PRELIMINARY METAHEURISTIC SCHEME FOR SOLVING THE SIGNAL CONFIGURATION OPTIMIZATION PROBLEM

Ariadna Claudia Moreno Román*, Mailyn Moreno Espino**†, Cynthia Porras Nodarse***, Yahima Hadfeg Fernández****

*Escuela Superior de Ingeniería Mecánica y Eléctrica, Instituto Politécnico Nacional, México.

**Centro de Investigación en Computación, Instituto Politécnico Nacional, México.

***Universidad Tecnológica de La Habana "José Antonio Echeverría", CUJAE, Cuba.

****Departamento de Administración, Universidad Católica del Norte, Chile.

ABSTRACT

Traffic planning and management is an important task for governments to achieve a proper functioning of the cities. The configuration of road signs and traffic lights is part of this process. The selection of an appropriate configuration is modeled as a variant of the traffic light cycle optimization problem (TLCOP), which is NP-Hard. To evaluate the impact of each configuration, agent-based simulations can be performed. Based on the importance and complexity of the problem, the objective of this paper is to present and evaluate a metaheuristic scheme to solve it and analyze solutions obtained. The algorithms were applied to six zones of Havana, Cuba. Among all the algorithms used, the genetic algorithm obtained the best performance.

KEYWORDS: metaheuristics, road signs, agent-based simulation, traffic lights, optimization

MSC: 68T20, 76A30, 90B20

RESUMEN

La planificación y gestión del tráfico es una importante tarea para los gobiernos en aras de lograr un correcto funcionamiento de las ciudades. Como parte de este proceso se encuentra la configuración de las señales alternativas y las luces de los semáforos. La selección de la configuración adecuada se modela como una variante del problema de optimización del ciclo del semáforo (TLCOP), el cual es NP-Duro. Para evaluar el impacto de cada configuración se puede utilizar la simulación basada en agentes. A partir de la importancia y complejidad del problema, el objetivo de este trabajo es presentar y evaluar un esquema metaheurístico para solucionarlo. Los algoritmos fueron aplicados a seis zonas de La Habana, Cuba. De los algoritmos aplicados, el Algoritmo Genético obtuvo el mejor rendimiento.

PALABRAS CLAVE: metaheurísticas, semáforo, señales alternativas, simulación basada en agentes, optimización

1. INTRODUCTION

Traffic light configuration is currently a complex task in traffic planning and management. This is known as the selection of the duration times that make up its light cycle (green, yellow and red). To

†Corresponding author: mailyn.moreno.espino@gmail.com

this problem add the decision of which vertical road signs to place, such as “Stop” and “Yield” signs, which are the most common [26]. Additionally, it should be seen where to place them within a given region, when a traffic light is not necessary.

The optimization of traffic control could reduce the effects such as traffic congestion or jams, traffic accidents, environmental pollution by gases or noise, among others. In addition, when a deficient signal configuration is installed, material and labor costs are wasted [18].

The generation of signal configurations derives in the well-known traffic light cycle optimization problem (TLCOP). This problem consists in selecting the best combination of duration times for the traffic lights [31] and belongs to the NP-Hard complexity class [13], as do its extensions. Additionally, in this work, vertical road signs are incorporated to this problem. These configurations must sometimes be generated in real time. This implies that new configurations must be generated in a short interval of time. The objective function would consist of minimizing the waiting time of the vehicles in a given configuration.

The placement of vertical road signs usually occurs only once and remains for a long period of time. However, in exceptional traffic situations, such as energy surges, lack of synchronization between traffic lights, or deterioration of the traffic light infrastructure, a method is needed to obtain configurations based on this road signs in a short time, and which can be modified [6].

In NP-Hard problems, the use of heuristics appears as an alternative to obtain good solution in less time. Specific variants of the problem have been presented and solved using hybrid metaheuristics [5, 2, 31].

Additionally, in order to evaluate the obtained configurations, simulation techniques are used. There are several types of simulation including Monte-Carlo, agent-based and dynamic system simulations [17, 37]. It is not possible to determine which type of simulation obtains the best results for all types of problems. Depending on the model to be used and the type of simulation applied, the results will be favorable or not [17].

Agent-Based Simulation (ABS) has been widely used for problems related to Traffic Engineering [23]. ABS allows modeling and simulation of realistic scenarios [16]. Intelligent agents have an autonomous and proactive character that fits perfectly with the social behaviors of drivers and pedestrians on urban roads [25].

Based on the above, the objective of this paper is to present an approximate scheme to solve the problem of generating traffic lights and vertical road signs configurations. Each configuration is evaluated using an ABS. Four classical metaheuristics [7] are used: Simulated Annealing [32], Threshold Limited [24], Evolution Strategy [28] and Genetic Algorithm [28]. Finally, the results show which of the algorithms gives the best performance.

The article is structured as follows: in Section 2. a literature review of the TLCOP and its solution techniques is presented. Then, in Section 3. the TLCOP variant and a simulation-based optimization problem are described. In Section 4. the heuristic scheme to be applied is proposed and in Section 5. an experimentation for six areas of Havana, Cuba, is carried out. Finally, Section 6. contains the discussion of the results and Section 7. presents the conclusions of the research.

2. REVIEW OF TLCOP AND ITS SOLUTION METHODS

The TLCOP has been a widely investigated problem [10, 11, 38]. The problem consists of finding the best combination for the duration times of the traffic lights in a given region. This is a multi-objective problem and the different objectives can be: maximize the number of vehicles arriving at their destination within a specific time period, maximize the average vehicle speed of all vehicles in the region, minimize the waiting time of all vehicles in this region, and minimize the queue length and thus time for all intersections in the region, among others [38].

There are several definitions of TLCOP with different objectives. Among them vary the objective function, as well as the constraints and the problem parameters. In [11] a TLCOP objective function is defined that includes all of the above objectives:

$$F(\tau) = \frac{\Sigma_{v=1}^{V(\tau)} j_v(\tau) + \Sigma_{v=1}^{V(\tau)+C(\tau)} w_v(\tau) + C(\tau) \cdot S_t}{V^2(\tau) + C_r} \tag{2.1}$$

where τ denotes a solution consisting of:
$V(\tau)$: the number of vehicles that arrives the destination;
$C(\tau)$: the number of vehicles that do not arrives the destination;
$j_v(\tau)$: the travel time of vehicle v;
$w_v(\tau)$: the waiting time of vehicle v and
S_t: the simulation time.

These values are calculated with internal equations from the times defined for the lights. For this expression all values must be in the range of $[1e + 0 \ldots 5e + 2]$ and the times of the lights must be integers greater than 0.

The decision variables of the problem are the duration times of the lights of each traffic light. In the extensions of the problem, the authors decide whether these times are synchronized or independent [35]; in this proposal are modeled independently. The general parameters of the problem include the number of existing traffic lights and their location, the vehicular flow of the region and the schedule of the day. Each new presentation of the problem may include other parameters that are considered influential in the evaluation of the solutions [15].

The solutions are represented by a list of integers that means the control plan of the duration times of all the traffic lights in TLCOP [31].

2.1. Solutions for TLCOP

Methods to solve TLCOP has been investigated from two main approaches: the application of simulation techniques [23] and the use of heuristic methods [15].

TRANsYT [34] is a tool for the simulation and optimization of traffic signals. It optimizes the configuration of the signals through an objective function that is a linear combination of the delay time and the number of stops of the vehicles in the simulated network. Microscopic simulation has also been used to imitate road networks and Simulation of Urban MObility (SUMO) [4] is an example. SUMO is a scalable agent-based traffic simulator in network size and number of vehicles simulated.

Another example of agent-based microscopic simulation is MATSim [33], which simulates the traffic of a vast region throughout the day. MATSim pursues an activity-based approach to demand generation. Approximate algorithms have been explored because TLCOP and its extensions are NP-Hard [19]. Developing an effective search algorithm for adjusting traffic lights in an urban area improves the performance of traffic lights and helps to reduce traffic congestion.

In [31], an integration between a grey wolf optimization algorithm (GWO) and a grasshopper optimization algorithm (GO) is proposed to obtain a new algorithm called grey wolf with grasshopper optimization (GWGO). The solutions are evaluated using SUMO and the resulting simulation shows that hybrid metaheuristics provides an effective way to simplify the control methods of the traffic light. In [5] a hybrid metaheuristics based on Tabu Search (TS) and Variable Neighborhood Search is proposed in order to optimize the model known as MAXBAND. The aim is to maximize the time that vehicles travel without being stopped by any traffic light through the placement of road signs. In [10] a model is proposed that minimize the total vehicle delay time and the total pedestrian delay time within a finite time range. Metaheuristics, such as Harmony Search (HS) and Ant Bee Colony (ABC), are used to solve the problem.

In [12] a tool that statistically studies the traffic flow and evaluates the lighting policy is proposed. It uses metaheuristic algorithms of the Simulated Annealing type for the optimization problem that can be single-objective or multi-objective. Simulation is also used to evaluate the proposal.

In [27], a Genetic Algorithm (GA) was implemented to minimize the delay at an intersection by searching for red and green cycle intervals. The objective function minimizes the delay at the intersection and increases the progressive traffic flows on the roads. A method for optimizing signal control parameters for intersection control based on GA is proposed in [21]. The control objective is the average vehicle delay, and the control variable is the ratio of green signal to cycle time. The GA with a fitness calibration method and an adaptive cross mutation function is used to solve the signal control model. The solutions are evaluated using a simulation in MATLAB [3]. In [20] the performance of different versions of a GA were used: standard GA, the sequential GA, and the voting GA. It was determined that the sequential GA is most efficient. In this proposal they use an agent-based simulation tool developed by the authors.

The survey did not find solutions that consider others road signals as an element of the TLCOP. The search for optimal solutions is focused on the correct selection and synchronization of the timing of the traffic lights.

3. MODELING SIGNAL CONFIGURATION OPTIMIZATION PROBLEM

This section presents an optimization scheme to solve an adaptation of the TLCOP. This adaptation includes the placement of traffic lights and others road signs in the available corners of the map.

The objective is to minimize the waiting time of vehicles. Ideally this objective can be achieved by setting all phases of the traffic lights to green. In practice this action is not possible. At a traffic-light intersection there must be at least one phase on red to guarantee the staggered passage of vehicles in order to avoid congestion and accidents.

To represent the solution of the problem, a *code* list with size C is used, where C is the number of signals located on the map. Each position in the list will have the form $\{index : signal\}$, where *index* represents the index of the corner (in the list of corners created by the simulation when loading the map) and *signal* will be the identifier of the placed signal. In each corner a vertical sign (V) or a traffic light (S) could be placed. The traffic lights would be represented by the times of each light of the cycle (Green, Red, Yellow) and the vertical signs would be "Stop" (P) and "Yield" (Y). Figure 3. shows an example of encoding the solution for a map with 22 corners where a "Stop" sign is placed in the corner 3 and a "Yield" sign is located in the corner 18. The traffic lights are placed at corners 12 and 22, and the green, red and yellow times are $(25, 15, 3)$ and $(68, 24, 3)$ respectively.

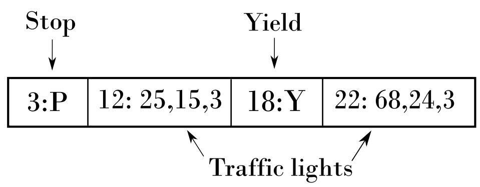


Figure 1: Example of codified solution

3.1. EVALUATION OF CONFIGURATIONS USING AGENT-BASED TRAFFIC SIMULATION

An agent-based simulation is used to evaluate the solutions of the problem. The simulation begins with the selection of the control parameters by the traffic analyst, such as: the duration times for each traffic light, the probability of vehicles committing a violation, the portion of the map to be simulated, and the weather conditions under which the simulation will take place.

Once the environment is opened, the analyst places the road signs at the desired intersections and starts the simulation. Different agents interact in the simulation. The behavior of each of the agents is described below:

- **Agent Traffic Light:** Checks if the light's time is finalized and changes for the next one. For each phase of the traffic light (one for each access road) and depending on the light that is active, there will be a group of streets allowed and others not.
- **Agent Arrival:** Checks time between arrivals selected by the analyst. Creates another vehicle type agent if time has elapsed. Otherwise it does not perform any action and continues waiting.
- **Agent Vehicle:** Behaves like a vehicle deciding whether to obey the signals or not. They can cause accidents, overtake lanes, break down and decide on the speed at which they travel.
- **Agent Environment:** The variables of temperature, humidity and rain are displayed by the environment and the vehicles receive the corresponding notification so that they moderate their speed taking into account the environmental state at all times.

Figure 3.1. shows an example of the simulation environment with the agents traffic light, vehicle and arrival. Also two vertical signs placed and the environmental state at a given point of the simulation are shown. The example corresponds to the solution previously showed in Figure 3..

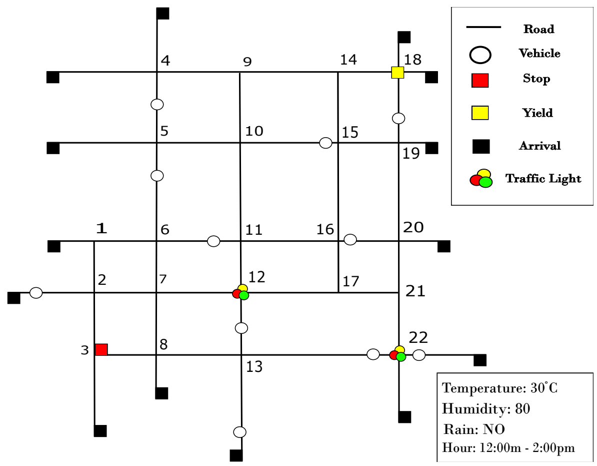


Figure 2: Simulation environment elements

When simulation ends, the waiting time of all simulated vehicles is averaged. This time is stored for each vehicle agent by adding up the seconds it has been stopped.

4. APPROXIMATE SOLUTION METHOD

Metaheuristics are high-level heuristics that can be used in a general way to solve any optimization problem in an approximate way. They are applied to problems that cannot be solved by exact methods, being popular in the solution of NP-Hard problems. According to the No Free Lunch theorem [36, 1] no metaheuristic is better than another for all optimization problems. Therefore, a scheme composed of initial solution or population generation methods and mutation and crossover operators is proposed to be used in classical metaheuristics.

4.1. INITIALIZATION METHOD AND NEIGHBORHOOD OPERATORS

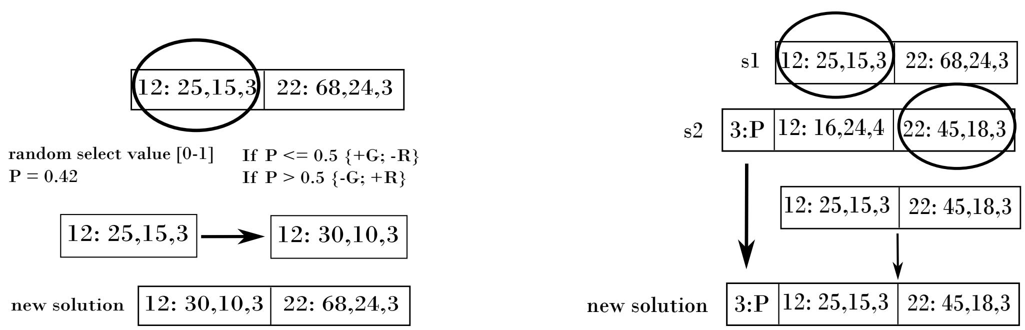
First, Algorithm 1 presents to the method of generation of the initial solution. This method follows a heuristic based on a factor H, with the objective of obtaining from the beginning possible good solutions of the problem. The H factor responds to the proportion between the time that the vehicles must wait with the time that they have to advance. The smaller this factor is, the longer the time to move forward, and therefore the less waiting time in the simulation. This ratio will never be equal to 0 because there must always be at least one phase in red.
Second, for the search of new solutions a mutation operator is applied. This operator selects one of the corners that compose the *code*. If the signal in the selected corner is a traffic light, the mutation is performed as shown in Figure 4.1.. If it is a vertical sign the value is changed (if there is a “Stop” it becomes “Yield” or reverse).

Algorithm 1 Initialization Method

Input: list of corners

Output: initial solution

```
for every corner in the map do
    get vertical sign set
    if there is a traffic light then
        Best = ∞
        repeat
            set TV, TR, TA in range and complete cycle of [35-120]
            compute H = (TR+TA+3ΣE_ij)/TV        ▷ variables that increase the waiting
            update Best                          ▷ time among variables that decrease it
        until H < Best
    end if
end for
return Best
```



(a) Mutation Operator (b) Crossover Operator

Figure 3: Operators Procedures

Finally, the application of the Genetic Algorithm needs to define a crossover operator. Figure 4.1. shows the procedure of this operator. This is a point-based crossover operator for common corners between solutions. Exclusive corners of a single solution are added directly to the new solution.

5. EXPERIMENTATION

The objectives of the experimentation are:

- To compare several metaheuristics using the propose scheme to solve the problem.
- Characterize the configurations generated by the algorithms.

The algorithms used are: Threshold Limited (TL) [24], Genetic Algorithm (GA) [29], Evolution Strategy (ES) [29] and Simulated Annealing (SA) [24].

To obtain the results, 15 executions with 5000 iterations were performed for each algorithm. The parameter's configuration of the algorithms is presented in Table 1. The average, maximum and minimum waiting time metrics are used to compare the results.

Table 1: Algorithms configuration

Algorithm	Parameter	Value
Simulated Annealing	alpha (α)	0.9
	Initial temperature	0.0
	Final temperature	20.0
	Same temperature iterations number	20
Threshold Limited	Threshold	0.5
Evolution Strategy	Selection by Truncation	2
	Replace Type	Generational Replace
	Count Ref	4
Genetic Algorithm	Selection by Truncation	2
	Replace Type	Generational Replace
	Size of Population	4
	Mutation Probability	0.5
	Crossover Probability	0.9

The simulation was performed on a PC with: Hewlet Packard Pavilion, 8GB RAM, Intel i5-6200U processor @ 2.3GHz and Windows Pro x64 OS. The simulation and heuristics were implemented using the Java programming language. The scheme for the metaheuristics was also developed in Java using the BiCIAM library [8].

5.1. DESCRIPTION OF THE INSTANCES

Six data-sets obtained from a map of Havana, Cuba, are used to evaluate the proposed methods. Table 2 shows the characteristics of the instances and the initial configuration.

Table 2: Instances description

Instance	Number of corners	Total of signals	Number of traffic lights	Number of road signs
a - 23 St. & L St.	43	17	10	7
b - Agua Dulce	35	10	6	4
c - Ciudad Deportiva	32	9	7	2
d - Carlos III & G St.	9	6	4	2
e - Túnel de la Bahía	18	6	4	2
f - Virgen del Camino	17	7	5	2

The agent's configuration is shown in Table 3. The used data correspond to the standard behavior of infractions and accidents in Cuba. Information on traffic light times, violation probabilities, accident probabilities, among other parameters directly related to traffic, are posted by traffic analysts based on their mastery of the subject. Other calculable parameters such as expected congestion and number of vehicles expected at each traffic signal are obtained from the mathematical model presented in [22][1].

[1] Test data set available at: https://raw.githubusercontent.com/amoreno98/simulation-model-data/main/parameters.json

Table 3: Configuration of simulation agents

Parameter	Value
Red light infringement probability	0.005
Vertical sign infringement probability	0.002
Collision probability	0.0001
Lane overtaking probability	0.02
Vehicle breakage probability	0.00001
Temperature - Humidity	30° C - 80%
Daytime hours	11:00 AM - 1:00 PM

5.2. COMPARISON OF THE RESULTS

In view of the first objective, Table 4 shows the results obtained by the algorithms for each of the evaluated instances. For each algorithm, the waiting time of the best and the worst solution and the average of the 15 executions are presented.

Table 4: Waiting times (s.) obtained by the algorithms

Instance	TL			GA			ES			SA		
	Best	Average	Worst	Best	Average	Worst	Best	Average	Worst	Best	Average	Worst
a	20.7	33.7	41.0	22.1	**29.6**	39.7	22.3	31.3	**43.4**	**20.2**	31.7	41.7
b	**21.2**	30.8	45.9	21.5	**30.2**	43.3	22.1	32.4	42.7	21.9	36.7	**46.1**
c	19.6	25.9	32.1	**17.3**	**25.1**	30.7.4	19.4	26.4	**33.0**	18.7	26.7	31.1
d	18.9	**22.4**	28.2	19.5	23.1	27.0	18.8	25.4	**33.8**	**15.5**	23.1	30.7
e	17.9	**21.3**	28.4	**17.5**	23.9	29.1	18.5	22.4	**29.9**	18.2	22.1	28.6
f	18.4	21.9	29.1	**18.1**	**21.6**	29.2	19.8	22.3	**30.2**	19.4	22.1	30.0

The GA algorithm is the one that obtains the best solution the most times, followed by SA and then TL, while the worst ones are generally generated by the ES algorithm. For most of the instances the best average results per execution are provided by GA.

The waiting time takes values between [15.5 - 46.1] seconds. The waiting times obtained by the metaheuristics are considerably shorter than the times obtained by the initial configurations performed manually for each instance.

The difference between the worst and best solutions obtained by the metaheuristics shows that the implemented operators converge correctly to good solutions for the problem. In the best solutions the total waiting time of a vehicle when leaving the simulation will be between two and three times shorter than the waiting time of the worst solutions.

To analyze if there are really significant differences between the algorithms we use the average metric. We apply the Friedman test and then the posthoc methods Holm [14] and Finner [9]. The Friedman test is performed with an $\alpha = 0.05$. The null hypothesis (H_0) is: all algorithms are equal. The alternative hypothesis (H_1) is: all algorithms are different. The tests were executed using the KEEL tool [30].

The Friedman statistic (chi-square distributed with 3 degrees of freedom) was obtained with a value of 5.75, and the $p-$value $= 0.024427$. This result evidences that there are significant differences between the algorithms. Table 5 shows the ranking of the Friedman test and the posthoc results. The ranking shows that the GA has the best performance for the instances.

Table 5: Friedman test results

Algorithm	Ranking	Posthoc
TL	2	-
SA	3.0833	$>$
GA	**1.75**	*
ES	3.1667	$>$

From this comparison we can concluded that the only algorithm that does not present significant differences with respect to GA is TL.

For achieving the second objective, we analyze the configurations that were generated by the algorithms.

Table 6 shows duration times obtained by the algorithms for best and worst solutions (green, red). The average times for those lights are also showed. The yellow light was maintained between [3 - 5] seconds for all configurations.

Table 6: Summary of the duration times generated by the algorithms

Instance	TL			GA			ES			SA		
	Best	Average	Worst	Best	Average	Worst	Best	Average	Worst	Best	Average	Worst
a	40, 20	58, 23	**30, 35**	40, 25	**60, 21**	28, 32	**45, 22**	55, 25	29, 25	38, 21	57, 22	30, 30
b	48, 28	50, 30	35, 18	**50, 26**	**55**, 29	45, 46	52, 30	53, **28**	**25, 30**	44, 20	52, 32	48, 50
c	46, 23	61, 28	55, 52	**51, 19**	60, **27**	36, 30	52, 22	60, 33	**33, 45**	50, 25	**62**, 30	50, 45
d	**55, 23**	59, 31	30, 40	45, 18	**66, 29**	55, 60	57, 30	63, 30	30, 33	50, 20	58, 34	**25, 45**
e	55, 28	63, 24	40, 42	52, 30	68, **22**	**19, 40**	**58, 31**	64, 23	30, 36	60, 35	**70**, 25	39, 42
f	**30, 16**	60, 32	45, 50	35, 22	**71, 30**	60, 55	30, 18	65, 35	35, 34	25, 15	68, 50	**26, 38**

We can notice that GA has generally lowest red time average and highest green time average. The balance between green and red times is essential to ensure a minimum waiting time.

The best and worst balances between green and red times are also shown. This duration times are generally in correspondence with best and worst waiting times observed in Table 4, with the exception of Instance (a), whose best balance corresponds to the second best waiting time.

This demonstrates the importance of selecting both synchronized times. Lower green time implies less time between red phases. Higher red times implies directly higher waiting times.

After analyzing the times obtained by the solutions, we will see also their structure, taking by reference Instance (a).

Figure 5.2. shows the signal configuration obtained in the best solution of the GA algorithm. It can be seen that the vertical road signs are generally placed at intersections close to the traffic lights. In this location they serve as flow controllers with constant waiting time for vehicles before reaching the traffic lights, and thus the waiting time at the red light decreases. In addition, their placement helps to avoid accidents, which are another waiting factor on the road.

The areas with more traffic lights correspond to primary streets with high vehicle demand, while the areas with no traffic lights correspond to secondary or residential streets.

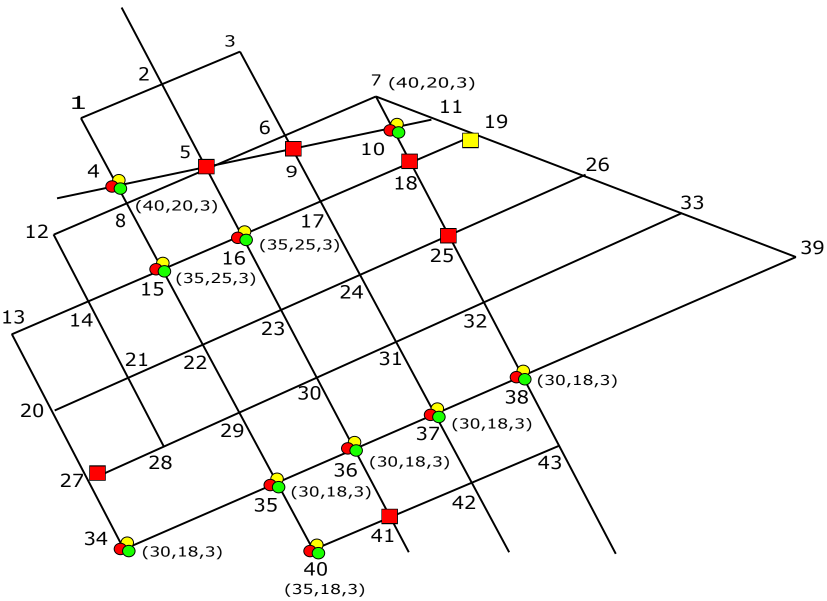


Figure 4: Configuration analysis for Instance (a)

For this instance, initially were placed "Yield" signs at intersections 5, 18, 41, and the times of traffic lights were not synchronized. Waiting time obtained by the initial configuration was 58 seconds, while optimized solution had a waiting time of 23 seconds. The application of the metaheuristics considerably improves waiting times experienced by the vehicles in the simulation. The initial configurations serve as a basis for the operators in their quest to obtain signal locations close to those located by the analysts. Optimization is essential to obtain correct roadway performance.

6. DISCUSSION

In the literature review, no models or proposals were found that include in the traffic configuration all the variants of vertical signals. Most of existing models and optimization methods include traffic light only. The main contribution of the presented work is to propose the optimization of traffic from traffic lights and other signaling alternatives when the firsts are not necessary or not feasible.
The creation of a specific heuristic including in them all the factors and parameters present in the definition of this problem adaptation could ensure a significant improvement in the results. As future work we will focus on the implementation of a new specific heuristic for the generation of signal configurations.

7. CONCLUSIONS

The presented problem is an extension of TLCOP that includes the placement of road signs. Agent-based simulation was used to evaluate the configurations. Four approximate algorithms were compared and after analyzing the results it can be concluded that:

- For the presented signal configuration problem, the algorithm with the best average performance was Genetic Algorithm.
- The Evolution Strategy algorithms offer the longest waiting times in the generated solutions.
- For the correct synchronization of traffic lights, it is necessary to achieve an optimal balance between red and green times.
- Vertical road signs are an essential element in the TLCOP because they regulate the traffic flow and prevent road accidents which influence the waiting time on the road.
- It is necessary a new specific heuristic that includes road signs and variables from traffic.

RECEIVED: AUGUST, 2022.
REVISED: MARCH, 2023.

REFERENCES

[1] ADAM, S. P., ALEXANDROPOULOS, S.-A. N., PARDALOS, P. M., AND VRAHATIS, M. N. (2019): No free lunch theorem: A review **Approximation and Optimization: Algorithms, Complexity and Applications**, pages 57–82.

[2] ALRIDHA, A. H. AND AL-JILAWI, A. S. (2022): Solving np-hard problem using a new relaxation of approximate methods **International Journal of Health Sciences**, 6:523–536.

[3] ARELLANO, M. A. (2013): **MATLAB y Simulink para Ingeniería.** PERU: Universidad de Ciencias y Humanidades UCH, 1st. edition.

[4] BEHRISCH, M., BIEKER, L., ERDMANN, J., AND KRAJZEWICZ, D. (2011): SUMO-simulation of urban mobility: an overview In **Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation**. ThinkMind.

[5] CABEZAS, X., GARCÍA, S., AND SALAS, S. (2019): A hybrid heuristic approach for traffic light synchronization based on the MAXBAND. **Soft Computing Letters**, 1(2019):100001.

[6] COSTA, M., SIMONE, A., VIGNALI, V., LANTIERI, C., AND PALENA, N. (2018): Fixation distance and fixation duration to vertical road signs. **Applied Ergonomics**, 69:48–57.

[7] DOKEROGLUA, T., SEVINCB, E., KUCUKYILMAZA, T., AND COSARB, A. (2019): A survey on new generation metaheuristic algorithms. **Computers and Industrial Engineering**, 137(2019):106040.

[8] FAJARDO, J. (2015): **Soft Computing en Problemas de Optimización Dinámicos** PhD thesis, Departamento de Ciencia de la computación e Inteligencia Artificial, Granada , España.

[9] FINNER, H. (1993): On a monotonicity problem in step-down multiple test procedures. **Journal of the American Statistical Association**, 88:920–923.

[10] GAO, K., ZHANG, Y., ZHANG, Y., SU, R., AND SUGANTHAN, P. (2019): Meta-heuristics for bi-objective urban traffic light scheduling problems. **IEEE Transactions on Intelligent Transportation Systems**, 20(7):2618–2629.

[11] GARCÍA-NIETO, J., OLIVERA, A., AND ALBA, E. (2013): Optimal cycle program of traffic lights with particle swarm optimization. **IEEE Transactions on Evolutionary Computation**, 17(6):823–839.

[12] GUILARTE, O., PUJOL FARIÑA, G., ALLENDE ALONSO, S. M., GEMAYQZEL, Y., AND BOUZA ALLENDE, G. (2015): Herramienta computacional para sugerir y validar políticas de luces de semáforos In **13º Simposio Argentino de Investigación Operativa, SIO 2015**.

[13] HARTMANIS, J. (1982): Computers and intractability: A guide to the theory of NP-Completeness (Michael R. Garey and David S. Johnson). **Siam Review**, 24(1):90.

[14] HOLM, S. (1979): A simple sequentially rejective multiple test procedure **Scandinavian Journal of Statistics**, 6:65–70.

[15] JALILI, S., NALLAPERUMA, S., KEEDWELL, E., DAWN, A., AND OAKES-ASH, L. (2021): Application of metaheuristics for signal optimisation in transportation networks: A comprehensive survey. **Swarm and Evolutionary Computation**, 63:100865.

[16] JIMENEZ, A., CARDENAS, P., CANALES, A., JIMENEZ, F., AND PORTACIO, A. (2020): A survey on intelligent agents and multi-agents for irrigation scheduling. **Computers and Electronics in Agriculture**, 176(2020):105474.

[17] KARAKIKES, I., NATHANAIL, E., AND SAVRASOVS, M. (2019): Techniques for Smart Urban Logistics Solutions' Simulation: A Systematic Review In Prentkovskis, O., editor, **Reliability and Statistics in Transportation and Communication. RelStat 2018. Lecture Notes in Networks and Systems**, volume 68, pages 551–561. Springer, Cham.

[18] KRZYSZTOF, M. (2016): The importance of automatic traffic lights time algorithms to reduce the negative impact of transport on the urban environment In **Transportation Research Procedia**, volume 16, pages 329–342.

[19] LENSTRA, J. K. AND KAN, J. K. (1981): Complexity of vehicle routing and scheduling problems. **Networks**, 11(2):221–227.

[20] LIANG, X. J., GULER, S. I., AND GAYAH, V. V. (2020): A heuristic method to optimize generic signal phasing and timing plans at signalized intersections using Connected Vehicle technology **Transportation Research Part C: Emerging Technologies**, 111:156–170.

[21] MA, C. AND LIU, P. (2019): Intersection signal timing optimization considering the travel safety of the elderly **Advances in Mechanical Engineering**, 11(12):1687814019897216.

[22] MORENO ROMÁN, A. C., MORENO ESPINO, M., PORRAS, C., AND PAVÓN, J. (2022): Modelling urban traffic configuration with the influence of human factors In Bicharra Garcia, A. C., Ferro, M., and Rodríguez Ribón, J. C., editors, **Advances in Artificial Intelligence – IBERAMIA 2022**, pages 53–64, Cham. Springer International Publishing.

[23] NGUYEN, J., POWERS, S. T., URQUHART, N., FARRENKOPF, T., AND GUCKERT, M. (2021): An overview of agent-based traffic simulators. **Transportation Research Interdisciplinary Perspectives**, 12:100486.

[24] RERE, L. M. R., WARDIJONO, B. A., CHANDRA, Y. I., AND LATIFAH (2019): A comparison study of three single-solution based metaheuristic optimisation for stacked auto encoder In **The 2nd International Conference on Data and Information Science**, 1192.

[25] ROMÁN, A. C. M. AND ESPINO, M. M. (2021): Configuración de señales viales: Herramienta de simulación basada en agentes. **Revista Latino-Americana de Innovación e Ingeniería de Producción**, 9(16):194–208.

[26] ROMÁN, A. M., ESPINO, M. M., AND FRAGA, D. B. (2021): Herramienta de simulación para evaluar configuraciones semafóricas. **Revista Cubana de Transformación Digital**, 2(1):102–114.

[27] SHARMA, B. AND KUMAR, S. (2019): Delay Optimization Using Genetic Algorithm at the Road Intersection **International Journal of Information Retrieval Research (IJIRR)**, 9(2):1–10.

[28] SLOWIK, A. AND KWASNICKA, H. (2020): Evolutionary algorithms and their applications to engineering problems **Neural Computing and Applications**, 32:12363–12379.

[29] TALBI, E.-G. (2009): **Metaheuristics: from design to implementation** John Wiley & Sons, Inc., Hoboken, New Jersey, 1st edition.

[30] TRIGUERO, I., GONZÁLEZ, S., MOYANO, J. M., GARCÍA, S., ALCALA-FDEZ, J., LUENGO, J., FERNANDEZ, A., DEL JESUS, M. J., SANCHEZA, L., AND HERRERA, F. (2017): KEEL 3.0: An Open Source Software for Multi-Stage Analysis in Data Mining. **International Journal of Computational Intelligence Systems**, 10(1):1238–1249.

[31] TSAI, C.-W., TENG, T.-C., LIAO, J.-T., AND CHIANG, M.-C. (2021): An effective hybrid-heuristic algorithm for urban traffic lightscheduling. **Neural Computing and Applications**, 33(24):17535–17549.

[32] VENKATESWARAN, C., RAMACHANDRAN, M., RAMU, K., PRASANTH, V., AND MATHIVANAN, G. (2022): Application of simulated annealing in various field **Materials and its Characterization**, 1(1):01–08.

[33] W AXHAUSEN, K., HORNI, A., AND NAGEL, K. (2016): **The multi-agent transport simulation MATSim** Ubiquity Press.

[34] WALLACE, C. E., COURAGE, K., REAVES, D., SCHOENE, G., AND EULER, G. (1984): TRANSYT-7F user's manual Technical report, U.S. Department of Transportation.

[35] WEI, H., ZHENG, G., YAO, H., AND LI, Z. (2018): IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control In **The 24th ACM SIGKDD International Conference**, number 10 in KDD '18, pages 2496–2505.

[36] WOLPERT, D. H. AND MACREADY, W. G. (1997): No free lunch theorems for optimization. **IEEE Transactions on Evolutionary Computation**, 1(1):67–82.

[37] YUNIAWAN, D., P., A. F., HARIYANTO, S., AND SETIAWAN, R. (2018): Traffic queue modeling using arena simulation software (a case study of Mergan 4-Way intersection in Malang City) In **MATEC Web of Conferences**, volume 204.

[38] ZHANG, Y., GAO, K., ZHANG, Y., AND SU, R. (2019): Traffic light scheduling for pedestrian-vehicle mixed-flow networks. **IEEE Transactions on Intelligent Transportation Systems**, 20(4):1468–1483.