Traffic lights synchronization for Bus Rapid Transit in Montevideo (Uruguay) using a parallel evolutionary algorithm

Sergio Nesmachnow, Renzo Massobrio, Efraín Arreche[1]

Faculty of Engineering, Universidad de la República, Julio Herrera y Reissig 565, Montevideo, 11300, Uruguay

Christine Mumford[2]

School of Computer Science & Informatics, Cardiff University, Queen's Buildings, 5 The Parade, Cardiff, CF24 3AA, Wales, UK

Ana Carolina Olivera, Pablo Javier Vidal[3]

Departamento de Ciencias Exactas y Naturales, Universidad Nacional de la Patagonia Austral, Ruta Nº3 Acceso Norte, Caleta Olivia, 9011, Santa Cruz, Argentina

Andrei Tchernykh[4]

Computer Science Department, CICESE Research Center, Carretera Ensenada-Tijuana No. 3918, Zona Playitas, 22860, Ensenada, Baja California, México

Abstract

This article presents a parallel evolutionary algorithm for public transport optimization by synchronizing traffic lights in the context of Bus Rapid Transit systems. The related optimization problem is NP-hard, so exact computational methods are not useful to solve real-world instances. Our research introduces a parallel evolutionary algorithm to efficiently configure and synchronize traffic lights and improve the average speed of buses and other vehicles. The Bus Rapid Transit on Garzón Avenue (Montevideo, Uruguay) is used as a case study. This is an interesting complex urban scenario due to the number of crossings, streets, and traffic lights in the zone. The experimental analysis compares the numer-

[1] {sergion, renzom, efrain.arreche}@fing.edu.uy
[2] MumfordCL@cardiff.ac.uk
[3] {aolivera, pjvidal}@uaco.unpa.edu.ar
[4] chernykh@cicese.mx

Preprint submitted to International Journal of Transportation Science and Technology April 10, 2018

ical results computed by the parallel evolutionary algorithm with a scenario that models the current reality. The results show that the proposed evolutionary algorithm achieves better quality of service when compared with the current reality, improving up to 15.3% the average bus speed and 24.8% the average speed of other vehicles. A multiobjective optimization analysis also demonstrates that additional improvements can be achieved by assigning different priorities to buses and other vehicles. In addition, further improvements can be achieved on a modified scenario simply by deleting a few bus stops and changing some traffic lights rules. The benefits of using a parallel solver are also highlighted, as the parallel version is able to accelerate the execution times up to 26.9× when compared with the sequential version.

Keywords: bus rapid transit, traffic light synchronization, evolutionary algorithms

---

1. Introduction

The number of vehicles has been growing steadily worldwide in the last twenty years. This growth is one of the main causes of serious problems related to traffic congestion, which severely affect the development of cities and the quality of life [1]. Urban intelligence methods have been widely applied to address several issues in modern smart cities [2]. One of the main problems in big urban areas is related to citizens mobility, especially when using public transportation. A large number of private vehicles in circulation impacts negatively on the efficiency of public transport service, thus lowering its acceptance. To deal with these problems, a number of intelligent solutions have been proposed, which are included in the paradigm of *Intelligent Transportation Systems* (ITS).

ITS integrate synergistic technologies, computational intelligence, and engineering concepts to develop and improve transportation. ITS are aimed at providing innovative services for transport and traffic management, with the main goals of improving transportation safety and mobility, and also enhancing productivity [3].

The ITS paradigm can be applied in combination with other innovative approaches for public transportation. *Bus Rapid Transit* (BRT) is a mass transit system that has gained popularity because it provides a good user experience and reduced implementation costs when compared against more expensive solutions, such as metros [4, 5].

In Montevideo, the capital city of Uruguay, there is a growing problem of traffic congestion, similar to the issues arising in many other cities in Latin America. Although the situation is not as dramatic as in larger cities, the local authorities have taken steps towards reducing the impact of this problem by implementing an *Urban Mobility Plan* to improve the efficiency of public transport and allow a more democratic access to it [6].

The Urban Mobility Plan proposes including BRTs, with priority for buses, in the city. One of the first elements of the Urban Mobility Plan was the BRT implemented in Garzón Avenue, located in the north of Montevideo. This avenue includes 24 intersections with traffic lights and exclusive lanes for buses. Open to the public since 2012, 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.

Traffic optimization methods aim at improving the flow of vehicles on the road network. The strategies are classified in two main categories: *i*) those that influence drivers' behavior (by setting traffic lights, installing signs, etc.) and *ii*) those that propose changing the infrastructure (adding new lanes, widening streets, etc.) [7]. Infrastructure modifications can significantly improve traffic flow, but they are expensive and need physical space that is not often available. For this reason, strategies to influence 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, improving the development of cities and the quality of life of citizens.

The traffic lights synchronization problem is complex when dealing with real-world scenarios. Thus, computational intelligence and metaheuristics are applied to find accurate solutions within reasonable execution times [8, 9, 10].

In this line of work, this article presents a nature-inspired computational intelligence methodology for traffic lights optimization. By applying a parallel Evolutionary Algorithm (EA), we are able to address a complex real-world problem, providing accurate solutions for decision-makers and authorities to implement. We aim to provide an efficient and innovative solution, improving the quality of service offered to citizens.

Several authors have addressed the traffic lights planning problem using computational intelligence methods before. However, traffic lights planning proposals in the context of BRTs are scarce in the related literature.

Our research contributes with a traffic planning method that takes into account the point of view of both public transportation users and city administrators. We focus on BRTs, which are relevant scenarios for modern cities, studying the problem of traffic lights synchronization to streamline public transportation. Several features are included in the problem model and also in the field research performed, including: time to board the buses (including time to pay the ticket, with and without smart card), time to alight from the buses, real traffic data gathered in situ, traffic lights phases/offset and traffic rules, etc. Furthermore, we apply a novel methodology that combines an efficient parallel evolutionary optimization technique with microscopic simulations, and study multiobjective variants of the traffic lights synchronization problem that account for different priorities for public transportation (buses) or private transportation (other vehicles). This approach provides the decision maker several options to speed up the travel times and improve the user experience on BRTs. As a case study we apply the optimization approach in a real world scenario, the BRT defined on Garzón Avenue, Montevideo, modeled using real data collected in situ.

The studied BRT scenario poses a complex challenge because it includes an extensive urban area, a large number of crossings and traffic lights, rules for exclusive lanes, and different types of traffic (on Garzón Avenue and crossing streets). We study different options for improving the speed of both buses and vehicles, analyzing trade-off solutions, and a new scenario that accounts for modifications to bus stop locations and traffic lights rules to further improve speed and travel times. The main results demonstrate that the proposed parallel EA is able to improve the average speed of buses and other vehicles when compared with the current scenario. Additional experiments demonstrate that further speed improvements can be achieved when considering different priorities for buses and other vehicles, and new traffic/bus stop settings in the new scenario. Furthermore, we also demonstrate the benefits of using a parallel model for evaluating the different configurations of traffic lights: the parallel version of the proposed EA improves the execution times up to 26.9× when compared with the sequential version. These results have been presented to the public transportation administrators in Montevideo.

The article is organized as follows. Section 2 presents the problem description and the optimization model. The EA proposed to solve the problem is described in Section 3. Section 4 presents the experimental analysis using realistic case studies on Garzón avenue. Section 5 reviews the main related works in the literature. Finally, the conclusions and the main lines of current and future work are discussed in Section 6.

2. Methodology for public transport optimization via traffic lights optimization

This section presents the problem formulation and describes the optimization model using EAs.

2.1. Problem model for traffic lights synchronization in BRT

The problem model simplifies the reality, considering only those features relevant for traffic lights synchronization. A map of the geographical area to study is built including real data collected in situ. Microscopic simulations are applied to evaluate the solutions. The methodology and tools used in the research are described in the following paragraph.

Map. The first step of the modeling process is to design a map of the area of study. For this purpose, the Open Street Map (OSM) service [11] is used to design a map of the studied area, which is compatible with the microscopic simulator used for evaluation. The Java OSM editor is used to correct and adapt the map, keeping only those elements that are relevant to the problem. The validation of the designed map can be assessed by comparing it with data gathered in situ and from other services (Google Maps/Bing Maps).

The map is downloaded from OSM and afterwards the *NetConvert* application is used to include real data for traffic lights collected in situ, as described in the next paragraph.

Field research for gathering real data from traffic lights, buses, and vehicles. The real mobility data for the area of study (e.g., number of vehicles, traffic lights data) may not be freely available. Thus, a field research might be needed to get the real traffic data corresponding to the studied area. For this purpose, the recommendations for vehicle counting proposed by Smith and McIntyre [12] need to be followed to avoid bias: normal traffic should be characterized counting vehicles on a working day, with normal weather, and in representative (non-peak) hours. For the case study presented in this article, a field research was performed to gather information about the traffic density and the traffic patterns in Garzón Avenue and surrounding streets (see details in Section 4.2).

Traffic simulator. Candidate solutions (i.e., traffic-light configurations) are evaluated using SUMO [13], a free open-source traffic simulator that allows modeling streets, vehicles, public transport, and traffic lights. SUMO applies a microscopic model, performing an explicit simulation of each element in a scenario. The simulator is simple to operate: it takes as input a set of configuration files that represents the road network, vehicles, traffic, and traffic lights, and generates output files with useful information from the simulated scenario: simulation time, number and speed of vehicles, travel duration, and other relevant metrics. SUMO also allows including specific features to model BRTs, including bus stops, bus trajectories and frequencies, number of passengers boarding buses, time to board, etc. Initial experiments were performed to analyze and validate the results of BRT scenarios simulated using SUMO, including different traffic lights phases, different traffic patterns, and specific modifications on the scenario. Results showed that the microscopic simulation offered by SUMO is able to accurately model the reality of urban traffic for buses and private vehicles, particularly in the context of BRTs, matching the results obtained in the field research.

2.2. Metaheuristics

Metaheuristics are generic strategies for designing computational methods to find approximate solutions for complex problems (e.g., search, optimization, and learning problems) [14, 15].

In practice, many optimization problems are $\mathcal{NP}$-hard, intrinsically complex, and demand a large amount of computing effort. Many of the problems arising in real-world applications from scientific and technological fields are within this high-complexity class of problems, due to a number of reasons: they have very large search spaces, they include hard constraints that make the search space very sparse, they are multimodal/multiobjective problems taking into account hard-to-evaluate optimization functions, they are time-varying problems including complex mathematical functions, or they manage very large volumes of data.

The problem addressed in this article, i.e., optimizing traffic lights to improve the speed of public transportation in a BRT, is an instance of a $\mathcal{NP}$-hard problem. In this context, metaheuristics provide efficient and accurate methods for solving realistic instances of the problem, which cannot be solved using classical exact resolution methods for optimization (e.g., enumerative search, backtracking/branch and bound, dynamic programming) which are extremely time-consuming.

2.3. Evolutionary algorithms

EAs are non-deterministic metaheuristic methods that emulate the evolution of species in nature to solve optimization, search, and learning problems [16]. In the past thirty years, EAs have been applied to solve many highly complex optimization problems. Algorithm 1 presents a pseudocode of a generic EA.

Algorithm 1: Pseudocode of an EA.

1 t $\leftarrow$ 0

2 initialize($P(t)$)

3 evaluate($P(t)$)

4 **while** *not stop_condition* **do**

5 $\quad P'(t) \leftarrow selection(P(t))$

6 $\quad P''(t) \leftarrow recombination(P'(t))$ {according to p_R}

7 $\quad P'''(t) \leftarrow mutation(P''(t))$ {according to p_M}

8 $\quad$ evaluate($P'''(t)$)

9 $\quad P(t) \leftarrow replacement(P'''(t), P(t))$

10 $\quad t \leftarrow t + 1$

11 **end**

12 **return** best individual found

EAs are iterative methods that apply stochastic operators on a set of *individuals* (the *population*). Each individual in the population encodes a candidate solution for the optimization problem. The initial population is generated by applying a random procedure or by using a specific heuristic for the problem (line 2 in Algorithm 1).

A *fitness* value is assigned to every individual by the evaluation function (line 3), indicating how good the corresponding solution is at solving the problem. The search is guided by a probabilistic selection-of-the-best technique (for

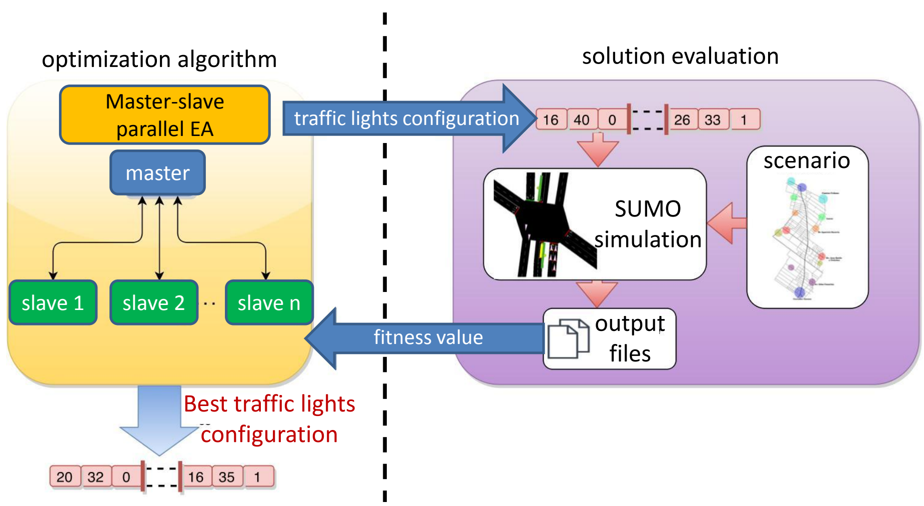

Figure 1: Diagram of the optimization model

both parents and offspring) towards tentative solutions of higher quality (line 5). Iteratively, new solutions are built during the search by applying probabilistic *variation operators*, including mixing parts of two individuals (*recombination*, line 6) or performing random changes in the individual (the *mutation* operator, in line 7). Specific policies are used to select the groups of individuals to recombine and to determine which new individuals are inserted in the population in each new generation (the criterion used by the *replacement* function, in line 9).

The stop condition usually involves a fixed number of generations or fixed execution time, a quality threshold on the best fitness value, or the detection of a stagnation situation. The EA returns the best solution found in the iterative process, taking into account the fitness function (line 12).

Parallel models for EAs have been proposed to accelerate the computing time required for the search, especially when dealing with complex objective functions or hard search spaces [17]. In this work, we apply a master-slave model for parallelization, in order to reduce the execution time of performing the traffic simulations for the studied scenario.

As suggested in related works, simple EAs such as basic genetic algorithms [18] are not powerful enough to find the best traffic lights configuration efficiently, mainly because the search space is intrinsically complex. Ad-hoc operators are needed to properly explore the search space and avoid getting stuck in local optima. Furthermore, a parallel model is needed to overcome efficiency issues when dealing with large real scenarios via simulations. Thus, we propose applying a custom EA implemented in C++, using the skeleton available in the Malva library for optimization [19]. We performed specific modifications of the Malva code in order to implement the parallel model for fitness evaluation using multiple threads, suitable for execution in modern multi-core computers. The main features of the proposed metaheuristic technique are described next.

2.4. Optimization model

The optimization model applied in this work is described in the following paragraphs.

Optimization criteria. The mathematical model for optimization 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 $f = w_B \times \overline{s_B} + w_O \times \overline{s_O}$, used for solution evaluation in the proposed EA ($0 \leq w_B, w_O \leq 1; w_B + w_O = 1$). This way, we can focus on assigning a higher priority to public transport (buses), by choosing appropriate values for weight w_B.

Optimization using evolutionary algorithms. The optimization process using evolutionary algorithms is described in the diagram on Figure 1. The diagram clearly separates the two main components of the resolution strategy: the optimization algorithm and the procedure using simulations for solution evaluation. The optimization algorithm, i.e., a master-slave parallel EA, performs the search of the best traffic lights configuration, considering the optimization criteria defined in the previous subsection. In turn, the simulation procedure is used to evaluate the speed objectives s_B and s_O for each solution considered by the parallel EA. Simulations are performed using SUMO, according to the problem model defined in Section 2.1, and using the specific problem features and real data (map, traffic patterns and volume, etc.). The interface between these two modules is via the communication of solutions and fitness values. Each slave process in the parallel EA sends solutions (i.e., traffic lights configurations) to the evaluation module, which performs the corresponding simulation and returns the fitness value taking into account the two evaluated speeds.

The conceptual separation between the problem and the optimization objective is twofold. On the one hand, the clear separation between problem and resolution method allows applying a modular design for the optimization software. This way, from the point of view of the software design, it is easy to incorporate new algorithms (e.g., heuristics, other metaheuristics, or ad-hoc methods) to solve the problem. On the other hand, the modular design allows applying the proposed optimization approach and the metaheuristic method to solve the traffic lights synchronization problem over different scenarios, by incorporating maps, traffic data, and traffic-lights location for a given area.

The traffic speed optimization is performed not only on the BRT, but over a portion of the road network including several surrounding streets. This global optimization approach is crucial to achieve a traffic lights configuration that guarantees a sustainable improvement on the mobility patterns. This improvement cannot be assured if the problem model considers only some streets or optimizes each intersection separately.

3. A parallel evolutionary algorithm for traffic lights synchronization

This section describes the proposed parallel EA for traffic lights synchronization.

3.1. The proposed implementation

Solution encoding. The proposed encoding includes the elements needed for traffic lights 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. In the proposed encoding, all values for traffic lights phases and offsets are natural and expressed in seconds. Offsets are in the range $[0, 60]$.

Figure 2 graphically explains the concept of traffic lights phases. Crossings are classified according the number of phases for traffic lights operation. For instance, in a crossing with two phases (Crossing 1 in the figure), one of them allows going forward in the main street and the other one allows turning right. In a crossing with three phases (Crossing 2 in the figure), one of them allows going straight in the main street, a second one allows vehicles coming from a specific direction in the secondary street (in the image, from the left) to go straight and turn (right, and left when allowed), and the third phase allows vehicles coming from the opposite direction in the secondary street (in the image, from the right) to go straight and turn.

The solution encoding logically groups this information into crossings, storing the time for each phase. Numeric values correspond to the duration of green

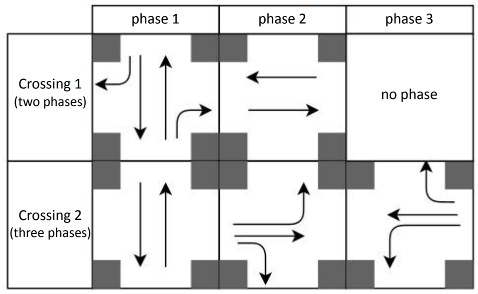

Figure 2: Phases in an intersection

lights, red lights, and the offset of initiation. Every traffic light starts on its first phase. Amber lights are omitted as they do not affect the times of passing vehicles; they are assumed to last for four seconds, as specified by international standards. Figure 3 presents an example of the solution encoding for the traffic lights in Garzón Avenue, the case study used for the experimental evaluation.

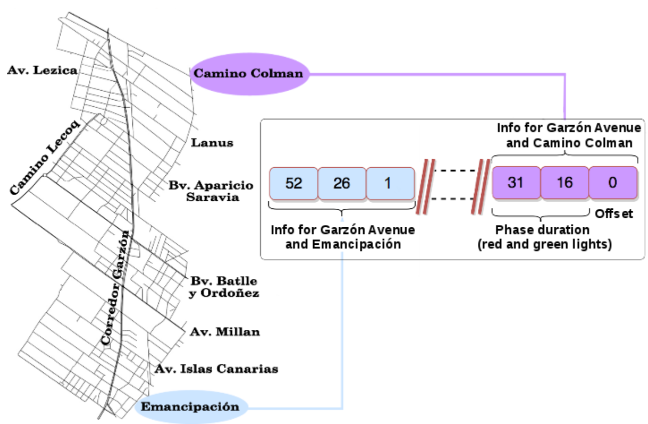

Figure 3: Example of the solution encoding applied in the parallel EA.

The length of an encoded solution depends on the number of crossings and the number of phases defined to optimize in the scenario. Using this encoding allows optimizing the complete scenario: all intersections are optimized simultaneously, unlike some proposals in the related works where intersections are configured and optimized separately.

Fitness function. The fitness function, defined in Section 2, accounts for the optimization of the average speed of buses and other vehicles over the defined scenario. Several weight combinations are used to explore different priorities between buses and other vehicles, according to suggestions provided by city administrators.

Population initialization. A set of initial solutions for the problem is built using the data collected from the current (non-optimized) reality on the area of study, following similar approaches from the related literature [10, 20]. Small perturbations are applied to provide diversity to the initial population, including modifying the phase time using a Gaussian distribution with deviation $\sigma^2 = 0.4$, and modifying the phase offset applying a uniform distribution. These settings allow generating an appropriate diversity to start the search.

Recombination. We apply a one point crossover, considering the information of each street crossing as a group, and only taking into account the positions between groups as possible crossover points.

Mutation. Two mutation operators are applied: *i*) Gaussian mutation to modify the values of phases; and *ii*) 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 the standard tournament selection operator, configured to consider three individuals that participate in the tournament, where the best one survives. Regarding the replacement policy, the proposed EA applies the $(\mu + \lambda)$ evolution model, where parents and offspring compete for survival.

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 be evaluated on slave processes, executing in those threads. Slaves perform the simulations to evaluate each traffic lights 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 on Garzón Avenue.

4.1. Computational platform

The analysis was performed on an AMD Opteron 6272 at 2.09GHz (64 cores, 48GB RAM, CentOS Linux 6.5), from Cluster FING, the High Performance Computing facility at Universidad de la República, Uruguay [21].

4.2. Problem instances

The studied area includes the BRT in Garzón Avenue and two alternative paths running on parallel streets and internal roads on both sides of Garzón Avenue. Each alternative path includes two-way streets or two one-way streets to guarantee connectivity. Figure 4 presents the studied area. Crossings in the studied scenario have traffic lights with two and three phases.

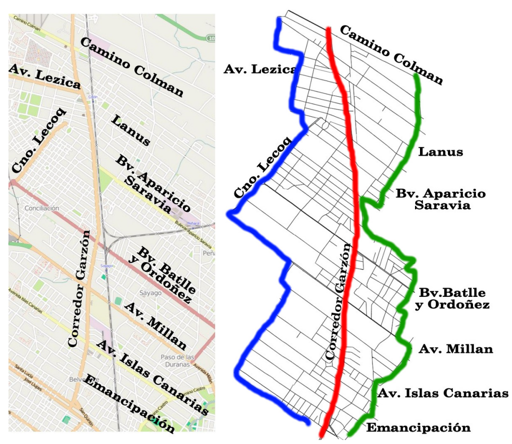

Figure 4: Left: OSM map. Right: processed version compatible with SUMO (Garzón Avenue in red, alternative parallel paths in blue and green).

The real mobility data for Garzón Avenue is not available from the local government of Montevideo. Thus, a field research was needed to get the real traffic data. The field research included many activities devoted to gather information about the traffic density and the traffic patterns in Garzón Avenue and surrounding streets. Several static measurements were performed in situ using manual counting methods and also automatic counting using video cameras to determine the traffic density. In addition, dynamic counts were performed in the studied area (especially the BRT in Garzón Avenue), traveling in public transportation (bus) and private transportation (car) to evaluate the travel times using each vehicle type. The number of vehicles and several other relevant traffic data were gathered in different days an different hours.

Table 1 summarizes the number of vehicles counted in the field research on five representative intersections of the studied area.

Table 1: Summary of the traffic data gathered in the field research in the five major intersections in Garzón area.

intersection	*traffic direction*			
	Garzón South	*Garzón North*	*West*	*East*
Camino Colman	410	390	140	230
Plaza Vidiella	400	444	292	0
Aparicio Saravia	390	450	40	90
Batlle y Ordoez	510	390	470	300
Camino Ariel	436	226	177	203

A *baseline scenario* was built using the real data and the actual configuration of traffic lights on Garzón Avenue and surrounding streets. The baseline scenario is used as a reference to compare the results computed by the proposed parallel EA for traffic lights planning.

Three XML files are used in the SUMO simulation: *i*) *traffic lights configuration*, defining the geographical location, phases and offsets; *ii*) *vehicle routes*, built using real data and the *Traffic Modeler* software [22]; and *iii*) *public transport details*, including paths, frequencies, stop locations, and delay times in each stop. We decided to use a between-areas mobility model in the problem scenario, which provides an appropriate granularity to define the traffic density. We collected data from all urban bus lines in the zone (*G*, *D5*, *2*, *148*, and *409*). We also analyzed one month of GPS data (position/speed) from buses to determine mobility patterns and average speed on Garzón Avenue (14.5 km/h). We evaluated the average times for phases and the offsets of the current traffic lights configuration. Finally, we studied videos from cameras in the zone to compute the average delays due to the times that passengers need to board and alight from the buses (validate card, pay ticket in cash, etc.), which are between 20 and 35 seconds, depending on the bus stop. All these data were included in the proposed simulation model.

Three traffic patterns are studied: *i*) *normal traffic*, with the main bulk of data from the field research (e.g., working day, sunny weather, non-peak hour), including 2000 vehicles and 70 buses; *ii*) *low traffic*, using data collected during weekends and night hours, with 1000 vehicles and 70 buses, and significantly shorter delays on the bus stops because fewer people use public transportation; and *iii*) *high traffic* using data from rush hours, including 3000 vehicles and 70 buses. Bus frequencies change according to the city schedule and are not affected by the traffic density. All data were contrasted and verified with the information provided by the city administration.

Our main goal is to advance in designing a methodology to be used operationally (as close as possible to real time) over different traffic patterns.

4.3. Parameters setting

EAs are stochastic methods, so a parameter setting analysis is needed to find the configuration that allows computing the best results. We studied the values for population size, stopping criterion, recombination probability (p_R), and mutation probability (p_M) in the parallel EA. We also studied the simulation time in SUMO for the proposed scenarios.

In order to avoid bias in the results, 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 in the parameter setting experiments. The main results are summarized next.

Simulation time. The best results were obtained using 4000 simulation steps, which represent 66 minutes in the real scenario. Using this simulation time, more than 85% of the vehicles are able to reach destination. The execution time to perform each simulation is between 10 and 30 seconds, depending on the details and features of the scenario.

Stopping criterion. A specific goal of the optimization is to achieve a trade-off between solution quality and execution time. Results showed that the best fitness values did not vary significantly after performing 400 generations. Thus, we decided to use a limit of 500 generations as stopping criteria. Using this limit, the parallel EA demanded between 1 and 24 hours of execution time.

Population size. We considered the quality of results, the execution time, and the computing elements available in the platform, to find the best population size in the proposed EA. We analyzed using 32, 48, and 64 individuals in the population. The results indicated that no significant improvements are achieved in the fitness values when using larger populations, so we decided to use 32 individuals, in order to have the shortest execution times. Table 2 presents an example of the results obtained in the population size analysis.

Table 2: Population size analysis

#P	*fitness*		*execution time (m)*
	best	*average* $\pm\sigma$	
32	17.18	16.36±0.48	80.8±6.7
48	16.69	15.84±0.32	112.8±5.5
64	17.27	16.37±0.60	169.7±8.0

Operator probabilities. We explored all the combinations of the following candidate values: $p_R \in \{0.5, 0.8, 1\}$, and $p_M \in \{0.01, 0.05, 0.1\}$. We performed a statistical analysis of the results applying the Student's t-test, and concluded that the best results are computed when using ($p_R = 0.5$, $p_M = 0.1$) and ($p_R = 0.5$, $p_M = 0.01$). Finally, we decided to choose the parameter configuration ($p_R = 0.5$, $p_M = 0.01$), which provides the fastest execution times. Table 3 presents an example of the results for the analysis of the operator probabilities.

Table 3: Operator probability analysis

p_R	p_M	*average fitness*$\pm\sigma$
0.5	0.01	16.09±0.30
0.5	0.05	15.60±0.27
0.5	0.1	**16.36±0.22**
0.8	0.01	16.04±0.45
0.8	0.05	15.82±0.32
0.8	0.1	16.12±0.34
1	0.01	16.08±0.25
1	0.05	15.83±0.34
1	0.1	16.04±0.25

4.4. 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 main results are summarized and discussed in the following paragraphs.

Simulations of the baseline scenario. We performed a set of simulations over the current scenario in order to obtain the baseline results for the comparison. Table 4 presents the numerical results for the baseline scenario, reporting the average speed for buses and the average speed for vehicles (in km/h), as well as the corresponding fitness value according to the linear aggregation function used for evaluation. The simulations confirmed that the results for average speed and time travel matches those computed when processing the GPS data from the city authorities; thus validating the proposed approach using simulations.

Table 4: Simulations of the baseline scenario.

traffic	$\overline{s_B}$	$\overline{s_O}$	*fitness*
low	15.9	32.5	13.4
medium	14.6	28.8	12.1
high	14.3	26.4	11.3

Results of the proposed parallel EA. Table 5 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.

Table 5: Numerical results of the proposed parallel EA on the base scenario, for different traffic patterns.

traffic	*baseline scenario*			*parallel EA results*					
						fitness		*fitness improv.*	
	$\overline{s_B}$	$\overline{s_O}$	*fitness*	$\overline{s_B}$	$\overline{s_O}$	average$\pm\sigma$	best	average	best
low	15.89	32.45	13.42	17.92$\pm$0.18	34.30$\pm$0.40	14.50$\pm$0.14	14.88	8.0 %	10.8 %
medium	14.59	28.81	12.00	16.95$\pm$0.32	33.29$\pm$0.29	13.95$\pm$0.15	14.19	15.7 %	17.7 %
high	14.31	26.36	11.30	16.51$\pm$0.61	32.90$\pm$0.25	13.72$\pm$0.17	14.04	21.4 %	24.2 %

Results in Table 5 indicate that the parallel EA allows improving the average speed for the three traffic patterns studied. Speed improvements are up to **24.2%** (in fitness values), up to **15.3%** (in average bus speed), and up to **24.8%** (in average speed of other vehicles). 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 (with a confidence level of 99%).

Analysis of travel times on Garzón Avenue. We also evaluated the travel times for buses and other vehicles on Garzón Avenue (6.5 km). The comparison between the optimized traffic lights configuration and the baseline scenario is summarized in Figure 5 (for buses) and in Figure 6 (for other vehicles).

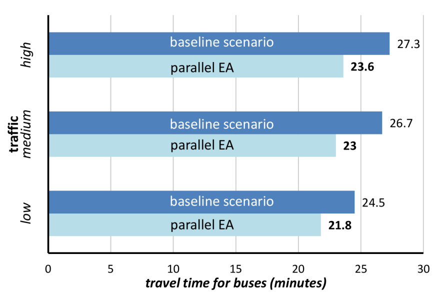

Figure 5: Travel times for buses: optimized traffic lights configuration (parallel EA) vs. baseline scenario.

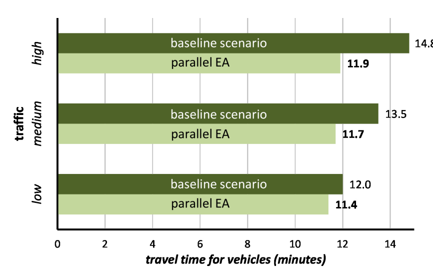

Figure 6: Travel times for other vehicles: optimized traffic lights configuration (parallel EA) vs. baseline scenario

According to the results reported in Figure 5, the optimization using the parallel EA allowed reducing the travel times for buses on Garzón Avenue from 27.3 to 23.6 minutes in the high traffic scenario. Similar results were obtained for the other traffic patterns. Vehicles also moved faster when considering the optimized traffic lights configuration: the travel times have a significant improvement from 14.8 minutes to 11.9 minutes in the high traffic scenario, and similar improvements for the other traffic patterns.

Multiobjective optimization analysis. Table 6 reports the results computed by the proposed parallel EA when using different weights to prioritize the speed of buses or vehicles. The studied weights were defined according to suggestions provided by both bus operators and city administrators. These weights allow modeling different priorities for buses and other vehicles in the BRT, which can be implemented in practice.

Table 6: Numerical results when using different weights in the fitness function of the parallel EA, for different traffic patterns.

traffic	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	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	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	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%

The comparative results indicate that choosing different weights has a rather significant influence on the optimization results. An additional 2% of improvement in the speed of buses can be achieved when optimizing with the proposed parallel EA for the combination $w_B = 0.7$, $w_O = 0.3$. This improvement comes with a negligible reduction on the speed of other vehicles (results in bold font). This result is statistically significant according to the Kruskal-Wallis test used to analyze the results distributions (with confidence level 99%).

4.5. Optimization in a modified scenario.

We also performed an experimental evaluation of the proposed traffic lights synchronization using EAs in a modified scenario. The new urban scenario is built considering slight modifications on the locations of the bus stops and an improved traffic lights management. The main details about the modified scenario and the experimental evaluation are reported next.

4.5.1. The modified scenario for the BRT on Garzón avenue

The main characteristics of the modified scenario are described next.

Alternate bus stops. One of the main problems related to BRTs in general is that, due to their slow acceleration, buses demand a significant time to reach an acceptable speed after stopping in traffic lights or bus stops. This is a specific inconvenience that arises in the BRT on Garzón Avenue, where bus stops are located near each other, and all bus stops are shared by all bus lines. Thus, in addition to optimizing the traffic lights configurations, in the modified scenario we consider alternating bus stops for line 'G'. Line G is one of the main bus lines traveling across the BRT in Garzón Avenue, and it is operated by two bus companies: CUTCSA and COETC. We propose a modified scenario alternating bus stops for buses of different companies. As both companies operate the same line, the modification will have a minimal impact on the quality of service for users. If needed, additional optimization of bus timetabling can be performed to reduce the average waiting times for passengers in each bus stop.

Figure 7 presents a description of the bus stop changes performed in the modified scenario: the current stops are marked with blue circles. We propose eliminating bus stop 'Casavalle' (marked in grey) in the original path of line 'G' and alternating every other stop. The resulting new paths for buses consider odd bus stops for COETC (marked with red circles) and even bus stops for CUTCSA (marked with green circles). The base map/figure for the Garzón BRT in Figure 7 is from Intendencia de Montevideo [6].

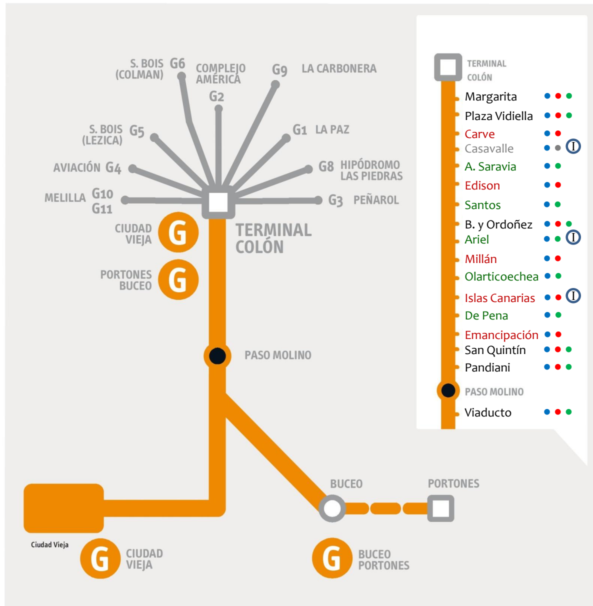

Figure 7: Description of the modified scenario. Blue circles: original bus stops for line 'G'; grey: dropped bus stop ('Casavalle'); red circles: line 'G'–COETC; green circles: line 'G'–CUTCSA. 'I' stands for the three junctions where the improved traffic lights management was implemented. The base map for the Garzón BRT is from Intendencia de Montevideo.

Improved traffic lights management. During the field research we observed that in some intersections where a bus line traveling through Garzón turns left, the current rules for traffic lights force the vehicles traveling on the right lane to stop, while vehicles in the central lane has green light to advance. According to a personal communication from administrators from Intendencia de Montevideo, this rule is applied to make the traffic control easier, because it allows the simultaneous operation of traffic lights for the lanes on both sides (right and left) of the central lane, reserved for buses. Allowing a separate operation of these two lanes improves the speed of vehicles circulating on both ways of Garzón Avenue. In the modified scenario, this modification was implemented and evaluated in three intersections: *i*) Garzón and Islas Canarias, where line 409 turns left, in direction to Colón (North); *ii*) Garzón and Camino Ariel, where lines 2 and 148 turn left, in direction to Paso Molino (South); *iii*) Garzón and Casavalle, where line 174 turns left, in direction to Paso Molino (South).

4.5.2. Experimental results for the modified scenario

The main results of the studied traffic metrics for vehicles and buses are reported in Table 7. We analyze the average speed for vehicles and buses, the average and best fitness values obtained using the proposed parallel EA and the improvements of the results computed by the parallel EA over the baseline scenario (including the optimization), as reported in Section 4.4.

Table 7: Numerical results for the parallel EA on the modified scenario.

traffic	$\overline{s_B}$	$\overline{s_O}$	*fitness*		*fitness improv.*	
			average$\pm\sigma$	*best*	*average*	*best*
low	23.15±0.36	34.43±0.33	15.99±0.08	16.10	19.1%	19.9%
medium	21.83±0.50	33.89±0.22	15.47±0.09	15.65	28.3%	29.8%
high	21.46±0.54	33.41±0.38	15.24±0.19	15.50	34.8%	37.1%

Results in Table 7 indicate that the parallel EA computes traffic lights configurations that account for accurate speed values for both buses and other vehicles in the modified scenario. The average speed for buses is over 21 km/h, and a maximum value of 23.15 km/h is obtained for the instance with low traffic. Regarding the speed for other vehicles, the values are between 33 km/h and 34.5 km/h in all scenarios. The improvements on the fitness values are between 19.9% and 37.1%, when compared to the baseline scenario. Furthermore, the best improvements are obtained for the high traffic scenario, indicating that the proposed strategy is useful to speed up vehicle flow and avoid traffic jams and congestions in the studied BRT in peak hours and under high traffic density.

Figure 8 graphically reports the time (in minutes) needed for buses and other vehicles to travel along Garzón Avenue (total length 6.5 km). The travel times achieved by the parallel EA on the modified scenario are compared against the baseline scenario. A significant reduction in travel times for buses can be noticed for all traffic patterns. For other vehicles, the best improvement over the baseline scenario is achieved in the high traffic scenario. The study of the results distribution applying the Kruskal-Wallis test indicated that the observed improvements against the baseline scenario are statistically significant.

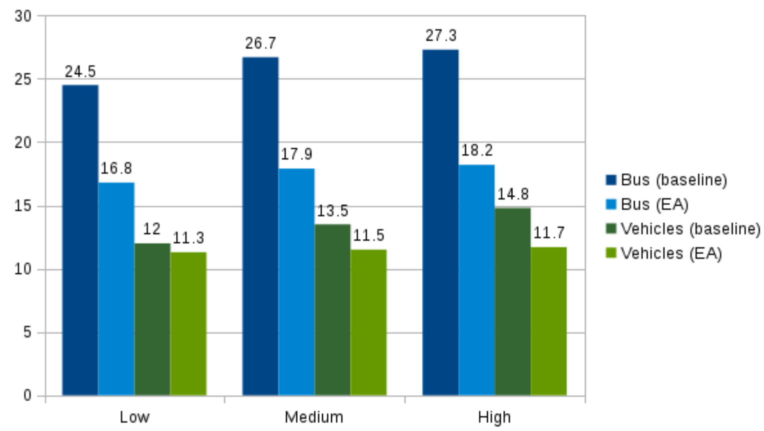

Figure 8: Comparison of travel times (in minutes) between baseline scenario and the solution computed by the EA in the modified scenario.

4.6. Computational efficiency analysis

We studied the execution time improvements when applying the master-slave parallel model in the proposed EA. We evaluated two relevant metrics for performance improvement: *speedup* and *computational efficiency* [23]. The speedup metric evaluates how much faster the parallel EA is when compared to the sequential implementation. It is defined as the ratio of the execution time of the sequential algorithm (T_1) and the parallel version executed on m computing elements (T_m) (Equation 1). The computational efficiency is the normalized value of the speedup. It is the result of dividing the speedup by the number of computing resources (Equation 2).

$$S_m = \frac{T_1}{T_m} \qquad (1) \qquad\qquad e_m = \frac{S_m}{m} \qquad (2)$$

Figure 9 reports the execution time analysis for a set of representative scenarios used in the experimental evaluation. The execution time of the sequential EA (T_1) is compared with the execution time when using 32 computing elements (T_{32}, where one computing element is used for each solution to evaluate). All times are reported in minutes. The comparison between speedup and computational efficiency values for the parallel and sequential version of the proposed EA is reported in the graphic on the right.

#	*traffic*	T_1	T_{32}	*speedup*	*efficiency*
1		1572	59	26.6	0.83
2	*low*	1571	59	26.6	0.83
3		1183	44	**26.9**	**0.84**
4		3002	119	25.2	0.79
5	*normal*	2195	82	26.8	0.84
6		3007	120	25.1	0.78
7		2920	110	26.6	0.83
8	*high*	4365	183	23.9	0.75
9		4276	177	24.2	0.75
		average±σ		25.7±1.1	0.80±0.03

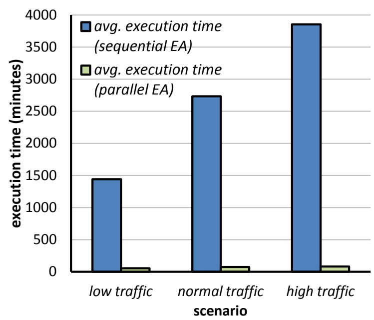

Figure 9: Execution time analysis of the proposed parallel EA (minutes).

From the results presented in Figure 9, we conclude that the parallel EA is **26.9** times faster when using 32 computing resources. The parallel EA allows executing in 44 minutes the optimization that requires 20 hours of execution time when using the sequential version. A sublinear speedup behavior is observed, but computational efficiency is 0.84, very close to the ideal value of 1.

The efficiency results demonstrate that the proposed methodology can be applied to compute traffic lights planning based on real data to be used operationally (for example, to compute a planning to be applied in the next hour).

5. Related works

Computational intelligence has been applied to traffic lights planning, as this is a complex nonlinear stochastic problem and exact algorithms cannot compute solutions efficiently [24]. Furthermore, exact methods require mathematical models to model the traffic dynamics, which are hard to build. Thus, combining computational intelligence and simulations provides a robust methodology to handle stochastic events, uncertainty, and dynamic environments [25].

Early related works focused on small problem scenarios [26]. Adaptive methods [27] and vehicular networks [28, 29] are useful in real urban areas but usually demand large infrastructure investments (vehicles and roadside) to guarantee on-line information exchange.

Another popular strategy for traffic lights planning is green wave [30], which coordinates traffic lights in the same street to achieve continuous traffic in one direction. The main problem of this methodology is the limited urban topologies where it can be applied.

Regarding BRTs, a few recent proposals applied bioinspired computing for traffic lights planning. López et al. [31] proposed a multiagent simulation model for Transmillenio BRT in Bogotá, Colombia, using Petri nets to describe the dynamic of the system (people, road traffic dynamics, bus network operations). The capabilities of the system to fulfill mobility demands in rush hours minimizing the number of buses was studied. The main finding is that there is an optimal number of buses to attend the demand. Beyond that number, using more buses does not reduce the traveling times of passengers. This can lead to a more efficient system from the point of of view of environmental protection, resources utilization, etc.

Zhou et al. [32] proposed a real-time signal priority control algorithm for single intersections based on vehicles communicating with signal controllers on BRTs. Buses location and speed are sent in real-time to roadside units and the algorithm computes the estimated arrival time of buses to each intersection and the timetable deviations. This information is used to implement signal priority at intersections for delayed buses, to improve quality of service. Eight different strategies are proposed according to the traffic lights phase when the bus arrives. A BRT in Jinan China with simulated traffic data was studied. Results indicated that average passenger delay can decrease up to 25.3% and speed of BRT vehicles can be improved in up to 7.6%.

Closer to our research, Sánchez et al. [10] applied EA for traffic lights synchronization to improve traffic flow in Santa Cruz de Tenerife, Spain. The road network has 42 traffic lights, 26 input roads, and 20 output roads. Nine hand-made solutions from traffic administrators are used as initial population and a two-point crossover is applied to explore the search space. The fitness function evaluates the travel time for vehicles in the simulated road network. Results from the experimental evaluation indicate that the EA was able to improve up to 26% the trip times over the baseline solutions, but no details about the benefits for public transportation are reported.

Olivera et al. [20] applied Particle Swarm Optimization for traffic lights planning and reducing pollution in Seville and Málaga, Spain. Objectives are integrated in a single objective function applying a linear aggregation approach. Results are compared with Differential Evolution over two scenarios of 0.75 km^2. Results show significant improvements in fuel consumption, time delay, and pollutant emissions. The obtained traffic lights configurations reduce CO and NO_x concentrations by 25%. Improvements on fuel consumption reached 18.2%. However, the single objective approach does not model a global vision of the traffic network: solutions with traffic jams are wrongly considered as "good" solutions, because vehicles that do not move produce low emissions and have minimal fuel consumption.

The analysis of related work indicates that computational intelligence has been applied to solve traffic lights planning problems. However, specific solutions for BRTs are scarce. Our research proposes applying to BRTs a model that considers several features previously used for traffic lights planning in generic urban scenarios. As case study, the methodology is applied to Garzón BRT. This scenario is larger than the ones studied in most related works: it includes 6.5 km and a total area of more than 30 km^2. Several distinctive features are also included: a significantly larger number of intersections, all 28 bus stops in the zone, real traffic data collected in situ, and specific mobility logic due to the BRT regulations (exclusive lines, priorities, and allowed/forbidden turning corners).

6. Conclusions and future work

This article presented a parallel EA for traffic lights synchronization to optimize public transport in BRTs.

The proposed solution takes into account several complex features of a real urban zone including real maps and real mobility data. The devised methodology includes analysis of GPS information, traffic modeling, simulation, and computational intelligence for optimization. A real scenario is presented as a case study: the BRT on Garzon Avenue in Montevideo, Uruguay. This is an innovative approach in Uruguay, where urban intelligent systems have not been applied to public transport until now.

The experimental analysis compared the results computed using the proposed parallel EA against a baseline scenario that models the current reality. Results show that the parallel EA allows computing traffic lights plannings that provide a better quality of service than the current reality. The optimized traffic lights configuration allows improving up to **15.3%** the average bus speed and **24.8%** the average speed of other vehicles. An additional improvement of 2% in the speed of buses is achieved when assigning a higher priority to the first objective.

Besides optimizing traffic lights configurations, we proposed specific modifications to the current reality in Garzón Avenue to improve travel times, by defining an alternative scenario that alternates bus stops and performs minor changes to traffic lights rules. Under this modified scenario, the experimental results show that the proposed EA is able to reduce travel times for buses from 27.3 to 18.2 minutes and from 14.8 to 11.7 minutes for other vehicles.

The master-slave parallel model was effective in reducing the execution times needed to compute the traffic lights configurations, achieving speedup values of up to **26.9** when using 32 cores. This model allows reducing from 20 hours to 44 minutes the execution time, when compared against a sequential version of the algorithm.

Results show that the proposed optimization approach is useful to help authorities with long-term urban planning that has significant impact in citizens mobility. Software simulation results must be tested before applying the proposed approach in real scenarios. Our validation results suggest that real improvements on traffic flow and speed can be obtained indeed. Furthermore, the proposed approach can be applied to optimize other urban scenarios and different problem variants.

The main lines for future work are related to improve the proposed approach by considering different problem objectives and an explicit multiobjective optimization method. In addition, we also plan to apply the proposed methodology for traffic optimization in other urban scenarios.

Acknowledgment

The research was partly supported by Science and Innovation Fund (Uruguay), Foreign & Commonwealth Office, UK. The work of R. Massobrio and S. Nesmachnow was partly funded by ANII and PEDECIBA, Uruguay. C. Olivera and P. Vidal thank CONICET for grant PCB-I CONICET-ANII and ANPCyT for grant PICT 2014-0430.

References

[1] A. Bull, Congestión de tránsito: el problema y cómo enfrentarlo, no. 87 in Cuadernos de la CEPAL, United Nations Publications, 2003.

[2] J. Fernández, S. Guzmán, M. Collado, V. Fernández, How to incorporate urban complexity, diversity and intelligence into smart cities initiatives, in: Smart Cities: First International Conference, 2016, pp. 85–94.

[3] J. Sussman, Perspectives on Intelligent Transportation Systems (ITS), Springer Science + Business Media, 2005.

[4] E. Bañobre, A. Romero, Los BRT en corredores segregados como sistema óptimo de transporte urbano, in: F. Cossío Silva (Ed.), Administrando en entornos inciertos, Escuela Superior de Gestión Comercial y Marketing, Spain, 2009.

[5] L. Wright, W. Hook, Bus rapid transit planning guide, Institute for Transportation & Development Policy New York, 2007.

[6] Intendencia de Montevideo, Plan de movilidad urbana, http://www.montevideo.gub.uy, June 2017.

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

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

[9] N. Rouphail, B. Park, J. Sacks, Direct signal timing optimization: strategy development and results, in: XI Pan American Conference in Traffic and Transportation Engineering, 2000, pp. 195–206.

[10] J. Sánchez, M. Galán, E. Rubio, Applying a traffic lights evolutionary optimization to a real case: Las Ramblas area in Santa Cruz de Tenerife, IEEE Transactions on Evolutionary Computation 12 (1) (2008) 25–40.

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

[12] D. Smith, J. McIntyre, Handbook of simplified practice for traffic studies, Tech. Rep. 455, Iowa State University (2002).

[13] M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz, SUMO–Simulation of Urban MObility: An Overview, in: 3^{rd} International Conference on Advances in System Simulation, 2011, pp. 63–68.

[14] F. Glover, Future paths for integer programming and links to artificial intelligence, Computers & Operations Research 13 (5) (1986) 533–549.

[15] S. Nesmachnow, An overview of metaheuristics: accurate and efficient methods for optimization, International Journal of Metaheuristics 3 (4) (2015) 320–347.

[16] T. Bäck, D. Fogel, Z. Michalewicz, Handbook of evolutionary computation, Oxford University, 1997.

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

[18] D. Goldberg, Genetic Algorithms in search, optimization, and machine learning, Addison-Wesley, 1989.

[19] G. Fagúndez, R. Massobrio, The Malva Project: A framework for computational intelligence in C++, http://themalvaproject.github.io, July 2017.

[20] A. Olivera, J. García-Nieto, E. Alba, Reducing vehicle emissions and fuel consumption in the city by using particle swarm optimization, Applied Intelligence 42 (3) (2015) 389–405.

[21] S. Nesmachnow, Computación científica de alto desempeño en la Facultad de Ingeniería, Universidad de la República, Revista de la Asociación de Ingenieros del Uruguay 61 (2010) 12–15.

[22] L. Papaleondiou, M. Dikaiakos, Trafficmodeler: A graphical tool for programming microscopic traffic simulators through high-level abstractions, in: 69^{th} Vehicular Technology Conference, 2009, pp. 1–5.

[23] I. Foster, Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering, Addison-Wesley, Boston, MA, USA, 1995.

[24] S. Araghi, A. Khosravi, D. Creighton, A review on computational intelligence methods for controlling traffic signal timing, Expert Systems with Applications 42 (3) (2015) 1538–1550.

[25] D. Zhao, Y. Dai, Z. Zhang, Computational intelligence in urban traffic signal control: A survey, IEEE Transactions on Systems, Man, and Cybernetics, Part C 42 (4) (2012) 485–494.

[26] L. Peng, M. Wang, J. Du, G. Luo, Isolation niches particle swarm optimization applied to traffic lights controlling, in: 48^{th} IEEE Conference on Decision and Control, 2009, pp. 3318–3322.

[27] J. Chen, L. Xu, Road-junction traffic signal timing optimization by an adaptive particle swarm algorithm, in: 9^{th} International Conference on Control, Automation, Robotics and Vision, 2006, pp. 1–7.

[28] R. Massobrio, S. Nesmachnow, J. Toutouh, E. Alba, Infrastructure deployment in vehicular communication networks using a parallel multiobjective evolutionary algorithm, International Journal of Intelligent Systems 32 (8) (2017) 801–829.

[29] R. Massobrio, S. Nesmachnow, J. Toutouh, Multiobjective evolutionary algorithms for smart placement of roadside units in vehicular networks, in: N. Nedjah, L. D. M. Mourelle, H. S. Lopes (Eds.), Evolutionary Multi-Objective System Design: Theory and Applications, Chapman & Hall/CRC Computer and Information Science Series, 2017, pp. 1–36.

[30] X. Wu, S. Deng, X. Du, J. Ma, Green-wave traffic theory optimization and analysis, World Journal of Engineering and Technology 2 (3) (2014) 14.

[31] D. Lopez, A. Triana, H. Chamorro, Simulation model of public transportation system using multiagent approach by means of petri nets: Bogotá study case, in: IX Latin American Robotics Symposium and IEEE Colombian Conference on Automatic Control and Industry Applications, 2011, pp. 1–6.

[32] L. Zhou, Y. Wang, Y. Liu, Active signal priority control method for bus rapid transit based on vehicle infrastructure integration, International Journal of Transportation Science and TechnologyIn press, available online 16 June 2017.