Reduction of traffic time in Medellin city

Juan Sebastián Pérez Salazar
Universidad Eafit
Colombia
jsperezs@eafit.edu.co

Yhoan A. Guzmán García
Universidad Eafit
Colombia
yaguzmang@eafit.
edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACT

In this paper, we were asked to propose a solution to the principal problem that was presented to us. The aforementioned problem being; how can we reduce the traffic in Medellin and avoid having highly contaminated air which affects the quality of life for the citizens of Medellin. First you will read the introduction, afterwards we will introduce you to the problem, later we will present problems which are similar and lastly we will present you with a possible solution which will show a more stable future for Medellin.

KEYWORDS

A*, DFS, BFS, data structures, complexity, travels, shared cars, reduction of traffic, improving paths, life quality, air pollution.

ACM CLASSIFICATION KEYWORDS

Theory of computation → Design and analysis of algorithms → Graph algorithm analysis → Shortest path, Dynamic graph algorithms.

1. INTRODUCTION

One of the biggest problems that Medellin faces, or any other growing city for that matter, is congestion along the main routes of the city. This affects how easily someone can move within the city which has been shown to be a factor in the quality of life for the citizens of the city. As the number of cars along these routes increases the air pollution and overall contamination of the city increases, when this happens, the government is enforced to restrict the usage of cars by the citizens according to license plate numbers. One way that we can decrease these extremes in contamination is by carpooling.

One suggested solution is that within each company, the employees that own cars will rotate and carpool to work which will reduce the traffic in the city and improving mobility within the city.

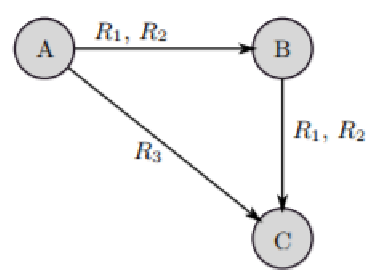
2. PROBLEM

The presented problem is to improve the current traffic situacion in the city by carpooling. We are to assume that everyone that has a car currently goes to work alone. The idea is that every driver can carpool with 4 other people to work without drastically increasing their commute time to work.

3. RELATED WORK

3.1 CrowdRoute: a crowd-sourced routing algorithm in public transit networks

Crowd-sourced (CS) systems recruit a crowd of human beings to solve a range of problems. There are various CS systems that already exist. Popular systems include Wikipedia, Linux, Mechanical-Turk based systems and many more systems are being built. We can classify CS systems by several dimensions such as the nature of collaboration, type of target problem, standalone or piggyback. For a complete list of dimensions, readers are referred to [1]. But the two most important dimensions are the nature of collaboration and type of target problem. The passengers could collaborate explicitly or implicitly towards solving a target problem, which can be any problem defined by the system owner. Some other important dimensions include: how to recruit passengers, what passengers do, how to combine them, and how to evaluate them. There have been works on using crowd-sourced information to improve the quality of the routing algorithm, one great example is [3] which is recently acquired by Google [4]. [2]



R_1: A → B → C from 8:00 till 10:00 every 0.5 hr

R_2: A → B → C from 8:10 till 9:10 every 1 hr

R_3: A → C from 9:00 till 11:00 every 1 hr

Time to travel between stops is 20 minutes

Transfers time is 5 minutes for every stop

3.2 Traffic Measurement and Route Recommendation System for Mass Rapid Transit (MRT)

Singapore’s public transport system is among the world’s most ecient ones in terms of the density of the network, crowdedness and travel times [5, 6]. Its core component is the Mass Rapid Transit (MRT), a railway system spanning the entire city-state, which has been in operation since 1987. Eciency metrics such as the crowdedness of stations and travel times between stations are instrumental to decision making for operators in ensuring quality of service. Currently, available technical solutions to estimate the crowdedness of stations include: • Estimations from video footage of camera surveillance systems [7]. However, the performance of these systems depends on various external factors such as the lighting conditions inside stations. Video footage cannot be used to measure travel times between stations. • Estimations from WiFi captures [8] since Singapore’s subway stations are currently equipped with free WiFi [9]. However, this measure does not di↵erentiate between passengers on platforms and passengers on trains. Similarly, travel times cannot be estimated by WiFi captures. • Smart card system records from when passengers enter of exit the platforms of subway stations, which can be used to determine the number of boarding and disembarking at a particular station. Such records include the time of departure and arrival of each passenger. However, these data do not capture activity such as travel routes and transfer activity between lines— factors of importance when considering the crowdedness of platforms, especially in Singapore, where the density of stations and lines is high and route alternatives available to each passenger are numerous.

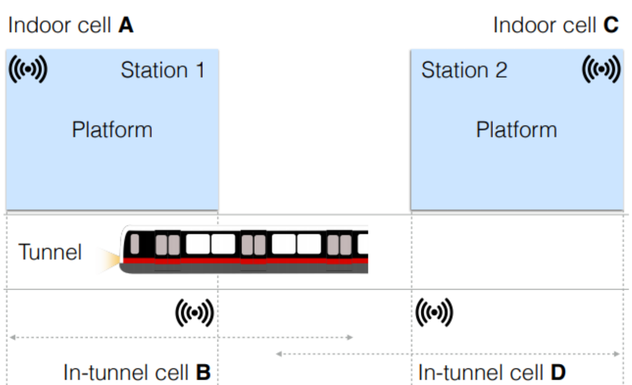


Figure 1: Dedicated indoor cell towers serve the platforms (A and C) and tunnels (B and D) of indoor MRT stations exclusively.

3.3 Enabling Smart Transit with Real-time Trip Planning

With the rise of urbanization, modern cities are facing big problems such as providing efficient and effective public transportation services. Innovative ideas in technology and citizen engagement are therefore essential to create smarter cities and improve lives. Intelligent transportation systems enable city-dwellers to quickly find optimal and relevant routes in a transit network between specified origin and destination when planning daily trips. Nevertheless, traffic congestion has always been a long-winded issue in big cities, especially during peak hours, and hence in a realistic transit network bus arrival times are not as accurate as scheduled since they are largely dependent on real traffic conditions. As a result, state-of-the-practice trip planners, which mainly rely on static schedules of public transport, may cause user dissatisfaction as the expected travel times of their recommended routes are often inaccurate in real world. Therefore, in order to provide smart traffic services in modern cities it is important to develop a practical transit trip planner that leverages both historical and real-time data to recommend routes that are adapted to traffic situations. Time-dependent characteristic of urban transit. We perform an analysis on the historical data to understand the time-dependent characteristic of bus travel time. The data are collected from real smart card integrated bus network in Singapore and comprise of the recorded usage of all bus lines in a period of three months.

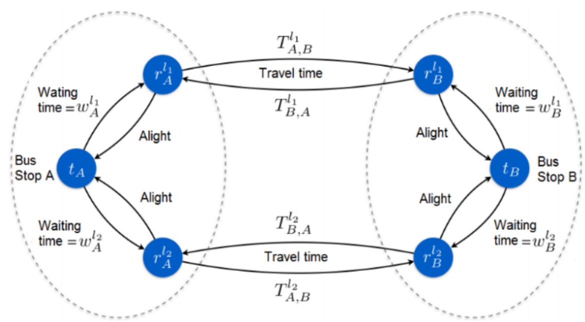


Figure 2: Time-dependent transportation graph.

3.4 Instant Graph Routing: Lightweight Graph Generation Scheme

Recently, the application services are extended. A typical WSNs consist of a large number of low powered, low cost, and low memory and computation performance, intelligent sensor devices. These sensors are involved in monitoring and measuring some industrial, militarily and other ubiquitous fields. It is one of important factor for evaluate network efficiency that decide the route for transfer packets. There are many researches about deciding a route in various and limited sensor network like AODV [10]. The graph routing is a one of source routing mechanism that adapted ISA100.11a [11] standard for wireless industrial monitoring and controlling system. The graph ID involved some sensor node ID on the one route. It makes possible that reduce the inefficient extend of routing header size. In addition, instead of routing the sensor node just forwarding the pickets using graph tables. These graphs are built and managed by the system manager, central network management unit in ISA100.11a network. However, it is difficult to change graph route and react with network re-consist flexibly. The rest of this paper is organized as follows. Section 2 outlines related work in this area. Section 3 describes our network model and assumptions and explains instant graph routing mechanism. Section 4 focuses on the comparison of the proposed topology with existing graph generation overhead in typical WSN communications.

Solution:

Graph Routing: The ISA100.11a standard is one of the sensor network application based on IEEE802.15.4 standard. In this standard, layer 2 routing called graph routing is used for the sensor network efficiency. The graph routing is based on source routing which doesn't have routing tables. Source routing has several advantages like every node does not keep the routing tables. Every address of next node in the paths are stored in the network header. However, the size of header increases when the paths get longer. So, in ISA100.11a routing mechanisms have mingled source routing with graph routing. The source routing is basic routing method of this standard and some paths are represented just by one graph ID. The graph routing plays the role that it binds every node on one route (graph) with one ID and reduces the size of a header in order to overcome this disadvantage. In every node, there exists the graph table which has next hop information of each graph ID for transmission. Node determines the next hop by comparing with graph table and packet header. The graph information and graph table are described and distributed by the system manager to each node according to each graph.

octets	bits							
	7	6	5	4	3	2	1	0
1 octet	Compress=0	Priority (Unsigned4)				DlForwardLimit (Unsigned3)		
0-1 octet	DlForwardLimitExt (Unsigned8)							
1 octet	N (number of entries in routing table; Unsigned8)							
2*N octets	Series of N GraphIDs/addresses (Unsigned16, LSB)							

Figure 3. Structure of Uncompressed DROUT Subnet header

A routing is supported from the DROUT sub-header of the data link subnet layer existing between the data link layer and network layer. Nodes communicate with system manager through the predefined alternative route. The system manager advertises to the node the route of source to destination and the node comprises the DROUT Sub-header using the received routing information A policy and the method of drawing this graph were not determined by the standard and the self-regulated graph generation which is recommended by each industry system.

octets	bits							
	7	6	5	4	3	2	1	0
1 octet	Compress=1	Priority (Unsigned4)				DlForwardLimit (Unsigned3)		
0-1 octet	DlForwardLimitExt (Unsigned8)							
1 octet	GraphID (Unsigned8)							

Figure 4. Structure of Compressed DROUT Subnet header

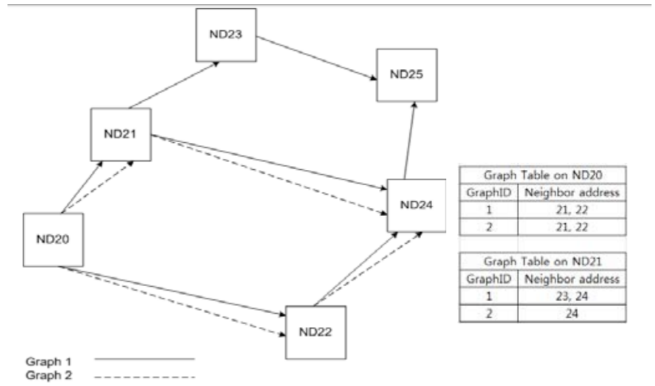


Figure 5. Example of graph routing and graph tables

Network Topology: Proposed network topology is two-tiered hybrid type of traditional mesh and star topology sensor network. In the first level, there deployed reduced functional sensor nodes (end nodes) having star topology. In the second level, there deployed fully functional sensor nodes for routing packet to the root using mesh topology. In this sensor network fields are consist with cells like hexagon network model. In the one cell, there are only one cluster headers (level 2) and many end nodes (level 1). In the level 1, they do not need any routing mechanisms. However, in the level 2, they use ISA100.11a graph routing mechanism.

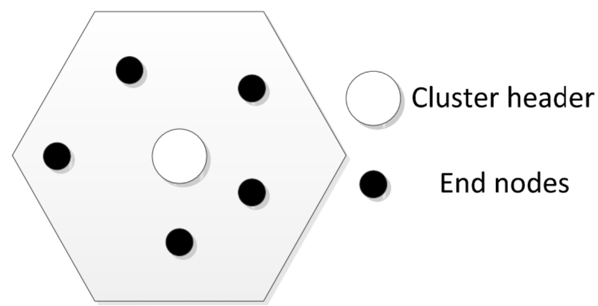


Figure 6 cell, cluster header, end node

Cluster header nodes: The Cluster header nodes have high performance for routing and deploying 16bit short address to end nodes. These routing nodes communicate cell to cell with other adjacent cluster header node. The system manager deploy the cluster headers consider regional information in each cells.

Figure 6 show 16 bit short addresses of each region based the cluster header in the cell.

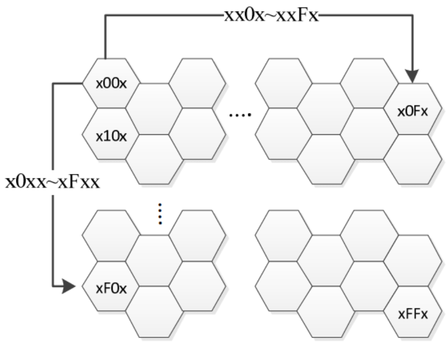


Figure 7 Regional information based cluster header ID

Route Stored Graph ID: In the proposed scheme, the graph ID is 16bit length like ISA100.11a standard. 000bit prefix – first 3bit of 16bit graph ID decide next route from network root node. 000~101 are indicating 6 adjacent cells with clock cycle. 110 and 111 means that ID is not graph ID (It is short address of one node). 00bit body – following 2bit set of 16bit graph ID decide next route from ongoing direction. 00~10 are indicate 3 on going direction cells with clock cycle. There are any route that rapidly changed path in the planar network. 11 mean that graph is finish. 0bit tail – the last 1bit indicate that graph is upstream or downstream. 0 mean downstream graph (from root to end node) and 1 means upstream graph (from end node to root) By above rules, the graph ID includes routing path information. First 3bit indicates direction from root and following 2bit shows changing of directions maximum 7hops. Last 1 bit decide the up/down stream information.

Route Stored Graph ID In the proposed scheme, the graph ID is 16bit length like ISA100.11a standard. 000bit prefix – first 3bit of 16bit graph ID decide next route from network root node. 000~101 are indicating 6 adjacent cells with clock cycle. 110 and 111 means that ID is not graph ID (It is short address of one node). 00bit body – following 2bit set of 16bit graph ID decide next route from ongoing direction. 00~10 are indicate 3 on going direction cells with clock cycle. There are any route that rapidly changed path in the planar network. 11 mean that graph is finish. 0bit tail – the last 1bit indicate that graph is upstream or downstream. 0 mean downstream graph (from root to end node) and 1 means upstream graph (from end node to root) By above rules, the graph ID includes routing path information. First 3bit indicates direction from root and following 2bit shows changing of directions maximum 7hops. Last 1 bit decide the up/down stream information.

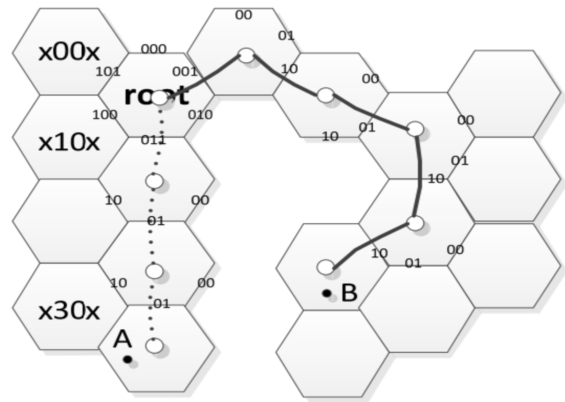


Figure 8 example of route stored graph routing

4. ORDER FINDER

4.1 Data structure

For storing the graph we used a matrix. The matrix contains Arcs, this arcs have the distance, the weight (time) and the angle with respect to the target node.

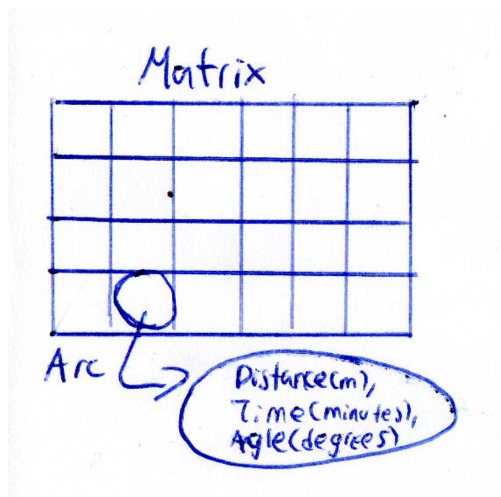


Figure 9: Matrix of arcs. An Arc is a class that contains a time, distance and angle.

4.2 Operation of the data structure

First we read the file. The first part of the data set has the information of the nodes (coordinates, id and sometimes a name). The second part of the file has the information about each connection (since we analyze complete graphs). When reading the first part of the file, we store each node's information in an object of a class named Node, this class contains a latitude, longitude, an ID and a name.

When reading the second part we calculate the distance between nodes (using the coordinates), the angle between them (using the arc sin function). When the calculations are completed we store the information as an Arc in the matrix.

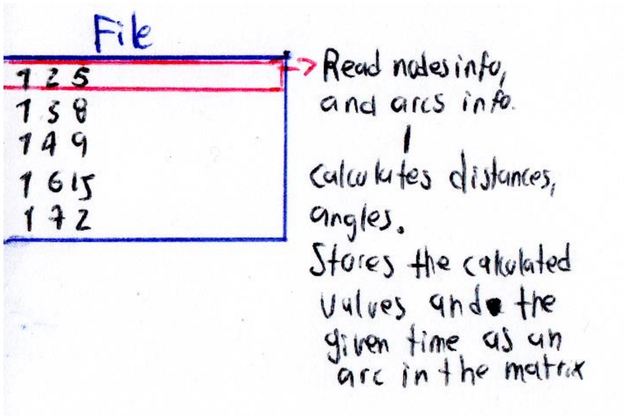
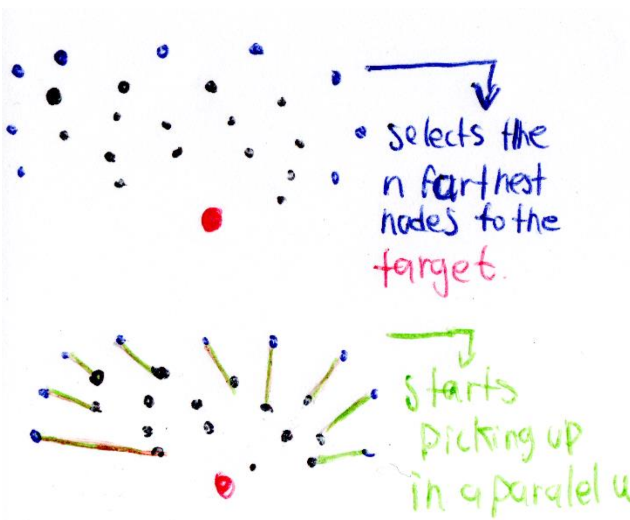


Figure 10: File reading

When calculating the way people should pick up colleges, we start from selecting the n farthest nodes (where n is number of user divided by 5), and then we start picking up the closest user, taking into account that the person to pick up can't be farther than the picker from the target, and that the angle should be as close to 0 or 90 as possible. This last part is because when the angle is close to 0 or 90 it is more likely that the person to pick up is in the regular rout of the picker, and this helps with P constraint.



4.3 Design criteria of the data structure

We decided to use a matrix to store the graph since we have a complete graph. In the matrix we store an Arc. The arcs have the distance, the weight (time) and the angle with respect to the target node. This helps out with the ordering definition part of the algorithm because we can access the information in O(1) complexity, and this is key because we have to access several times in the ordering part of the process.

4.4 Complexity analysis

Method	Complexity
Read file	O(n)
Distance between nodes	O(1)
Distance between latitudes	O(1)
Sort by distances	O(n*log (n))
Order finder	O(n^3)

Table 1: Table to report complexity analysis

4.5 Execution time

	Dataset 1 (4) ms	Dataset 2 (205) ms
Best case	0	563
Average case	0	693
Worst case	0	781

Table 2: Execution time of the operations of the data structure for each data set.

4.6 Memory consumption

	Dataset 1 (4) mb	Dataset 2 (205) mb
Memory consumption	4	45

Table 3: Memory used for each operation of the data structure and for each data set data sets.

REFERENCES

[1] A. Doan, R. Ramakrishnan, and A. Y. Halevy. Crowdsourcing Systems on the World-Wide Web. Communications of the ACM, 54(4):86–96, 2011.Francis, E., Mendez, L., Sofrony, J. Collision-free path planning in free path planning in multi-dimensional environments, Ingeniería e investigación, VOL 31, 2011, pages 5-14.

[2] To Tu Cuong. CrowdRoute: a crowd-sourced routing algorithm in public transit networks. Freie Universität Berlin, Berlin, Germany, 2013. Pages 1-2.

[3] Waze. www.waze.com, July 2013.

[4] Yahoo! News. http://news.yahoo.com/googlediscloses-paid-966-million-waze-220918812.html, July 2013.

[5] V. Shah and M. Manibo. Cities with ecient public transport highly attractive and competitive: study. http://www.eco-business.com/news/citiesefficient-public-transport-highly-attractiveand-competitive-study/, retrieved on 2015-02-02.

[6] R. Sim. Study: Singapore's public transport system one of world's most ecient. http://www.straitstimes.com/news/singapore/ transport/story/study-singapores-publictransport-system-one-wo

[7] B. Ozer and M. Wolf. A train station surveillance system: Challenges and solutions. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 652–657, 2014.

[8] L. Schauer, M. Werner, and P. Marcus. Estimating crowd densities and pedestrian flows using Wi-Fi and Bluetooth. In Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MOBIQUITOUS 2014), pages 171–177, Brussels, Belgium, 2014. ACM.

[9] Roll-out of free WiFi at 32 MRT stations to be completed by end-2015. http://www.todayonline.com/singapore/roll-out-freewifi-32-mrt-stations-be-completed-end-2015, retrieved on 2015-02-05

[10] ISA-100.11a-2009 Wireless systems for industrial automation: Process control and related applications, —http://www.isa.org/ISA100-11a

[11] Wireless Devices Specification, HCF SPEC 290 Revision 1.0. HART communication foundation. September, 2007.