour represents the shortest path between given two points. [3]

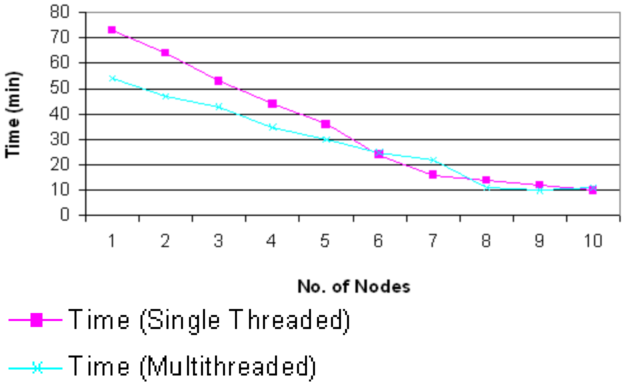


Figure: 6 Nodes vs. Time Chart

It is inferred from the results that the division of tasks to multiple nodes decreases the computation time upto a certain extent as the division into tasks and collection of results leads to overhead upon the hadoop framework. Both the type of applications declined to yield better results upon further division and increase in number of nodes.[13]

Table: 4 Time taken to find shortest path by single & multiple threads.

Number of Nodes (Single Threaded)	Time (Single Threaded)	Number of Nodes (Multithreaded)	Time (Multithreaded)
1	73	1	54
2	64	2	47
3	53	3	43
4	44	4	35
5	36	5	30
6	24	6	25
7	16	7	22
8	14	8	11
9	12	9	10
10	10	10	11

10.APPLICATIONS

Distributed Computing using Hadoop can be used in various areas like rendering 3D imagery, computing the energy in a system in a molecular model, data mining and in scientific analysis.

11.LIMITATIONS

Distributed Computing using Hadoop is suitable for large sized files especially for scientific data analysis but for small sized files the performance is negligible as compared to the time required for the setup of distributed environment.

12.CONCLUSIONS

Applications where Mission critical solutions are required with real time immediate response, Distributed Computing with Apache Hadoop is suitable platform. Hadoop can also be used with personal computers and low end servers.

13.ACKNOWLEDGEMENTS

The authors would like to thank to the Director, BISAG, T .P. Singh for his inspiration and motivation.

REFERENCES

[1] Lam, Chuck (July 28, 2010). Hadoop in Action (1st Ed.). Manning Publications. ISBN 1-935182-19-6.

[2] J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. OSDI ’04, pages 137–150.

[3] Courtesy form BISAG

[4] Z. Mahmood and R. Hill (eds.), Cloud Computing for Enterprise Architectures, Computer Communications and Networks, DOI 10.1007/978-1-4471-2236-4_2, © Springer-Verlag London Limited 2011

[5] VerticaInputFormat. http://www.vertica.com/mapreduce

[6] Tom White. Hadoop:The Definitive Guide[M]. United States of America: O’Reilly Media, Inc. 2009.

[7] Jeffrey Dean, Sanjay Ghemawat. MapReduce:Simplied data processing on large clusters[C]. Proceedings of the 6th Symposium on Operating System Design and Implementation. New York: ACM Press. 2004:137-150.

[8] Raghavendra, C. S., Kumar, V. K. P. and Hariri S., “Reliability analysis in Distributed systems”, IEEE Transactions on Computers, Volume 37, Issue 3, Pg. 352 – 358, March 1988.

[9] Kin-Sun-Wah and McAlister D.F, Reliability optimization of computer communication network, IEEE Trans. On Reliability, Vol.37, No. 2, Pp.275- 287 (1998) Dccember.

[10] GPFS in the Cloud: Storage Virtualization with NPIV on IBM System p and IBM System Storage DS5300.

[11] Bo Peng, B. C. (2009). Implementation Issues of A Cloud Computing Platform. IEEE , 8.

[12] R. Agrawal and J.C. Shafer , "Parallel Mining of Association Rules," Distributed Systems Online March 2004.

[13] D.W. Cheung , et al., "Efficient Mining of Association Rules in Distributed Databases,"IEEE Trans. Knowledge and Data Eng., vol. 8, no. 6, 1996,pp.911-922.