Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Improved Snake Optimizer Approach to Enhance Traffic Signal Timing Optimization by SUMO

Kadda Zerrouki[1], Siham Kouidri[2]
[1] *GeCoDe Laboratory, Department of Computer Science, University of Saida Dr. Moulay Tahar, Saida 20000, Algeria*
(e-mail: zerrouki.kadda.2023@gmail.com)
[2] *GeCoDe Laboratory, Department of Computer Science, University of Saida Dr. Moulay Tahar, Saida 20000, Algeria*
(e-mail : skouidri2023@gmail.com)

ARTICLE INFO

Received: 31 Dec 2024

Revised: 20 Feb 2025

Accepted: 28 Feb 2025

ABSTRACT

Introduction

Drag on city traffic has escalated to be a pivotal problem in almost all cities in the world, and commuter travel time, fuel consumption and emission have also soared. Fixed-time signal control systems have been traditionally used in traffic signal control, but their optimality highly depends on the time of day they are applied and they are not capable of real-time traffic control. In order to overcome these drawbacks, intelligent and adaptive traffic control approaches are being explored by both scientists and traffic developers.

Objective

The purpose of this research is to create and assess a dynamic traffic light control system by utilizing Snake Optimization (**SO**), a novel meta-heuristic algorithm. In order to minimize vehicle queues, cut down on delays, and enhance overall traffic flow, the main goal is to examine how well **SO** optimizes traffic signal timings based on real-time traffic data. by contrasting conventional static control techniques and meta-heuristics with the **SO**-based dynamic approach.

Proposed Method

To find the best traffic light phase durations at intersections, the suggested approach uses the Snake Optimization (**SO**) algorithm, a bio-inspired meta-heuristic technique. Real traffic data from the Algerian city of Algiers, which depicts realistic traffic volumes and patterns, is used to test the method. The **SUMO** (Simulation of Urban Mobility) simulator, an open-source program popular for simulating microscopic traffic, is used to construct the experimental framework. Traffic scenario modeling, control algorithm implementation, and performance evaluation in a simulated urban setting are all made possible by **SUMO**. The following steps are part of the optimization process: gathering of data, using the **SUMO** simulator to model traffic flow and intersections, incorporating the **SO** algorithm to modify traffic light timings in real time in response to inputs, evaluation of performance using metrics like vehicle throughput, average waiting time, reduction of CO2 emission and total travel time, and evaluation of the benefits obtained by the **SO**-based approach through comparison with a baseline static control system and **ACO-DSOS** meta-heuristic.

Conclusion

According to the simulation results, the suggested approach, which is based on the Snake Optimization algorithm, performs better than other meta-heuristics and conventional static control systems. The algorithm's effectiveness in controlling traffic flows is validated by quantitative comparisons. According to the study's findings, **SO** is a practical and successful strategy for maximizing traffic light control in practical settings.

Keywords: Traffic Flow, Average Waiting Time, Traffic Light Program, Snake Optimizer (**SO**), Green time, Travel time, CO2 emission, **SUMO**.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

INTRODUCTION

Planning for urban transportation is one of the biggest problems facing contemporary cities. A number of issues, including pollution, traffic congestion, safety concerns, and parking challenges, can result from inadequate traffic light planning [1], [2]. Rapid population growth and the rise in private automobiles per resident, along with uneven urbanization and inadequate infrastructure, make the issues large cities face worse. Urban transport system failures lead to serious economic difficulties as well as a host of social issues that impact day-to-day living. With time, urban traffic management gets more complicated. Vehicles and people must be able to travel swiftly, comfortably, affordably, and safely. Congestion zones in the transportation network are frequently represented by intersections, where several roads converge. As a result, the entire road system may benefit from any optimization carried out at these crucial locations. At intersections, traffic flow is controlled by traffic light management systems. These signaling device-equipped intersections are essential components of urban infrastructure. The order and length of the crossing phases are orchestrated by the various colored lights. Two types of traffic light systems are currently in use: those that operate for set periods of time and those that adjust in real time to the flow of traffic [3].

All parameters, such as cycle length, phase distribution, and sequence order, are pre-programmed when controlling traffic lights with preset durations; the number of cars or pedestrians on the road is not taken into consideration. This method is not very adaptable to traffic fluctuations. To overcome these limitations, various traffic management strategies have been developed. This system adjusts cycle duration, phase distribution, and sequence order in real-time based on collected data and prior knowledge.

Traffic management becomes even more complex when both immediate changes in traffic conditions and dynamic traffic parameters must be considered in designing signal sequences to optimize the efficiency of the control system. This complexity makes it difficult to calculate optimal phase durations within a short time frame using traditional optimization methods. To address the need for real-time solutions, the use of advanced optimization algorithms enables more efficient management of intersection traffic signals in urban environments [4].

In this study, the control of traffic lights is considered an optimization problem. Optimization is simply a process of finding the best solution for a problem or system. The main objective of optimization is to find the optimal traffic light parameters that minimize the average vehicle waiting time in the studied area.

The Snake Optimizer (**SO**) is a powerful metaheuristic algorithm that offers several advantages when applied to traffic light optimization. By mimicking the adaptive behavior of snakes, **SO** efficiently balances exploration (searching for better cycle programs) and exploitation (refining the best solutions), leading to improved traffic flow management. This ensures the algorithm does not get trapped in local optima, making it suitable for dynamic and complex urban road networks, this iterative improvement process enhances traffic flow efficiency and reduces vehicle wait times [9].

This study is structured as follows: Section 2 a review of related works in the literature is presented. In Section 3, the fundamental principles and overall structure of the **SO** algorithm are discussed. Section 4 introduces the proposed approach for optimizing traffic light cycle programs. The experimental setup used to assess the effectiveness of the optimization technique is detailed in Section 5. Finally, Section 6 presents the conclusion of the framework.

RELATED WORK

Traffic light optimization based on metaheuristics is a compelling research area that addresses the complex interplay of traffic dynamics, offering solutions to urban congestion through sophisticated computational algorithms. Metaheuristics, known for their capability to tackle non-linear, multi-variable, and multi-objective problems, provide a diverse range of strategies for optimizing traffic signal timings.

Authors in [12] have shown robust performance in optimizing traffic light scenarios, especially when dealing with large solution spaces. For instance, Yektamoghadam et al. demonstrate the use of GAs combined with teaching-learning-based optimization to improve vehicle throughput by 10%, highlighting their efficacy in real-world implementations like the crossroads in Mashhad, Iran. Another popular method for traffic signal optimization.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

García-Nieto et al. [13] evaluated PSO in metropolitan areas like Bahía Blanca and Málaga, achieving substantial enhancements in vehicle destination counts and reducing journey times significantly compared with traditional cycle programs.

Ant Colony Optimization inspired by the natural behavior of ants, is applied to the dynamic control of traffic lights. Liao et al. integrated a look-ahead mechanism to predict traffic flow using IoV systems, effectively reducing both waiting times and carbon emissions, notably in scenarios with high traffic flow [14].

Differential Evolution (DE) stands out for its simplicity and efficiency, with studies like Lin et al. combining DE with fuzzy control to produce a multi-objective optimization model that minimizes average queue lengths and CO_2 emissions [15].

Metaheuristic methods provide adaptable and powerful solutions to urban traffic challenges. As research progresses, integrating these algorithms with cutting-edge technologies such as connected and autonomous vehicles is essential to enhancing traffic management systems potency and sustainability further. These explorations into the use of metaheuristics underscore their pivotal role in managing traffic system complexities.

Integrating PSO and SA for traffic light optimization in a **SUMO** simulation framework presents a compelling strategy to balance global exploration and local exploitation. This hybrid approach not only improves overall traffic flow by reducing average delay and queue lengths, but it also offers a flexible method adaptable to multi-intersection networks and varying traffic conditions.

The study in [17] proposes a hybrid method combining Ant Colony Optimization (**ACO**) and Discrete Symbiotic Organism Search (**DSOS**) to improve traffic flow at intersections in Algiers, Algeria. **ACO** is inspired by ant foraging behavior, while **DSOS** is based on symbiotic relationships in ecosystems. Results indicate a 20% reduction in the mean number of queued vehicles compared to fixed-time control methods. The main limitation of this work lies in its focus on a fixed number of intersections. As the number of intersections increases, the system's complexity also grows.

Table 1: Comparative study.

Title	**Optimization method**	**Advantages and disadvantages**
Time Optimization for Traffic Signal Control Using Genetic Algorithm [18] [19]	Genetic Algorithm (**GA**)	-Adaptive and efficient in real time, but extension time is too low; 5s is insufficient for high congestion. Also, the model focuses on a single isolated intersection, which does not account for interactions between multiple intersections in a network.
On Tuning the Particle Swarm Optimization for Solving the Traffic Light Problem [20]	Particle Swarm Optimization (PSO)	Significant gains via PSO parametric tests. But neglected coordination with other intersections causes traffic jams, and real-time traffic fluctuations (e.g., peak hours..) are not dynamically incorporated into the optimization model.
Analysis of Simulated Annealing to Decongest Traffic in a Multi-Road Coordinated Intersection [21]	Simulated Annealing (SA)	SA avoids local optima. It focuses on minimizing vehicle waiting time, another important objective such as CO_2 emissions are not considered.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

A hybrid algorithm based on GWO and goa for cycle traffic light timing optimization [22]	Gray Wolf Optimizer (GWO)+ GOA	- The hybrid algorithm leverages the combined strengths of GWO and GOA. - The complexity of the algorithm is not mentioned.
Hybridization of Racing Methods with Evolutionary Operators for Simulation Optimization of Traffic Lights Programs [23]	IRACE + DE/GA	Efficient schedule optimization. Computationally expensive due to extensive simulations also the experiments appear to focus on small size intersections
Hybrid Fuzzy Genetic Algorithm for an Adaptive Traffic Signal System Lights Programs [24]	Fuzzy logic controller & genetic algorithm	Minimization of the number of vehicles remaining after the green light. The paper does not compare the hybrid method against optimization methods
ACO-DSOS Hybrid Approach to Enhance Traffic Signal Optimization [17]	**ACO** & **DSOS**	Minimize vehicle waiting times. Its focus on a fixed number of intersections

SNAKE OPTIMIZER FOR TRAFFIC LIGHT OPTIMIZATION

This section describes our optimization approach proposed for the optimal cycle programs of traffic lights. It details the solution encoding, the fitness function, and finally the global optimization procedure. Previous to this, basic notions about the **SO** algorithms are given.

1. Snake In Nature

As illustrated in Figure 1, snakes are remarkable reptiles that stand out by their lengthy, legless bodies. As ectothermic vertebrates, they, like all other squamates, rely on external heat sources to regulate their body temperature. Most snake species are distinguished by possessing heads that are incredibly flexible and multi-jointed. They are able to consume prey bigger than their own heads due to this unique adaptation.

Snakes' unusual mating habits are among the most fascinating features of their behavior. Multiple mating, seasonal breeding, and a variety of reproductive modes are among the reproductive traits displayed by female snakes. In addition to controlling physical characteristics by controlling body temperature and selecting nesting locations, they can also affect genetic outcomes through mate selection and sperm competition. However, male snakes compete fiercely for mates, acting in ways like mate guarding and even imitating females. Which male eventually succeeds in mating is largely determined by physical encounters [25][26].

Figure 1: Snake in nature.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. Snake Optimizer (SO)

The Snake Optimizer (**SO**) is a metaheuristic optimization algorithm inspired by the hunting behavior of snakes. It mimics how snakes move, search for prey, and adapt their movement strategies to efficiently explore and exploit the search space.

The Snake Optimizer (**SO**) is modeled after the reproductive and survival strategies of snakes. When conditions are favorable, such as low temperatures and abundant food, snakes engage in mating. However, in less ideal situations, their primary focus shifts to searching for or consuming available food. Drawing from this natural behavior, the **SO** algorithm defines its search process through two key phases: **exploration** and **exploitation** [9].

Exploration: corresponds to environmental conditions where essential factors, such as a cold habitat and food, are absent. In this situation, the snake focuses solely on searching for food within its surroundings [9].

Exploitation: this phase consists of multiple transition stages aimed at achieving a more optimal global solution. When food is present but temperatures remain high, snakes prioritize consuming the available food. However, if both food and a cold environment are present, mating occurs. The mating process follows two possible scenarios: **fight mode** or **mating mode**. In fight mode, males compete to secure the best female, while females seek the strongest male. In mating mode, pairings take place based on food availability. If mating occurs within the search space, there is a chance that the female will lay eggs, which will eventually hatch into new snakes [9].

3. Mathematical Model of Snake Optimizer

The SO algorithm uses mathematical equations to simulate these behaviors.

Exploration Phase (Position Update)

During exploration, snakes move randomly to discover better solutions. The position update is defined as [9]:

$$\mathrm{X}_i^{t+1} = \mathrm{X}_i^t + \alpha.\mathrm{R}.(X_{best} - \mathrm{X}_i^t) \quad \textbf{(1)}$$

Where

- X_i^t is the best solution of the i^{th} Snake at iteration t
- X_{best} is the best solution found so far.
- α is an exploration coefficient controlling search diversity.
- R is a random number in $[-1, 1]$ ensuring randomness.

Exploitation Phase (Position Update)

When food is found, snakes refine their movements to exploit the best solutions. The update equation is:

$$X_i^{t+1} = X_i^t + \beta.(X_{best} - X_i^t) + \gamma.R \quad \textbf{(2)}$$

Where

- β is the exploitation coefficient, guiding the snake toward optimal solutions.
- γ introduces small perturbations to avoid premature convergence.

Mating Process and Offspring Generation

If environmental conditions are ideal, snakes reproduce. The probability of mating is given by:

$$P_m = \frac{1}{1 + e^{-(F_i - F_{avg})}} \quad \textbf{(3)}$$

Where

- F_i is the fitness of snake i.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

- F_{avg} is the average fitness of the population. A higher P_m increases the chance of reproduction.

New offspring positions are calculated as:

$$X_{offspring} = \lambda X_{male} + (1 - \lambda) X_{female} \qquad (4)$$

Where

- λ is a weighting factor controlling the contribution of parents. This process increases population diversity and enhances solution quality.

The Snake Optimizer algorithm is described as follows:

Algorithm 1 : Snake Optimizer Algorithm
1: Initialize Problem Setting *(Dim, UB, LB, and Pop_Size(N), Max_Iter(T), Curr_Iter t)*
2: Initialize the population randomly
3: Divide population N to 2 equal groups N_m and N_f
4: **while** ($t \leq T$) **do**
5: Evaluate each group N_m and N_f
6: Find best male *f_best,m*
7: Find best female *f_best,f*
8: Define *Temp*
9: Define food Quantity Q
10: **if** ($Q < 0.25$) **then**
11: Perform exploration
12: **else if** ($Q > 0.6$) **then**
13: Perform exploitation
14: **else**
15: **if** ($rand > 0.6$) **then**
16: Snakes in Fight Mode
17: **else**
18: Snakes in Mating Mode
19: Change the worst male and female
20: **end if**

4. Problem Definition and Formulation

An urban traffic scenario primarily consists of intersections, traffic signals, roads, and vehicles following their predefined routes. Traffic lights are located at intersections and control the flow of vehicles by following their programs of color states and cycle times. In the studied area, all traffic lights located at the same intersection are governed by a common and fixed program.

The Snake Optimizer is inspired by the adaptive and agile movement strategies of snakes, leveraging a dynamic balance between exploration and exploitation.

In this context, our main objective is to develop optimized cycle plans for all traffic lights within a selected urban area. These plans determine the optimal duration for each intersection. Additionally, they must synchronize the lights at adjacent intersections to improve overall traffic flow while ensuring compliance with traffic regulations. In our study, a **SO** solution is represented by a pair of integers indicating the green light duration for each direction (East-West, North-South). The size of the **SO** solution, or the number of intersections considered, depends on the selected study area.

An example of this mechanism can be observed in **Figure 2**. Implementing the snake optimization algorithm across several intersections presents unique challenges and opportunities. The algorithm can be designed to coordinate

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

signal timings among multiple intersections, ensuring that traffic flows smoothly through interconnected areas. This coordination is essential for reducing congestion and improving travel times across urban networks.

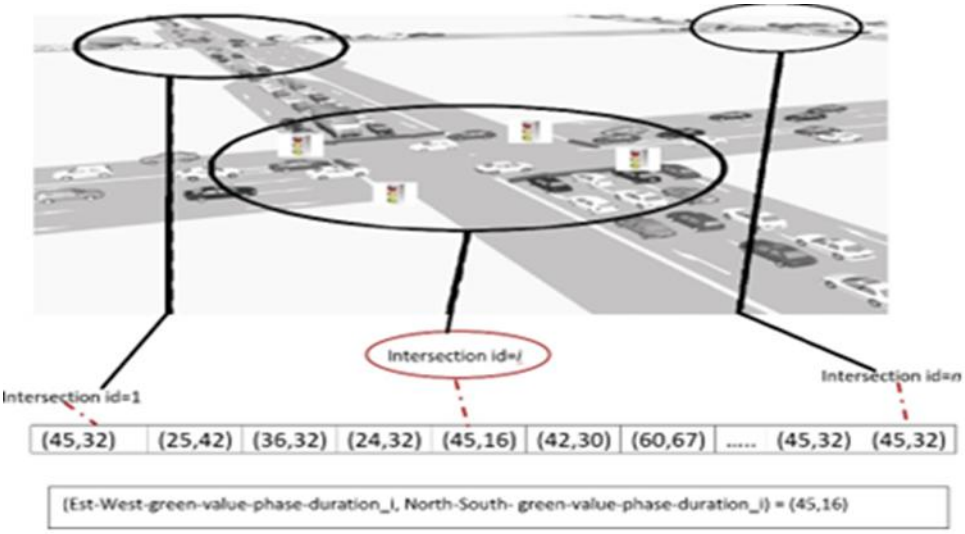

Figure 2: Traffic Light signal instance SO solution number j: TLS-SO-j_

1. Fitness function

To assess each cycle program generated by our SO, the following fitness function is applied, taking into account data gathered from events occurring throughout the simulation:

$$Min\ F(X_j) = \left(\frac{1}{N}\right)\sum_{i=1}^{N} W_i \qquad (5)$$

Where:

X_j: decision variable for a specific intersection j;

W_i: waiting time of car i during a cycle in intersection j.

N : total number of cars during a cycle (if N = 0 then $F(X)$ = 0 means there is no waiting cars).

The main objective (Eq. (5)) is to minimize average waiting time of cars in a specific intersection.

Constraints

It is essential to consider the specific constraints. Below are some common constraints that may arise in traffic light scheduling problem instances:

Decision variable: X

$Phase(NS)_i$: green time phase for North-South in intersection i ;

$Phase(EW)_i$: green time phase for Est-West in intersection i ;

$X_i = [Phase(NS)_i, Phase(EW)_i]$

$X = [X_1, X_2, X_3, \ , X_n]$

Definition of cycle C:

$Cycle_i : C_i = Phase(NS)_i + Phase(EW)_i$

$V_{min} <[\ Phase(NS),\ Phase(EW)\ ]< V_{max}$ and $Cmin < C < Cmax$

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Where:

V_{min} = 15 and V_{max} = 50 , C_{min} = 30 and C_{max} = 100.

V_{min}, V_{max}: Minimum and maximum allowed light values;

C_{min}, C_{miax}: Minimum and maximum cycle time.

The proposed optimization approach consists of two steps: an optimization algorithm and a simulation procedure. In the first step, optimization is carried out using the SO algorithm, specifically adapted for optimizing traffic light cycle programs. Next step, simulations with SUMO are then performed to collect data and compute the fitness function for each solution.

5. General architecture: SO-SUMO

First, each individual (solution) in the population represents a snake, and its **position in the search space** represents a **possible solution** to the problem being optimized which is a vector initialized with a pair of random integer values within the range [15,100], representing (phase (NS), phase (EW)). The size of each particle corresponds to the number of intersections. This interval represents the range of possible durations (in seconds) during which a traffic light can remain green. The red-light duration is derived, while the yellow light duration is a fixed value at 3 second added to the green phase. We defined this interval based on several real traffic light schedules provided by the city of Algiers, Algeria. The population is divided int two equal populations, the first represents the male snake, and the second represents the female snake.

Secondly, the simulation process serves as a means to collect quantitative data for calculating the fitness and determining the global average vehicle wait time for all intersections. When **SO** generates a new solution Xnew using Formula 1, it is used to update the cycle program. Then, **SUMO** is launched to simulate the new scenario, incorporating new traffic lights. After the simulation, **SUMO** provides the essential global data required to calculate the fitness function. The process continues until a predefined number of iterations or a convergence criterion is met. The best solution is selected as the optimized traffic light schedule.

It is important to note that each new cycle program is statically loaded for each intersection and time period. Our aim is to develop optimized cycle programs for a specific intersection scenario and timetable. In this regard, real-world traffic light planners primarily require fixed cycle programs for specific areas and predefined time periods (such as peak hours). This necessity led us to focus on addressing this issue. Figure 3 illustrates the full optimization strategy employed in this study.

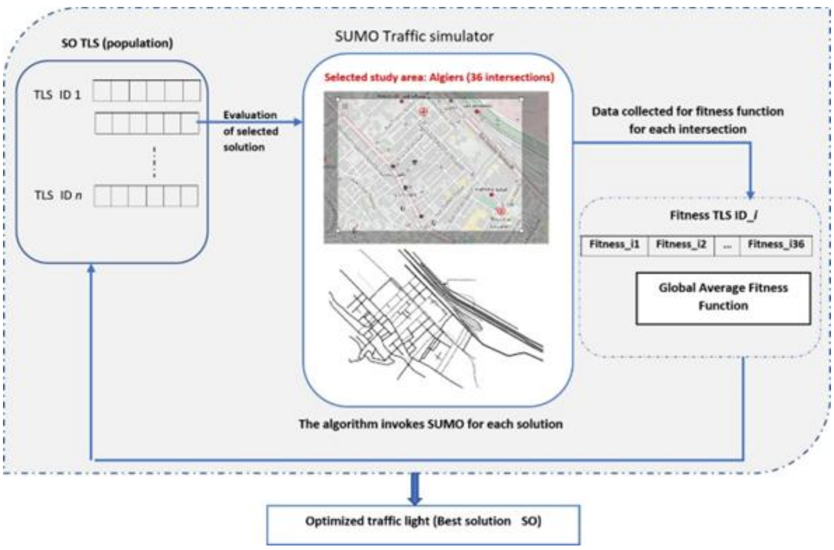

Figure 3: Optimization strategy **SO** for traffic lights.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Applying the Snake Optimizer to multi-intersection traffic signal optimization presents a promising approach characterized by adaptive exploration and exploitation in complex, dynamic environments. By integrating with a simulation environment like SUMO. The algorithm's ability to dynamically adjust to real-time feedback positions it as a potential tool for scalable, network-wide traffic management.

RESULTS

To develop an optimization solver capable of handling realistic and versatile urban areas, we designed a scenario using real data extracted from OpenStreetMap. Figure 4. displays the selected areas of Algiers city, along with their corresponding snapshots from OpenStreetMap, and **SUMO**. The number of studied intersections is 36 for this instance, and circulating 500 vehicles through the instance. The simulation time was set to 500s (iterations of microsimulation). These scenarios cover similar areas of approximately 1km².

The remaining parameters are summarized in Table 2. These parameters were set after preliminary executions of our approach with the Algiers instance.

Parameters	Value
Simulation time	500 s
Simulation area	1 km²
Number of vehicles	[500, 3000]
Number of intersections studied	36
Total population size	100
Female snake population size	50
Male snake population size	50
Snake size (number of traffic lights)	36

DISCUSSION

In this section, we present the results obtained from simulations of our proposed **SO** approach, static traffic light control approach and **ACO-DSOS** Approach [17]. A total of 63 simulations were conducted over a 7-day period for each approach, with 9 simulations per day, in order to cover a wide range of traffic conditions. The collected data includes key performance indicators such as CO_2 emissions, global average waiting time, and average travel time. The simulation scenarios were designed to represent various traffic intensities, ranging from smooth traffic during off-peak hours to heavy congestion during peak hours (e.g., from 12:00 PM to 2:00 PM). This diversity allows for an effective evaluation of the performance and adaptability of each approach within a realistic and dynamic urban context. The results are presented below.

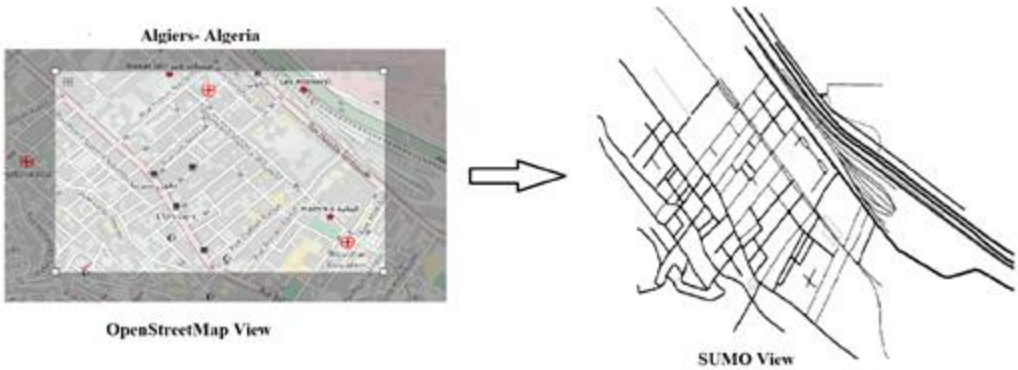

Figure 4: Process of creation of real-word instances under study.

After selecting our area, it is interpreted by OpenStreetMap tool and exported to **SUMO**.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Result1: Global average waiting time for each day

The data provided in figure 5 shows the average waiting times, calculated from real-time traffic data. It can be observed that the dynamic approach (**SO** approach) consistently results in lower global waiting times compared to the static approach and **ACO-DSOS** approach [17] for each day analysed.

These results suggest that dynamically adapting traffic light durations based on current traffic conditions can significantly reduce overall waiting times.

Figure 5: Global Average Waiting Time for each day

Result2: CO2 emission for each day

The graph compares the CO_2 emissions generated by the static approach, **ACO-DSOS** approach and the **SO** approach over different days. It clearly highlights a systematic reduction in emissions with the **SO** strategy compared to the static method and **ACO-DSOS** approach.

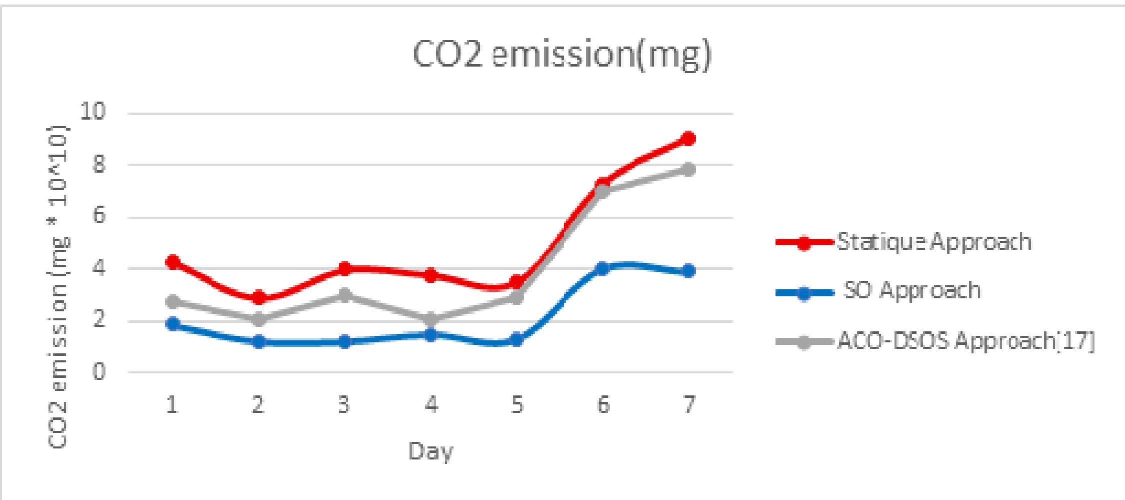

Figure 6: reduction of Co2 emissions for each day.

These results demonstrate the superiority of the **SO** Approach in optimizing traffic light cycles for reduced CO_2 emissions. By dynamically adapting to real-time traffic conditions, this method outperforms both: the static approach, which is rigid and outdated and the **ACO-DSOS** method, which although better than static, does not reach the same efficiency level.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Result3: Number of vehicles leaving the simulation

The number of cars that depart the traffic network (arrive at their destinations) throughout the experiment is now compared. The dynamics of the number of vehicles leaving the network (in Algiers) during the simulation of the optimal cycle programs discovered with SO, Fixed time, and ACO-DSOS [17] are then plotted in Fig. 7.

We can say that SO's cyclic programs record a greater number of autos arriving at their destination during simulation. Over the course of the simulation, the resource-intensive fixed cyclic program performs worse, even outperforming ACO-DSOS for the Algiers example. It is evident that the creation of traffic bottlenecks during simulation has a direct impact on how well cyclic systems manage them. However, the SO learning process makes it possible to create cyclic programs that can avoid lengthy lines, which enhances traffic flow in general.

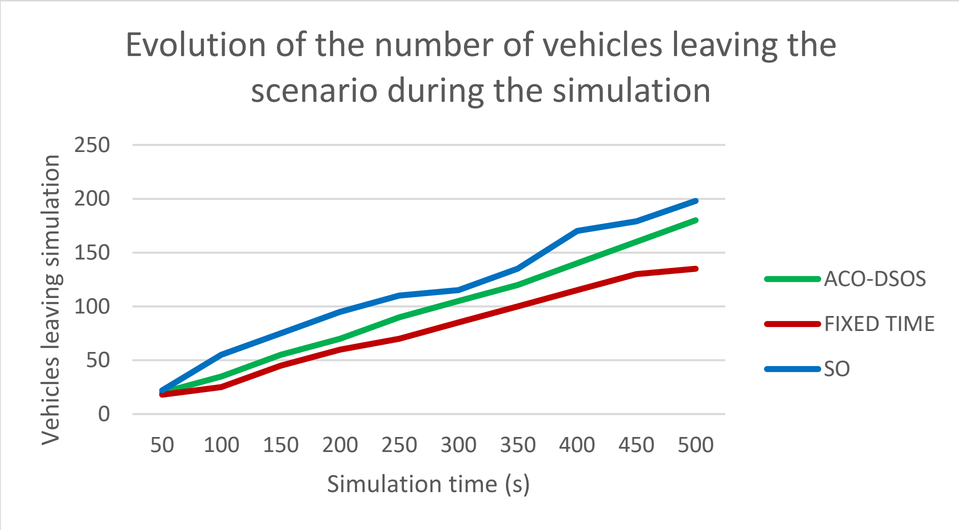

Figure 7: Number of vehicles leaving the simulation.

Result4: Progress of the best fitness values in 9 independent runs of SO tackling the Algiers city

Figure 8 is the executed plot showing the fitness over 500 iterations across 9 runs. The fitness values start higher and gradually decrease, simulating an optimization process when solving the Algiers instance. we can observe that, for all runs, our algorithm has practically converged after the first 200 iterations.

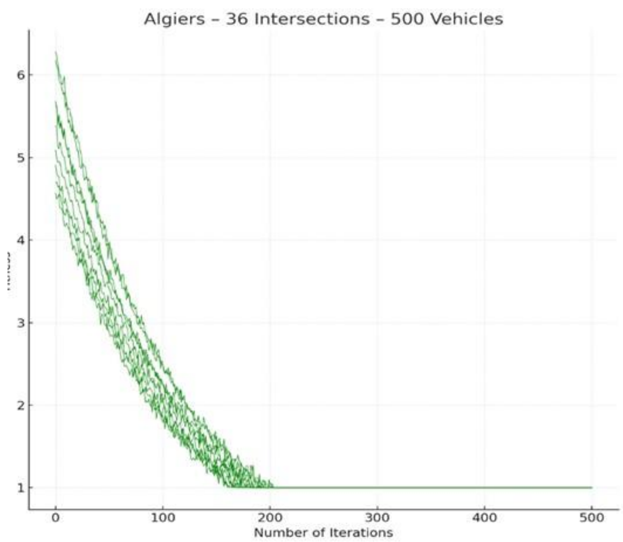

Figure 8: Progress of the best fitness values in 9 independent runs of SO

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

CONCLUSION

This study presents a Snake Optimizer (**SO**)-based traffic signal control system. In terms of lowering global average waiting times over several cycles, this method performs noticeably better than the fixed approach and ACO-DSOS approach. This demonstrates how adaptive signal control can improve urban traffic management, ease congestion, and boost productivity.

An approach that shows promise for adaptive exploration and exploitation in dynamic; complex environments is the Snake Optimizer's application to multi-intersection traffic signal optimization. Researchers can verify the system's performance against predetermined standards by integrating it with a simulation environment such as **SUMO**. As a potential tool for scalable, network-wide traffic management, the algorithm can adapt dynamically to real-time feedback. Further research should look into combining the Snake Optimizer with other methods to improve its performance even more. It should also confirm how well it works in cooperative multi-agent settings and under various traffic demand scenarios.

REFRENCES

[1] Spall, J. C., & Chin, D. C. (1997). Traffic-responsive signal timing for system-wide traffic control. Transportation Research Part C: Emerging Technologies, 5(3-4), 153-163.
[2] McCrea, J., & Moutari, S. (2010). A hybrid macroscopic-based model for traffic flow in road networks. European Journal of Operational Research, 207(2), 676-684.
[3] Zheng, X., & Recker, W. (2013). An adaptive control algorithm for traffic-actuated signals. Transportation Research Part C: Emerging Technologies, 30, 93-115.
[4] Garcia-Nieto, J., Olivera, A. C., & Alba, E. (2013). Optimal cycle program of traffic lights with particle swarm optimization. IEEE Transactions on Evolutionary Computation, 17(6), 823-839.
[5] Latchoumi, T. P., Balamurugan, K., Dinesh, K., & Ezhilarasi, T. P. (2019). Particle swarm optimization approach for waterjet cavitation peening. Measurement, 141, 184-189.
[6] Parvin, J. R., & Vasanthanayaki, C. (2019). Particle swarm optimization-based energy efficient target tracking in wireless sensor network. Measurement, 147, 106882.
[7] Sangeetha, A. L., Bharathi, N., Ganesh, A. B., & Radhakrishnan, T. K. (2018). Particle swarm optimization tuned cascade control system in an Internet of Things (IoT) environment. Measurement, 117, 80-89.
[8] Norouzi, N., Sadegh-Amalnick, M., & Alinaghiyan, M. (2015). Evaluating of the particle swarm optimization in a periodic vehicle routing problem. Measurement, 62, 162-169.
[9] Hashim, F. A., & Hussien, A. G. (2022). Snake Optimizer: A novel meta-heuristic optimization algorithm. Knowledge-Based Systems, 242, 108320.
[10] Eberhart, R. C., Shi, Y., & Kennedy, J. (2001). Swarm Intelligence (Morgan Kaufmann series in evolutionary computation). Morgan Kaufmann Publishers.
[11] Clerc, M., & Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE transactions on Evolutionary Computation, 6(1), 58-73.
[12] Yektamoghadam, H., Nikoofard, A., Behzadi, M., Khosravy, M., Dey, N., & Witkowski, O. (2024). Multi-criteria evolutionary optimization of a traffic light using genetics algorithm and teaching-learning based optimization. Expert Systems, 41(2), e13487.
[13] Garcia-Nieto, J., Olivera, A. C., & Alba, E. (2013). Optimal cycle program of traffic lights with particle swarm optimization. IEEE Transactions on Evolutionary Computation, 17(6), 823-839.
[14] Liao, S., Wu, Y., Ma, K., & Niu, Y. (2023). Ant colony optimization with look-ahead mechanism for dynamic traffic signal control of IoV systems. IEEE Internet of Things Journal, 11(1), 366-377.
[15] Lin, H., Han, Y., Cai, W., & Jin, B. (2022). Traffic signal optimization based on fuzzy control and differential evolution algorithm. IEEE Transactions on Intelligent Transportation Systems, 24(8), 8555-8566.
[16] Cardoso, M. F., Salcedo, R. L., & De Azevedo, S. F. (1996). The simplex-simulated annealing approach to continuous non-linear optimization. Computers & chemical engineering, 20(9), 1065-1080.
[17] Kouidri, C., Bouiadjra, R. B., & Mahi, F. (2024). ACO-DSOS Hybrid Approach to Enhance Traffic Signal Optimization. International Journal of Transport Development and Integration, 8(1).

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

[18] Kouidri, S., & Kouidri, C. (2022). Bi-Objective Optimizing for Data-Intensive Scientific Workflow Scheduling in Cloud Computing. International Journal of Organizational and Collective Intelligence (IJOCI), 12(1), 1-12.
[19] Singh, L., Tripathi, S., & Arora, H. (2009). Time optimization for traffic signal control using genetic algorithm. International Journal of Recent Trends in Engineering, 2(2), 4.
[20] Silva, G. O., Rocha, A. M. A., Witeck, G. R., Silva, A., Durães, D., & Machado, J. (2022, July). On tuning the particle swarm optimization for solving the traffic light problem. In International Conference on Computational Science and Its Applications (pp. 68-80). Cham: Springer International Publishing.
[21] Akanni, A. O., & Omidiora, E. O. Analysis of Simulated Annealing to Decongest Traffic in a Multi-Road Coordinated Intersection.
[22] Teng, T. C., Chiang, M. C., & Yang, C. S. (2019, October). A hybrid algorithm based on GWO and GOA for cycle traffic light timing optimization. In 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC) (pp. 774-779). IEEE.
[23] Cintrano, C., Ferrer, J., López-Ibáñez, M., & Alba, E. (2021, March). Hybridization of racing methods with evolutionary operators for simulation optimization of traffic lights programs. In European Conference on Evolutionary Computation in Combinatorial Optimization (Part of EvoStar) (pp. 17-33). Cham: Springer International Publishing.
[24] Odeh, S. M., Mora, A. M., Moreno, M. N., & Merelo, J. J. (2015). A hybrid fuzzy genetic algorithm for an adaptive traffic signal system. Advances in Fuzzy Systems, 2015(1), 378156.
[25] Shine, R. (2003). Reproductive strategies in snakes. Proceedings of the Royal Society of London. Series B: Biological Sciences, 270(1519), 995-1004.
[26] Shine, R., Phillips, B., Waye, H., LeMaster, M. T. M. R., & Mason, R. T. (2001). Benefits of female mimicry in snakes. Nature, 414(6861), 267-267.