

Tracer Trails Urban Transport System of the City of Villahermosa: Case of Transbus

Emmanuel Palomera May, Gerardo Arceo Moheno, Guillermo de los Santos Torres,
Martha Patricia Silva Payro, Pablo Payró Campos, and José Adán Hernández Nolasco

Division Académica de Informática y Sistemas, Universidad Juárez Autónoma de Tabasco,
Villahermosa, Mexico

emanuelpmay@gmail.com

Abstract. The city of Villahermosa, considered the gateway to southern Mexico, has experienced growth due to the economic activities of the region. This growth has not been based on an urban development plan where a suitable system of streets and avenues is contemplated, which impacts the service of the public transportation system, which consists of the mass transit system urban (vans), the rental car service (taxis), Transbus system (with routes within the city) and Transmetropolitano system (covering the metropolitan area of the city). According to the above, it is necessary to have an application support to determine the transfer of a part of the city to another in an efficient and intuitive way. The proposed tool (in the first phase only Transbus system) makes use of the PRM (Probabilistic Road Map) and Dijkstra algorithm to find a path Shuttle routes Transbus using interfaces in his version of HTML 5, jQuery, CSS and GoogleMaps, having the ability to adapt to different screen sizes of devices, thus enhancing the ways in which the user can have the tool.

Keywords: Tracer, probabilistic roadmap, Dijkstra algorithm.

1 Introduction

Transportation systems are key components in the social, economic and physical structures of an urban area [1]. In economic terms, the urban transport enables the reproduction of the labor force through the massive displacement of labor, by increasing the economies of scale and overall productivity of the city. Town planning, urban transport has effects on the size and configuration socio - spatial city. Culturally, enables different social relations to the strictly productive and generates spaces in which citizens can present and imagine the city and others [2].

The city of Villahermosa, as the capital of Tabasco state, receives a large tributary of people from different parts of the state, or outside, being necessary to provide adequate means of transport to travel within the city, in addition to signal properly routes and stops covering these transportation services; between these services is the so-called Transbus.

At its inception, the service Transbus consisted of 210 buses equipped with air conditioning and television screen, which replaced 1,293 vans with over 12 years providing the service. Having not only the challenge of modernity urban public transport but expedite the move within the city of Villahermosa, it is important to mention that in just four years of operation Transbus system has mobilized nearly 122 million users until December 2012, mobilizing about 150 thousand users daily [3].

Moreover, as published by the INEGI [4], Tabasco in 2011, less than 35% of the population had access to computers, less than 30% had access to fixed Internet services and only 16% have access in their households; however, 56% have access to mobile phone services and 99.1% of households have an internet connection broadband.

Although no official data updated by gender and age, in a poll by Consulta Mitofsky [5] shows that 56% of men, 51% of women and two of every three Mexicans under 30 have access to a cellular phone. As for the age range 18 to 29 years, 61% said they know all the features of your cell phone [6].

In the National Development Plan (NDP) promotes sustainable urban mobility projects supported by public and mass transit, and promotes the use of non-motorized transport [7]. Thus, following the guiding principles of the NDP, in the State Development Plan (EDP) arises stimulate sustainable human mobility options that favor the use of non-polluting means of collective transport [8].

In this way having the commitment to promote the use of public transport and the growth media in the state of Tabasco, the idea of having a tool based on information technology, which allows the method to move within the city of Villahermosa through service offering Transbus units, allowing greater utilization from persons who, supported by the tool, Transbus can make use of economic resources. The development was guided by adaptive software development methodology to implement the probabilistic roadmap method for obtaining results.

2 Description of the Method

2.1 Methodological Approach

We used an action research approach. The purpose of it is solve everyday problems and immediate; it is understandable to make the social world and to improve the quality of life of people and immediate solving [9]. This is defined as "the study of a social situation with a view to improving the quality of action within it". It involves understanding the teaching office, integrating reflection and intellectual work in the analysis of the experiences carried out. Similarly, Sampieri et al. [10] define it as a "small-scale intervention in the functioning of the real world and the thorough examination of these interventions".

The sources of information were the primary and secondary type. The instruments of data collection were unstructured interviews made to the officer in charge of public transport area and documentation provided by the Ministry of Communications and Transport of the State of Tabasco (SCT).

2.2 Development Methodology

The methodology used for development is Adaptive Software Development (ASD) that is an agile methodology that has the following characteristics [11]:

- Iterative.
- Cyclic.
- Fault tolerant.
- Guided by the risks.

The review of the components allow us to learn from mistakes and not force commitments to restart the development cycle. This methodology consists of three stages (speculate, collaborate and learn), which are shown in Figure 1, same to be used repeatedly until the routes obtained are satisfactory in quality testing.

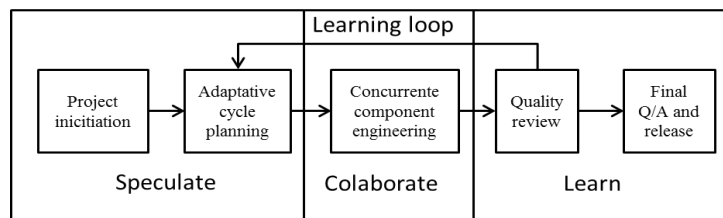


Fig. 1. Description of our methodology.

2.3 Probabilistic Roadmap

As Kaviraki [12] mention, probabilistic roadmap method (PRM) has two phases: the learning and the query, each one comprised of different steps. The PRM method was implemented in conjunction with the Dijkstra algorithm to offer the user the shortest route to your destination:

- Learning phase
 - Mapping routes of Transbus,
 - Correction and debugging routes,
 - Connect the nodes generated.
- Query phase
 - Determine the start and end point,
 - Switch start and end with the roadmap generated,
 - Apply Dijkstra's algorithm,
 - Show results.

First, mapping routes of Transbus is performed to obtain a map that has paths routes. Post-mapping and integrating routes in a single environment, we corrected the routes by eliminating irrelevant points. Then, we proceed to enter the vicinity of the nodes generated within the database for later use in the consultation phase.

Already developed the roadmap is possible to make queries, i.e., ask if there is a way to move from a start point to an end point. The start point and end point are determined depending on the proximity there is a node in the roadmap.

To run a query, first the initial and final roadmap, after this the Dijkstra algorithm is implemented to obtain the shortest path between nodes connected configurations. Finally, the sequence of nodes becomes a feasible route, that is, the path consists of three sub-paths connecting the initial configuration to a node in the roadmap, the sub-path present in the roadmap between the two connection nodes and the sub-path connecting a node in the roadmap to the final configuration. This is shown through the interfaces previously elaborated.

3 The Learning Phase

3.1 Mapping Routes of Transbus

Transbus routes were obtained from different sources such as page Juarez Autonomous University of Tabasco provides students and the general public a means to know the Transbus routes through a PDF file. In figure 2 shown the M1 route comprising of Pino Suarez Market Ixtacomitan.



Fig. 2. Route M1 Ixtacomitan.

With the routes already obtained, we proceeded to develop the course of each route using the lite version of Google Maps Engine, as shown in Figure 3.

3.2 Correcting and Debugging Routes

With the possibility that the user can observe the path of the individual paths, the route was elaborated in detail using a high amount of nodes in each route, including the direction of each route.

The number of nodes used on each route is decreased by replacing the segments where the journey is made in both directions, with one segment that represents both directions through adjacencies in the next step of the PRM.

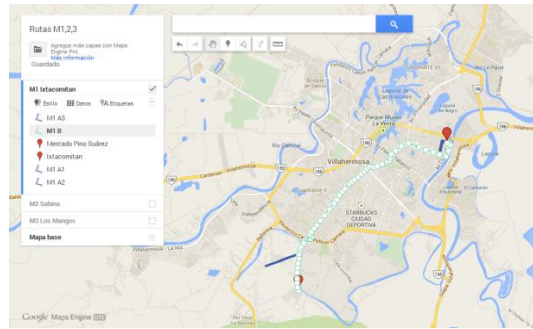


Fig. 3. Development of the route M1.

Many nodes in the path are removed, leaving only the nodes representing established by the Transbus stops, which are located at an approximate distance of 200 feet between them. Decreasing the number of nodes involved in the roadmap, decrements the running time Dijkstra algorithm, which is shown in Figure 4.

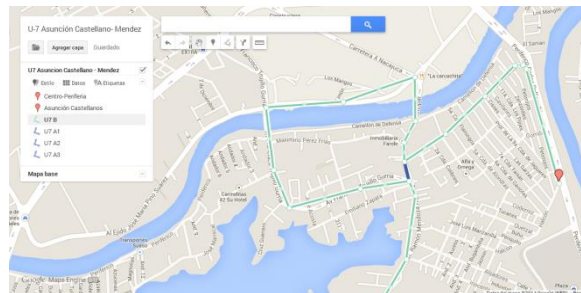


Fig. 4. Debug output node.

Having made all the routes, we proceeded to perform an integration of the 30 routes under a single environment, because Google Maps Engine imposes a limit on the number of layers available for use in the lite version. The integration was developed by modifying routes KML files generated for each route, extracting code segment with the sequence of nodes of a route identified with the <LineString> shown. The result of the integration can be seen in Figure 5.

After performing the integration of routes, were grouped according to with the sense, i.e., the group where the tour takes place from the periphery to the center and the other group where the tour takes place from the center to the periphery of the city, obtaining a map as shown in Figure 6, which was used to identify existing adjacency between the nodes.

Once the integration and aggregation of the routes are done, the route is cleared in a single layer to unify the paths where it passes over other routes, therefore, simplifying the roadmap. The result is seen in Figure 7.

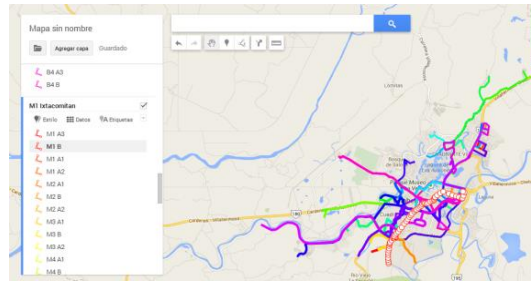


Fig. 5. Integrating routes.

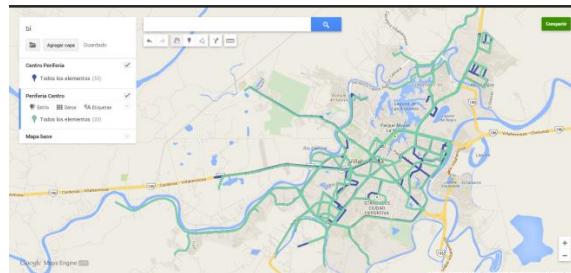


Fig. 6. Grouping routes.

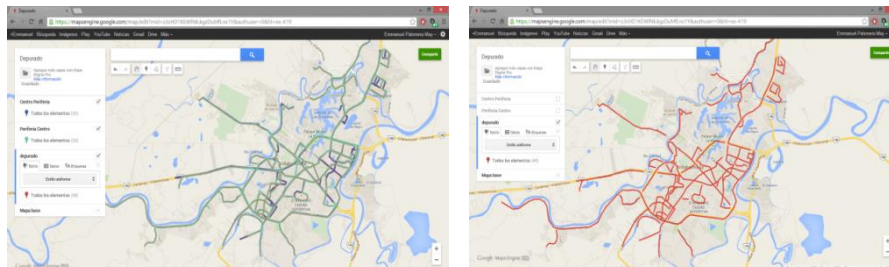


Fig. 7. Debugging nodes.

Once the debugging of route is done, the KML file containing the roadmap is processed with a tool available online (kml2x), converting the KML file to a plain text file that can be handled more easily.

3.3 Connecting Nodes

The adjacency of the nodes was determined according to the senses of Transbus routes and was introduced to the database. Using PHP to generating an array in

JavaScript that contains the location of the nodes, latitude and longitude, as well as adjacencies. It is possible to display input and adjacent nodes on a map as shown in Figure 8.

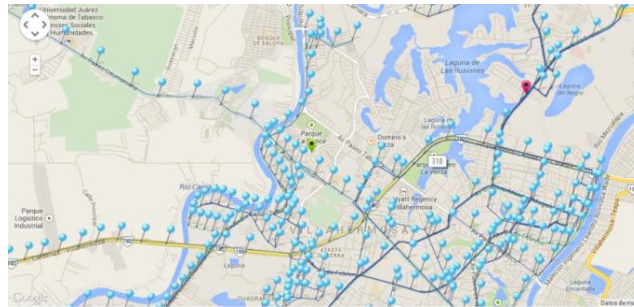


Fig. 8. Nodes on the map.

4 The Query Phase

4.1 Setting the Start and End Points

The user -through the user interface- determines the start and end points by using the integrated library of Google Maps Place. It has a feature called autocomplete that shows a list of the places that match the name you enter. It is noteworthy that the places listed are located solely within the city of Villahermosa (see Figure 9).

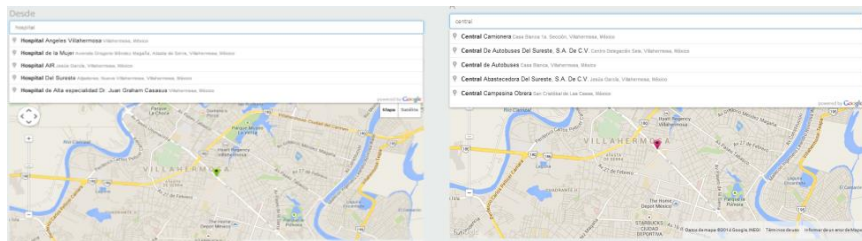


Fig. 9. Setting the start and end points.

The user adjusts the start and end points by moving the marker in the map in order to increase the accuracy of the results.

4.2 Connecting Start and End Points with the Roadmap

To connect the start and end points given by the user to the roadmap, is to determine the closest node to the start and end points, having to calculate the distance between all nodes with respect to the start point and go always keeping the node with the smallest distance. The pseudocode for determining the nearest node is shown in Figure 10.

```
1  program begin
2  var distance, tmp = infinity, closer, u
3  u = lat and lng
4  start for cycle i < numNodes
5  distance = distance between nodes u and [i]
6  if distance < tmp
7  tmp = distance
8  closer = i
9  end if
10 increase i
11 end for cycle
12 program end
```

Fig. 10. Pseudocode for determining the nearest node.

4.3 Applying the Dijkstra Algorithm

After determining the start and end points in the roadmap, Dijkstra's algorithm is run. The algorithm uses as an input the start and end values determined in the previous step, obtaining the identifier of the nodes that are present in the shortest path between them. After obtaining the identifier of the nodes, we proceed to generate a sequence with the location of the nodes, i.e., latitude and longitude of each node on the path generated to be represented to the user via a map (see Figure 12). The way to proceed is shown in Figure 11.

```
1  program begin
2  var x = determine start node
3  var y = determine end node
4  run Dijkstra (x, y)
5  path obtained from x, and y
6  show path between x, y
7  show path between start - x
8  show path between end - and y
9  program end
```

Fig. 11. Algorithm for generating the sequence of location nodes.

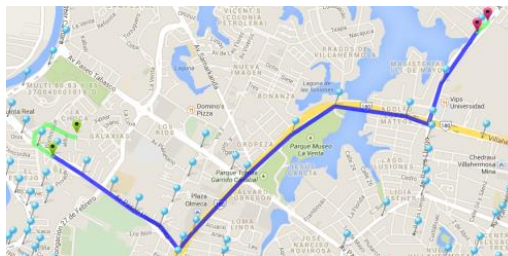


Fig. 12. Shortest path.

Distance between nodes is used as the only criterion to determine the shortest path, as the service has Transbus lanes on the avenues more vehicular affluent. Also, it is not possible to determine the transfer time since the speeds are not constant, nor are monitored.

4.4 Show Results

Finally, it should show the user the path to follow from the start point to the starting node in the roadmap and the way forward between the end node and the end point roadmap introduced by the user (see Figure 13).

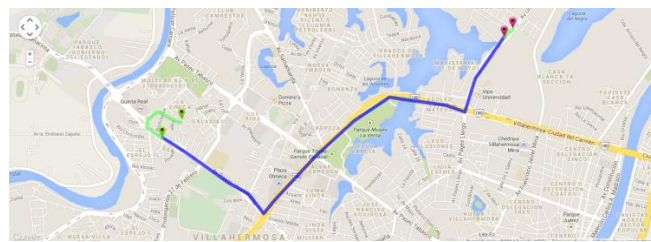


Fig. 13. Result displayed to the user.

Besides showing graphically the shortest path between the start and end nodes, also displayed in textual sequence of steps to go from source to destination (see Figure 14).



Fig. 14. Route instructions.

4.5 Performance Results

A roadmap consisting of 530 nodes representing the path of the 30 service routes Transbus was generated. The roadmap adjacency node was pointing in both directions, indicating that each segment of the roadmap could go either way, i.e., center to periphery or periphery to center. This roadmap primarily benefit was the processing time, which averaged 233 milliseconds. However, it was not possible to determine the paths involved in the route generated so a second roadmap which had 1041 nodes was developed. The execution time was not significantly affected: the average time was 273 milliseconds, i.e., there was only an increase of 40 milliseconds processing time, which is irrelevant in view of the increase in the number of nodes.

The major problem of the roadmap of 1041 nodes was that if the tour began in a segment where the addresses were not separated, the identification of appropriate routes for the transfer was not done correctly. That is the reason for separating the management of the routes in the entire roadmap. So, 1325 roadmap nodes were separate of the paths and more accurate results were added. The processing time for the 1325 nodes test runs was an average 279 milliseconds, i.e., there was an increase of only 6 milliseconds with respect to the processing time of the roadmap of 1041 nodes and an increase of 46 milliseconds relative to the roadmap 531 nodes. By using this roadmap, unlike the previously generated, it was possible to correctly identify Transbus routes present in the path.

Table 1. Performance tests.

| Number of Nodes | Execution time |
|-----------------|------------------|
| 530 | 233 milliseconds |
| 1041 | 273 milliseconds |
| 1325 | 279 milliseconds |

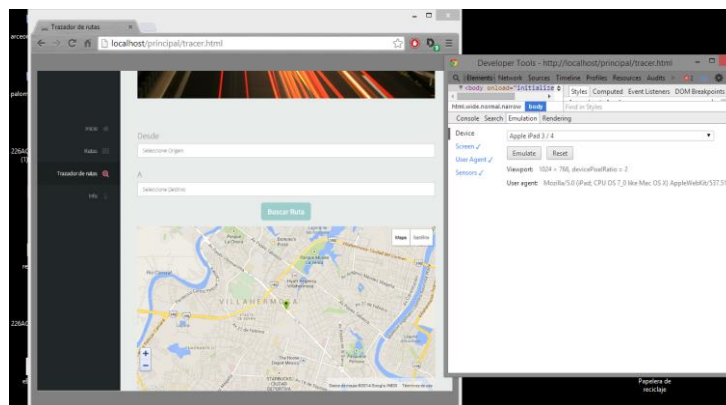


Fig. 15. Interface adapted to the dimensions of an iPad.

5 Interfaces

Interfaces were developed using Html5, CSS and jQuery, getting results in a site that conforms to the resolution of the device from which it is viewed. As seen in Figure 15, the interface adapts to the screen resolution of an iPad.

6 Conclusions

The solution proposed in this paper describes how we can apply the method of probabilistic maps in conjunction with the Dijkstra algorithm, to solve the problem of travel within the city of Villahermosa using the routes offered by Transbus. Among the benefits that users can contribute to mention the ease of use of the site, the pleasant manner in which indications are provided for transfers using Transbus routes, proper way to adapt interfaces different devices and that the site is freely accessible via the Internet.

In tests, users felt satisfied with the results generated by the site, as of today there is a similar tool to make transfers within the city of Villahermosa using the public transport system. The solution is applicable to other means of transportation, and while the introduction of routes does not occur automatically but manually, the results displayed to the user are performed properly.

There is also the option to develop a mobile application, although better to control the user's experience, they are costly to develop and are complicated to migrate platforms or operating systems as appropriate.

References

1. Krishna Rao, K. V., Muralidhar, S., & Dhingra, S. L.: Public transport routing and scheduling using Genetic Algorithms. In 8th international conference on computer aided scheduling of public transport, Berlin, Germany (2000)
2. Canclini, N. G., Castellanos, A., & Mantecón, A. R.: La ciudad de los viajeros: travesías e imaginarios urbanos, México, 1940-2000. Grijalbo (1996)
3. De la Cruz, A.: Movilización Transbus a 122 millones de usuarios en 4 años. *De Facto*. http://www.defacto.com.mx/v2/sureste_notas.php?id=7186 (2012)
4. INEGI (Instituto Nacional de Estadística, Geografía e Informática). Estadísticas sobre disponibilidad y uso de tecnología de información y comunicaciones en los Hogares, 2011. México, INEGI (2012)
5. Consulta Mitofsky.: La generación milenio... Cuando el destino nos alcance (2013)
6. Consulta Mitofsky. México: usuarios de telefonía celular (2013)
7. Peña-Nieto, E.: Plan Nacional de Desarrollo. Diario Oficial de la Federación. México: Secretaría de Gobernación (2013)
8. Núñez, A.: Plan estatal de desarrollo. Gobierno del Estado de Tabasco (2013)
9. Álvarez-Gayou, J. L.: Cómo hacer investigación cualitativa. Fundamentos y metodología. Métodos básicos. Ed. Paidós. México, 127-8 (2005)

Emmanuel Palomera May, Gerardo Arceo Moheno, Guillermo de los Santos Torres, et al.

10. Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P.: Metodología de la investigación. México: Editorial Mc Graw Hill (2010)
11. Highsmith, J.: Adaptive software development: a collaborative approach to managing complex systems. Addison-Wesley (2013)
12. Kavraki, L. E., Svestka, P., Latombe, J. C., & Overmars, M. H.: Probabilistic roadmaps for path planning in high-dimensional configuration spaces. Robotics and Automation, IEEE Transactions on, 12(4), pp. 566–580 (1996)