A *PSO-Optimized Fixed* and a *PSO-Optimized Neural Network-Adaptive* Traffic Signal Controllers for Traffic Improvement in Santo Domingo, Dominican Republic

Eddy Martínez[1][0000-0002-3641-7954], Neil Buckley[1], and Emanuele Lindo Secco[1][0000-0002-3269-6749]

[1]Robotics Laboratory, School of Mathematics, Computer Science & Engineering, Liverpool Hope University, L16 9JD, UK
20209153@hope.ac.uk, bucklen@hope.ac.uk, seccoe@hope.ac.uk

Abstract. Satisfying the mobility demand is one of the biggest concerns arising with the increase of urban population. With many people in the road network, traffic congestions are present in most of the cities in the world. The Distrito Nacional in Santo Domingo, capital city of Dominican Republic, is a notorious example of this phenomenon. Unfortunately, all the efforts to improve traffic experience there have had little success.
With this work, two models have been developed using Particle Swarm Optimization (PSO): a *PSO-optimized Fixed Traffic Signal Control* (PSO-FTSC) and a *PSO-optimized Neural Network-Adaptive Traffic Signal Control* (PSO-NN-ATSC) that uses 4 Neural Networks to predict phase times.
The intersection of 27 de Febrero Avenue corner with Winston Churchill Avenue was simulated using *Simulation of Urban Mobility* (SUMO), minimizing the Time Loss per vehicle during optimisation.
The models, PSO-FTSC and PSO-NN-ATSC, present reductions of 17% and 24% of Mean Time Loss, respectively.

Keywords: Traffic Signal Control, Simulation of Urban Mobility, Particle Swarm Optimisation, Neural Network.

1 Introduction

The increase of traffic flow is an unavoidable consequence of population growth. The resulting traffic congestions are the cause of long waits for commutants and higher fuel consumption, which cause higher expenses and increased air pollution globally. Only in the United States of America (USA), 3.6 billion vehicle hours were lost due to traffic delay in year 2000, together with 21.6 billion liters of fuel and US$67.5 billion because of decreased productivity [1]. The same costs in Greater Santo Domingo, Dominican Republic, exceeded RD$48 million (USD$ 855 thousand) daily in 2016 [2]. Apart from these expenses, the long waits and air pollution result in mental and respiratory health issues, respectively. One third of mortality related to fine particulate matter pollution in North America is caused by these emissions [3].

Throughout the last 30 years, many attempts have been made to find solutions to this issue. The work done in this matter can be classified in one of the following fields:

1. Improving road infrastructure
2. Coordinated vehicle rerouting
3. Diversifying mobility
4. Traffic light signal optimisation

Unfortunately, improving road infrastructure and diversifying mobility are not achievable in short term and require large investments [4]. Coordinated vehicle re-routing is only possible if all vehicles are connected to a centralized system and if drivers comply with the system.

Traffic Signal Timing (TST) optimization, on the other hand, is considered by Qadri et al. [4] as the fastest and cheapest way to tackle the traffic congestion problem. Hence, this project proposes a new method based on neural networks optimized with Particle Swarm Optimization (PSO) that adjust the phase of a traffic light signal in real-time. As a comparison, a model of optimum fixed signal times was developed using PSO as well. Both models were developed and tested on Simulation of Urban Mobility (SUMO) program with a sample of Santo Domingo's road network as an environment.

2 Literature Review

2.1 Approaches to Traffic Signal Control

Qadri et al. [3] also categorized three different approaches of Traffic Signal Control (TSC) [4]:

- **Fixed TSC**: This approach (referred by FTSC) mostly uses offline optimization that returns predetermined TST parameters; hence, it is only appropriate for settings where there are predictable patterns of traffic.
- **Actuated TSC**: It alters TST directly after present data is captured by sensors.
- **Adaptive TSC**: Here called ATSC, it is the modified form of the Actuated TSC, as the TSC is trained to optimize the overall traffic in real-time using present data.

Any of these approaches can be achieved with a variety of computational techniques, the most common listed below.

Fuzzy Logic. Adacher [5] formulated a discrete delay cost function and converted it into a continuous function with a surrogate method in order to solve it with gradient algorithms. Alvarez Gil et al. [6] also used a surrogate model simulated in IT MICROSIM, which was optimized using fuzzy logic. Hartanti et al. [7] also optimized their model using the Fuzzy Mamdani method. Another fuzzy logic approach was presented by Tchuitcheu et al. [8] using the data from smart cameras that are able to identify special vehicles to prioritize them. They also suggested this approach would be able to identify traffic violations and emergency situations.

Fuzzy Logic, as many other approaches, can be enhanced by neural networks. This was demonstrated by the work of Araghi et al. [9] [10], who developed an Adaptive Neuro-Fuzzy Inference System (ANFIS) that was optimized using metaheuristics.

Metaheuristics. García-Nieto et al. [11] used PSO to find optimum traffic light cycles in an adaptive approach for Bahía Blanca city in Argentina and Málaga city in Spain and tested the results in SUMO. Malecki et al. [12] also used SUMO to for different metaheuristic algorithms including Differential Evolution (DE), Covariance Matrix Adaptation Evolution Strategy (CMA-ES), Monte Carlo, Archipelago, Genetic Algorithm (GA) and PSO, where CMA-ES obtained the best results.

Zhang et al. [13] used a multi-objective optimization for signal cycle and the proportion of green light on a given intersection in a modified Webster function, using GA after converting it to single-objective and achieving a delay reduction of 15.64%. Yu et al. [14] had a similar multi-objective approach using fuzzy compromise programming, which assigns weight coefficients to each optimization objective depending on traffic measurements.

Multi-objective GA (NSGA-II) was used by Nguyen et al. [15] and Zheng et al. [16]. The first one implemented a Local Search (LS) within the algorithm, forming NSGA-II-LS which obtained good results in early steps of optimization [15]. The latter used it to optimize surrogate models developed from simulations in PTV Vissim program [16].

On a different approach, Bemas et al. [17] enhanced evolution algorithms with neural networks, proving their results with SUMO.

Sanchez-Medina et al. [18] and Inoue et al. [18] demonstrated that metaheuristics combined with powerful hardware result in high-performance models. The former one used a Beowulf Cluster (a multi-instruction multi-data (MIMD) computer of good performance/price ratio) to run cellular-automata-based microsimulators that were optimized using GA, obtaining good results when the number of vehicles per hour increased in simulations set in La Almorzara (Saragossa, Spain) [18]. The latter one modelled a road network as an Ising model of 50x50 nodes and to optimize it they went a step further by developing a Quantum Annealing algorithm (a variation of Simulated Annealing for quantum particles) that they ran on a quantum machine called Quantum Annealer 2000 from D-Wave Systems Inc [19]. However, the number of parameters in this model is limited by the Quantum Annealer hardware and the whole approach is based on the transformability of the road network to an Ising model.

Despite having these effective models, Ferrer et al. [20] state that no traffic simulation is an accurate representation of a real system due to the complexity and variability of a city, thus, the fitness of any solution will vary when deployed in a real setting. They claim to overcome this issue by using larger scenarios (58 intersections, 275 traffic-lights and 4827 vehicles derived from real data from the city of Málaga, Spain) and incorporating classical and iterative resampling strategies for simulating multiple scenarios in SUMO for each optimization run [20]. These strategies were tested with DE, GA, PSO and irace, which obtained the best results. irace is a software package that combines heuristic optimization with a racing method, based on F-race, that iteratively optimizes candidate solutions for noisy problems [21].

Reinforcement Learning (RL) is one of the most popular techniques for developing ATSC. Unfortunately, for centralized methods the computational complexity grows exponentially with the number of considered intersections. Multi-Agent Reinforcement Learning (MARL) decentralizes the approach, reducing the complexity with the cost of reducing the observability of the agents, as each intersection is optimized by a local RL agent. To solve this problem, Chu et al. [22] developed a MARL algorithm based on Advantage Actor Critic (A2C), where neural networks are used to estimate Q values and policies in RL.

Ozan et al. [23] used a Q-Learning RL combined with Transfyt-7F to find optimum TST. It obtained higher results than other RL approaches reviewed by Qadri et al. [4] because it was able to create sub-environments for each learning event.

Genders and Razavi [24] implemented some of the most used ATSC approaches including Webster function, Max-pressure and Self-Organizing Traffic Lights, along with deep Q-network and deep deterministic policy gradient reinforcement learning. The different controllers were optimised and tested in SUMO and the best performance was achieved by Max-pressure algorithm, stating that insights from heuristic approaches result in higher performance than many deep learning algorithms [24]. Nevertheless, they also suggest that learning methods can be further developed to outperform the other methods.

Dynamic Programming. Compared to the Control Optimization of an Intersection (COP) algorithm, with a time complexity of $O(T^3)$, Samra et al. used dynamic programming for a traffic cost function defined in time steps T that included all possible states of an intersection TSC, minimizable in $O(T)$ [25].

He et al. [26], Mehrabipour & Hajbabaie [27] and Yan et al. [28] used Mixed Integer Linear Programming (MILP) to model traffic: the first as a lane-based optimization solved with branch and bound algorithm, the second modelled network-level signal timing and found the solution with rolling horizon algorithm and the third modelled a network-level multiband signal coordination and used vehicle trajectory records to compute progression bands for high traffic demands. In contrast, Mohebifard & Hajbabaie [29] and Yu et al. [30] used Mixed Integer Non-Linear Programming (MINLP): the former based their approach on the Cell Transmission Model (CTM) and achieved a narrow convergence area and the latter implemented a double queue traffic flow model to keep real-time track of traffic dynamics and queue spillback. Yu et al. [31] also used quadratic programming to optimize vehicular and pedestrian TST for an isolated intersection.

2.2 Traffic in Greater Santo Domingo

In Dominican Republic, more than 2 million people are concentrated in the Greater Santo Domingo with a 545.4 km² area [32], having in this area the largest population density in the Caribbean [33]. The Distrito Nacional, capital city of Dominican Republic and located in the center of Greater Santo Domingo, has only 92 km^2 and hosts more than 1 million vehicles [34]. This creates the need of providing transportation to 3,097,106 trips per day [33]. Of these trips, 40% are made in a private vehicle, 40% in public transport and 20% by foot [33]. Because of this, traffic congestions occur daily, especially in the peak hours that take place from 7:30am to 9:30am, from 12:00pm to 1:30pm and from 5:00pm to 7:00pm [35].

In 2018, the local newspaper Diario Libre used data from General Direction of Transit and Land Transport Security (DIGESETT in Spanish) to create a map of the ten key points of traffic congestion in Santo Domingo [36], shown in Fig. 1.

Fig. 1. Ten key traffic congestion points in Santo Domingo [36]. The one with the white dot is the focus of this work.

The primary option to address traffic bottlenecks in Dominican Republic has always been the construction of new highways and tunnels, but in the longer term, these alternative structures haven't solved the problem [37].

Recently, in June 2020, the National Institute of Transit and Land Transport (INTRANT in Spanish) installed cameras that detect traffic levels in some key intersections of Santo Domingo city and adjust TST accordingly [38]. However, no effects of this are perceived, as communicated by Eusebio Rivera Almódovar [39] when he stated earlier this year that Santo Domingo still needs "real intelligent traffic lights".

3 Methodology

This project proposes two TSC optimization models:

1. A FTSC model with TST optimized using PSO.
2. An ATSC model with a PSO-optimized neural networks that control traffic signals.

Both models were developed and tested using a SUMO simulation based on the intersection of 27 de Febrero Avenue corner with Winston Churchill Avenue, one of the key traffic congestion points of Santo Domingo [36]. These two models were compared against the actual traffic signal timings currently used in the given intersection.

The compared metrics are the Mean Time Loss among all vehicles, provided by SUMO at the end of the simulation, and the fuel consumption and CO_2 emission per vehicle, estimated by SUMO during the simulation.

The PSO was implemented using PySwarms library which connects via python programming to the API provided by SUMO.

3.1 Simulation

The selected intersection joins two of the largest avenues in Santo Domingo, hosting high traffic levels variable throughout the day. The proximity that this intersection has to other key points means that improving traffic in this intersection can possibly result at improving traffic in the neighbouring areas.

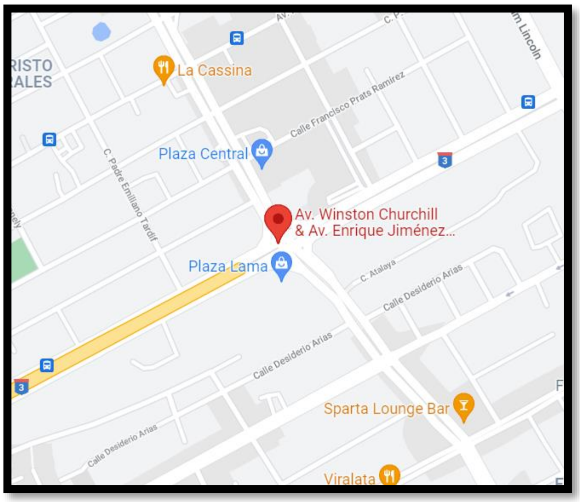

Fig. 2. 27 de Febrero Av. corner with Winston Churchill Av. from Google Maps [40].

For simulation purposes, green light phases are even numbers 0, 2, 4, 6 and the consecutive odd numbers 1, 3, 5, 7 are the respective yellow light phases. The traffic signal operates with fixed TST as specified in Figures 3-6 and Table 1.

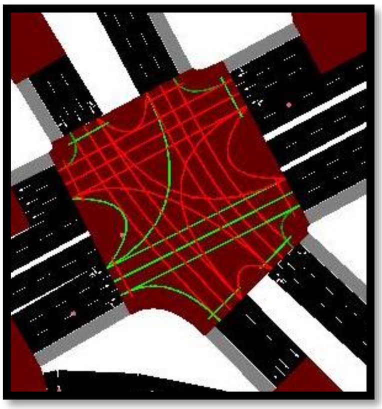

Fig. 3. Traffic Phase 0: West to East (WE)

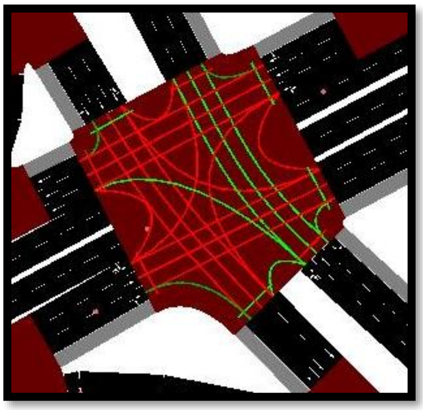

Fig. 4. Traffic Phase 2: South to North (SN)

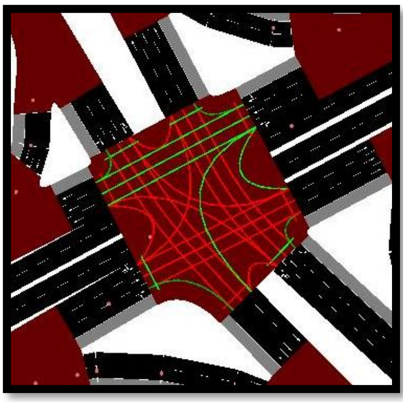

Fig. 5. Traffic Phase 4: East to West (EW)

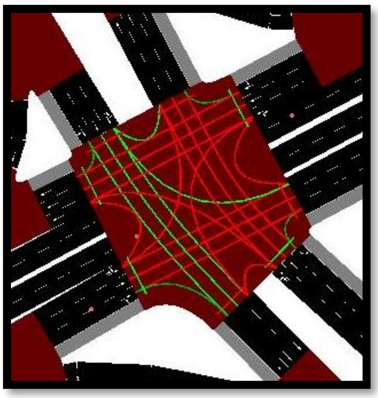

Fig. 6. Traffic Phase 6: North to South (NS)

Table 1. Current TST in seconds for 27 de Febrero Av. corner with Winston Churchill Av. from INTRANT [40]

Phase	SN	NS	WE	EW
Green	28	36	36	34
Yellow	4	4	4	4
Red	118	110	110	112
Cycle	150			

Even though this intersection has just one set of TST for the whole day, the crossing times vary a lot, which demands a variable TST, as visible in Table 2.

Table 2. Crossing times in seconds for 27 de Febrero Av. corner with Winston Churchill Av. from INTRANT [41]

Phase	SN	NS	WE	EW
Crossing time in AM hours	175	240	32	No data
Crossing time in PM hours	647	325	438	705

Because of the lack of data of actual vehicle flow in the intersection, an arbitrary variable vehicle flow is generated for the simulation by defining nine *Traffic Assignment Zones* (TAZs) including road segments (called edges in SUMO) on the border of the road network and defining *Origin-Destination* (OD) matrices between those TAZs for a total of 5244 vehicles during 3600 seconds of simulation runtime.

3.2 Traffic Optimizer Models

PSO-Optimized Fixed Traffic Signal Control (PSO-FTSC). The FTSC operates with the following algorithm:

Table 3. Algorithm for Fixed Traffic Signal Control

Pseudocode for FTSC

```
while active:
    if current_traffic_phase in [0, 2, 4, 6]: //green phases
        if time_in_phase ≥ phase_time[current_traffic_phase]:
            current_traffic_phase += 1
    else if current_traffic_phase in [1, 3, 5, 7]: //yellow phases
        if time_in_phase ≥ 4 seconds:
            current_traffic_phase += 1
end while
```

phase_time is an array of four dimensions, only indexable by even numbers between 0 and 6. The PSO-FTSC model is a 4-dimensional optimization of the green phases time, which is bound to a minimum value of 7 seconds for safety purposes [41]. Each optimization iteration requires a new simulation instance. The mean time loss value provided by SUMO at the end of simulation is the cost function to be minimized with PSO algorithm (see Fig. 7).

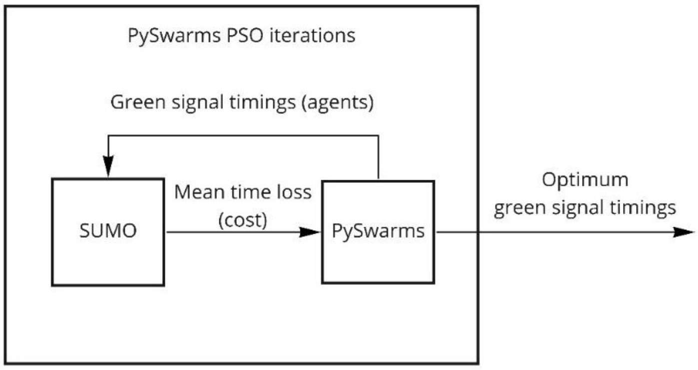

Fig. 7. PySwarms and SUMO interactions in PSO-FTSC

PSO-Optimized Neural Network for Adaptive Traffic Signal Control (PSO-NN-ATSC). In the proposed approach here, the traffic lights are controlled by the same rules of previous algorithm, except that phase_time varies on every cycle with the following algorithm:

Table 4. Algorithm for the proposed Adaptive Traffic Signal Control.

Pseudocode for ATSC
1. for phase in [0, 2, 4, 6]: //green phases
2. time_predictor[phase] = neural_network(parameters[phase])
3. while active:
4. data = get_vehicle_count()
5. if changing_to_phase in [0, 2, 4, 6]: //green phases
6. current_traffic_phase = assign_phase_number()
7. phase_time = time_predictor.predict(data)
8. end while

time_predictor is an array of four neural networks which predict the phase time respective to the four green phases. Parameters is an array of four groups of 49 parameters respective to each neural network. Thus, it is required to optimize 196 parameters in total.

When the traffic light turns green, vehicle counts are taken from edges adjacent to the intersection and processed by the correspondent neural network, which returns the amount of seconds that the green phase should last.

Hence, The PSO-NN-ATSC requires a 196-dimensional optimization (see Fig. 8).

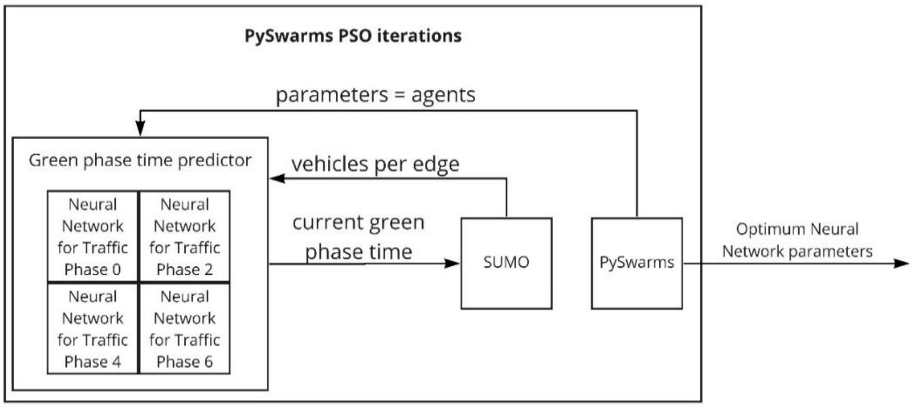

Fig. 8. PySwarms, SUMO and Neural Network interactions

Neural Network for traffic times predictions. The proposed neural network is illustrated here:

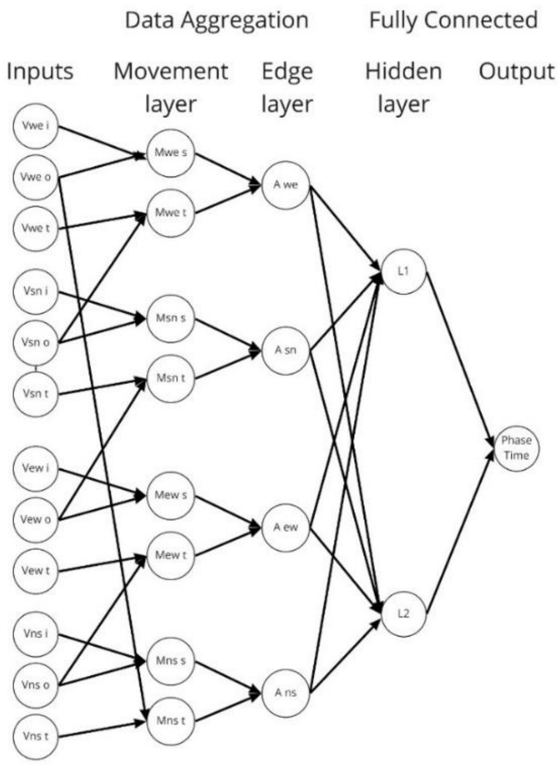

Fig. 9. Proposed Neural Network

The **input layer** receives data from SUMO. It has twelve input placeholders for the number of vehicles of the eight adjacent edges of the intersection, i.e., before (***_i*** suffix) and after (***_o*** suffix) the intersection in each of the four directions, and four more for the vehicles that are aligned on the left waiting to turn left on each side of the intersection (***_t*** suffix).

This is done because, in the simulation, sometimes most of the vehicles want to turn left, thus, they align to the left lane and leave the middle and right ones empty. Without this data, the neural network might mistakenly perceive that there are not many jammed vehicles on that side of the intersection.

In addition, the vehicle count after the intersection lets the neural network know if the edge ahead is empty or full, avoiding the mistake of allocating much green phase time for vehicles that will find no space to cross.

As predictions are taken every 5 seconds, SUMO adds the counts taken in each second.

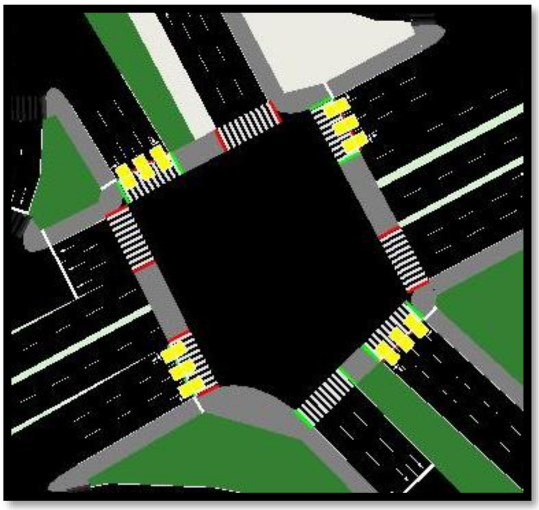

Fig. 10. Data collection points (in yellow)

For each side of the intersection, the **Movement Aggregation layer** computes the weighted sum of:

- The number of vehicles that will go straight with the number of vehicles after the intersection (***_s*** suffix).
- The number of vehicles that will turn left with the number of vehicles after the intersection on its correspondent side (***_t*** suffix).

This is formulated in Equation 1.

$$\begin{bmatrix} m_{WE_s} \\ m_{WE_t} \\ m_{EW_s} \\ m_{EW_t} \\ m_{NS_s} \\ m_{NS_t} \\ m_{SN_s} \\ m_{SN_t} \end{bmatrix} = \begin{bmatrix} w_{WE_i} * v_{WE_i} + w_{WE_o} * v_{WE_o} + b_{WE_s} \\ w_{WE_{it}} * v_{WE_t} + w_{WE_{ot}} * v_{SN_o} + b_{WE_t} \\ w_{EW_i} * v_{EW_i} + w_{EW_o} * v_{EW_o} + b_{EW_s} \\ w_{EW_{it}} * v_{EW_t} + w_{EW_{ot}} * v_{NS_o} + b_{EW_t} \\ w_{NS_i} * v_{NS_i} + w_{NS_o} * v_{NS_o} + b_{NS_s} \\ w_{NS_{it}} * v_{NS_t} + w_{NS_{ot}} * v_{WE_o} + b_{NS_t} \\ w_{SN_i} * v_{SN_i} + w_{SN_o} * v_{SN_o} + b_{SN_s} \\ w_{SN_{it}} * v_{SN_t} + w_{SN_{ot}} * v_{EW_o} + b_{SN_t} \end{bmatrix} \tag{1}$$

With this equation, the Movement Layer sums up to 24 parameters, counting 2 weights and 1 bias for each of the options, straight and turning, for each of the 4 directions.

The **Edge Aggregation layer** takes the movement data and aggregates it one step further by computing the weighted sum of the straight and turning values for each direction (see Equation 2).

$$\begin{bmatrix} a_{WE} \\ a_{EW} \\ a_{NS} \\ a_{SN} \end{bmatrix} = \begin{bmatrix} w_{WE_s} * m_{WE_s} + w_{WE_t} * m_{WE_t} + b_{WE} \\ w_{EW_s} * m_{EW_s} + w_{EW_t} * m_{EW_t} + b_{EW} \\ w_{NS_s} * m_{NS_s} + w_{NS_t} * m_{NS_t} + b_{NS} \\ w_{SN_s} * m_{SN_s} + w_{SN_t} * m_{SN_t} + b_{SN} \end{bmatrix} \tag{2}$$

The Edge Aggregation layer has 12 parameters, including 2 weights and 1 bias for each of the 4 directions.

The **hidden layer** is a standard fully connected layer with 2 neurons.

$$\begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} w_{1_{WE}} & w_{1_{EW}} & w_{1_{NS}} & w_{1_{SN}} \\ w_{2_{WE}} & w_{2_{EW}} & w_{2_{NS}} & w_{2_{SN}} \end{bmatrix} \times \begin{bmatrix} a_{WE} \\ a_{EW} \\ a_{NS} \\ a_{SN} \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{3}$$

In order to add some non-linearity, the hidden layer neurons arepassed through a vectorised sigmoid function,

$$\begin{bmatrix} l_{1_S} \\ l_{2_S} \end{bmatrix} = sigmoid\left(\begin{bmatrix} l_1 \\ l_2 \end{bmatrix}\right) \tag{4}$$

where the sigmoid function is:

$$sigmoid(l) = \frac{1}{1+e^{-l}} \tag{5}$$

The hidden layer has 10 parameters: 2 weights and 1 bias for each of the 2 outputs.

The **output layer** computes the current green phase time as the weighted sum of the 2 neurons from the hidden layer:

$$Phase\ time = w_1 l_{1_S} + w_2 l_{2_S} + b \tag{6}$$

The output layer has only 2 weights and 1 bias, thus, only 3 parameters.

3.3 Particle Swarm Optimization Implementation

PSO is a metaheuristic optimization algorithm proposed for the first time by Kennedy and Eberhart in 1995 [43] that uses the way in which different swarm-organized animals (e.g., birds) move in group in order to converge on a certain point. A set of agents (particles) $\boldsymbol{x} \in \boldsymbol{X}$ are defined which move in the search space of solutions in every iteration, all influenced by their local best and the overall global best cost among all the particles.

In PSO-FTSC, each agent is a set of the four green phase timings, and in PSO-NN-ATSC, each agent is a set of 196 parameters for the neural networks. Each agent is aiming to compute the lowest Mean Time Loss in a simulation. and obtaining the cost function requires running a complete simulation with a given set of timings to compute the mean time loss at the end of the simulation.

Table 5. Algorithm for Particle Swarm Optimization

Pseudocode for PSO
instance_n_agents
i = 0 // Iteration number
while terminating criteria not reached:
for a in agent do:
instance_simulation(agents)
run_simulation()
cost = computeMeanTimeLoss()
$v_a^{i+1} = wv_a^i + rand(0,c1) * \left(x^{best} - x_a^i\right) + rand(0,c2) * (x_a^{best} - x_a^i)$
$x_a^{i+1} = x_a^i + v_a^{i+1}$
end for
end while

In this algorithm:

- $\boldsymbol{x_a^i}$ is the value of agent *a* in the iteration *i*.
- $\boldsymbol{x_a^{best}}$ is the value of agent *a* correspondent to its lowest mean time loss.
- $\boldsymbol{x^{best}}$ is the agent value correspondent to the lowest mean time loss among all agents.
- $\boldsymbol{v_a^i}$ is the velocity of each agent in the swarm.
- $\boldsymbol{rand(0, c)}$ is a function that generates a uniform stochastic number between 0 and *c*.
- $\boldsymbol{w}$, $\boldsymbol{c1}$ and $\boldsymbol{c2}$ are hyperparameters of PSO and have the same dimensionality of the agents.

In that way, the value of each agent in the next iteration x_a^{i+1} is influenced by:

1. The current value.
2. The weighted sum of:
   a. The current velocity.
   b. the difference between the current value and the global best.
   c. the difference between the current value and the agent best.

The following PSO parameters were defined:

- $w = 0.9$
- $c1 = c2 = 0.2$

4 Results

4.1 Baseline model

As baseline model, the real FTSC was modelled in SUMO, generating the following traffic pattern:

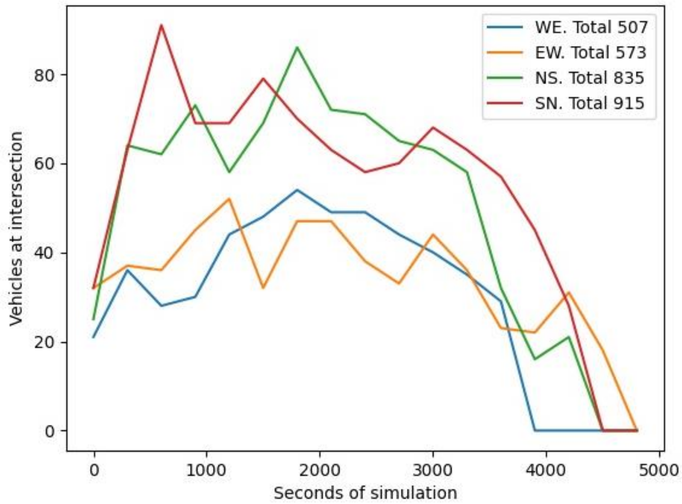

Fig. 11. Traffic pattern of baseline model

4.2 PSO-FTSC

The PSO-FTSC model optimized the green signal timings using 3 agents through 20 iterations of PSO, creating the Mean Time Loss curve in Fig. 12.

PSO converged in the solution in Table 6.

Table 6. PSO-optimised FTSC timings

Phase	SN	NS	WE	EW
Green	48	59	47	55
Yellow	4	4	4	4
Red	173	162	174	166
Cycle	225			

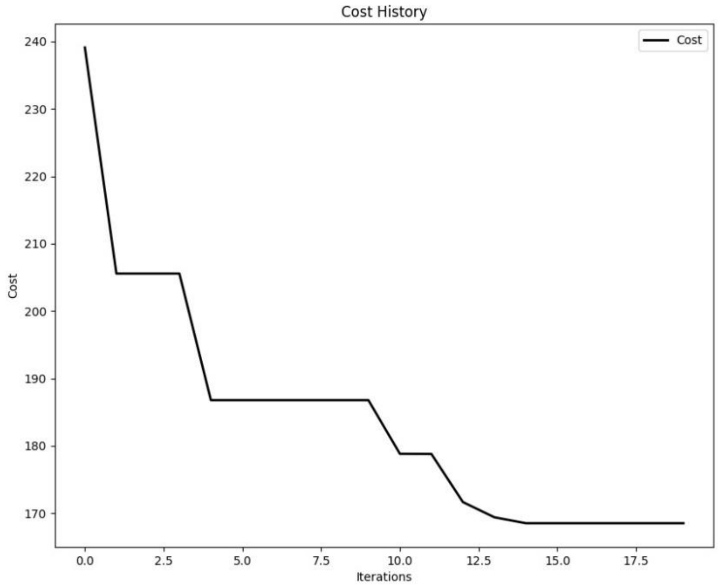

Fig. 12. PSO-FTSC cost history

This is the resulting traffic pattern in the intersection:

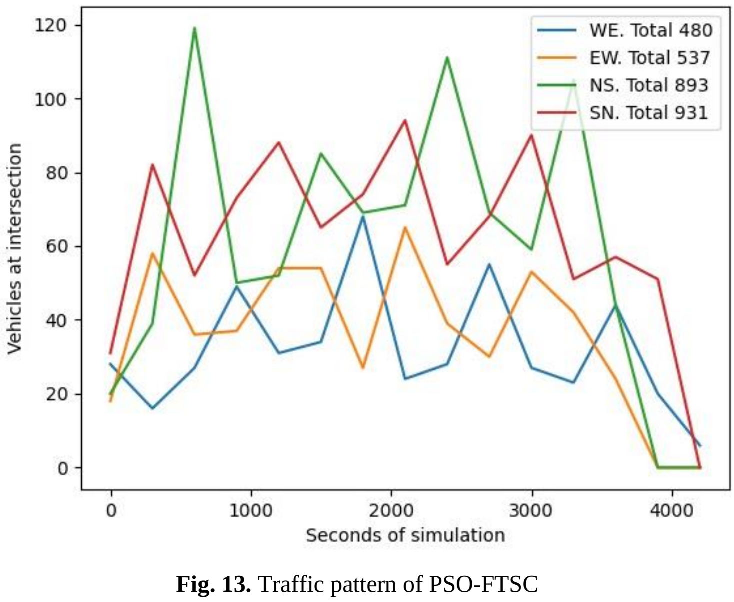

Fig. 13. Traffic pattern of PSO-FTSC

4.3 PSO-NN-ATSC

The PSO-NN-ATSC model optimized the 196 parameters for four neural networks using 6 agents through 10 iterations of PSO, creating the Mean Time Loss curve in Fig. 14.

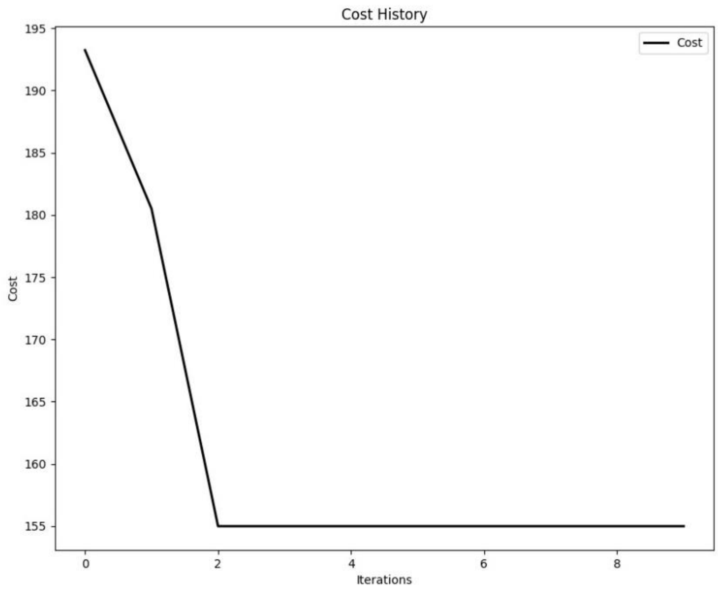

Fig. 14. PSO-NN-ATSC cost history

Unlike the other models, it is not practical to show the solution here, as it is an array of 196 parameters. The neural networks decide the green phase duration in each cycle, generating the traffic pattern in Fig 15.

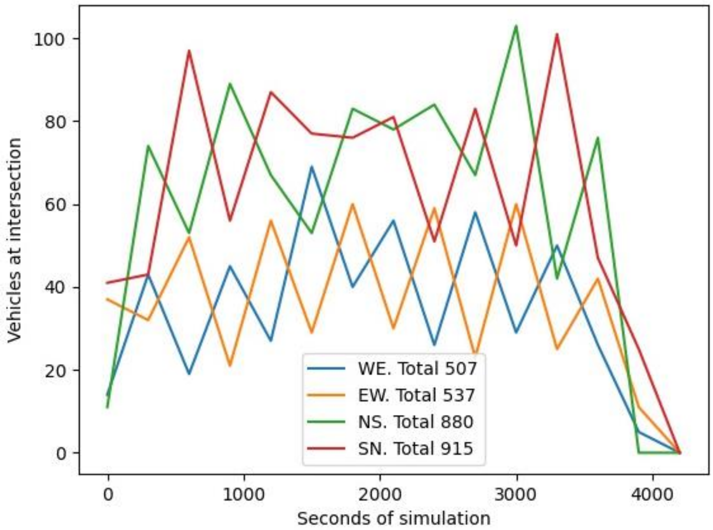

Fig. 15. Traffic pattern of PSO-NN-ATSC

4.4 Comparison between models

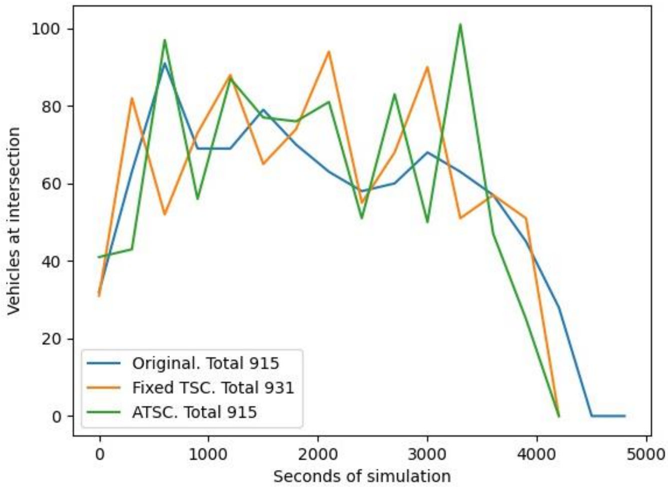

Fig. 16. Comparison of Traffic Pattern in the NS side

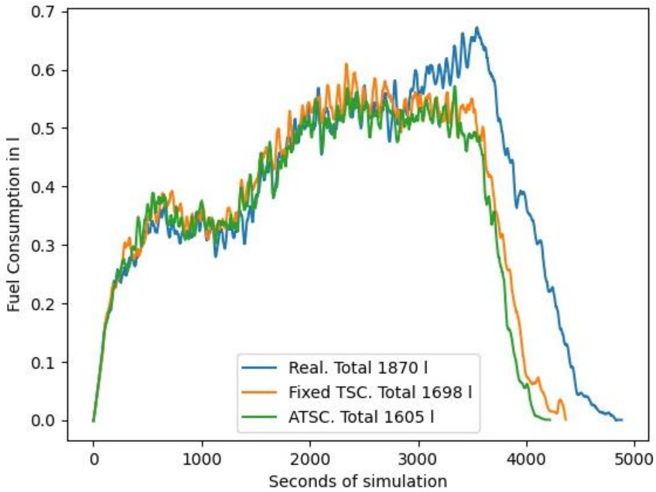

Fig. 17. Comparison of Fuel Consumption pattern

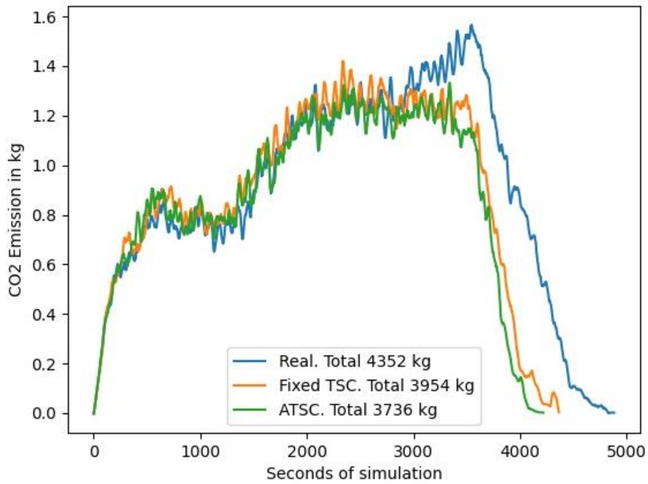

Fig. 18. Comparison of CO_2 Emission pattern

Table 7. Overall results

Metric	Baseline model	PSO-Fixed TSC	PSO NN-ATSC
Number of vehicles	5244	5244	5244
Last vehicle appearance time	3600 seconds	3600 seconds	3600 seconds
Last vehicle finishes on	4885 seconds	4367 seconds	4221 seconds
Simulation software running time*	55.83 seconds	49.76 seconds	47.04 seconds
Mean Time Loss	204 seconds	169 seconds	155 seconds
Estimated Total Fuel Consumption	1870 litters	1698 litters	1605 litters
Estimated Total CO_2 Emission	4352 kilograms	3954 kilograms	3736 kilograms

* These runtime results depend on hardware specifications.

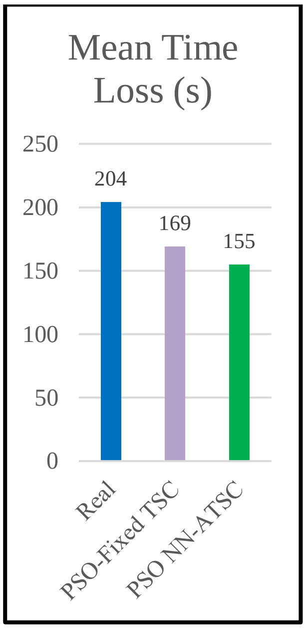

Fig. 19.. Mean Time Loss differences

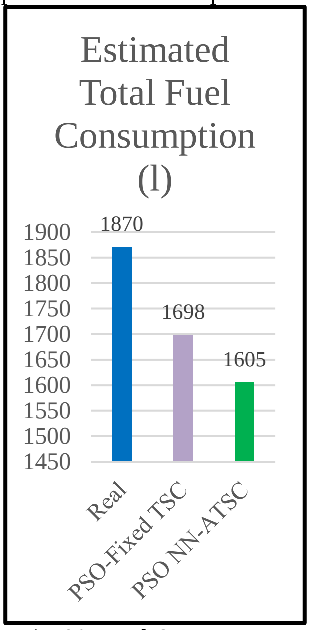

Fig. 20.. Fuel Consumption differences

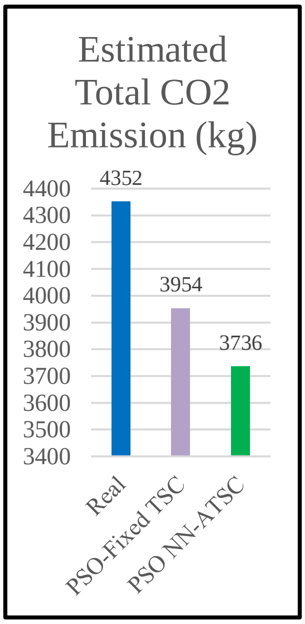

Fig. 21. Fuel Consumption differences

5 Discussion

The results shown here are subject to SUMO driving algorithm and thus, the numbers can change when analyzing specific driving patterns of drivers in Dominican Republic. The variability can be huge when adding human behavior concerns to the variable.

The PSO-FTSC was able to reduce 17% of the time loss from the original model, reducing up to 35 seconds of mean loss time per vehicle. Apart from that, it also reduced 172 litters of fuel consumption and 398 kilograms of CO_2 emission.

The PSO NN-ATSC on its side was able to reduce the numbers even further, taking off 24% of mean time loss (49 s) per vehicle. The difference in fuel consumption between the real TSC and PSO NN-TSC is 265 litters, and in CO_2 emission is 616 kilograms.

A mean time loss reduction of 49 seconds means that in average, each of those 5244 vehicles could arrive 49 seconds faster in their short path through the simulation. That is: the **PSO NN-TSC saved 256,956 s = 71.38 h of vehicle time** for the total 5244 vehicles.

Compared to the PSO-Fixed TSC, the PSO NN-TSC saved 14 s of Mean Time Loss. For the total of 5244, this scales up to 71,3416 = 20.39 h of vehicle time.

The perceived traffic congestion is observed from the simulation in Figures 22-27.

Fig. 22. Real model traffic congestion 1

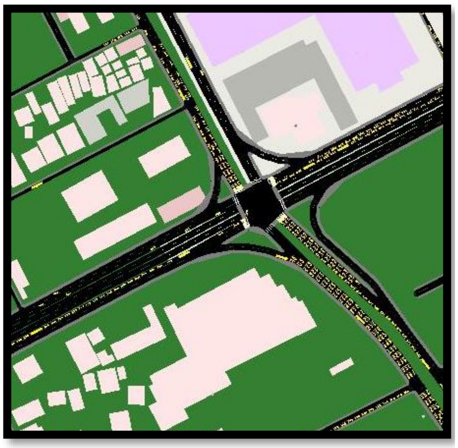

Fig. 23. Real model traffic congestion 2

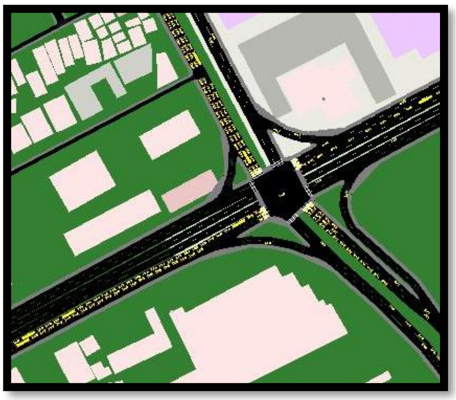

Fig. 24. PSO-Fixed TSC traffic congestion 1

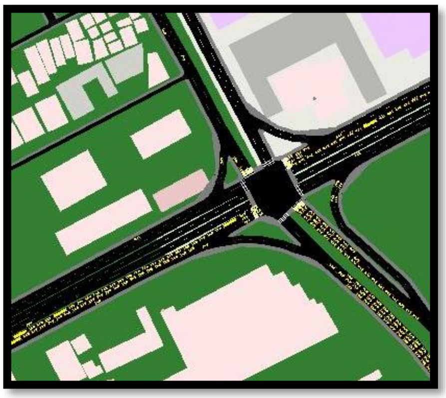

Fig. 25. PSO-Fixed TSC traffic congestion 1

Fig. 26. PSO NN-ATSC traffic congestion 1

Fig. 27. PSO NN-ATSC traffic congestion 2

- In the real model, a constant congestion is perceived.
- In the PSO-FTSC, larger congestions are accumulated sometimes but they disappear once the correspondent side turns green. However, after the green phase, these vehicles can cumulate again on the next intersection.
- In the PSO-NN-ATSC, the congestions seem equal to the PSO-Fixed TSC, but the road network is able to receive those vehicles without causing more congestion after the intersection.

6 Conclusions

Both the proposed models can improve the current traffic situation in Santo Domingo, with PSO-NN-ATSC having the best results.

Creating an ATSC supposes the investment in Internet of Things (IoT) for vehicle count system, data processing and traffic signal controls. Because of this, providing stakeholders with a PSO-FTSC could temporarily improve the transit in Santo Domingo as a preliminary solution while the infrastructure for a more complex system is prepared. However, as mentioned by Skanda Vivek [44], there is always the possibility of having events such as sports, concerts. The city needs to be prepared even to emergencies like evacuations, accidents and more. In such events, the traffic is unpredictable and might present a very severe variability. Because of this, having an Adaptive Traffic Signal Control is always recommended.

7 Future Work

This work can be extended by researching real traffic data of Santo Domingo and modelling it in SUMO, measuring the performance of both, PSO-FTSC and PSO-NN-ATSC on this data.

Consecutively, the models can be applied. To implement the PSO NN-ATSC in real world, there are, among others, two ways in which the data currently generated by SUMO could be collected in real-time:

- Sensors (e.g., smart cameras)
- Data from a satellite data provider (e.g., Google Maps)

Additionally, the PSO-NN-ATSC can be applied to multiple intersections simultaneously. In this project, although only one intersection was optimized, the overall time loss of the road network was taken as a cost function, guaranteeing that this intersection was optimized to its best to improve the overall traffic. However, bigger improvements could be achieved if multiple signalized intersections are considered. For this, PSO can be scaled to optimize parameters to all intersections parallelly.

8 ACKNOWLEDGEMENTS

This work was presented in dissertation form in fulfilment of the requirements for the MSc in Robotics Engineering for the student Eddy Martínez from the Robotics Lab, School of Mathematics, Computer Science and Engineering, Liverpool Hope University.

9 References

1. Q. Dai, "Traffic Flow Optimization," Mitacs, [Online]. Available: https://www.mitacs.ca/en/projects/traffic-flow-optimization-1. [Accessed 18 September 2021].
2. R. Pérez Reyes, "'Tapones' provocan RD$48 MM en pérdidas," 11 May 2016. [Online]. Available: https://listindiario.com/economia/2016/05/11/418788/tapones-provocan-rd-48-mm-en-perdidas. [Accessed 5 December 2021].
3. R. A. Silva, Z. Adelman, M. M. Fry and J. J. West, "The Impact of Individual Anthropogenic Emissions Sectors on the Global Burden of Human Mortality due to Ambient Air Pollution," *Environ Health Perspect,* vol. 124, no. 11, pp. 1176-1784, November 2016.
4. S. S. S. M. Qadri, M. A. Gökçe and E. Öner, "State-of-art review of traffic signal control methods: challenges and opportunities," *European Transport Research Review,* vol. 12, no. 55, 2020.
5. L. Adacher, "A Global Optimization Approach to Solve the Traffic Signal Synchronization Problem," *Procedia - Social and Behavioral Sciences,* vol. 54, pp. 1270-1277, 2012.
6. R. P. Alvarez GIl, Z. C. Johanyák and T. Kovács, "Surrogate Model based Optimization of Traffic Lights," *International Journal of Artificial Intelligence,* vol. 16, no. 1, April 2018.
7. D. Hartanti, R. N. Aziza and P. Siswipraptini, "Optimization of smart traffic lights to prevent traffic congestion using fuzzy logic," *TELKOMNIKA (Telecommunication, Computing, Electronics and Control),* vol. 17, pp. 320-327, February 2019.
8. W. C. Tchuitcheu, C. Bobda and M. J. H. Pantho, "Internet of smart-cameras for traffic lights optimization in smart cities," *Internet of Things,* vol. 11, September 2020.
9. S. Araghi, A. Khosravi and D. Creighton, "Intelligent cuckoo search optimized traffic signal controllers for multi-intersection network," *Expert Systems with Applications,* vol. 42, no. 9, pp. 4422-4431, 1 June 2015.
10. S. Araghi, A. Khosravi, D. Creighton and S. Nahavandi, "Influence of meta-heuristic optimization on the performance of adaptive interval type2-fuzzy traffic signal controllers," *Expert Systems with Applications,* vol. 71, pp. 493-503, 1 April 2017.
11. J. García-Nieto, A. C. Olivera and E. Alba, "Optimal Cycle Program of Traffic Lights With Particle Swarm Optimization," *IEEE Transactions on Evolutionary Computation,* vol. 17, no. 6, pp. 823-839, December 2013.

12. K. Malecki, P. Pietruszka and S. Iwan, "Comparative Analysis of Selected Algorithms in the Process of Optimization of Traffic Lights," *Asian Conference on Intelligent Information and Database Systems,* vol. 10192, pp. 497-506, 2017.
13. H. Zhang, H. Yuan, Y. Chen, W. Yu, C. Wang, J. Wang and Y. Gao, "Traffic Light Optimization Based on Modified Webster Function," *Journal of Advanced Transportation,* vol. 2021, 2021.
14. D. Yu, X. Tian, X. Xing and S. Gao, "Signal Timing Optimization Based on Fuzzy Compromise Programming for Isolated Signalized Intersection," *Mathematical Problems in Engineering,* vol. 2016, 2016.
15. P. T. M. Nguyen, B. N. Passow and Y. Yang, "Improving anytime behavior for traffic signal control optimization based on NSGA-II and local search," in *2016 International Joint Conference on Neural Networks (IJCNN),* Vancouver, 2016.
16. L. Zheng, C. Xu, P. J. Jin and B. Ran, "Network-wide signal timing stochastic simulation optimization with environmental concerns," *Applied Soft Computing,* vol. 77, pp. 678-687, April 2019.
17. M. Bernas, B. Placzek and J. Smyla, "A Neuroevolutionary Approach to Controlling Traffic Signals Based on Data from Sensor Network," *Sensors,* vol. 19, no. 8, p. 1776, 2019.
18. J. J. Sanchez-Medina, M. J. Galan-Moreno and E. Rubio-Royo, "Traffic signal optimization in "La Almozara" district in Saragossa under congestion conditions, using genetic algorithms, traffic microsimulation, and cluster computing," *IEEE Transactions on Intelligent Transportation Systems,* vol. 11, no. 1, pp. 132-141, March 2010.
19. D. Inoue, A. Okada, T. Matsumori, K. Aihara and H. Yoshida, "Traffic signal optimization on a square lattice with quantum annealing," *Scientific Reports,* vol. 11, no. 3303, 2021.
20. J. Ferrer, M. López-Ibáñez and E. Alba, "Reliable simulation-optimization of traffic lights in a real-world city," *Applied Soft Computing,* vol. 78, pp. 697-711, May 2019.
21. M. López-Ibáñez, J. Dubois-Lacoste, L. Pérez Cáceres, M. Birattari and T. Stüztzle, "The irace package: Iterated racing for automatic algorithm configuration," *Operations Research Perspectives,* vol. 3, pp. 43-58, 2016.
22. T. Chu, J. Wang, L. Codecà and Z. Li, "Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control," *IEEE Transactions on Intelligent Transportation Systems,* vol. 21, no. 3, pp. 1086-1095, March 2020.
23. C. Ozan, O. Baskan, S. Haldenbilen and H. Ceylan, "A modified reinforcement learning algorithm for solving coordinated signalized networks," *Transportation Research Part C: Emerging Technologies,* vol. 54, pp. 40-55, May 2015.
24. W. Genders and S. Razavi, "An Open-Source Framework for Adaptive Traffic Signal Control," *Journal On Transactions Of Intelligent Transport Systems,* vol. X, no. X, August 2019.
25. S. Samra, A. El-Mahdi and Y. Wada, "A Linear Time and Space Algorithm for Optimal Traffic-Signal Duration at an Intersection," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, no. 1, pp. 387-395, February 2015.

26. Q. He, R. Kamineni and Z. Zhang, "Traffic signal control with partial grade separation for oversaturated conditions," *Transportation Research Part C: Emerging Technologies,* vol. 71, pp. 267-283, October 2016.

27. M. Mehrabipour and A. Hajbabaie, "A Cell-Based Distributed-Coordinated Approach for Network-Level Signal Timing Optimization," *Computer-Aided Civil and Infrastructure Engineering,* vol. 32, no. 7, pp. 599-616, 21 June 2017.

28. H. Yan, F. He, X. Lin, J. Yu, M. Li and Y. Wang, "Network-level multiband signal coordination scheme based on vehicle trajectory data," *Transportation Research Part C: Emerging Technologies,* vol. 107, pp. 266-286, October 2019.

29. R. Mohebifard and A. Hajbabaie, "Optimal network-level traffic signal control: A benders decomposition-based solution algorithm," *Transportation Research Part B: Methodological,* vol. 121, pp. 252-274, March 2019.

30. H. Yu, R. Ma and H. M. Zhang, "Optimal traffic signal control under dynamic user equilibrium and link constraints in a general network," *Transportation Research Part B: Methodological,* vol. 110, pp. 302-325, April 2018.

31. C. Yu, W. Ma, K. Han and X. Yang, "Optimization of vehicle and pedestrian signals at isolated intersections," *Transportation Research Part B: Methodological,* vol. 98, pp. 135-153, April 2017.

32. World Population Review, "Dominican Republic Population 2021 (Live)," World Population Review, 2021. [Online]. Available: https://worldpopulationreview.com/countries/dominican-republic-population. [Accessed 13 October 2021].

33. Inter-American Development Bank, "Support Program for Mobility, Ground Transportation and Road Safety in the Dominican Republic (IADB-DR-L1132)," 3 March 2020. [Online]. Available: https://ewsdata.rightsindevelopment.org/projects/dr-l1132-support-program-for-mobility-ground-transportatio/. [Accessed 13 October 2021].

34. J. L. Ravelo, "Cities and NCDs: In Santo Domingo, cars cause more than congestion," Devex, 08 June 2018. [Online]. Available: https://www.devex.com/news/cities-and-ncds-in-santo-domingo-cars-cause-more-than-congestion-92895. [Accessed 13 October 2021].

35. D. J. Edelman, "Managing the Urban Environment of Santo Domingo, the Dominican Republic," *Current Urban Studies,* vol. 7, no. 1, pp. 76-142, March 2019.

36. J. Pérez Orsini, "Los diez puntos críticos del tránsito del Distrito Nacional en las horas pico," 11 September 2018. [Online]. Available: https://www.diariolibre.com/actualidad/ciudad/los-diez-puntos-criticos-del-transito-del-distrito-nacional-en-las-horas-pico-ID10641393. [Accessed 13 October 2021].

37. Instituto Tecnológico de Santo Domingo, "How to reduce vehicular traffic jams?," 12 April 2018. [Online]. Available: https://www.intec.edu.do/en/prensa/notas-de-prensa/item/como-reducir-los-embotellamientos-vehiculares. [Accessed 13 October 2021].

38. Listín Diario, "Intrant instala cámaras tipo sensores para detección de vehículos," 21 June 2020. [Online]. Available: https://listindiario.com/la-republica/2020/06/21/622889/intrant-instala-camaras-tipo-sensores-para-deteccion-de-vehiculos. [Accessed 13 October 2021].

39. E. Rivera Almodóvar, "Los semáforos inteligentes," 21 April 2021. [Online]. Available:

https://hoy.com.do/los-semaforos-inteligentes/. [Accessed 13 October 2021].

40. Google Maps, "Av. Winston Churchill & Av 27 de Febrero," 13 October 2021. [Online]. Available: https://www.google.com/maps/place/Av.+Winston+Churchill+%26+Av+27+de+Febrero,+Santo+Domingo,+Dominican+Republic/@18.4624816,-69.9406238,15.96z/data=!4m5!3m4!1s0x8ea5620a17ad443d:0x38608525ec1b70ab!8m2!3d18.4627037!4d-69.9361725. [Accessed 13 October 2021].
41. Instituto Nacional de Tránsito y Transporte Terrestre (INTRANT), "Tiempo de Espera de los Vehículos en Intersecciones Semaforizadas," Instituto Nacional de Tránsito y Transporte Terrestre (INTRANT), Santo Domingo, 2017.
42. Green Signals Consulting Limited, "UTC/SCOOT and Pedestrian Pushbuttons," Green Signals Consulting Limited, [Online]. Available: http://www.greensignals.co.uk/news/utcscoot-and-pedestrian-pushbuttons/. [Accessed 15 October 2021].
43. J. Kennedy and R. Eberhart, "Particle swarm optimization," in *International Conference on Neural Networks*, Perth, 1995.
44. S. Vivek, "How To Simulate Traffic On Urban Networks Using SUMO," July 2018. [Online]. Available: https://towardsdatascience.com/how-to-simulate-traffic-on-urban-networks-using-sumo-a2ef172e564. [Accessed 17 October 2021].
45. German Aerospace Center, "SUMO User Documentation," German Aerospace Center, 22 September 2021. [Online]. Available: https://sumo.dlr.de/docs/index.html. [Accessed 13 October 2021].
46. P. Alvarez Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner and E. Wiessner, "Microscopic Traffic Simulation using SUMO," in *21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, 2018.