A Methodology for the Automatic Regulation of Intersections in Real Time using Soft-computing Techniques

Eusebio Angulo, Francisco P. Romero, Ricardo García,
Jesús Serrano-Guerrero, and José A. Olivas

Escuela Superior de Informática,
Universidad de Castilla - La Mancha,
Paseo de la Universidad, 4, 13071, Ciudad Real, España
{Eusebio.Angulo@alu.uclm.es}{FranciscoP.Romero,Ricardo.Garcia,
joseangel.olivas,Jesus.Serrano}@uclm.es

Abstract. This work presents an application of diverse soft-computing techniques to the resolution of semaphoric regulation problems. First, clustering techniques are used to discover the prototypes which characterize the mobility patterns at an intersection. A prediction model is then constructed on the basis of the prototypes found. Fuzzy logic techniques are used to formally represent the prototypes in this prediction model and these prototypes are parametrically defined through frameworks. The use of these techniques supposes a substancial contribution to the significance of the prediction model, making it robust in the face of anomalous mobility patterns, and efficient from the point of view of real-time computation.

Key words: Regulating traffic lights, soft-computing, clustering, estimation models

1 Introduction

The semaphoric regulation problem seeks to optimize i) the cycle lengths of a set of traffic-lights, ii) the percentage of time devoted to each of the phases in a cycle and iii) the transitions between consecutive sets of lights. This problem has been tackled in two temporal planning contexts. In the medium term, the stationary situation of the traffic is considered, and the objective is to obtain the semaphoric regulation of a set of intersections within the network. This problem has been formulated through a mathematical program with equilibrium constraints (MPEC). The results of these models are semaphoric regulations with fixed times for the cycles. The short term methods, which consider the dynamic aspect of the problem, have been fundamentally tackled through the application of optimization techniques to simulation models [1].

Various works using soft-computing techniques exist, and such works have fundamentally used Genetic Algorithms [2], whose objective has been the optimization of semaphoric transitions [3]. Numerous fuzzy logic approximations

have also been carried out, particularly in the field of the fuzzy control of traffic-lights [4],[5], [1], [6] and [7]. Many of these developments have been carried out in an off-line context. The appearance of new traffic control technologies permits the real-time availability of precise data with regard to traffic conditions and makes the development of on-line methodologies possible.

Besides, Sanchez [8] presents architectures for traffic light optimization based on Genetic Algorithms with greater stability. It is designed and tested an evolutive architecture which optimizes the traffic light cycles in a flexible and adaptive way. These tests were of medium size and took place in a zone of Santa Cruz de Tenerife (Spain), thus improving the results of fixed cycle traffic lights.

In spite of these approximations, problems still remain which must be solved. One of these problems is that of tackling non-stationary mobility patterns, which is to say, the changing demands at various times of the day. This paper tackles this problem by proposing a methodology for the adaptive control of semaphoric intersections by using on-line traffic light counts.

The methodology here proposed is based on the extraction of mobility patterns on the basis of prototypes through the use of diverse soft-computing techniques which are implemented as an approach of the classic process of Knowledge Discovery on Databases (KDD) [9]. The use of diverse techniques, such as fuzzy logic and clustering, are incorporated in to this model and these techniques allow us to obtain more comprehensible and useful results for the prediction process.

The remainder of the work is organized as follows: Section 2 describes the different tasks that have been carried out to design the mobility patterns-based model. Section 3 explains the necessary stages to apply the above-designed model at a real intersection. To assess the methodology here proposed, an experiment has been developed in section 4. Finally, some conclusions and future works are pointed out.

2 Methodology

The objective of the methodology here proposed is that of adaptively regulating an intersection, as is shown in Figure 1. The intersection has sensors which measure all four lanes and permit the existence of entrance and exit traffic linkcounts in both directions at each of the time intervals considered. Moreover, that intersection has a semaphoric regulation. To build the model with which to determine adaptive regulation, it is first necessary to extract the intersection´s mobility patterns. These patterns will be extracted from the vehicle flow observations obtained from the sensors.

The following stages are carried out to build the model:

1. Observations of the entrance/exit flows by use of the sensors.
2. Estimation model for *traffic dynamic O-D matrix*: This model permits the estimation of turns at the intersection. The O-D matrix is defined as being the matrix which contains, in the *i-j row*, the flow (number of people per time unit) which is incorporated into the intersection of lane i, and which

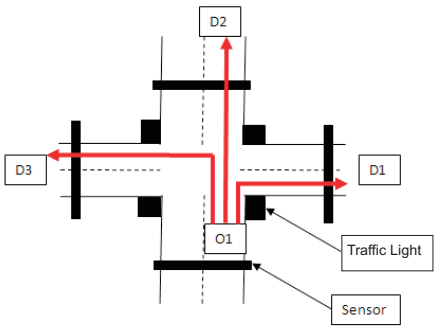

Fig. 1. Four lane intersection with no U-turns

leaves that lane via lane j. It is assumed that U-turn movements are not allowed, i.e., the main diagonal entries (i-i) of the O-D matrix are all zero.

3. Extraction of mobility Patterns: This stage models the mobility patterns using the O-D matrix and represents them by means of fuzzy deformable prototypes.
4. Traffic light regulation model: An expert can model the optimum behaviour of the semaphores following the above-mentioned prototypes.

In a concrete moment of the day, it can be seen different O-D matrices. The differences among themselves are random, so it can be considered that all matrices represent the same behaviour. So, such concept is here known as *mobility pattern* during a concrete time period and the exact representation of this pattern is each above-mentioned matrix.

2.1 Flow Observations

The entrance/exit sensors situated in each lane calculate the number of vehicles that are been driven in the instant t. We thus obtain the number of vehicles which pass each sensor, although their destinations are unknown owing to the turns that they may make.

Let an intersection be a tuple composed by m entrances and n exits and considering the time divided into N intervals ($t = 1, \ldots, N$), the inputs of the model would be:

- Entrance flows: $q_i(t)(i = 1, ..., m)$; $q(t) = [q_1(t), ..., q_m(t)]^T$;
- Exit flows: $y_j(t)(j = 1, ..., n)$; $y(t) = [y_1(t), ..., y_n(t)]^T$;

These data are the linkcounts in the intersection during the time period t. We thus obtain the number of vehicles which pass each sensor, but their destinations (turns) are unknown.

2.2 Estimation Model: Dynamic O-D Matrix

The estimation model shown in this sub-section is used to obtain the O-D matrix with complete predictions (including turns). This estimation can be carried out instantaneously by using the sensors information.

The model´s variables are the following:

- I_j : A set of values in which entrance i allows user to take exit j. I_j $(j = 1, \ldots, n)$.
- The probability that a vehicle enters via i and takes the exit j. $b_{ij}(i = 1, \ldots, m; j = 1, \ldots, n)$.
- The probability vector from each entrance i to the exit j. $b_j = [b_{ij}] \forall i \epsilon I_j$ y $Q_j = [q_i] \forall i \epsilon I_j$; $b = [b_1^T, ..., b_n^T]^T = [b^{(i)}]$

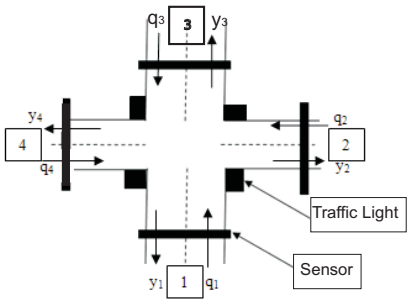

Fig. 2. q_i entrances and y_i exits in a four lane intersection with prohibited U-turns

For an intersection such as that shown in Figure 2, in which $n = m = 4$, the variables are as follows:
$b_1 = [b_{21}, b_{31}, b_{41}]^T$, $b_2 = [b_{12}, b_{32}, b_{42}]^T$, $b_3 = [b_{13}, b_{23}, b_{43}]^T$, $b_4 = [b_{14}, b_{24}, b_{34}]^T$
$Q_1 = [q_2, q_3, q_4]^T$, $Q_2 = [q_1, q_3, q_4]^T$, $Q_3 = [q_1, q_2, q_4]^T$, $Q_4 = [q_1, q_2, q_3]^T$

$$b = [b_{21}, b_{31}, b_{41}, b_{12}, b_{32}, b_{42}, b_{13}, b_{23}, b_{43}, b_{14}, b_{24}, b_{34}]^T = [b^{(1)}, ..., b^{(12)}]^T$$

Where b should fulfil:

$$b \geq 0, \sum_{j=1}^{n} b_{ij} = 1, b_{ii} = 0 \tag{1}$$

Therefore, the sum of probabilities from each entrance i to an exit j must be 1 and each probability must be greater than 0. For the observed linkcounts $y(t)$ and $q(t)$ in each time interval t, the estimation problem of b is resolved with:

$$J_i(b) = \sum_{s=1}^{t} \sum_{j=1}^{n} \{y_j(s) - Q_j(s) b_j\}^2, t = 1, ..., N \tag{2}$$

Where J_i is the set of values in which exit j permits users to enter entrance i, being $(i = 1, \ldots, m; j = 1, \ldots, n)$.

The estimation model creates an O-D matrix taking into account the turns. These vectors are the input of the phase called mobility patterns extraction.

2.3 Mobility patterns extraction

A clustering process is carried out to find relationships among the O-D matrices and after this process, the mobility patterns are detected. The goal of the clustering process is to reduce the amount of data by categorizing or grouping similar data items together. Firstly the process must be build a similarity matrix based on the matrices returned by the estimation model, i.e., the inputs of the prototypes extraction process. The euclidean distance is the measure chosen to calculate the similarity among vectors.

Once the similarity matrix has been created, the two stages of the clustering process are carried out. Firstly the goal is finding groups of similar flows data detected in successive instants. This goal is reached following a graph-based clustering method [10]. In the second stage, to detect other similar groups that exist in non-successive instants, is carried out a hierarchical clustering algorithm based on fuzzy graph connectedness [11]. The nodes of the graph are the clusters of the first stage.

Every cluster represents a mobility pattern found at the intersection. Every pattern is described by a fuzzy deformable prototype that finally will be represented by a fuzzy numbers set. The fuzzy numbers set is modeled by a normalization and aggregation process using the O-D matrices of each cluster. This process permits to calculate the center and the length of the base of the fuzzy triangular numbers, the unique necessary data to represent each fuzzy number. In figure 3 are shown five prototypes that are the output of the clustering stage.

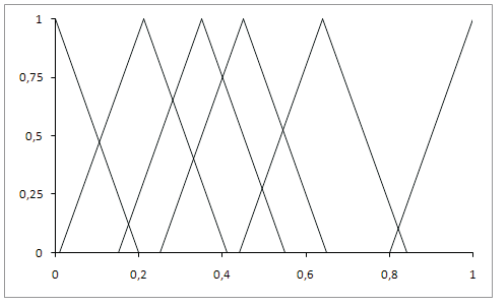

Fig. 3. Formal representation of the prototypes.

Using a fuzzy numbers-based representation, it is easy to calculate the membership degree (in range [0-1]) between real situations and the prototypes detected.

2.4 Semaphoric Regulation

Once the mobility patterns have been detected and defined by means of fuzzy prototypes, the behaviour of each semaphore is analyzed by an expert depending on each pattern. If there are N prototypes then there are N optimum system responses and the value set of each response will be represented by a frame.

Table 1. Parametric Description of the prototypes

Prot	Congestion Level	Demand Direction
P1	Congested (High)	go
P2	Semicongested (Medium)	go
P3	Without congestion (Very Low)	go
P4	Semicongested (Medium)	return
P5	Without congestion (Very Low)	return

Fuzzy deformable prototypes and the parametric definition of the semaphoric regulation permits to design a flexible solution for the problem of traffic tie-ups. This idea could be especially important in critical moments such as great sport or cultural events, where the traffic can be a serious problem.

3 Model performance

Once the model has been calculated, it can be applied to the daily management of the intersection. The regulation system´s entrance data will be the real-time flow observations, and the exits will express the type of regulation that must take place at each moment.

3.1 Real-time flow observations

At an intersection, sensors located in every entrance/exit of the lanes catch information about the number of vehicles driving in every moment. These data feed the system to discover the optimum semaphoric regulation parameters.

3.2 Estimation model

The estimation model permits us to obtain the complete O-D matrix (including turns) from the linkcount estimations. This is obtained in exactly the same manner as in the model construction phase (off-line). The elements and calculations specified in sub-section 2.2 will thus also be applicable in this step.

3.3 Inference in prototypes

The mobility pattern is calculated using the values of the O-D matrix by means of an inference process based on the fuzzy deformable prototypes of the model. The algorithm is:

1. Normalization of the values of the entrance O-D vector.
2. Aggregate the normalized values (X value).
3. Calculate the membership degrees of each prototype represented by fuzzy numbers. To assess a concrete situation (Figura 4) is necessary relevant information. This relevant information is achieved by calculating an affinity degree with the prototypes(μ_i).

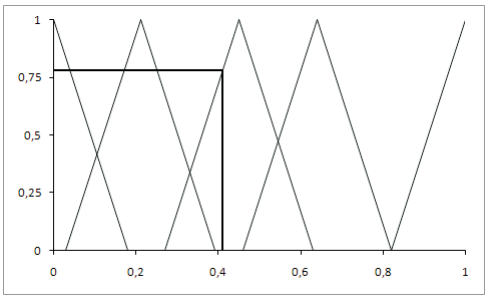

Fig. 4. Similitude of a vector to a prototype ($X = 0.41$, $\mu_3 = 0,78$)

Once the membership degrees between the real environment and the prototypes of the model have been calculated, the definition of the prototype, that represents the optimum behaviour of the semaphores, must be returned.

3.4 Implementation of the optimum control

The most similar prototype among the above-calculated prototypes will be chosen as the most suitable one to simulate the semaphoric regulation in an exact moment.

The system´s exit thus contains all the parameters that define the traffic lights, behaviour whilst the detected mobility pattern remains. These values will be transmitted to the electronic component in charge of transmitting orders to each of the traffic lights in the intersection.

So, the output of the system is composed by all the parameters that are necessary to describe the behaviour of the semaphores while the mobility pattern is happening. The values of these parameters will be transmitted to the control process unit to manage the semaphores that are at the intersection.

4 Computational experience

The data used in these numerical tests has been generated by simulation. The traffic density at each time interval is the same as that used in the demand which supports the urban railway network in Madrid. The graph in Figure 5 shows this hourly demand distribution.

Let q_i be the entrance traffic density in the approach i and e let y_i be the exiting traffic density in a determined time period. The estimation of the entrance flows q_i to the intersection is carried out by using the following expression:

$$q_i = D * p(t) * u(1 - \varepsilon, 1 + \varepsilon) \tag{3}$$

D: is the total entrance demand to the intersection, namely, the total number of vehicles passing through the intersection throughout the day and we consider 10000.

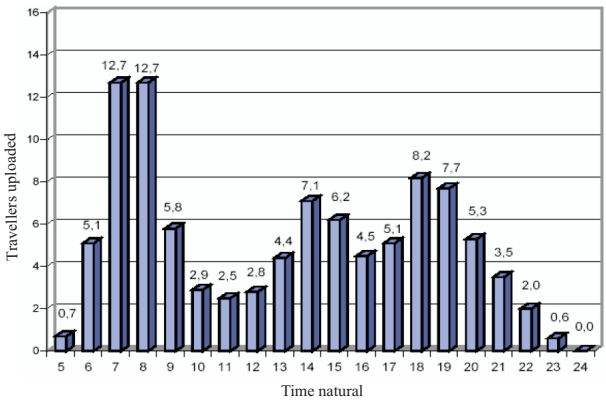

Fig. 5. Hourly demand distribution of the urban railway network, Madrid.

$p(t)$: is the proportion of turns dependant on the time instant. This parameter allows us to take into account the direction of the traffic flow in each instant.
$u(.)$: is the uniform random variable.
ε: takes the value of 0.15.

The estimation of the exits y_i at the intersection is carried out by using the expression:

$$y_j = \sum_{j \neq i} (P_{ij}(t) * q_i) \tag{4}$$

$P_{ij}(t)$ is calculated by using the expresion:

$$P_{ij}(t) = \left(\frac{t-5}{24-5}\right) * P_1 + \left(1 - \frac{t-5}{24-5}\right) * P_2 \tag{5}$$

where P_1 and P_2 are:

$$P_1 = \begin{pmatrix} 0 & 0.2 & 0.6 & 0.2 \\ 0.1 & 0 & 0.5 & 0.4 \\ 0.2 & 0.4 & 0 & 0.4 \\ 0.25 & 0.25 & 0.5 & 0 \end{pmatrix} \quad P_2 = \begin{pmatrix} 0 & 0.4 & 0.2 & 0.4 \\ 0.5 & 0 & 0.25 & 0.25 \\ 0.6 & 0.2 & 0 & 0.2 \\ 0.5 & 0.4 & 0.1 & 0 \end{pmatrix}$$

Once the entrance and exit estimations for each lane have taken place for all the 5 minute time intervals between 05:00 and 24:00, the predicted origin-destination matrix is estimated by using the resolution of the proposed optimization model and by using GAMS software.

The estimation model allows us to obtain the complete O-D matrix (including turns) from the linkcount estimations and is calculated by using the elements and calculations specified in sub-section 3.2. Figure 6 shows the results of the linkcount estimation model, as opposed to those of the prediction model shown in sub-section 3.2, for the entrance turn in 1 and the exit in 2.

Figure 6 shows the results obtained. Note the high adjustment quality. This algorithm offers results which allow us to group the elements into 6 different mobility patterns. Figure 7 shows the different assignation of each element to the different groups obtained.

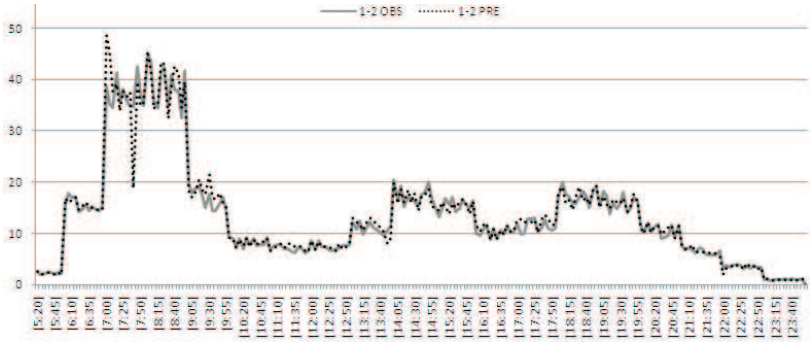

Fig. 6. Matrix $(1, 2)$ observed compared to the predicted matrix

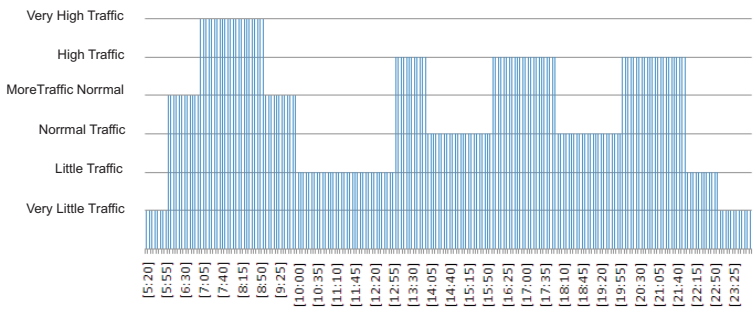

Fig. 7. Grouping distribution

5 Conclusions and future work

A new methodology has been presented to define and extract mobility patterns by means of optimization and fuzzy logic techniques. These techniques have been used to discover knowledge useful to design a formal, meaningful and useful model.

The methodology presents an automatic and adaptive control for intersections, achieving controlled outputs of the system and avoiding wrong responses. Besides, the requirements to develop a system based on these ideas are very simple due to the fact that the implementation of the system is really easy, the technology requirements are not expensive and the performance is very efficient.

To validate the proposed methodology, an experiment has been carried out simulating the behaviour of vehicles following known distributions. The performance of the experiment has been satisfactory.

In future works, the main goals are 1) testing a system developed following this methodology using real data and 2) refining the parameters used to described the mobility patterns.

Acknowledgments This research has been partially supported by PAC06-0059 SCAIWEB project and PCC08-0081-4388-2, JCCM, Spain, TIN2007-67494 F-

META project, MEC-FEDER, Spain, and FIT-340001-2007-4 BUDI project, MICYT, Spain.

References

1. Wiering M., Vreeken J., Van Veenen J., Koopman A.: Simulation and optimization of traffic in a city, In: IEEE Intelligent Vehicles Symposium, Proceedings, 453–458 (2004)
2. Rouphail N., Park B., Sacks J.: Direct signal timing optimization: Strategy development and results. In: XIth Pan American Conference on Traffic and Transportation Engineering (2000)
3. Lim G.Y., Kang J.J., Hong Y.S.: The optimization of traffic signal light using artificial intelligence, In: Proc. 10th IEEE Int. Conf. Fuzzy Syst., Dec. 2-5, 2001, vol. 3, pp. 1279–1282 (2001)
4. Lei C., Guojiang S., Wei Y.: The traffic flow model for single intersection and its traffic light intelligent control strategy. In: Proceedings of the World Congress on Intelligent Control and Automation (WCICA) 2, art. No. 1713650, 8558–8562 (2006)
5. Van Leeuwaarden J.S.H.: Delay analysis for the fixed-cycle traffic-light queue, Transportation Science 40(2), 189–199 (2006)
6. Lim G.Y., Kang J.J., Hong Y.S.: The optimization of traffic signal light using artificial intelligence, In: IEEE International Conference on Fuzzy Systems 3, 1279–1282 (2002)
7. Hoyer R., Jumar U.: Fuzzy control of traffic lights, In: IEEE International Conference on Fuzzy Systems 3, 1526–1531 (1994)
8. Sánchez J., Galán M., Rubio E.: Applying a Traffic Lights Evolutionary Optimization Technique to a Real Case: "Las Ramblas" Area in Santa Cruz de Tenerife, In: IEEE Transactions on evolutionary computation, pag. 25-40 VOL. 12, NO. 1 (2008)
9. Fayyad U., Piatetsky-Shapiro G., Smyth P.: The KDD Process for Extracting Useful Knowledge from Volumes of Data, In: Communications of the ACM, 39(11), pp. 27–34 (1996)
10. Kawaji H., Yamaguchi Y., Matsuda H., Hashimoto A.: A Graph-Based Clustering Method for a Large Set of Sequences Using a Graph Partitioning Algorithm. Genome Informatics 12: 93–102 (2001)
11. Dong Y., Zhuang Y., Chen K., Taib X.: A hierarchical clustering algorithm based on fuzzy graph connectedness, Fuzzy Sets and Systems, vol. 157, pp. 1760–1774 (2006)