A Similitude Algorithm through the Web 2.0 to Compute the Best Paths Movility in Urban Environments

Christian J. Abrajan[1], Fabian E. Carrasco[1], Adolfo Aguilar[1], Georgina Flores[1], Selene Hernández[1], and Paolo Bucciol[2]

[1] DSC, Instituto Tecnológico de Puebla,
Av. Tecnológico 420, Maravillas 72220, Puebla, Mexico
[2] French-Mexican Laboratory of Informatics and Automatic Control,
Ex Hacienda Sta. Catarina Mártir s/n, 72820 S. Andrés Cholula, Pue., Mexico
{christian.abrajanf,node.fecc,adolforico2,kremhilda,
hdezrod}@gmail.com
bucciol@ieee.org

Resumen In this paper we present a similitude algorithm based on fuzzy relations to support the movement of a user of the Urban Public Transportation System (UPTS) in the Puebla City. The algorithm computes the best paths in order to support and to optimize the user mobility within urban environments based on three QoS metrics: spatial distance, security and number of transfers. The algorithm feedback uses the knowledge gained through the Web 2.0 to allow the user to query and to exchange experiences. This virtual system incorporates: a decision algorithm of the best paths, search algorithms and fuzzy relations algorithms of the UPTS, in order to benefits local and foreign travelers of the Puebla City or cities with similar characteristics.

Key words: Fuzzy Classification, Similarity Relations, Public Transportation, Shortest Path Algorithms, Shortest Path Optimization, Personal Safety, Web 2.0.

1. Introduction

In this paper a set of algorithms to find the best mobility paths using the Urban Public Transportation System (UPTS) in the City of Puebla is proposed. The objective of our approach is to support the UPTS user in order to provide the best transfer options with respect to: the shortest distance, the lowest cost (number of transfer) and the higher level of security. Our approach is supported on Web 2.0, enabling users to access and exchange knowledge taking advantage of interoperable standards. Our work is based on a novel approach to find the best path between starting and destination points of UPTS users. The path search and selection algorithm choose the best path based on road safety constraints and on a fuzzy classification algorithm, with the ultimate goal to improve the safety of the UPTS users. In Puebla, as happens in major cities, personal safety is of utmost importance to visitors and residents. Helping them to choose the safest mobility path is therefore one of the primary goals that a modern public transportation system has to achieve. However, path selection algorithms for public transportation systems are actually based on only two types of metrics: temporal (time to reach the destination) and spatial metrics (distance to reach the destination). The case of the Puebla city is even worse. Information on the public transportation system is provided by means of leaflets sold in newspapers stores. Moreover, such information is generally inconsistent and vague. For this reason, UPTS users take decisions on their mobility paths based on intuition and incomplete information, often wasting more resources than needed (such as time and money) and putting into risk their physical safety. The proposed framework has the objective of providing information on the possible mobility paths between various locations in the city, taking into consideration the urban bus routes and safe transfer points. The remainder of this paper is organized as follows: Section 2 presents the state of the art. The theoretical framework is discussed in Section 3. Section 4 describes the architecture of the virtual web platform. Section 5 presents the path search and classification algorithms. Section 6 presents the simulation testbed and the results of the simulation. Finally, Section 7 concludes the paper.

2. State of the Art

2.1. Other Similar Applications of Fuzzy Classifications Algorithms

Fuzzy classification is used in several applications like medicine [3][4], urban remoting sensing [5], intrusion detection systems [6], among other. Some classifiers are based on several approaches, some of them are: fuzzy rules [7][8], evolutionary algorithms [6], maximum–likehood classification [5] and neuro–fuzzy models [9]. In this paper, a fuzzy classifier based on similarity relations is used to select from a set of paths that meet desired characteristics (spatial distance and number of transfers) those which meet certain restrictions (such as security level).

2.2. Similar Works

Páginas Amarillas:[3] Páginas amarillas is a Spanish site that has a section called *Callejero* that focuses on helping visitor and citizens of Spain. It provides information on traffic and allows real-time viewing of traffic webcams.
ViaDF:[4] ViaDF is a route planning web site for the public transportation system in México City, Distrito Federal. The system minimizes the path cost between two points based on the information of the whole public transportation network (Metro, Metrobus, bus) in the city.

[3] http://callejero.paginasamarillas.es
[4] http://www.viadf.com.mx/

EL UNIVERSAL:[5] El Universal is a national newspaper that has developed a software tool for anonymous complaint via Internet when threats to personal safety, such as assaults, occur. The site of EL UNIVERSAL uses the Google Maps API [10] to show the place where the threat occurred, with the long-term goal of raising public awareness and preventing more threats in dangerous locations.

3. Theoretical Framework

3.1. *K* Shortest Path

"The K shortest paths"means K loopless paths from the origin to the sink that have the shortest lengths, and "the Kth shortest path"means the last of "the K shortest paths."[2]

The K-th Shortest Path Problem consists on the determination of a set $\{p_1, \ldots p_k\}$ of paths between a given pair of nodes when the objective function of the shortest path problem. That is, not only the shortest path is to be determined, but also the second shortest, the third shortest, and so up to the K-th shortest path.

3.2. Fuzzy Classification

The fuzzy classification could be based on Fuzzy Equivalence Relations. A fuzzy relation, $\underset{\sim}{R}$, on a single universe X, maps elements of X to X through the Cartesian product, where the strength of the relation between (x_1,x_2) ordered pairs of X is measured with a membership function $\mu_{\underset{\sim}{R}}(x_1, x_2) \in [0, 1]$. The Cosine Amplitude [1] is a useful similarity method to assing values to a fuzzy relation when a set of m data samples, $X = \{x_1, x_2, ..., x_m\}$, should be compared with each other. If each of the m data samples, x_i, is characterized by a set of n attributes, $x_i = \{x_{i_1}, x_{i_2}, ..., x_{i_n}\}$, the Cosine Amplitud method computes r_{ij} as

$$r_{ij} = \frac{\|\sum_{k=1}^{n} x_{ik}x_{jk}\|}{\sqrt{(\sum_{k=1}^{n} x_{ik}^2)\left(\sum_{k=1}^{n} x_{jk}^2\right)}}, \quad (1)$$

where $i, j = 1, 2, ..., m$ and $0 \leq r_{ij} \leq 1$; the resulting fuzzy relation $\underset{\sim}{R}$ is reflexive ($\mu_{\underset{\sim}{R}}(x_i, x_i)$=1) and symmetric $\mu_{\underset{\sim}{R}}(x_i, x_j) = \mu_{\underset{\sim}{R}}(x_j, x_i)$). In order to ensure that $\underset{\sim}{R}$ is an equivalence relation, the transitivity, given by

$$\mu_{\underset{\sim}{R}}(x_i, x_j) = \lambda_1 \text{ and } \mu_{\underset{\sim}{R}}(x_j, x_k) = \lambda_2, \text{ then } \mu_{\underset{\sim}{R}}(x_i, x_k) \geq min\{\lambda_1, \lambda_2\}, \quad (2)$$

is achieved by at most $m - 1$ fuzzy max–min composition of $\underset{\sim}{R}$, where the fuzzy max–min composition of two fuzzy relations $\underset{\sim}{R}$ and $\underset{\sim}{S}$, $\underset{\sim}{T} = \underset{\sim}{R} \circ \underset{\sim}{S}$, is computing by $\mu_T(x_1, x_3) = max_{x_2 \in X}(min(\mu_R(x_1, x_2), \mu_R(x_2, x_3)))$.

[5] http://www.eluniversal.com.mx/graficos/graficosanimados10/EU_mapa/mapa.html

Now, if $\underset{\sim}{R}$ is a fuzzy equivalence similarity relation, the data relationed in $\underset{\sim}{R}$ can be classified applying a Lambda–Cut[1] (λ–cut) as follows: given a λ, where $0 \leq \lambda \leq 1$,

$$R_\lambda = \left\{ x | \mu_{\underset{\sim}{R}}(x) \geq \lambda \right\}, \tag{3}$$

then R_λ contains the equivalence classes of the data samples X.

4. Web 2.0-based Virtual Web Architecture

The use of Information and Communication Technologies (ICT) is a very promising field to solve the routing issues (or at least minimize their impact) of public transportation systems. On one side, it aims to obtain information on the safety of the UPTS system through its most active and dynamic resource, that is, their users. On the other side, it aims to assist UPTS users in the planning of their mobility paths based on safety constraints. Through the design of an ad-hoc web 2.0-based virtual web architecture we propose to improve the information exchange processes between the users of public transportation systems and the correlated governmental agencies such as Secretaria de Comunicaciones y Transportes, Procuraduría General de Justicia, Instituto Nacional de Estadística, Geografía e Informática. The main goal of the platform is twofold: (i) to ease the feedback procedures from the users and (ii) to provide standardized mechanisms to take advantage of the feedback information in the UPTS users' path selection processes.

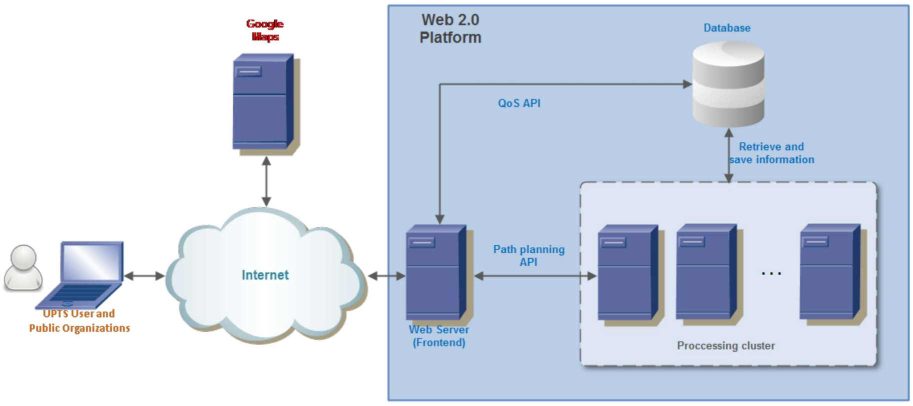

Figura 1. Architectural diagram of the proposed web platform.

The global architecture of the virtual web platform (see figure 1) is composed of:

Web services provider The web services provider is the system front-end to the user. It guarantees the correct service execution through two main API's: the Quality of Service API and the SafeTraveler API.

Algorithm 1 Yen's Algorithm that generates all shortest paths.

Input: s source, t target, P adjacency matrix.
Output: Q paths.

```
For every arc u − v on P:
ăăă ăRemove u − v and all nodes preceding u in P from graph.
ăăă ăăP'_1 ← subpath from s to u in P.
ăăă ăăP'_2 ← shortest path from u to t in modified graph.
ăăă ăăP' ← append P'_2 to P'_1
ăăă ăăAdd P' to Q.
ăăă ăăRestore graph.
```

Processing cluster The processing cluster performs the system processing tasks. In particular, it pre-processes the user queries and executes the real-time path selection algorithms.

Database Within the database the pre-processed information on the most queried paths is stored. This information is constantly updated with the results of the new queries.

Traveler This is the user that will (i) query and (ii) share information with the system.

The platform also includes web 2.0-based mechanisms such as polls and fora (not shown in the figure) to easily obtain feedback from the UPTS users on the quality of the public transportation systems, and interfaces (Web Services) to public organizations to share information related with the UPTS system.

5. Search and Classification Algorithms of Public Transport Path

5.1. Yen's Algorithm

Algorithm of Y. Yen [2] was used to find the shortest routes to reach a destination.

Yen's algorithm calculates new paths from start node s to target node t based on a path P as follows in algorithm 1.

The algorithm of Yen has a time complexity of: $O(kn(m + nlogn))$ with n the number of nodes and m the number of arcs. [11]

5.2. Transfer Algorithm

The general complexity of the algorithm 2 is: $O(\prod_{i=0}^{Nt}(2Nr * (Nn/2^i))$
where:
Nt: Number of transfers Nr: Number of routes Nn: Númber of nodes to go
For the case study, we have $Nt = 2$
$O(Nn^3 * Nr^3)$
where Nr is a small value.

Algorithm 2 Algorithm that generates the possible transfer options at UPTS for a road and number of transfers given.

Globals: $Rutas$, m lists of nodes of the UPTS paths,
$Path$, list of n nodes with the path from source A to destination B
Input: $indice$, current index of $Path$ list
$results$, array of k triplets that contain $Rutas_x$, $Path_{abordo}$,$Path_{descenso}$
$transbordos$, transfer number allowed
Output: R, l type-$results$ arrays that contain the possible transfers
Base Case If $indice = Path_B$ add $results$ to R and return
Pruning Case If $transbordos = 0$ return
Search Recursive Case For each $r \in Rutas$
identify those containing a $(Rutas_{r,i}, ..., Rutas_{r,i+n})$
list equals to $(Path_{indice}, ..., Path_{indice+n})$,
where $n \geq 1$, it means, selecting all lists with more than one element,
creating for each list a triplet $Rutas_r, Path_{indice}, Path_{indice+n}$ in $results$.
Finally, the $TransferAlgorithm(indice + n, results, transbordos - 1)$ is called.

Algorithm 3 Algorithm that computes the equivalence classes of a data sample.

Input: X, $m \times n$ sample data matrix of m paths from A to B,
each one of n attributes
λ, the restriction of classification of m paths
Output: R_λ, $m \times m$ equivalence classes matrix of the m paths
1. Compute the similarity matrix $\underset{\sim}{R}$ using (1)
2. Verify if $\underset{\sim}{R}$ is transitive in accordance with (2)
3. If $\underset{\sim}{R}$ is not an equivalence relation
then compute $\underset{\sim}{R} = \underset{\sim}{R} \circ \underset{\sim}{R}$ at most $m - 1$ times
4. Compute the equivalence class matrix R_λ using (3)

5.3. Fuzzy Classification Algorithm

In order to compute equivalence classes of the paths resulting from the transfer algorithm 2, the algorithm 3 is presented. This algorithm applies the Cosine Amplitude method (step 1) to compute the similarity among the paths given by algorithm 2. In steps 2 and 3 the similarity matrix obtained is transformed into an equivalence relation matrix applying the max–min composition. Finally, in accordance with a restriction level (λ) with respect to safety, distance and transfer numbers given by user, in the step 4 a defuzzification process (λ–cut) is applyed to equivalence relation matrix, obtaining the classification of paths that allow a user to move from point A to point B in Puebla City using UPTS.

Since the complexity of steps 1, 2, 3 and 4 of the Algorithm 1 is $O(mn^2)$, $O(m^3)$, $O(m^4)$ and $O(m^2)$, respectively, the time complexity of algorithm 1 is $O(m^4)$, where m is the paths number from A to B.

6. Results

In the following example (Figure 2), 21 reference points in a part of the City of Puebla are listed in the Table 1. The entry data required by the Virtual System are: origin, destination and security parameters.

Figura 2. Nodes in the graph

The routes and the nodes shown in Table 2 are done by bus lines. This list of nodes follow a defined sequence as mentioned in the Table. Boardings and transfers the UPTS user needs to make to complete his journey, from the origin to a destination, as well as the route to take, are calculated using the information of the routes given in the graph in Figure 2.

In some cases the list in Table 2 includes 0's just to have the same length in each line of the route.

For example: a user needs to reach from the origin node 1 (CAPU) to the destination node 10 (Pza. Dorada). To do so, he has up to 6 possible routes (k-shortestPath) and a maximum of 3 transfers to make.

The obtained results are shown in Table 3 where: **Path** n: is a numeric sequence which identifies each node, and the order that needs to be followed to go

Cuadro 1. List of Nodes in the graph.

Nodo	Nombre	Nodo	Nombre
1	CAPU	12	Karts
2	Pza. San Pedro	13	China Poblana
3	H. San Alejandro	14	H. San José
4	Reforma	15	Centro Convenciones
5	Fuente Los Frailes	16	Reforma y Blvrd. 5 Mayo
6	25 pte y TELMEX	17	CENCH
7	31 pte Marriot	18	Diagonal y 11 Nte
8	Hosp. Universitario	19	Casa del Abue
9	31 pte y 11 sur	20	Museo Ferrocarril
10	Pza. Dorada	21	Paseo Bravo
11	Soriana		

Cuadro 2. List of Routes.

Route	**Way**
R1	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 0 - 0 - 0 - 0 - 0 - 0
R2	10 - 9 - 8 - 7 - 6 - 5 - 4 - 3 - 2 - 1 - 0 - 0 - 0 - 0 - 0 - 0
R3	1 - 11 - 12 - 13 - 14 - 15 - 16 - 17 - 10 - 0 - 0 - 0 - 0 - 0 - 0- 0
R4	10 - 17 - 16 - 15 - 14 - 13 - 12 - 11 - 1 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R5	8 - 7 - 6 - 5 - 4 - 3 - 2 - 1 - 11 - 12 - 13 - 14 - 15 - 16 - 17 - 10
R6	1 - 11 - 18 - 19 - 20 - 21 - 9 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R7	9 - 21 - 20 - 19 - 18 - 11 - 1 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R7	15 - 20 - 21 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R8	21 - 20 - 15 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R9	20 - 21 - 4 - 5 - 6 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0
R10	6 - 5 - 4 - 21 - 20 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0

Cuadro 3. Results of k-shortest path and transfers.

Path 1:	$1-11-12-13-14-15-16-17-10$			Weight:	$20u$
Sol 1:	$3-1-10$	Sol 2:	$5-1-11$ $3-11-10$	Sol 3:	$5-1-12$ $3-12-10$
Sol 4:	$5-1-13$ $3-13-10$	Sol 5:	$5-1-14$ $3-14-10$	Sol 6:	$5-1-15$ $3-15-10$
Sol 7:	$5-1-16$ $3-16-10$	Sol 8:	$5-1-17$ $3-17-10$	Sol 9:	$6-1-11$ $3-11-10$
Sol 10:	$6-1-11$ $5-11-12$ $3-12-10$	Sol 11:	$6-1-11$ $5-11-13$ $3-13-10$	Sol 12:	$6-1-11$ $5-11-14$ $3-14-10$
Sol 13:	$6-1-11$ $5-11-15$ $3-15-10$	Sol 14:	$6-1-11$ $5-11-16$ $3-16-10$	Sol 15:	$6-1-11$ $5-11-17$ $3-17-10$
Path 2:	$1-2-3-4-5-6-7-8-9-10$			Weight:	$24u$
Sol 1:	$1-1-10$				
Path 3:	$1-2-3-4-21-9-10$			Weight:	$26u$
No solutions for Path 3 with 3 transfers.					
Path 4:	$1-11-18-19-20-15-16-17-10$			Weight:	$28u$
Sol 1	$6-1-20$ $9-20-15$ $3-15-10$				
Path 3:	$1-11-18-19-20-15-16-17-10$			Weight:	$30u$
No solutions for Path 6 with 3 transfers.					
Path 6:	$1-11-18-19-20-21-9-10$			Weight:	$30u$
Sol 1:	$3-1-11$ $6-11-9$ $1-9-10$	Sol 2:	$5-1-11$ $6-11-9$ $1-9-10$	Sol 3:	$6-1-9$ $1-9-10$

from the origin node to the destination node in each of the routes.
Weight: is the path weight given in "u" units.
Sol n: contains a group of (R, O, D) triplets, where R indicates the route to board, O is the bus stop R, and D is either a bus stop to transfer or the destination. It is worth mentioning that the order of the triplets in the result list is the sequence in which the buses need to be boarded.
The 20 found solutions are mapped obtaining a $m \times n$ matrix.

Given the following $m \times n$ matrix $X = [X1 \;\; X2]$, of $m = 20$ (columns) paths from A (CAPU) to B (Pza. Dorada), each one of $n = 5$ security levels: total, high, medium, low and poor (rows):

$$X1 = \begin{bmatrix} 0 & 0{,}33 & 0 & 0 & 0 & 0 & 0 & 0 & 0{,}33 & 0{,}25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0{,}50 & 0{,}33 & 0{,}67 & 0{,}67 & 0{,}67 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}50 \\ 0 & 0 & 0 & 0 & 0 & 0{,}33 & 0{,}33 & 0{,}33 & 0 & 0 \\ 0{,}50 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}33 & 0{,}25 \end{bmatrix},$$

$$X2 = \begin{bmatrix} 0{,}25 & 0{,}25 & 0{,}25 & 0{,}25 & 0{,}25 & 0 & 0 & 0{,}25 & 0{,}25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0{,}50 & 0{,}50 & 0{,}25 & 0{,}25 & 0{,}25 & 0{,}50 & 0{,}50 & 0{,}50 & 0{,}50 & 0{,}67 \\ 0 & 0 & 0{,}25 & 0{,}25 & 0{,}25 & 0 & 0{,}25 & 0 & 0 & 0 \\ 0{,}25 & 0{,}25 & 0{,}25 & 0{,}25 & 0{,}25 & 0{,}50 & 0{,}25 & 0{,}25 & 0{,}25 & 0{,}33 \end{bmatrix},$$

the Algorithm 3 computes the equivalence classes of Table 4. When $\lambda = \{0{,}9, 0{,}91\}$ the method computes two equivalence classes, grouping the 1-12 and 16-20 paths in the equivalence class 1: the paths with more unsafe transfers at a higher rate; and groupint the 13-15 paths in the equivalence class 2: the paths with a smaller proportion of unsafe transfers. In accordance with these results, the paths of the equivalence class 2 are preferable to the paths of the equivalence class 1.

Cuadro 4. Equivalence Classes Obtained by Fuzzy Classification Algorithm

λ	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
0.90,0.91	1-12, 16–20	13-15	–	–	–	–	–
0.92,0.93, 0.94	1,3-5, 16,20	2,9-12, 18,19	6-8, 17, 13–15	–	–	–	–
0.95,0.96, 0.97,0.98,0.99	1,16	2,9	3-5, 20	6-8	10-12, 18,19	13-15	17

Doing a similar analysis as above, in Table 4, when $\lambda = \{0{,}92, 0{,}93, 0{,}94\}$, the Algorithm 3 computes three equivalence classes: the first class comprises the paths with medium security transfers at most; the second class comprises the paths with safe, fairly safe and unsafe transfers at the same rate; most of the third class paths have a balance of medium security, low security and unsafe transfers. View of these results, the paths of the second equivalence class are preferable.

Finally, when $\lambda = \{0{,}95, 0{,}96, 0{,}97, 0{,}98, 0{,}99\}$, there are seven equivalence classes and the most recommended are first the fifth class and second the third class, because in the fifth class paths most of their transfers have medium security, and the third class paths have medium security transfers in greater proportion than unsafe transfers. All other classes are less desirable because the transfers with low security and unsafe are significant.

7. Conclusions

One of the main characteristics of the cities in developing countries is insecurity in certain urban areas. This Web platform designed to improve the integrity of UPTS traveler.

Mexico City and developing countries with similar characteristics, using a network of public disorderly public transport, which requires the traveler a precise knowledge of the public transport routes, safe areas for transfers and potential career options to ensure their integrity during their trip.

The creation of a social network based on the concepts of Web 2.0 provide greater speed in the exchange of experiences among travelers, with these preferences stored to support future travelers.

The connection of this site via Web Services, provides mechanisms for the exchange of knowledge among end users (travelers) and the opportunity that other sites use this service, incorporating the platform into the cloud computing.

Tackling a complex problem with a fuzzy classification algorithm provides a new approach to research to automatically find the path between two points, considering fuzzy parameters and the integrity of travelers.

Despite being a cubic algorithm, the pre-processing routes, helps to reduce the dimension of the problem. The use of robust computational architecture and parallelization of these, improves performance.

Acknowledgments. Thanks to CONACyT, LAFMIA and the Instituto Tecnológico de Puebla for their support, without which this research would have been possible.

Referencias

1. Ross, J.: Fuzzy Logic with Engineering Applications. John Wiley & Sons (2004)
2. Yen J.:Finding the K shortest loopless paths in a network. Management Science (1977)
3. Sun, R., Wang, Y.; Atrial Arrhythmias Detection Based on Neural Network Combining Fuzzy Classifiers. Advances in Neural Networks 4492, 284–292 (2007)
4. Keles,A., Hasiloglu, A.S., Keles, A., Aksoy, Y.: Neuro-fuzzy classification of prostate cancer using NEFCLASS-J. Computers in Biology and Medicine, vol. 37, issue 11, 1617–1628 (2007)

5. Shackelford, A.K., Davis, C.H.: A hierarchical fuzzy classification approach for high-resolution multispectral data over urban areas. IEEE Transactions on Geoscience and Remote Sensing, vol. 41, issue 9, 1920–1932 (2003)
6. Özyer, T., Alhajj, R., Barker, K.: Intrusion detection by integrating boosting genetic fuzzy classifier and data mining criteria for rule pre-screening. Journal of Network and Computer Applications 30(1), 99–113 (2007)
7. Sánchez, L., Couso, I., Corrales, J.A.: Combining GP operators with SA search to evolve fuzzy rule based classifiers. Information Sciences 136, 175–191 (2001)
8. Evsukoff, A.G., Costa, M.C.A, Ebecken, N.F.F.: High Performance Computing for Computational Science–VECPAR 2004. LNCS 3402, 63–96 (2004)
9. Ozturk, A., Arslan, A., Hardalac, F.: Comparison of neuro-fuzzy systems for classification of transcranial Doppler signals with their chaotic invariant measures. Expert Systems with Applications 32(2), 1044–1055 (2008)
10. API Google MAPS.: Google Maps API Family., available at `http://code.google.com/apis/maps/index.html`
11. Vanhove, S., Fack, V.: Fast Generation of Many Shortest Path Alternatives, Belgium