USE OF GENETIC ALGORITHM FOR FUZZY SIGNAL CONTROLLER DESIGN – AN EXPLORATORY STUDY

Felipe Labanca – University of Brasilia
Paulo Sérgio França de Sousa Jr.- University of Brasília
Marcelo Vale Asari – University of Brasília
Maria Alice Prudêncio Jacques – University of Brasília

ABSTRACT

The fuzzy signal controller considered in this study is activated by traffic and its internal programming is based upon fuzzy logic by making use of the fuzzy extension principle. Each controller element can be implemented in different ways. The elaboration of the controller project is therefore hampered by the sizeable number of options available in defining its components. This study proposed the use of Genetic Algorithms (GAs) so as to conduct a search for an option that can be better adjusted to the prevailing control situation. Tests with the GA proposed involved different traffic volume levels to be controlled and proved the instrument's effectiveness.

Keywords: fuzzy signal controller, genetic algorithm, signalized intersections

1 - INTRODUCTION

Traffic controllers acting in response to traffic prevailing characteristics have been adopted at many road intersections all around the world as a surrogate for the conventional fixed time controller type. The major advantages of the so-called actuated signal controllers is their capability to respond to traffic flow variations along the control time, especially when they are hard to be predicted for the purpose of fixed time plan generation. However, to do so, its programming is not a trivial task, asking in many cases for mathematical modelling of the complex interactions among traffic and controller characteristics (Andrade and Jacques, 2006). Previous works have proven that fuzzy logic is an effective alternative to simplify the representation of these interactions for actuated signal controller applications. Starting with the work of Pappis and Mamdani (1977), several research studies have demonstrated that traffic signal controllers operating under the fuzzy extension principle provide significant advantages for controlled traffic as compared to the fixed time and other actuated controllers' type. However, the definition and implementation of the major components of these fuzzy signal controllers is a critical aspect.

Fuzzy signal controller design is hampered by the sizeable number of options available in defining its components. Studies published on this issue indicate that the controller responses and, consequently, the traffic controlled performance are strongly influenced by distinct combination possibilities for these elements' implementation (Jacques et al., 2002b and 2002c; Vaz, Oliveira and Jacques, 2005; Jacques et al., 2005). Thus, a tool providing an automated manner for identifying a combination better adjusted to controlled traffic needs will fairly assist the designer's task, hence contributing to the quality of his/her job.

This study aims to investigate the use of Genetic Algorithms (GAs) so as to conduct a search for a controller that can be better adjusted to the traffic control situation, given that the GAs are effective search instruments in very large spaces. For this exploratory study, the decision was to first of all limit controller parameters to be defined by the GA thus making it possible to assess GA performance parallel to the complete study of all possible combinations. The GA search process is carried out based on traffic performance measures provided by traffic micro simulation. The study also aims at the GA developed to allow the evaluation of the responses of the selected controller to the prevailing traffic, represented through a control surface. This surface should be smooth and coherent so that it can be correctly interpreted by specialists and easily assimilated by drivers. Therefore, throughout the analysis of the shape of the control surface, it will be possible to identify if some procedure must be added to the original GA in order to discard solutions not feasible according to the control actions' perspective.

This paper is organized into 7 (seven) sections as follows. In the first section, the motivation for this study's development and its major objectives are clearly stated, as well as the paper's structure. Section 2 presents the major characteristics of fuzzy signal controllers based on fuzzy extension principle by focusing on the controller studied. Subsequently, Section 3 shows the basics of genetic algorithms and their application as heuristic search methods. In Section 4, the traffic simulation program used in this study is presented. The development of the genetic algorithm proposed is showed in Section 5, while its application's results and analyses are included in Section 6. The last section deals with the major conclusion of this study and recommendations for future studies.

2 – FUZZY SIGNAL CONTROLLERS

According to the seminal work of Pappis and Mamdani (1977), fuzzy signal controllers use fuzzy logic principles for defining green extension times to current green at signalized intersections. The basic components of this signal controller type are: fuzzification interface, processing module e defuzzification interface. Although different applications of fuzzy logic to traffic signal control are found in the literature (see Jacques et al., 2002a), this work refers to only those controllers based on the extension principle. The following subsections describe the major characteristics of the controller used in the present work.

2.1 Fuzzification interface and linguistic variables

At the fuzzification interface occurs the so-called fuzzification process. According to this process, the numerical input traffic data collected by traffic detectors are transformed into linguistic values, based on the membership functions of the fuzzy sets related to each controller's linguistic input variables. The fuzziyfication process also considers the input variables' partition of the universe of discourse. The input variables of the controller studied, which are related to the state of the traffic controlled, are: arrival (*chegada*) and queue (*fila*).

Arrival is related to the number of vehicles at the intersection approach receiving green indication, and its fuzzy sets are: zero, few, medium, long and any. The variable queue refers to the vehicles at the approach receiving red indication, and considers the following fuzzy sets: small, medium, long and any.

The linguist control variable (output variable) is called extension (*extensão*). It means the length of the green extension to be provided by the signal controller to the current green according to the prevailing state of the traffic controlled. The fuzzy sets associated to this variable are: zero, short, medium and long.

All fuzzy sets related to the input and output variables have their corresponding partition in the universe of discourse and type of membership functions defined by the controller designer. Previous studies have indicated that the controller response and, consequently, the traffic performance are significantly affected by these definitions (Jacques et al., 2005; Vaz et al., 2005).

2.2 Processing module

The processing module comprises the knowledge base, represented by the rules defined by the traffic expert to relate the input variables with the output variable, and the decision-making logic. The fuzzy signal controller was developed in the context of fuzzy inference and according to generalised modus ponens (GMP) implication inference rules. A control action is determined by the inputs observed and the knowledge base and the consequent of one rule does not serve as an antecedent to another.

The relation among the state variables (rule antecedent) and the control action (rule consequent) is defined by the general form "*If (rule antecedent) then (rule consequent)"*. The response of the fuzzy controller is produced by the aggregation of all defined rules. This is done with the sentence connective "also", which can be implemented by different operators. A weight might be given to each rule. The greater the weight is the more relevant the rule is at the aggregation procedure.

In addition, if two or more conditions are considered based upon the rule's antecedent and/or consequent, connectives "*and"* or "*or*" must be used. These connectives, as well as the implication function, may be implemented by different operators. Jacques et al. (2002b) showed that different operators result into different controllers, thus producing distinct impacts on traffic performance.

The Fuzzy Logic Toolbox of MATLAB (FLT-M), used as reference for the present study, allows for the graphic representation of the knowledge base (see Figure 1). It provides a relatively fair interpretation and analysis of the controller response, expressed as a numerical value to extension (*extensão*) variable, for the values observed for the variables queue (*fila*) and arrival (*chegada*). Each horizontal line represents a rule created by the controller designer. It is formed by three boxes, one for each of the linguistic values. In the example shown in Figure 1, for 13 vehicles in the queue and 17 vehicles arriving during the current green the controller will produce 12 seconds of green extension. This latter value is the result produced by the defuzzification process.

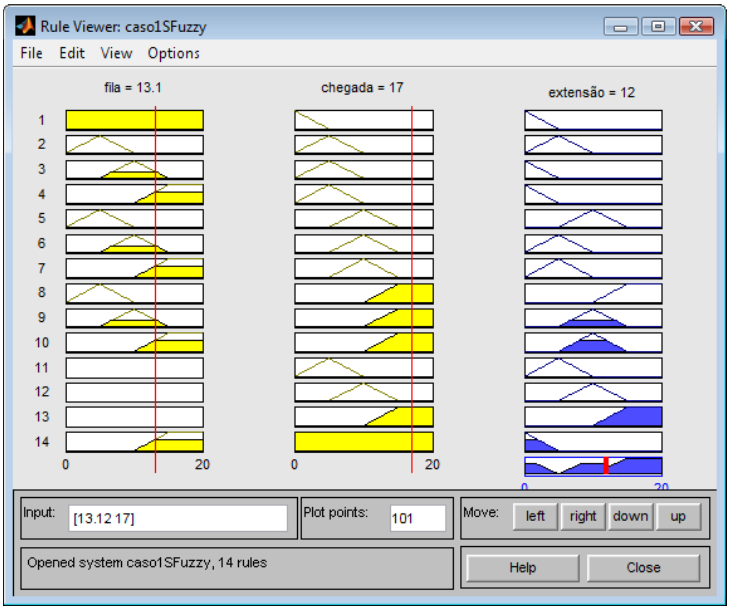


Figure 1: Rule Viewer at Fuzzy Logic Toolbox of MATLAB

2.3 Defuzzification interface

Defuzzification is the process of transforming the linguistic value given by the processing module into a numerical one. This process varies according to the type of fuzzy controller considered: Mamdani and Sugeno.

The knowledge base of Mamadani controller type encompasses rules with both the antecedent and the consequent being fuzzy sets (Chambers, 2001). The GAgregation of the linguistic result of each rule returns a unique linguistic value, named the fuzzy set of response. One example of this fuzzy set can be seen at the right bottom box of Figure 1.

The principal defuzzification methods available at FLT-M for defining the numerical value from the fuzzy set of response, applied to Mamdani controller type, are: (i) the largest of maximum method (LOM); (ii) the bisector method; (iii) the mean of maximum method (MOM); (iv) the centre of gravity method (COG); (v) the smallest of maximum method (SOM). Study related to the impact of defuzzification method on controller response shows that it is significant, directly affecting traffic controlled performance (Jacques et al., 2002c).

The Sugeno fuzzy controller considers the same fuzzification process for the input data and applies the rules' operators as is done in the Mamdani type. The major difference between these two types of controllers is that in the Sugeno controller, the membership functions related to the output linguistic variable are linear functions or constant values (Chambers, 2001). In this case, the defuzzification methods applied to the Mamdani controller are no longer valid. The defuzzification in the Sugeno controller consists in the weighted mean of the numerical values given by each rule output, where the weights are the corresponding membership values.

2.4 Global representation of the results provided by the signal fuzzy controller

As previously described, the fuzzy signal controller considered in this work produces a numerical value for each pair of values of the input variables. Therefore, it is possible to define the extension values for all possible values the input variables can assume. These overall results can be shown in a matrix form as per Figure 2, where the extension values are expressed in tenths of second. As the universe of discourse of the input variables varies from 0(zero) to 20, the matrix order is 21 per 21. The graphic representation of the matrix values in a surface form is available at the FLT-M, as shown in the example in Figure 3.

		CHEGADA																				
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
FILA	**0**	16	27	35	41	46	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	1	16	28	37	42	47	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	2	16	30	38	43	47	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	3	16	31	39	44	47	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	4	16	32	40	45	48	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	5	16	33	40	45	48	50	60	70	80	90	100	117	131	143	153	162	162	162	162	162	162
	6	16	31	39	43	46	48	58	67	77	86	95	112	126	138	148	157	157	157	157	157	157
	7	16	30	37	41	44	46	56	65	73	82	90	106	121	133	143	153	153	153	153	153	153
	8	16	29	35	39	42	44	53	62	70	78	85	101	115	127	138	148	148	148	148	148	148
	9	16	27	33	37	39	41	50	59	66	73	80	96	109	122	133	143	143	143	143	143	143
	10	16	25	31	34	37	38	47	55	63	69	75	90	103	116	127	137	137	137	137	137	137
	11	16	24	29	33	35	37	46	53	60	66	72	86	99	111	122	132	132	132	132	132	132
	12	16	23	28	31	34	36	44	51	58	64	69	83	96	107	118	128	128	128	128	128	128
	13	16	23	27	30	33	35	42	49	56	62	67	80	93	104	114	124	124	124	124	124	124
	14	16	22	26	29	32	34	41	48	54	60	65	78	90	100	111	120	120	120	120	120	120
	15	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116
	16	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116
	17	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116
	18	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116
	19	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116
	20	16	21	25	28	31	33	40	46	52	58	63	75	87	97	107	116	116	116	116	116	116

Figure 2: Matrix of the extension values (in tenths of seconds)

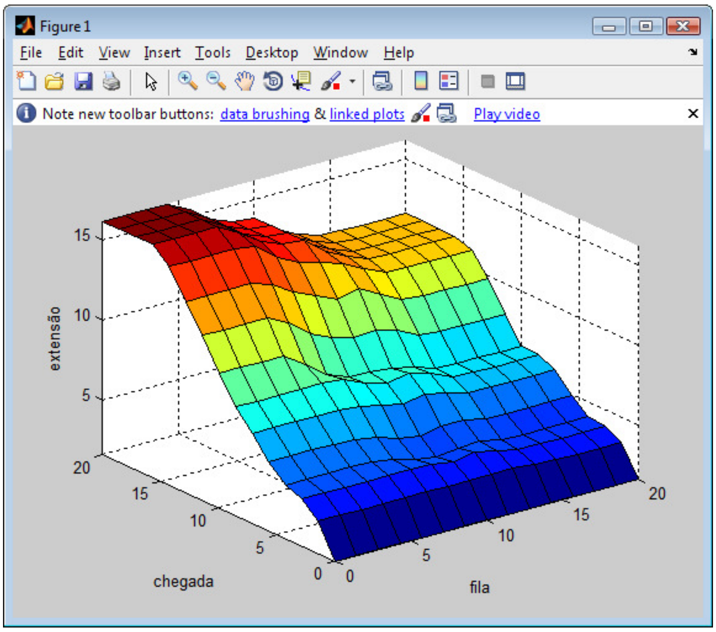


Figure 3: Graphic representation of the extension values

3 – GENETIC ALGORITHMS

Varying the parameters of any quantitative model causes direct impact on its results. Therefore, the determination of these parameters must be done accordingly to better represent the situation to be modelled or to optimize the model results. As the model complexity increases, computational tools can be used to improve the model parameters determination process. Automatic learning, artificial neural networks and genetic algorithms are just examples of tools that can be used in the search for more efficient solutions.

Genetic algorithms have shown themselves to be quite effective for solving problems in which the number of parameters to be defined makes the optimization process complex, with a large number of variables to be taken into account. It has a simple logic, which allows for its programming in many different computational languages.

3.1 Basic concepts

Genetic algorithm (GA) is a computational technique for search and optimization based on the theory of evolution and genetic science. It produces in a iterative way, sets of solutions for a given problem that tends to converge towards an optimal solution (Miranda, 2000). The principal natural evolution mechanisms considered for the GA's development are: natural selection and reproduction (Soares, 1997).

The principal concepts from genetic science applied to GA's development, according to Miranda (2000), are:

(i) member or individual: set of parameters that represent a possible solution to the system;

(ii) population: set of members;

(iii) generation: complete iteration of the algorithm, which generates a new population;

(iv) gene: representation of each parameter of a given population member;

(v) chromosome: sequence of genes that represents a possible solution to the problem;

(vi) allele: possible parameter values for a given gene

(vii) *locus*: gene position at the chromosome.

In this present study, a member is represented by a unique chromosome. Because of this, the definitions of member and chromosome given above are the same. However, it is possible to use more than one chromosome per individual as long as this situation betters the organization of the information at more complex situations.

3.2 Algorithm

The development of a genetic algorithm (GA) starts with the definition of the parameters to be adjusted during the searching process. Following, the general structure (chromosomes) of the population members must be defined. That is, the sequence of the parameters (genes) and the length of each one (number of locus for each gene) must be specified.

After the basic structure of each member is defined, a number of individuals must be randomly generated to constitute the initial population of the GA. As the initial members correspond to initial solutions to the problem, this initial population must be comprehensive. The definition of the number of individuals to form the initial population must consider that if this number is very short, the variety of initial solutions is small; large initial populations, on one side, will make the algorithm running process excessively slow.

The next step consists in the evaluation of each population's member to verify how much it has become adapted to the system. To do so, a measure of performance is associated with the member, indicating the result of its evaluation. The relationship between the individual and its performance is called fitness function. The member for whom the calculated performance is suitable for the problem, is considered a desired solution. If no desired solution is found at one given population, the GA performs a selection of pairs of members according to the performance.

Each selected pair of members will face a reproduction process. This reproduction happens according to two operations: crossover and mutation. The crossover is responsible for the interchange of genetic information between two members (called parents), generating two new members (descendents) with characteristics from both parents. The mutation is used to test characteristics that may not be included in the current population. After all the descendents have been defined, a new generation is formed. The new population will then return to the beginning of the process. It will be interactively repeated until the generated population satisfies the problem's conditions, or until a pre-defined number of interactions has been performed (Miranda, 2000).

The accepted level of fitness is defined based on the objectives of the problem to be solved by the GA. In the present work, the fitness will be represented by the average delay per vehicle, measured for all vehicles controlled by the fuzzy signal controller. This fitness measure will be calculated with the aid of a microscopic traffic simulation program, UnB-Sitracs, which is introduced in the next section.

4 – TRAFFIC SIMULATOR: UNB-SITRACS

Computational programs are able to predict the traffic operations at road segments and intersections by taking into account algorithms that consider models of vehicles' behaviours (car-following, lane-changing, etc.). These programs allow among other capabilities, for testing the impact of signal controllers on traffic performance at signalized intersections, before the device is installed at the conflict point. It is also possible to test the controller under different conditions, in order to evaluate its behaviour for new possible traffic and local physical conditions.

This work is developed with the aid of the simulation program named UnB-Sitracs (Asari et al., 2008). It is a free distributed program, available for downloading at the site <http://sourceforge.net/projects/unb-sitracs/>. At the same site, the corresponding user manual is also available. This manual presents in details, all the traffic models inserted in the program and the performance measures provided. The UnB-Sitracs simulates the traffic operations at isolated intersections and arterial routes, but only for intersections controlled by traffic lights. Three different signal controllers can be simulated: fixed-time, semi-actuated and fuzzy. The fuzzy signal controller represented at the UnB-Sitracs is the one that operates under the green extension principle, which is the case of this study.

Among the traffic performance measures delivered by UnB-Sitracs, the GA proposed will consider the average control delay per vehicle as the fitness measure for each population member. This measure is commonly used in Traffic Engineering for evaluating the traffic performance at signalized intersections. The Highway Capacity Manual (TRB, 2000), for instance, classifies the level of service for signalized intersections based on this measure.

The joint work with the simulator developers and the consequent possibility of having any necessary modifications to the program made, give that the necessary simulations were made directly on the MATLAB platform. The GA proposed implementation was made on the MATLAB language, which is compatible to the JAVA language used for the development of UnB-Sitracs.

5 – GENETIC ALGORITHM PROPOSED

The development of the genetic algorithm proposed started with the definitions of the components of the fuzzy signal controller that should be adjusted. Following, the logic structure of the algorithm was defined and, finally, it was implemented with the MATLAB software. This latter step will not be presented in this section; it might be found, along with the details of the previous steps, in Labanca (2009).

5.1 Components of the fuzzy signal controller to be adjusted

The number of components to be adjusted was limited to 5 (five), in order to allow for the intended evaluation of all the possible controllers to be generated by their distinct combinations. In light of this, just the following components will be analysed by the GA proposed: (i) Connective *"and"* – the GA will choose between the minimum (min) and product (prod) methods; (ii) Connective "*or"* – the GA will consider the maximum (max) and probabilistic "*or*" (probor) methods; (iii) Implication function – the minimum (min) and product (prod) methods will be considered; (iv) Connective "also" (aggregation) – the GA will choose among the maximum (max), sum (sum) and probabilistic "or" (probor) methods; (v) Defuzzification method – it will be considered the COG, bisector, MOM, SOM and LOM methods, referred to in section 2.3.

In this exploratory study it was chosen to work only with Mamdani controller type and, as the rules do not include the connective "*or*", the method for implementing this connective does not affect the GA search process. These constraints cause the search space to be constituted by 60 (sixty) possible solutions.

5.2 Logic structure of the GA proposed

The logic structure of the GA proposed is presented in the flowchart in Figure 4.

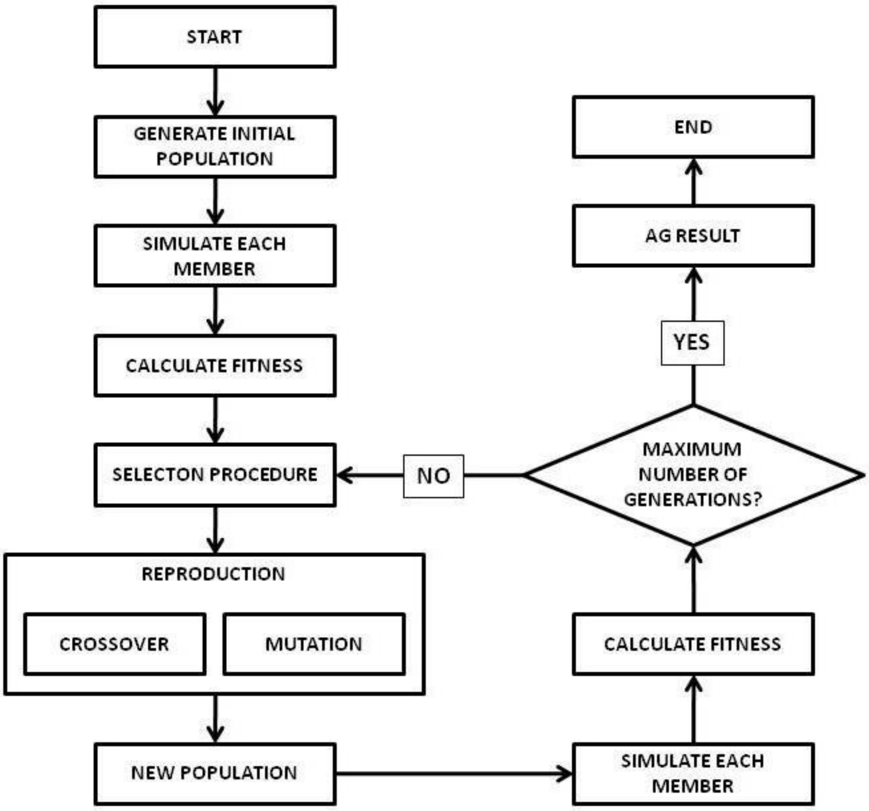


Figure 4: Logic structure of the GA proposed

This logic structure requires the development of the chromosome to be considered and its related fuzzy signal controller, as well as the traffic simulation method and the characteristics of reproduction methods to be adopted (crossover and mutation).

5.3 Components of the GA proposed

The following text presents the procedures adopted in the definition of the main components of the GA proposed.

5.3.1 Chromosome

The chromosome contains all the information related to a possible solution for the problem. In this case it is equal to a population member. This means that the chromosome will contain all the parameters of the fuzzy signal controller that the GA will adjust. Each of its genes represents one controller's component. The gene's number of locus must be able to cover all the options for the component implementation (alleles). In this work the chromosomes have a decimal structure; their genes have only one locus which can assume values from 0 (zero) to 9 (nine). The binary structure, usually presented in the literature, is not necessary here due to the small number of options available for each gene.

Each member (or individual) is formed by only one chromosome, coded by letter "*f*" followed by 06 (six) numbers, which represent the following characteristics of the controller:

(i) Controller type: 0 = Mamdani;

(ii) Connective “and” (conjunction method): 0 = min; 1 = prod;

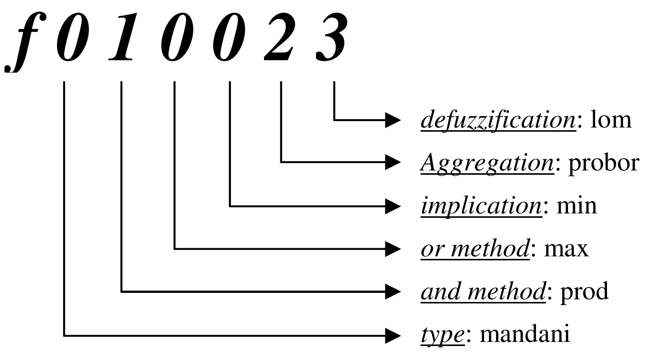
(iii) Connective “or” (disjunction method): 0 = max;

(iv) Implication function method: 0 = min; 1 = prod;

(v) Connective “also” (aggregation method): 0 = max; 1 = sum; 2 = probor;

(vi) Defuzzification method: 0 = centroid; 1 = bisector; 2 = MOM; 3 = LOM; 4 = SOM.

The letter “f’, the initial letter of the word fuzzy merely indicates that the chromosome is a text sequence and it is used just for facilitating the algorithm programming procedure. Therefore, a chromosome coded by ***f010023*** represents a fuzzy signal controller with the following characteristics:



The initial population is formed by a set of chromosomes which are randomly generated by the GA proposed.

5.3.2 Fuzzy signal controller

Controller responses to traffic conditions are generated by fuzzy signal controller components processing. Therefore, each chromosome previously defined given birth to a fuzzy signal controller which provides a matrix of extensions and its corresponding graphic surface.

In this phase of the GA proposed, the following occurs : (i) the generated surface is saved into two files, having extensions .jpg and .fig, respectively; (ii) the surface values are multiplied by 10 and rounded to the next integer in order to represent the extensions in tenths of seconds; and (iii) the matrix with the extension values (also expressed in tenths of seconds) is saved in a table form into two different text files; one file is to be used by the simulation program and the other for the purpose of this analysis.

5.3.4 Traffic simulation software: UnB-Sitracs

The simulation program uses the table with the green time extensions related to the fuzzy signal controller under analysis for simulation of traffic operations. Given that many aspects of the simulation procedure are randomly defined, UnB-Sitracs considers reference seeds for evaluating each replication of the different traffic volumes and intersection geometry to be studied. Identical controller and situation to be simulated will produce the same simulation results for a given seed. Therefore, the fair comparative analysis among the impact of the different controllers on traffic performance is guaranteed by the use of previously defined simulation seeds.

5.3.5 Performance measures

Performance measures are used by the GA proposed for evaluating the effectiveness of the different fuzzy signal controllers generated during the reproduction process. These measures are defined as GA's fitness function. In the GA case proposed, the fitness function is represented by the average control delay per vehicle.

For each controller evaluated at any traffic situation analysed, 10(ten) replications of the simulation process were made. This aimed at preventing that atypical situations, which may occur during the simulation process, produce misleading results in the evaluation of the controller impact on the controlled traffic performance. The replications' results were then treated accordingly, returning to the GA a representative average control delay per vehicle, related to the situation analysed.

5.3.6 Selection

Among the selection methods mentioned in literature for performing the GAs' selection, this work considers the roulette wheel method. In this method, the probability a member is selected for reproduction is proportional to its fitness, according to intervals built based on the fitness of all population members. For members with small fitness having a chance for being selected, a window can be defined for guaranteeing a minimal interval for these members.

5.3.7 Choice of the operator and elitism

After the definition of the intervals for the roulette method, the conditions for the reproduction process must then be specified. They basically consist of probability definition for choosing the crossover operator or mutation operator. The sum of these two probabilities must be equal to 1 (one). The operators are applied until a new population, with the same size of the previous one, being generated.

It is recommended that an accelerated fitness convergence occurs among the actual characteristics and their possible mutations at the beginning of the GA. This means higher crossover rates. Along further new generations, an increase in modification rates (produced by mutation) is desirable, aiming to leave good, but not optimal, fitness intervals.

Therefore, it is recommended to start the GA with the probability of crossover greater than the probability of mutation, providing that these probabilities will vary during the process with the increase of the mutation rate. In this work, the initial probabilities for crossover and mutation are defined as 70% and 30%, respectively. At the end of the process, the corresponding probabilities are 40% and 60%. These variations are planned to happen linearly and at each 5 (five) generations, as the mutation operator prevent the trend for population stabilization.

After the choice of the crossover operator, two population members (parents) must be selected according to the previously referred selection method. After that, crossover operator is again tested according to a second probability level for its occurrence. In case this operator is not chosen, the descendents will be identical to their parents. In this present work, this second level probability was set to 1 (one). That is, once the crossover was selected as the operator to be applied, the change in the genetic material will happen for sure.

The inclusion of new characteristics in future populations, provided by the mutation operator, is desirable. However, it is important to prevent a total transformation of the current population. Because of that, a second level of mutation probability is also specified, which should guarantee a few but always present modifications in the genetic characteristics of population members. The mutation rate is applied to all locus belonging to the parent selected. Based on different preliminary tests with the GA proposed, it was verified that the fitness evolution usually converged to local minima and that this situation could be changed by means of modifications to the second level of the mutation rate. The following rates were tested: 4%, 10%, 15% and 20%. The latter produced the better results and was chosen for this work.

Another relevant aspect considered in the GA proposed is the use of the elitism concept. According to this, if the current population has no member with fitness equal to or greater than the highest fitness in the previous population, the worse member in the current population is excluded and the best fitted member of the previous population takes its place.

5.3.8 Stop criterion

The stop criterion adopted for the present work is that the GA running must be ended as soon as the maximum predefined number of generation is reached. For the application describe herein, this value was set to 50 (fifty).

The application of the elitism concept guarantees that the fitness of any population is equal to or greater than the fitness of the previous one. Due to this situation, it is reasonable to assume that the greater the number of generation the better the results provided by the algorithm. However, when number of generations is very high, depending on the number of population members and the chromosome number of genes, the computer processing time can be prohibitive.

5.3.9 Process registration

In order to further the analysis of each GA interaction results, one text file registering information related to each generation is created. This file contains: (i) chromosomes and corresponding fitness measure value; and (ii) crossover and mutation operations, indicating parents, descendents and changes suffered. A CSV (Comma-Separated Values) files is also created, encompassing data related to the ten replica of the simulation process for each chromosome at all generations. That is, the data from which the average control value per vehicle were defined.

5.3.10 Genetic algorithm output

The final solution to the problem delivered by the GA, its output, is the best fitted member of the last simulated population. The information related to this member's characteristics must be presented in a friendly manner to the user, along with its fitness value.

6 – APPLICATION OF THE GA PROPOSED

The GA proposed was applied to fuzzy signal controller definition for three volume levels: low, medium and high. Taking into account the objectives of this study, two different analyses were conducted for each volume level: one comprehensive analysis and 5(five) running of the GA proposed. The comprehensive analysis leads to the evaluation of all 60 (sixty) possible fuzzy signal controllers referred to at Section 5.1.

6.1 Intersection and traffic for the study

This study considers an isolated intersection of two one-way streets. It has two lanes per approach and its legs are 500 meters long. The width of each lane is equal to 5 meters. The traffic to be controlled includes only passenger cars, presenting the following distribution for free-flow speed: 2% at 40km/h; 15% at 50km/h; 70% at 60km/h; and 13% at 70km/h.

The three traffic volume levels considered are:
- (i) high: major street approach = 1800 veh/h; minor street approach = 600 veh/h;
- (ii) medium: major street approach = 1200 veh/h; minor street approach = 400 veh/h;
- (iii) low: major street approach = 600 veh/h; minor street approach = 200 veh/h.

Details in all the parameters required for traffic simulation with UnB-Sitracs are available in Labanca (2009).

6.2 GA parameters

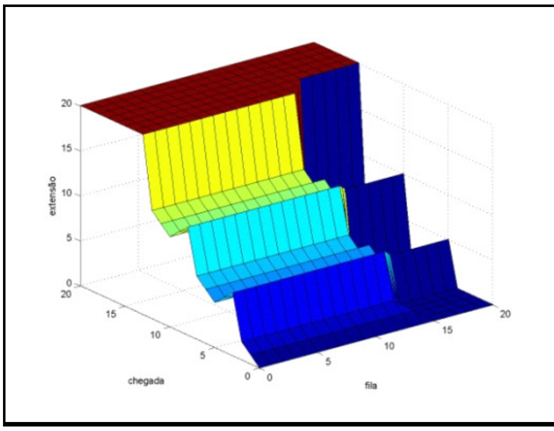
The present application of the GA proposed considers the following parameters:

(i) Population size: 06 members;
(ii) Selection window: 0.3;
(iii) Initial probabilities for operators' choice: crossover = 70%; mutation = 30%;
(iv) Final probabilities for operators' choice: crossover = 40%; mutation = 60%;
(v) Reduction of crossover rate at each 5 (five) generation;
(vi) Second level probability for crossover: 100%;
(vii) Second level probability for mutation: 20%;

6.3 Comprehensive analysis results

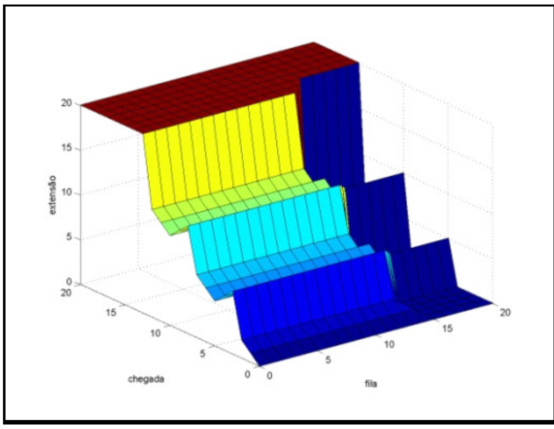
The results of the comprehensive analysis for the three volume levels are presented in Figures 5, 6 and 7. They referred to the best fuzzy controller found (chromosome) for each traffic volume level. The characteristics and results of the 59 non-optimal controllers (chromosomes) are described in Labanca (2009).

For the low volume situation, two chromosomes produced the same smallest average control delay, equal to 4.71 seconds. They differ from each other only for the operator selected for connective "*and*" implementation (see Figures 5a and 5b).



f000003	
Parameters	
Type	mandani
And	min
Or	max
Imp	min
Agg	max
Def	lom

(a) First controller selected

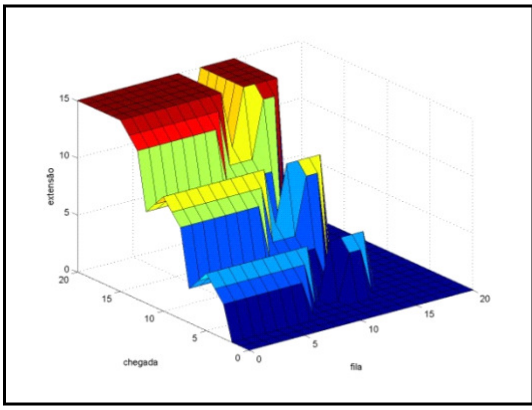


f010003	
Parameters	
Type	mandani
And	prod
Or	max
Imp	min
Agg	max
Def	lom

(b) Second controller selected

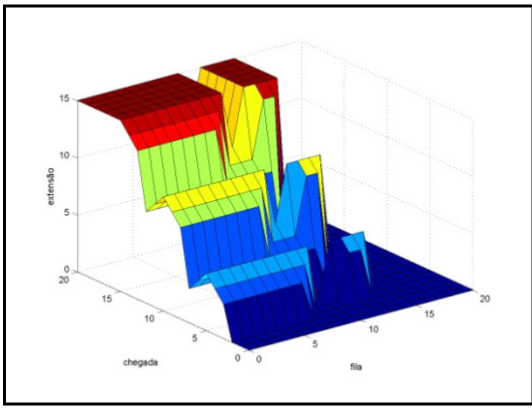
Figure 5: Characteristics and control surface for controllers selected for low volume

In case of medium volume level, the smallest average control delay was produced by the controller described in Figure 6 and is equal to 10.34 seconds. For the high volume level, the comprehensive analysis revealed that the best controller (see Figure 7) caused an average control delay of 11.65 seconds to traffic controlled.



f000004	
Parameters	
Type	mandani
And	min
Or	max
Imp	min
Agg	max
Def	som

Figure 6: Characteristics and control surface for controller selected for medium volume



f000004	
Parameters	
Type	mandani
And	min
Or	max
Imp	min
Agg	max
Def	som

Figure 7: Characteristics and control surface for controller selected for high volume

The results presented for the low traffic volume show that the conjuction methods "*min*" and "*prod*" produced identical control surfaces for the considered combinations of the other controllers' components. In both cases the defuzzification method selected was the largest of maximum method (LOM).

For the other two volume levels (medium and high) studied, the best defuzzification method was the smallest of maximum method (SOM). The controllers that produce the best performance measure for these volumes were the same (Figures 6 and 7 show identical controllers).

Based on the comprehensive analysis performed, it was observed that for the studied situarions, the fuzzy signal controllers with the best performance are related to irregular control surfaces. The analysis also showed that the definition of the best controller depends on the traffic volume level at the intersection approaches. This indicates that the characterístics of the traffic to be controlled, and probabily the geometric elements of the intersection, should be considered for fuzzy signal controllers design purpose.

6.4 GA results

The same simulation conditions used for the comprehensive analysis were adopted for the work with the GA proposed. This causes that identical controllers will produce equal performance measures for the same traffic volumes.

Tables 1, 2 and 3 presents the results produced by 5 (five) independent running of the GA proposed. They show the characteristics of the chromosome with the best performance for each running, the corresponding average control delay, and generation number in which this delay was first found (the generation where the best controller was effectively chosen). Figures 8, 9 and 10 present the convergence process for one running of the GAs applied to low volume (GA3), medium volume (GA3) and high volume (GA4), respectively.

Table 1: Results of the GA proposed - Low Volume

GA	Chromosome	First Occurrence	Average Control Delay (s)
1	f010003	02nd generation	4.7137
2	f010003	46th generation	4.7137
3	f010003	09th generation	4.7137
4	f000014	01st generation	4.7255
5	f010012	02nd generation	4.7255

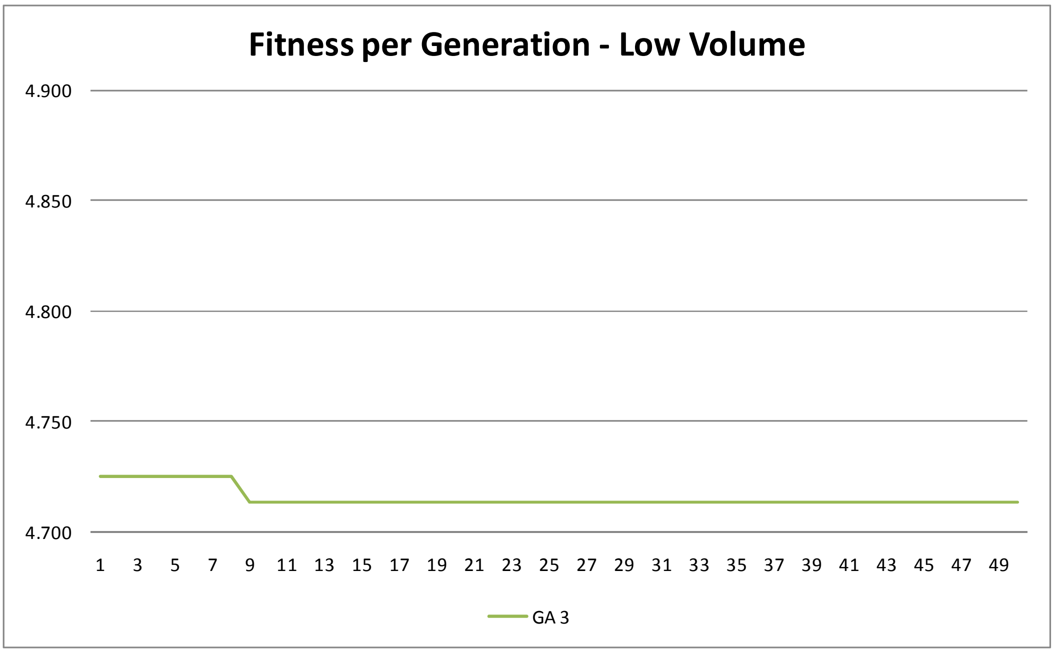


Figure 8: Graphic analysis of the GA proposed – Low Volume (AG3)

Table 2: Results of the GA proposed - Medium Volume

GA	Chromosome	First Occurrence	Average Control Delay (s)
1	f000004	01st generation	10.3383
2	f000004	15th generation	10.3383
3	f000004	18th generation	10.3383
4	f000004	32th generation	10.3383
5	f000004	38th generation	10.3383

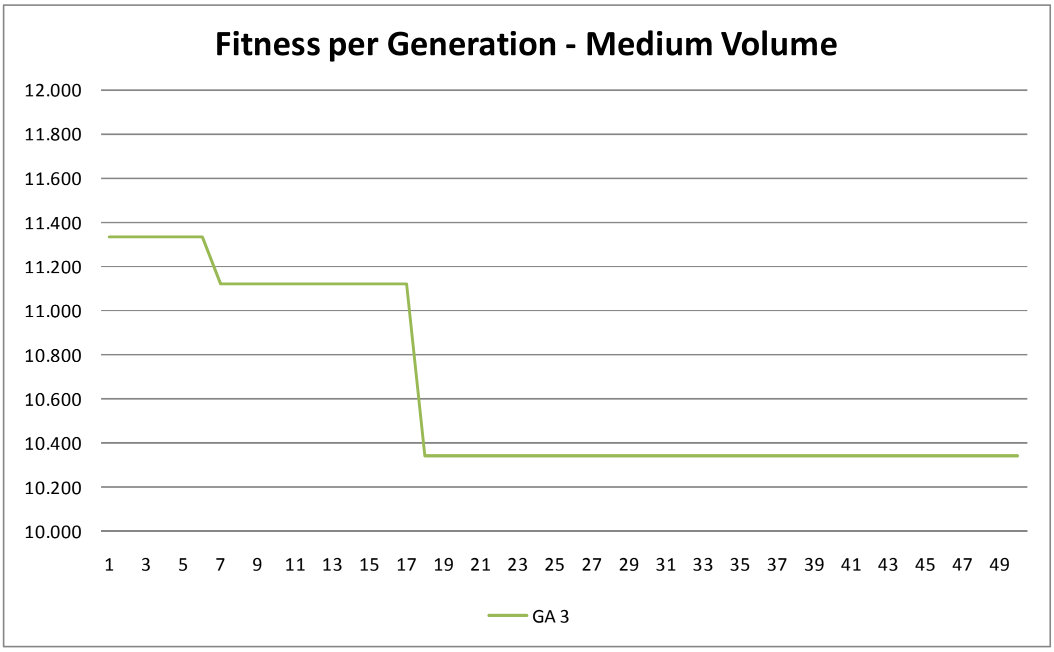


Figure 9: Graphic analysis of the GA proposed –Medium Volume (AG3)

Table 3: Results of the GA proposed - High Volume

GA	Chromosome	First Occurrence	Average Control Delay (s)
1	f000112	01st generation	11.8780
2	f000004	46th generation	11.6490
3	f000004	04th generation	11.6490
4	f000004	13th generation	11.6490
5	f000004	12th generation	11.6490

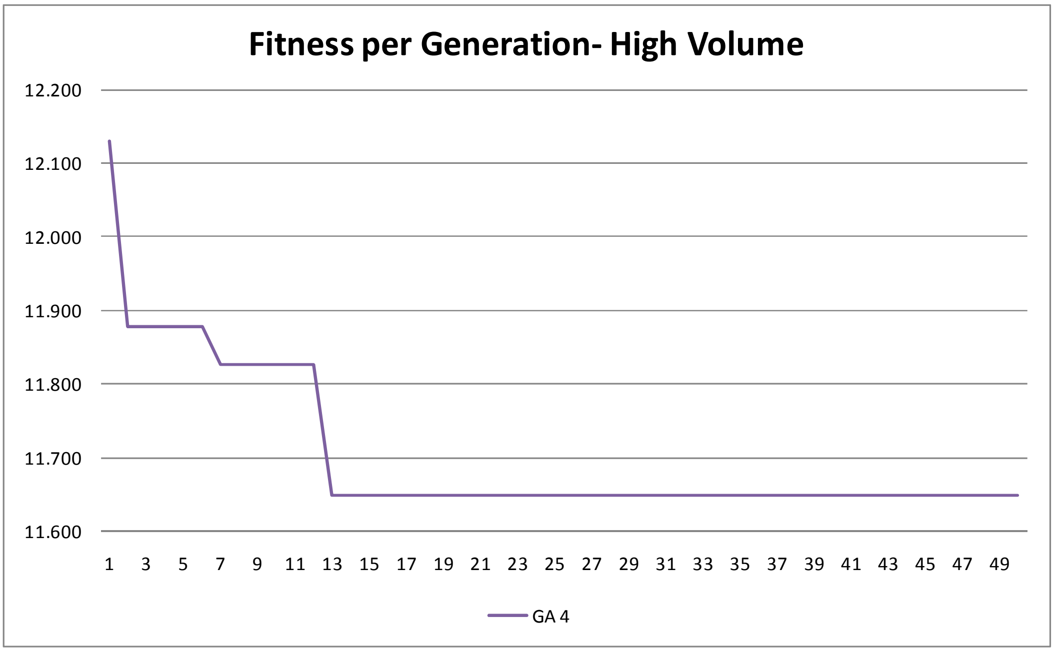


Figure 10: Graphic analysis of the GA proposed – High Volume (AG4)

Among the 15 (fifteen) runnings of the GA proposed, 12 (twelve) of them selected fuzzy signal controllers identical to those defined by the previous comprehensive analysis. For the medium volume level, all the GA results were related to the optimal fuzzy signal controller identified by the global analysis (compare the GAs selected controllers to that shown in Figure 6). Comparing the results presented in Table 1 with the controllers showed in Figure 5, it can be observed that for the low volume situation, 2 (two) of the 5 (five) runnings do not reach optimal controllers. Only one running of the GA for the high volume did not reach the optimal controller. It is important to highlight that for the 3 (three) runnings with non-optimal results, the fuzzy signal controller selected produced average control delay extremelly closed to the corresponding optimal result. That is, the GA proposed returned the optimal controller in 80% of the runnings performed and its results were quite satisfactory for the other 20%.

Other important findings of the GA present application were: (i) there was no regular pattern for the number of necessary generations to reach the optimal solution; and (ii) the occurrence of constant fitness level all through significative number of generations, as per Figures 8, 9 and 10. This latter situation makes clear the positive impact of the elitism concept implementation in the GA proposed. It prevent modifications made to consective populations from causing a decline in current fitness level in any generation. The first finding shows how difficult it is to preview the number of generations necessary to reach the optimal solution, even for a simple application as the one presented here.

In spite of the randomness of the GA convergence process, during the present application no relationship was found between the fitness level of the initial populations and the AG convergence process itself.

7 – CONCLUSIONS

The results of the application of the GA proposed in this work corroborate the algorithm's genetic effectiveness for aiding the design of fuzzy signal controllers. However, it is important to point out that the fitness function of the GA proposed only takes into account the average control delay. For future applications, depending on the traffic control goals, other traffic performance measures should be included.

One important aspect that must be considered is that the control surfaces of the optimal controllers are excessively irregular and not completely coherent under the traffic engineer point of view. For example, the control surfaces presented in Figures 5, 6 and 7 reveal that the controller response is less sensitive to variations in queue length than it should be expected. Also, in many cases, the green extension decreases when the number of arrival increases at identical queue level. Therefore, it is recommended that the fitness function, in some degree, takes into account the smoothness and coherence of the control surface.

In terms of the algorithm genetic per se, the preliminary studies regarding the definition of the second level mutation probability indicate that the calibration of the algorithm parameters is not a trivial task. However, the calibration effort is necessary to provide that the genetic algorithm reaches as much as possible its objectives. Some parameters, as is the case with population size, number of generations and probabilities of crossover and mutation (first and second levels), are very critical to the GA performance in large search spaces. Even in a small search space as that considered in the present study, the importance of a suitable definition of the mutation rate was clearly observed.

Finally, the results of this work recommend further research toward overcoming the detected limitations of the GA proposed and, specially, to test its general framework for selecting other components of fuzzy signal controllers. Among them, it is relevant to consider: (i) the partition of the universe of discourse for the definition of the input and output variables' fuzzy sets; (ii) the membership function to each fuzzy set; (iii) the type of fuzzy controller (Mamdani and Sugeno).

REFERENCES

Andrade, M. and M. A. P. Jacques. (2006). Study to Improve the Operation of Fuzzy Signal Controllers for Isolated Intersections (*in Portuguese*). In: Transporte em Transformação, pp. 79-95. LGE Editora Ltda, Brasília.

Asari, M., M. A. P. Jacques, M. Ladeira, P. C. M. Silva, and M. Andrade (2008). Development of a Traffic Micro Simulation Program for Studies Related to Signalized Intersections (*in Portuguese*). In: *XIV Congresso de Iniciação Científica da Universidade de Brasília*, PAINEL A-107.

Chambers, L. D. (2001). The Practical Handbook of Genetic Algorithms, Applications, 2nd ed: Chapman & Hall/CRC.

Jacques, M. A. P., J. Niittymäki, and M. Pursula (2002a). Analysing Different Fuzzy Traffic Signal Controllers for Isolated Intersections. Paper accepted for presentation at TRB 81st Annual Meeting. January 13-17, Washington, DC.

Jacques, M. A. P., M. Pursula, J. Niittymaki, and I. Kosonen (2002b). The Impact of Different Approximate Reasoning Methods on Fuzzy Signal Control In: *The 13th Mini-EURO Conference*, 2002, Bari. The 13th Mini-Euro Conference – Handling Uncertainty in the Analysis of Traffic and Transportation Systems. Bari: Local Organizing Committee – Politecnico di Bari, V.1 pp.184 – 191.

Jacques, M. A. P., M. Pursula, J. Niittymaki, and I. Kosonen (2002c). Analysis of Different Deffuzification Methods Applied to Fuzzy Signal Controllers. In: *XII Congresso Panamericano de Ingeniería de Tránsito y Transporte*, Quito.

Jacques, M.A.P., D. B. L. M. Santos, M., Pursula, and I. Kosonen (2005). I. Importance of Fuzzy Sets Definitions for Fuzzy Signal Controllers. Applied Research in Uncertainty Modeling and Analysis, Chapter 18.. Edited by Nii O. Attoh-Okine and Bilal M. Ayyub. Springer.

Labanca, F. (2009). Design of Fuzzy Signal Controllers Using Genetic Algorithm (*in Portuguese*). Monografia de Projeto Final em Transportes. Universidade de Brasília, Departamento de Engenharia Civil e Ambiental. Brasília.

Miranda, M. N. (2000) Genetic Algorithms: Fundamentals and Applications (*in Portuguese*). Universidade Federal do Rio de Janeiro. Disponível em: http://www.gta.ufrj.br/~marcio/genetic.html.

Pappis, C. P. and H. Mamdani (1977) A Fuzzy Logic Controller for Traffic Junctions. IEEE Transactions on Systems, Man and Cybernetics. Vol. SMC-7, Nº 10.

Soares, G. L. (1997). Genetic Algorithms*: Study, New Techniques and Applications* (*in Portuguese*). Escola de Engenharia da Universidade Federal de Minas Gerais.

TRB- Transportation Research Board (2000). *Highway Capacity Manual*. Transportation Research Board, National Research Council, Washington, D.C.

Vaz, A. M., E. R. de Oliveira, and M. A. P. Jacques (2005). Impact of Membership Functions on Fuzzy Signal Controllers' Operation: Exploratory Study (*in Portuguese*). In: *XIX ANPET – Congresso de Pesquisa e Ensino em Transportes*, Rio de Janeiro, v.1. pp. 853-862.