Traffic light optimization for Bus Rapid Transit using a parallel evolutionary algorithm: the case of Garzón Avenue in Montevideo, Uruguay

Efraín Arreche, Sergio Nesmachnow, Renzo Massobrio
Universidad de la República, Montevideo, Uruguay
{*efrain.arreche,sergion,renzom*}*@fing.edu.uy*

Christine Mumford
Cardiff University, Wales, UK
MumfordCL@cardiff.ac.uk

Ana Carolina Olivera, Pablo Vidal
Universidad de la Patagonia Austral, Argentina
acolivera@conicet.gov.ar, pjvidal@uaco.unpa.edu.ar

Abstract

This article proposes the design and implementation of a parallel evolutionary algorithm for public transport optimization on Garzón Avenue, Montevideo, Uruguay. This is an interesting complex urban scenario, due to the number of crossings, streets, and traffic lights in the zone. We introduce an evolutionary algorithm to efficiently synchronize traffic lights and improve the average speed of buses and other vehicles. The experimental analysis compares the numerical results of the evolutionary algorithm against a baseline scenario that models the current reality. The results show that the proposed evolutionary algorithm achieves better quality of service improving up to 15.3% average bus speed and 24.8% the average speed of other vehicles.

Keywords: traffic light synchronization; evolutionary algorithms; bus rapid transit.

1 Introduction

The number of vehicles has been growing steadily in the last twenty years worldwide. This growth severely affects the development of cities and also the quality of life of people [4]. Traffic problems progressively decrease the average speed of vehicles, which in turn lowers the acceptance of public transport. A number of intelligent solutions for transport systems have been proposed. *Bus Rapid Transit* (BRT), for example, has gained popularity because it provides a good user experience and reduced implementation costs when compared against other solutions.

Montevideo has a growing problem of traffic congestion. The local authorities have implemented a *Urban Mobility Plan* to reduce the impact of this problem [8]. The transport system in the Urban Mobility Plan is inspired by BRTs: several streets and avenues with priority for buses have been proposed for the city. The first element of the Urban Mobility Plan was implemented in Garzón Avenue (north of Montevideo). This avenue includes 24 intersections with traffic lights and exclusive lanes for buses. The BRT on Garzón Avenue has been much criticized for failing to streamline public transport, which is one of the main objectives of the Urban Mobility Plan.

Two main strategies are applied for traffic optimization: influencing drivers' behavior (by setting traffic lights, installing signs, etc.) and changing the infrastructure (adding new lanes, widening streets, etc.) [9]. Infrastructure modifications can significantly improve the traffic flow, but they are expensive and need physical space that is not often available. Strategies to influence the drivers' behavior are usually a better (or even the only viable) option in many scenarios. Methods for synchronizing traffic lights are among the most effective in speeding transit and avoiding congestion. The synchronization problem is complex when dealing with real-world scenarios, thus computational intelligence techniques are applied to find good quality solutions efficiently [6, 13, 14].

This article presents a parallel evolutionary algorithm (EA) for traffic optimization by synchronizing traffic lights in Garzón Avenue. We aim to provide an efficient and innovative solution, improving the quality of service offered to the citizens. Garzón poses a complex challenge due to an extensive urban area, a large number of crossings and traffic lights, rules for exclusive lanes, and different types of traffic on the roads. The experimental results show that the proposed parallel EA improves the average speed of buses and vehicles when compared with the current scenario.

The main contributions of the research reported in this article include: *i*) a study of the problem of traffic light synchronization to streamline public transport; *ii*) the design and implementation of a parallel EA that is able to efficiently solve the problem; and *iii*) the experimental evaluation of the proposed EA over realistic instances of the problem, created using real data collected in-situ.

The article is organized as follows. Section 2 presents the problem description. The proposed EA is described in Section 3. Section 4 presents the experimental analysis over realistic case studies in Garzón area. The conclusions and the lines of current and future work are discussed in Section 5.

2 Public transport optimization via traffic light optimization

This section presents the problem model and its mathematical formulation.

2.1 Problem model

The problem model simplifies the reality, considering only those feaures relevant for traffic light synchronization. We build a map of the area and instances including real data collected in-situ. Simulations are used to evaluate the solutions. The methodology and tools used are described next.

Traffic simulator. We use SUMO [3], a free open-source traffic simulator that allows modeling streets, vehicles, public transport, and traffic lights. SUMO applies a microscopic model, explicitly simulating each element in the scenario. Based in a set of configuration files that represent the road network, vehicles, traffic, and traffic lights, SUMO generates output files with useful information from the simulated scenario: simulation time, number and speed of vehicles, travel durations, etc.

Map. We used the Open Street Map (OSM) service [7] to design a map of the Garzón area, compatible with the SUMO simulator. The Java OSM Editor was used to adapt the map, keeping only the relevant elements for the problem. We validated the map by comparing it with data gathered in-situ and from other services (Google Maps/Bing Maps). All inconsistencies detected were corrected in the map used in the research. The studied area includes Garzón Avenue and two parallel paths on each side (there are no real parallel streets, so we included a set of parallel and internal roads, see Figure 1). Each parallel path includes two-way streets or two one-way streets to guarantee connectivity. We imported the map from OSM and used the *Net-Convert* application to include real data for traffic lights collected in situ (see next paragraph).

Field research: gathering real data from traffic lights and vehicles. The real data available from the local government is scarce, so field research was needed to get the real traffic data on five representative intersections of the studied area. We applied the recommendations for vehicle counting by Smith and McIntyre [15] to avoid bias: counting vehicles on a working day (Wednesday), with sunny weather, and in representative hours (15:00 to 17:00). Additional data from traffic light configurations and travel times were collected by traveling (in bus and in car) on the studied scenario and also from videos recorded in the zone.

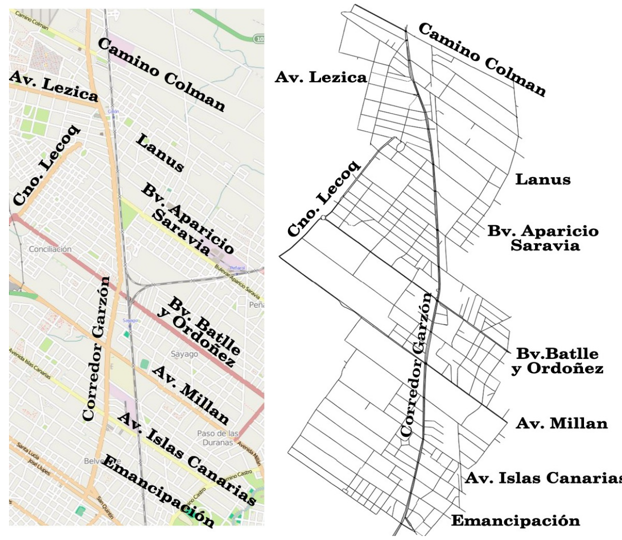

Figure 1: OSM (left) and SUMO map (right).

2.2 Mathematical formulation

The mathematical model is based on combining two relevant problem goals regarding the quality of service provided to the users: the average speed for buses ($\overline{s_B}$) and the average speed for other vehicles ($\overline{s_O}$) in the studied scenario. We optimize (i.e., maximize) both speeds simultaneously, by applying a linear aggregation approach defined by the fitness function $f = w_B \times \overline{s_B} + w_O \times \overline{s_O}$, to be used for solution evaluation in the proposed EA. This way, we can focus on assigning a higher priority to public transport (buses), by choosing appropriate values for weight w_B.

The traffic speed optimization is performed not only on Garzón Avenue, but over the entire road network. Applying this global optimization approach is crucial to achieve a traffic light configuration that guarantees a sustainable mobility improvement. This cannot be assured if the problem model only considers some streets (e.g., only Garzón Avenue) or optimizes each intersection separately.

2.3 Related works

A number of recent articles have addressed the optimization of BRTs by traffic light synchronization using evolutionary computing techniques. The most closely related publications are described next.

Sánchez et al. [14] applied an EA for traffic light synchronization to improve the traffic flow in a city scenario with 42 traffic lights and 20 output roads in Santa Cruz de Tenerife, España. Nine hand-made solutions from the City Hall are used as the initial population, and a traditional two-point crossover is applied. The fitness function evaluates the travel time for vehicles in the simulated road network. The evolutionary approach was able to improve up to 26% the trip times over the City Hall solutions, but no details about the benefits for public transportation are reported.

Rouphail et al. [13] studied a small traffic network (9 crossings) in Chicago, USA, including bus stops and real traffic data. The authors proposed an EA to control traffic lights, using a fitness function that takes into account the delays and the length of the queues in each crossing. The EA was able to reduce the delays in up to 44% when comparing against the non-optimized scenario.

The previous works apply a similar approach to the one we apply in our research. However, our study is performed over a larger scenario (6.5 km in Garzón Avenue, more than 30 km^2), a significantly larger number of intersections, 28 bus stops included in the zone, and specific mobility logic due to the BRT regulations (exclusive lines, priorities, and allowed/forbidden turning corners).

3 A parallel EA for traffic light synchronization

This section describes the methodology and the parallel EA for traffic light synchronization.

3.1 Metaheuristics and evolutionary algorithms

Metaheuristics are high-level strategies for designing computational methods to find approximate solutions for complex problems [11]. EAs are non-deterministic metaheuristic methods that emulate the evolution of species in nature to solve optimization, search, and learning problems [2]. In the past thirty years, EAs have been applied to solve many highly complex optimization problems.

EAs are iterative methods that apply stochastic operators on a set of *individuals* (the *population*) that encode candidate solutions for a problem. The initial population is generated randomly or using a specific heuristic. A *fitness* value is assigned to every individual, indicating how good it is at solving the problem. Iteratively, probabilistic *variation operators* (*recombination* of individuals or *mutations* in their contents) are applied for building new solutions during the search. The search is guided by a probabilistic selection-of-the-best technique to tentative solutions of higher quality.

Parallel models for metaheuristics and EAs have been proposed to speed up the computing time of the search when dealing with complex objective functions and hard search spaces [1]. In this work, we apply a master-slave model for parallelization, to reduce the execution time of performing the traffic simulations for the studied scenario. This work applies a traditional EA, implemented in C++ within the Malva library for optimization [5]. We included specific modifications in the Malva code in order to implement the parallel model for fitness evaluation using threads.

3.2 The proposed implementation

Solution encoding. The proposed encoding includes the elements needed for traffic light planning: *i*) the duration for each of the multiple *phases* allowed in every intersection, and *ii*) the *offset*, indicating the time the light cycle starts. All values are expressed in seconds. Figure 2 presents an example of the solution encoding: the information is logically grouped into crossings, storing the time for each phase (amber lights are omitted, as they do not affect the times of passing vehicles). The length of a representation depends on the number of crossings and the number of phases defined. This encoding allows the overall optimization of the scenario (all intersections are optimized simultaneously).

Fitness function. We apply the fitness function defined in Section 2, which accounts for the optimization of the average speed of buses and vehicles over the defined scenario. Several weight combinations are used to explore different priorities between buses and vehicles.

Population initialization. A set of initial solutions is built by using the data collected from the current (non-optimized) reality on Garzón Avenue. Small perturbations (i.e., phase time and offset changes) are applied to provide diversity to the initial population.

Recombination. We apply a modified one point crossover: after selecting the crossover point, offspring are built by combining phases and offsets, trying to keep nearby crossings synchronized.

Mutation. Two mutation operators are applied: *i*) a Gaussian mutation to modify the values of phases; and *ii*) a random modification (according to a uniform distribution) of the offset values. Both mutations are applied according to a given mutation probability.

Selection and replacement. We use tournament selection (three individuals participate, one survives). The $(\mu+\lambda)$ evolution model is applied, where parents and offspring compete for survival.

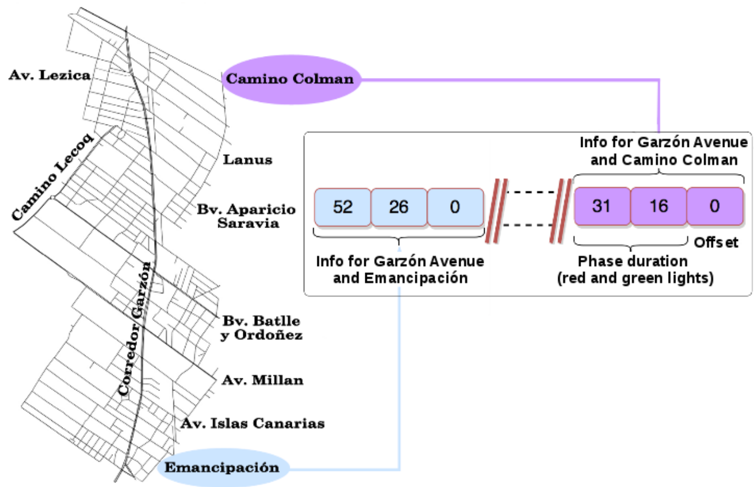

Figure 2: Example of the solution encoding applied in the parallel EA.

Parallel model. A master-slave model is applied for fitness function evaluation: a master process handles the population and a pool of threads. In each generation, the master assigns a set of solutions to slave processes, executing in those threads. Slaves perform the simulations to evaluate each traffic light configuration and return the results to the master, to be used in the evolution.

4 Experimental analysis

This section reports the experimental evaluation of the proposed EA for traffic optimization in Garzón Avenue. The analysis was performed on an AMD Opteron 6272 at 2.09GHz (64 cores, 48GB RAM, CentOS Linux 6.5), from Cluster FING, Universidad de la República, Uruguay [10].

4.1 Problem instances

We designed realistic problem instances using data from the field research and from the city authorities. A *baseline scenario* is built using the actual configuration of traffic lights in Garzón Avenue, to be used as a reference to compare the results computed by the proposed parallel EA.

Three XML files are used in the SUMO simulation: *i*) *traffic light configuration*, defining the location, phases and offsets; *ii*) *vehicle routes*, built in *Traffic Modeler* [12] using real data and a mobility model that provides good granularity for traffic density; and *iii*) *public transport details*, including paths, frequencies, stop locations, and delay times in each stop. We collected data from all urban lines in the zone (*G*, *D5*, *2*, *148* and *409*), analyzed one month of GPS data (position/speed) from buses to determine mobility patterns and average speed in Garzón Avenue (14.5 km/h), and studied videos to compute the delays (between 20 and 35 seconds, depending on the stop).

Three traffic patterns are studied: *i*) *normal traffic*, with data from the field research, including 2000 vehicles and 70 buses; *ii*) *low traffic*, using data from weekends and night hours, having 1000 vehicles and 70 buses, and significantly shorter delays on the bus stops because fewer people use the public transport; and *iii*) *high traffic* using data from rush hours, including 3000 vehicles and 70 buses. Bus frequencies are not affected by the traffic density. All data was contrasted and verified with the information provided by the city administration.

4.2 Parameters setting

We studied the *simulation time* for the proposed scenarios and the best values for population size, stopping criterion, and probabilities of recombination (p_R) and mutation (p_M) in the parallel EA.

To avoid bias, a different set of instances was used for the parameter setting analysis: *low traffic*, (500 vehicles/30 buses); *normal traffic* (1000 vehicles/60 buses); and *high traffic*, (2000 vehicles/120 buses). Ten independent executions of the proposed EA were performed for each problem instance.

Simulation time. The best results were obtained using 4000 simulation steps, which represent 66 minutes in reality, allowing more than 85% of the vehicles in the scenario to reach destination.

Stopping criterion. We try to balance the solution quality and the execution time of the proposed EA. As the fitness values did not vary significantly after 400 generations, we decided to use a limit of 500 generations, which demands between 1 and 24 hours of execution time.

Population size. We consider the quality of results, the execution time, and the computing elements available to find an appropriate population size in the proposed EA. We analyzed using 32, 48, and 64 individuals. No significant improvements in the fitness values were detected when using larger populations, so we decided to use 32 individuals, to have the shorter execution times.

Operators probabilities. We explored all combinations of the candidate values $p_R \in \{0.5, 0.8, 1\}$, and $p_M \in \{0.01, 0.05, 0.1\}$. A statistical analysis applying the Student t-test concluded that the combinations (p_R=0.5, p_M=0.1) and (p_R=0.5, p_M=0.01) computed the best results. We chose the parameter configuration (p_R=0.5, p_M=0.01), which provides faster execution times.

4.3 Numerical results for Garzón Avenue

We performed 30 independent executions of the proposed EA for each problem instance studied, and compared the results against those obtained for the baseline scenario. The baseline results for the comparison were computed by simulating the baseline scenario. We verified that the results for average speed and travel times matched those computed when processing the GPS data from the city authorities, thus validating the proposed approach using simulations.

Table 1 reports the results of the optimization using the proposed parallel EA. Speeds are expressed in km/h and improvements are computed over the results of the baseline scenario. The parallel EA allows improving the average speed over the baseline scenario (for the three traffic patterns studied) up to **24.2%** (fitness), up to **15.3%** (average bus speed), and up to **24.8%** (average vehicle speed). We applied the Kruskal-Wallis test to analyze the results distributions. The proposed parallel EA outperformed the baseline results with statistical significance in all scenarios.

Table 1: Numerical results of the proposed parallel EA.

instance	*baseline scenario*			*parallel EA results*					
	$\overline{s_B}$	$\overline{s_O}$	*fitness*	$\overline{s_B}$	$\overline{s_O}$	*fitness*		*fitness improvement*	
						average±σ	best	average	best
low traffic	15.89	32.45	13.42	17.92±0.18	34.30±0.40	14.50±0.14	14.88	8.04 %	10.8 %
medium traffic	14.59	28.81	12.0	16.95±0.32	33.29±0.29	13.95±0.15	14.19	15.7 %0	17.7 %
high traffic	14.31	26.36	11.30	16.51±0.61	32.90±0.25	13.72±0.17	14.04	21.40 %	24.2 %

Table 2 shows the results when using different weights to prioritize the speed of buses or vehicles. The results indicate that an additional 2% of improvement in the speed of buses can be achieved, with a negligible reduction on the speed of other vehicles (results in bold font).

Table 2: Numerical results when using different weights in the fitness function.

instance	w_B	w_O	$\overline{s_B}$	$\overline{s_O}$	*fitness*	$\Delta\overline{s_B}$	$\Delta\overline{s_O}$
	0.5	0.5	17.92±0.18	34.30±0.40	14.50±0.14	–	–
low traffic	0.7	0.3	17.93±0.23	34.06±0.17	12.65±0.11	+0.07	-0.7
	0.3	0.7	17.55±0.20	34.71±0.21	16.42±0.10	-2.06	+1.18
	0.5	0.5	16.95±0.32	33.29±0.29	13.95±0.15	–	–
normal traffic	0.7	0.3	**17.29±0.27**	33.08±0.14	12.24±0.12	**+2.0**	-0.62
	0.3	0.7	16.71±0.42	33.70±0.31	15.92±0.11	-1.41	+1.49
	0.5	0.5	16.51±0.60	32.90±0.25	13.72±0.17	–	–
high traffic	0.7	0.3	**16.72±0.14**	32.79±0.26	13.75±0.07	**+1.24**	-0.33
	0.3	0.7	15.48±0.42	33.20±0.25	15.49±0.16	-6.23	+0.92

Figure 3 compares the travel times in Garzón Avenue (6.5 km): it is reduced from 27.3 m to 23.6 m in the high traffic scenario (for buses) and from 14.8 m to 11.9 m (for vehicles) in that case.

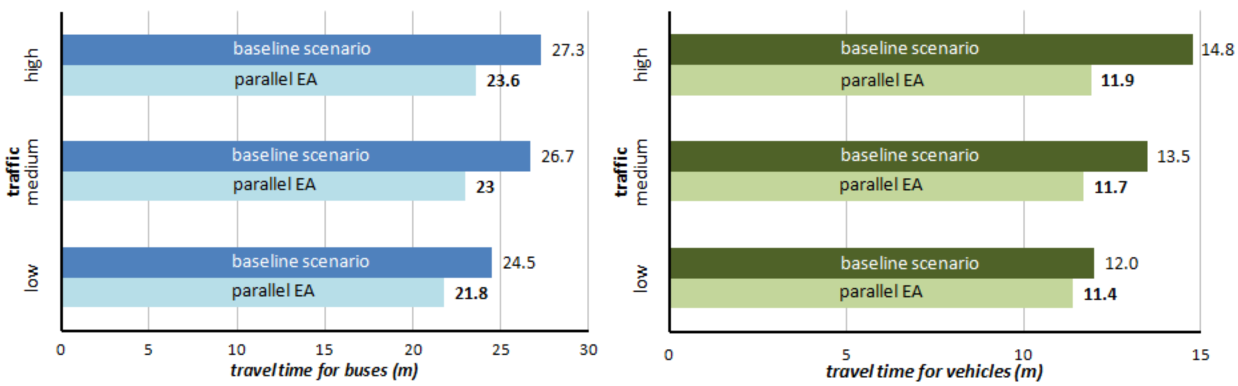

Figure 3: Travel time durations for travels in Garzón Avenue (in minutes).

4.4 Performance analysis

We studied the execution time of the master-slave parallel model. Figure 4 analyzes the *speedup* (i.e., how faster the parallel EA is when compared to the sequential implementation) for a set of representative scenarios. The *efficiency* is the normalized speedup (i.e., dividing it by the number of computing resources). The parallel EA is **26.9**× faster when using 32 computing resources, executing in 44 minutes the optimization that requires 20 hours for the sequential version.

#	*instance*	$t_{SEQ}(m)$	$t_{PAR}(m)$	*speedup*	*efficiency*
1		1572	59	26.6	0.83
2	*low traffic*	1571	59	26.6	0.83
3		1183	44	**26.9**	**0.84**
4		3002	119	25.2	0.79
5	*normal traffic*	2195	82	26.8	0.84
6		3007	120	25.1	0.78
7		2920	110	26.6	0.83
8	*high traffic*	4365	183	23.9	0.75
9		4276	177	24.2	0.75
			average	25.7±1.1	0.80±0.03

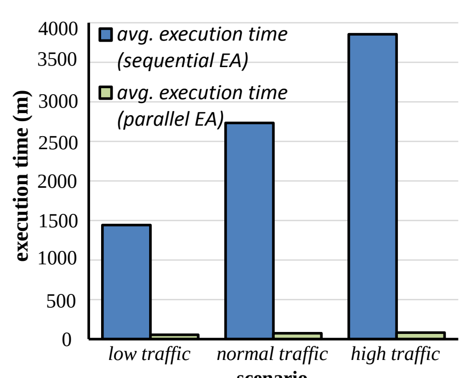

Figure 4: Execution time analysis of the proposed parallel EA (minutes)

5 Conclusions and future work

This article presented a parallel EA to optimize public transport by synchronizing traffic lights in Garzón Avenue, Montevideo, Uruguay. We considered several complex features of a real urban zone to devise a methodology that apply real maps and data, analysis of GPS information, traffic modeling and simulation, and computational intelligence for optimization. This is an innovative approach in our country, where intelligent transport systems have not been applied up to date.

The experimental analysis compared the parallel EA against the results from a baseline scenario that models the current reality. The results indicate that the parallel EA is able to compute traffic light configurations that accounts for a better quality of service, improving up to **15.3%** the average bus speed and **24.8%** the average speed of vehicles. An additional improvement of 2% in the speed of buses is achieved when assigning a higher priority to this objective.

The master-slave parallel model was effective to reduce the execution times needed to solve the problem, achieving speedup values of up to **26.9**×, allowing to execute in 44 minutes the optimization that would require 20 hours when using the sequential version.

The main lines for future work include studying infrastructure modifications in Garzón Avenue. The proposed methodology can also be applied for traffic optimization in other urban scenarios.

References

[1] E. Alba, G. Luque, and S. Nesmachnow. Parallel metaheuristics: Recent advances and new trends. *Int. Trans. in Operational Research*, 20(1):1–48, 2013.

[2] T. Bäck, D. Fogel, and Z. Michalewicz. *Handbook of evolutionary computation.* Oxford Univ., 1997.

[3] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. SUMO–Simulation of Urban MObility: An Overview. In *3rd Int. Conf. on Advances in System Simulation*, pages 63–68, 2011.

[4] A. Bull. *Congestión de tránsito: el problema y cómo enfrentarlo.* United Nations Publications, 2003.

[5] G. Fagúndez and R. Massobrio. The Malva Project: A framework for comutational intelligence in C++, 2014. Online, `http://themalvaproject.github.io`, March 2016.

[6] J. Garcia-Nieto, A. Olivera, and E. Alba. Optimal cycle program of traffic lights with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 17(6):823–839, 2013.

[7] M. Haklay and P. Weber. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.

[8] IMM. Plan de movilidad urbana, 2010. Online, `http://www.montevideo.gub.uy`, March 2016.

[9] D. McKenney and T. White. Distributed and adaptive traffic signal control within a realistic traffic simulation. *Engineering Applications of Artificial Intelligence*, 26(1):574–583, 2013.

[10] S Nesmachnow. Computación cientıfica de alto desempeño en la Facultad de Ingenierıa, Universidad de la República. *Revista de la Asociación de Ingenieros del Uruguay*, 61:12–15, 2010.

[11] S. Nesmachnow. An overview of metaheuristics: accurate and efficient methods for optimisation. *Int. Journal of Metaheuristics*, 3(4):320–347, 2014.

[12] L. Papaleondiou and M. Dikaiakos. Trafficmodeler: A graphical tool for programming microscopic traffic simulators through high-level abstractions. In *69th Vehicular Technology Conf.*, pages 1–5, 2009.

[13] N. Rouphail, B. Park, and J. Sacks. Direct signal timing optimization: strategy development and results. In *XI Pan American Conf. in Traffic and Transportation Engineering*, pages 195–206, 2000.

[14] J. Sánchez, M. Galán, and E. Rubio. Applying a traffic lights evolutionary optimization to a real case: "Las Ramblas" area in Santa Cruz de Tenerife. *IEEE Trans. on Evolut. Comput.*, 12(1):25–40, 2008.

[15] D. Smith and J. McIntyre. Handbook of simplified practice for traffic studies. Technical Report 455, Iowa State University, 2002.