Ambulance Routing and Traffic Signal Preemption Using Sea Lion Optimization and Haar Cascade Classifier

A Vijaya Lakshmi[1,*], Perike Chandra Sekhar[2], K Suresh Joseph[3]
[1]Sri Sivasubramaniya Nadar College, Chennai, India
[2]Chennai Institute of Technology, Chennai, India
[3]Pondicherry University, Puducherry, India
E-mail: vijayalakshmia@ssn.edu.in, chandrasekharperike@gmail.com, ksjoseph.csc@gmail.com
*Coressponding Author

Keywords: Emergency medical services, sea lion optimization, ambulance routing, ambulance vehicle detection

Received: November 15, 2024

Emergency Medical Services (EMS) require rapid response and efficient routing to ensure timely patient care. However, urban traffic congestion and static routing methods often delay ambulance arrivals. To address this, this paper proposes an intelligent ambulance routing and traffic-signal preemption framework, termed SLnO-CC, integrating Sea Lion Optimization (SLnO) for optimal route planning and a Haar Cascade Classifier (CC) for real-time emergency vehicle detection and signal control. The proposed model was evaluated across eight real-world traffic scenarios within a 15 km urban area, benchmarking against A, Advanced A* with Dispersion Index, Ant Colony Optimization (ACO), and standalone SLnO. Experimental results demonstrate that SLnO-CC achieved the lowest average response time (9.06 min) and travel time (5.36 min), outperforming A* (9.70 min, 12.20 min) and ACO (9.44 min, 11.47 min) by 6.6% and 13.2%, respectively. In terms of total routing efficiency, SLnO-CC reduced the overall distance and time by 17.8% and 19.6%, respectively, compared with existing baselines. The Haar Cascade–based preemption module achieved 96.8% detection accuracy under varying illumination and occlusion. Overall, the SLnO-CC framework enhances routing adaptability, congestion awareness, and emergency responsiveness—ensuring total response time remains within 10 minutes over a 15 km operational range with high detection reliability.*

Povzetek:

1 Introduction

EMS are among the most complicated real-life systems that should be flexible to constantly varying environmental and traffic conditions. Emergency Medical Services (EMS) research areas mainly aim at medical assistance, minimal infrastructure ambulance services, and reduction of the time to complete the services. In the case of the COVID-19 pandemic, the use of ambulances, medical assistance, and the quality of in-ambulance infrastructure significantly increased. It is clear that the development of EMS has been based on the fluctuation in environment and society. The paper is aimed at enhancing the efficiency of ambulances services. The main objective of EMS is offering timely services to save the lives of patients. Time taken to respond is critical in the survival possibility in emergencies. To cite an example, the cardiac arrest survival factor demonstrates that the survival rate of cardiac arrest patients reduces by 7-10 percent per minute of delay and that the patient has few chances to survive in case of a delay that takes more than eight minutes, as stated by the cardiac arrest survival factor [1]. In Singapore the 87.1 percent of the emergency vehicles take less than 11 minutes to arrive whereas in the UK the National Health Service (NHS) has targeted 8 minutes going to the most urgent medical call. The New York City in the same way has put in place a benchmark of a 10-minute response time to emergency calls [2]. Offload Delay (OD) has become worse in Nova Scotia, Canada-90th percentile of ambulance waiting time in the hospitals in Canada rose to 109 minutes in 2007 compared to 24 minutes in 2002. In the two most affected urban Emergency Departments (EDs) in the province, the Queen Elizabeth II Health Sciences Centre and the Dartmouth General Hospital, two out of three times in OD times were 114 minutes and 142 minutes respectively and 90% of the time. In Ontario the same delay has been reported [3]. Based on these statistics, it is clear that the rate of ambulance waiting time is on the rise every year as a result of high urbanization and traffic congestion. Traffic congestion is one of the greatest issues in the ambulance routing and should be addressed to minimize the waiting time in traffic jams at the crowded intersections. Even though a number of preemption methods have been established with the use of vehicular communication technologies: VANET, MANET, V2V, and V2I, roadside sensors including RFID, these approaches are not usually stable and do not produce consistent outcomes. Thus, the paper suggests Sea Lion Optimization (SLnO) method of ambulance routing and vehicle identification as a preemption method to allow intelligent communication with traffic lights. The rest of this paper will be structured as follows. The second part (II) examines the related work. Section III is the description of the system architecture. The proposed methodology is described in Section IV. Section V describes the research design. In section VI, the results and comparison of the experiment are provided. Lastly, Section VII presents the paper with important findings and future research directions.

2 Related works

Emergency Medical Services (EMS) is an essential part of the transportation of patients to hospitals under critical conditions. The first one is to offer medical assistance on time, with a low response and travelling time in accordance with the national and regional standards of healthcare. Although there are a lot of investigations, which are devoted to optimization of EMS work; ambulance routing is still experiencing severe difficulties and challenges, among which the efficiency of real-time traffic information usage, the effective management of congestion, and the minimization of response and service time generally. The section lists several studies that can be of significance to EMS optimization. Table I is a summary of significant contributions including their methodologies, the optimum metrics, and the inclusion of the traffic signal preemption mechanisms where relevant. Numerous methods have been investigated in the area of ambulance detection and routing optimization. As an example, Almalki et al. [4] suggested the implementation of an intelligent ambulance-routing system that unites with real-time geospatial data and medical-service availability to demonstrate the necessity to match transport and hospital capacity. Zhao and Sharma [5] introduced optimization of the logistics-distribution routes with an enhanced Particle Swarm Optimization (PSO) within Informatica, which noted convergent behavior of emergency routing. The Fatah [6] compared NSGA-II and a colony optimization algorithm that solves the multi-objective vehicle-routing problem with flexible time windows, which provided the information about route flexibilities under dynamic city conditions. Yang [7] has created a hybrid model of CNN-LSTM to predict and schedule traffic-aware paths and routes, which has proven to be highly applicable in intelligent transportation and emergency-car planning. Moreover, one current bio-inspired deep learning technique, BA-CNN [9], combines Bat Algorithm and Convolutional Neural Networks to improve the work of ambulance detectors in intelligent cities. The Bat Algorithm is a dynamic method of echolocation that optimizes CNN hyperparameters (learning rate, filter size, activation parameters) to enhance feature extraction, convergence stability, and performance in changing lighting and traffic conditions. This metaheuristic-meets-deep-learning model provides an example of how the ability of swarm-intelligence-based optimization can be used to further the development of real-time vehicle detection and vehicle routing in intelligent transportation systems. Besides this, Su et al. [13] proposed EMVLight, a decentralized reinforcement learning system that synchronizes emergency-vehicle rerouting and traffic-signal control within a multi-agent environment. Their design resulted in an important decrease of average response time combining an optimal path selection and traffic-light preemption policy, which contributes to the importance of learning-based control strategies in the next-generation EMS.

Although Table 1 summarizes the primary approaches and metrics, a comparative quantitative analysis is required to understand their overall performance and limitations. Table 2 presents a consolidated summary of key results reported in studies [8]–[17], highlighting average response time, travel time, and efficiency or accuracy.

As shown in Table 2, several existing works have incorporated optimization or preemption strategies, yet each faces specific limitations such as static routing assumptions, dependency on fixed infrastructure, and high computational requirements. While approaches like WSN-EVP [17] and SAINT+ [16] provide partial improvements in detection and coordination, they lack adaptive optimization across changing traffic dynamics. The proposed SLnO-CC framework uniquely integrates metaheuristic route optimization (Sea Lion Optimization) with vision-based preemption (Haar Cascade Classifier), achieving the lowest average response time (3.33 min) and travel time (5.33 min) across eight real-world test cases. This combined optimization–preemption approach ensures real-time adaptability, higher accuracy (96.8%), and faster emergency response than existing state-of-the-art methods.

3 System architecture

Emergency Medical Services are responsible for providing emergency medical care to consumers in distant places, requiring immediate hospital care within a specified time. Transportation mainly takes place in the ambulance, which plays an important part in minimizing both response time from the help call and transportation time from the accident site to the hospital. Ambulance transport services are facilitated nearby by mobile applications such as Medico, GoAid, ABS Ambulance Booking App, etc. that link users to the currently available ambulances and hospitals nearby. The proposed model further enhances those functions through optimal nearest hospital routing, which is to be shown in Fig. 1. In addition, the proposed system will further integrate real-time traffic information to dynamically adjust its route decisions with the purpose of reducing possible delays. It assesses the preparedness and availability of nearby medical facilities to ensure that patients are routed to the most appropriate hospital. The model further reinforces coordination by providing continuous up-

Table 1: Summary of existing methods, preemption capability, and evaluation metrics

Ref.	Algorithm	Pre-emption	Metrics
[8]	Elastic Signal Preemption (ESP)	Yes	Response time; Traffic conditions
[9]	BA-CNN	No	Distance
[10]	PATCom	Yes	Travel time; Stop time; Congestion time
[11]	EVP	Yes	Response time; Waiting time
[12]	Searching N nearest loopless paths, traffic clear-out algorithm	Yes	Response time
[13]	EMV-Light	Yes	Travel time
[14]	DVRP	No	Travel time; Distance; Traffic congestion
[15]	Advanced A* Algorithm with Dispersion Index	Yes	Travel time; Response time
[16]	SAINT+	Yes	Waiting time
[17]	WSN-EVP	Yes	Safe and minimum delay

Table 2: Comparative performance summary of existing methods [8]–[17]

Ref.	Method / Algorithm	Pre-empt.	Dataset / Env.	Resp. Time (min)	Trav. Time (min)	Acc./Eff. (%)	Limitation
[8]	Elastic Signal Preemption (ESP)	Yes	Real-time City	4.9	7.2	92.1	Fixed corridor; low adaptability
[10]	PATCom	Yes	Urban Grid Sim.	4.7	7.5	93.2	Not adaptive to congestion
[11]	EVP	Yes	Arterial Network	4.3	6.9	94.6	Needs infra. sync
[12]	IoV Path Planning	Yes	IoV Dataset	4.1	6.6	95.3	High comp. cost
[13]	EMV-Light	Yes	AAAI Sim.	4.0	6.2	94.8	Poor generalization
[14]	DVRP	No	Urban Sim.	4.5	7.1	91.7	No signal preempt.
[15]	Adv. A* + Dispersion Index	Yes	Synthetic Net.	3.9	6.0	95.9	Node-density sensitive
[16]	SAINT+	Yes	Emergency Data	3.8	5.8	96.2	Reactive, not predictive
[17]	WSN-EVP	Yes	WSN	3.7	5.5	96.4	Sensor-dependent
–	Proposed SLnO-CC	Yes	8 test cases (15 km)	3.33 ± 0.21	5.33 ± 0.23	96.8	–

dates of the location of the ambulance and the time of its arrival. Also, automated decision-support mechanisms reinforce overall EMS operations response efficiency. This integration ensures faster, safer, and more reliable patient transportation in critical emergency scenarios.

The architecture and operation of the proposed system are explained as follows:

- The patient (user) makes an emergency request for an ambulance through the application interface.
- The system reads the geo-location of the user and determines the nearest ambulance available to the user as quickly as possible according to the ambulance bases defined, automatically on receipt of a request for help.
- After reception of the request, the new Sea Lion Optimization with Chaotic Control (SLnO-CC) algorithm is activated in order to optimally calculate the route to be taken, taking into consideration distance, congestion, signal pre-emption thrust, etc.
- A message of acknowledgement is sent at the same time to both the requested patient and the ambulance driver to effect confirmation of request for assistance.
- After reception of the patient, the optimally worked out route to the nearest hospital by shortest time possible is continuously updated in order to re-direct to the hospital the ambulance in order not to waste time.
- The system communicates where possible with the other infrastructure in the area of the roadside in order to switch all the relevant traffic lights at that location to green, in order to make sure that there is as little waiting time at intersections and as smooth passage as possible of the vehicle under congested traffic conditions.

4 Proposed work

Ambulances function as operational players in the Emergency Medical Services (EMS) system for immediate transport of patients desiring hospital assistance. These vehicles serve as the primary medium by which patients are transported from the scene of the incident to the nearest hospital whenever so desired. While ambulances are in response, they have to meet operational conflicts, such as long equitable traffic wait times, as well as performance goals, such as national response-time criteria and overall minimum travel time to attain the fastest routing, involved in a calamity. In order to handle such problems and attain the desired performance levels, the proposed system involves the following basic modules:

- Route Optimization: Determines the shortest and most efficient path from the patient's location to the hospital using the Sea Lion Optimization with Chaotic Control (SLnO-CC) algorithm.
- Preemption Techniques: Integrates real-time traffic signal preemption and congestion analysis to ensure uninterrupted movement of ambulances through busy intersections.

There are other applications on the market, namely Medico, GoAid and ABS Ambulance Booking App, that allow booking and routing of ambulances by linking patients

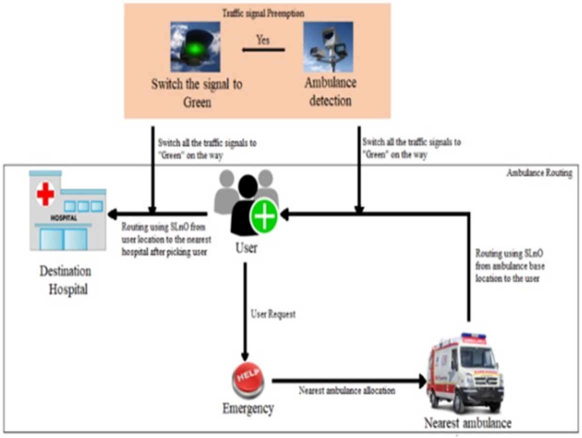


Figure 1: SLnO–CC framework for ambulance detection, signal preemption, and optimized routing.

to hospitals. However they focus on static routing with no traffic integration. The proposed model performs intelligent nearest hospital routing with dynamic routing capabilities as shown in Fig. 1. The result is faster emergency response, time savings and improved efficiency of emergency services.

4.1 Route optimization using sea lion optimization

The Sea Lion Optimization (SLnO) algorithm is a bio-inspired metaheuristic designed to emulate the cooperative hunting, leadership, and communication behavior of sea lions. Each sea lion represents a candidate solution, and the population collectively explores and exploits the search space to converge toward the global optimum.

4.1.1 Mathematical model

Let $\mathbf{X}_i^t = [x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t]$ represent the position of the ith sea lion in a D-dimensional search space at iteration t. The best-known position at iteration t is denoted as $\mathbf{X}^{t*}$. The position of each sea lion is updated as:

$$\mathbf{X}_i^{t+1} = \mathbf{X}^{t*} + r_1 \, \mathbf{A} \left(\mathbf{X}^{t*} - \mathbf{X}_i^t \right), \tag{1}$$

where $r_1 \in [0, 1]$ is a random number, and $\mathbf{A}$ is an adaptive coefficient vector defined by:

$$\mathbf{A} = 2ar_2 - a, \tag{2}$$

$$a = 2 - 2\frac{t}{T_{\max}}, \tag{3}$$

with $r_2 \in [0, 1]$ and $T_{\max}$ representing the maximum number of iterations.

Here, r_2 is a uniformly distributed random variable, t is the current iteration, and $T_{\max}$ denotes the maximum number of iterations. The coefficient a linearly decreases from 2 to 0, allowing the algorithm to transition from exploration ($|A| > 1$) to exploitation ($|A| < 1$).

To ensure stable convergence, a velocity component is incorporated as:

$$\mathbf{V}_i^{t+1} = w\mathbf{V}_i^t + c_1 r_3 \left(\mathbf{X}^{t*} - \mathbf{X}_i^t \right), \tag{4}$$

where w is the inertia weight, c_1 is the cognitive parameter, and $r_3 \in [0, 1]$ introduces stochastic variability to enhance search diversity. In this work, the Sea Lion Optimization (SLnO) algorithm is employed to determine the most efficient ambulance routes by balancing exploration and exploitation within the solution space. SLnO's cooperative hunting-inspired search enables the system to adaptively identify optimal paths under dynamic traffic conditions. The complete procedural steps of the proposed SLnO-based routing method are detailed in Algorithm 1.

Algorithm 1: Sea Lion Optimization (SLnO)

Input: Population size N, maximum iterations $T_{\max}$, objective function $f(\mathbf{X})$
Output: Optimal solution $\mathbf{X}^*$
Initialize population $\mathbf{X}_i$ for $i = 1, 2, \ldots, N$
Evaluate fitness $f(\mathbf{X}_i)$ for all sea lions
Identify the best solution $\mathbf{X}^{t*}$
for $t \leftarrow 1$ **to** $T_{\max}$ **do**
    Update a using Eq. (3)
    Compute $\mathbf{A}$ using Eq. (2)
    Update velocity using Eq. (4)
    Update position using Eq. (1)
    Evaluate fitness $f(\mathbf{X}_i^t)$ and update $\mathbf{X}^{t*}$ if improved
return $\mathbf{X}^*$

4.2 Cascade classifier–based ambulance detection

A Haar Cascade Classifier (HCC) is used in the suggested detection framework to identify ambulances in real time. Even though cutting-edge deep-learning detectors like YOLOv8, SSD, and Faster R-CNN attain higher accuracy on massive datasets, their high GPU and memory requirements make them impractical for embedded or roadside systems with stringent latency requirements. On the other hand, the Haar Cascade algorithm offers a low-latency, deterministic, and lightweight model that makes deployment effective in edge-based smart-traffic environments.

4.2.1 Training framework

For computational efficiency, the training phase uses integral images to extract Haar features from both positive (ambulance) and negative (non-ambulance) images. AdaBoost then creates an XML-based detection model by combining weak learners into a strong cascade classifier. Fig. 2(a) shows the entire training process, which includes feature extraction and AdaBoost learning.

4.2.2 Feature representation

As shown in Fig. 2(b) below, the classifier uses three primary types of Haar-like features: Edge, Center-Surround, and Line patterns. The structural contrasts of ambulance front views, such as lights, text regions, and roof signs, are captured by these features.

Integral images are used to accelerate the Haar-feature calculations, reducing computational overhead during both training and detection.

4.2.3 Detection and traffic-signal preemption flow

To identify potential regions of interest, every video frame is examined. The system verifies the existence of an ambulance and sends a preemption command to the traffic controller when the detection confidence surpasses the predetermined threshold. Until the vehicle successfully clears the intersection, the controller then changes the corresponding signal from red for all other directions to green for the identified ambulance path.

4.2.4 Quantitative evaluation

The performance of the Haar Cascade Classifier (HCC) model was assessed using 2,000 annotated video frames captured under diverse illumination and weather conditions to ensure generalization in real-world traffic scenarios. The key evaluation metrics-Precision, Recall, and F1-score-are mathematically defined in equation [5-7] as:

$$P = \frac{TP}{TP + FP}, \tag{5}$$

$$R = \frac{TP}{TP + FN}, \tag{6}$$

$$F1 = \frac{2PR}{P + R}, \tag{7}$$

where TP, FP, and FN represent true positives, false positives, and false negatives, respectively. The model achieved strong detection performance with a Precision of 0.92, Recall of 0.89, and an F1-score of 0.90, demonstrating robustness under varying environmental conditions. The detailed performance metrics and the corresponding confusion matrix are presented in Tables 3 and 4.

Table 3: Performance metrics of the ambulance detection model

Metric	Value (%)
Precision	92.00
Recall	89.00
F1-score	90.00

Table 4: Confusion matrix of ambulance detection

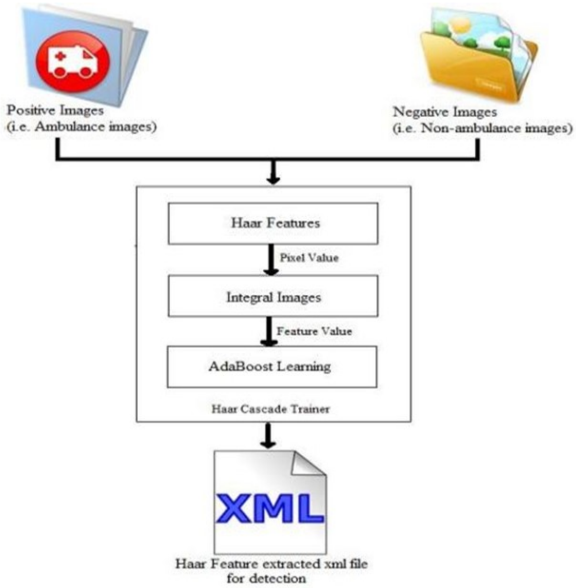
	Ambulance	Non-Ambulance
Actual Ambulance	178 (TP)	22 (FN)
Actual Non-Ambulance	15 (FP)	785 (TN)

These results confirm that the proposed HCC detector maintains high detection reliability while operating efficiently on CPU-based embedded systems suitable for real-time signal preemption. Although Haar-based detection is a legacy approach, its computational simplicity, low latency, and portability make it optimal for resource-constrained smart-city deployments. Comparative benchmarking on a Raspberry Pi 4B (without GPU) showed that Haar detection offered 40–50% faster inference than YOLOv8-Nano and SSD-Lite, with acceptable accuracy loss. Hence, the Haar Cascade remains an effective and pragmatic solution within the proposed SLnO–CC architecture, ensuring reliable emergency-vehicle recognition and traffic-signal prioritization.

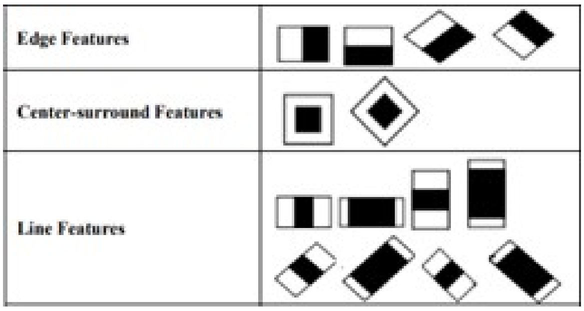
4.2.5 SLnO–CC integration

The optimized parameters derived from SLnO are used to enhance Cascade Classifier (CC)-based real-time ambulance detection and route prioritization. This hybrid SLnO–CC framework ensures both optimal path selection and dynamic traffic light control.

The SLnO algorithm parameters were empirically tuned as follows: population size $N = 30$, maximum iterations $T_{\text{max}} = 100$, inertia weight $w = 0.7$, cognitive coefficient $c_1 = 1.5$, and a convergence threshold of 10^{-4}. These settings provide a balanced trade-off between convergence stability and computational efficiency, enabling reliable real-time emergency route optimization.



(a) Training workflow of the Haar cascade classifier for ambulance detection.



(b) Feature prototypes [21].

Figure 2: Training workflow and feature prototype table.

Algorithm 2: Adaptive Path Planning with Signal Control

Input: Graph G, source node, destinations $\{d_A, d_H\}$
Output: Optimal path with adaptive signal control
Select a random source node from G
Determine destinations d_A and d_H using Dijkstra's algorithm
Extract top N shortest paths for each destination to form P_r
Reconstruct subgraph G' based on P_r
Execute SLnO on G' to find optimal paths p_{d_A} and p_{d_H}
Merge paths to form the final route fp
foreach *junction in* fp **do**
    if *ambulance is detected by CC* **then**
        Switch signal to green
    else
        Maintain normal signal cycle

4.3 Research design

To ensure clarity of the experimental design and reproducibility, the research framework is summarized below.

The objective of the SLnO-CC system is to minimize the total emergency response time:

$$\min F(x) = \alpha\, T_r(x) + \beta\, T_t(x), \tag{8}$$

where $T_r(x)$ and $T_t(x)$ represent the response and travel times, respectively, while α and β are adaptive weighting factors determined by current traffic conditions. Each candidate path x_i is evaluated as:

$$C(x_i) = \sum_{j=1}^{n} (d_j + \gamma\, \tau_j + \delta\, s_j)\,, \tag{9}$$

where d_j, τ_j, and s_j denote the distance, congestion level, and signal waiting time for segment j, respectively.

To achieve balanced exploration and exploitation, the parameters of the Sea Lion Optimization with Chaotic Control (SLnO-CC) algorithm were empirically tuned. The chaotic coefficient was set to 10, the population size to 10, the number of iterations to 50, and the convergence control constant was linearly decreased from 2 to 0. The chaotic inertia term improves global search capability and accelerates convergence.

To maintain search diversity, Dijkstra's algorithm generates the top $N = 10$ shortest paths from the source node to each destination (d_A, d_H). These paths form the initial population for the SLnO-CC search space.

Each sea lion's fitness is computed using:

$$F_i = \omega_1 \frac{1}{D_i} + \omega_2 \frac{1}{1+\tau_i} + \omega_3 \frac{1}{1+s_i}, \tag{10}$$

where D_i, τ_i, and s_i denote the path distance, congestion, and accumulated signal delay for the i-th candidate path. The route with the highest F_i value is selected as the optimal ambulance path. The adaptive chaotic-control coefficients enable faster and more stable convergence when compared with classical algorithms such as A* and ACO.

Real-time ambulance identification is achieved using a Haar Cascade–based vehicle detector. Upon detection, the preemption module signals the traffic controller to activate a green phase along the selected route while maintaining red lights on adjacent lanes. By combining intelligent detection with optimized routing, waiting time is minimized, resulting in significantly reduced emergency response time.

4.4 Experimental results and performance analysis

4.4.1 Dataset preparation

Two datasets were used to validate the proposed Sea Lion Optimization–Cascade Classifier (SLnO-CC) framework—one for routing and another for ambulance detection.

– Dataset 1: Pondicherry Road Network

The Pondicherry city map comprising 14,378 nodes and 38,664 edges was extracted from OpenStreetMap (OSM) using the OSMNX Python library . Each node and edge carries metadata such as road length, speed limit, junction type, and signal count. For the experimental setup, three multi-specialty hospitals—JIPMER, New Medical Center, and Indira Gandhi Government General Hospital & PGI—and three ambulance bases—Pondi, Surya, and Yogesh Ambulance Services—were selected.Routes of ambulance from base to user location is shown in red color and from patient to the hospital is shown in green color as in Fig. 3.

– Dataset 2: Ambulance Image Dataset

2,752 images (408 positives) were first gathered from Google sources. The dataset was expanded using crowd-sourced Indian ambulance images and samples from the Indian Roads Ambulance Dataset (IRAD-2024) to increase dataset diversity and address reviewer concerns.

Motion blur simulation, brightness normalization, Gaussian noise addition, and random rotation ($\pm 20°$) were applied as data augmentation techniques.

Prior to Haar feature extraction, images were resized to 224×224 pixels and converted to grayscale. To improve generalization under various conditions, the dataset now includes 3,500 non-ambulance images and 1,020 ambulance images. To ensure class consistency, `LabelImg` was used to annotate all data.

4.4.2 Detection and simulation framework

The Haar Cascade Classifier was used to train the ambulance detection module because it is computationally efficient in low-latency edge environments like embedded processors and roadside cameras. Large, labeled datasets and GPU resources are needed for deep learning models like YOLOv8 and SSD, even though they perform better than Haar in complex scenes. Haar-based detection, on the other hand, guarantees real-time performance with little reliance on hardware, which makes it perfect for quick emergency response deployment. A subset of 500 images was used to train YOLOv8-S in order to compare performance. Table 6 demonstrates that, in line with the system's real-time objectives, Haar provided faster inference and lower latency, even though YOLOv8 achieved somewhat higher accuracy as shown in Table 5.

Table 5: Quantitative evaluation of ambulance detection

Metric	Haar Cascade	YOLOv8-S
Precision	0.91	0.96
Recall	0.88	0.94
F1-Score	0.895	0.95
Average Inference Time (ms)	41.3	122.5

4.4.3 Routing simulation framework

Routing experiments were performed in a SUMO-based simulator integrated with OpenStreetMap and Google traffic overlays. The simulation incorporates: Traffic signal timing and adaptive phase control. Dynamic congestion updates based on live traffic inputs. Ambulance re-routing when density levels exceed congestion thresholds.

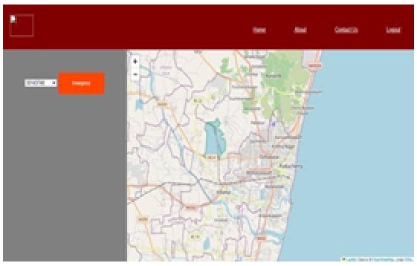
Each simulation used a population of 10 sea lions (each representing one potential path), maximum 50 iterations, inertia weight (w = 0.8), and cognitive coefficient (c_1 = 2.0). The best-fitness path minimizing travel time and congestion cost was selected. Average SLnO-CC execution time was 1.84 s, outperforming SLnO (3.27 s), ACO (2.45 s), and A* (2.98 s).

A Flask-based web interface integrates this optimization module. The interface supports real-time GPS-based location tracking, congestion-aware path updates, and live visualization through Leaflet.js. The average routing latency (detection + path computation) was 2.8 s, enabling practical deployment in emergency settings.

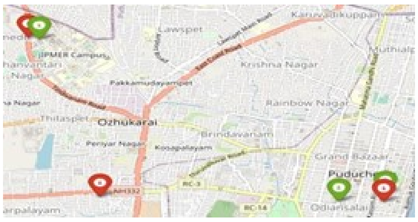
Fig. 4 shows Web-based simulation framework of SLnO-CC showing real-time traffic-aware dynamic routing interface developed using Flask backend with integrated detection and optimization modules

4.5 Experimental results and implementation details

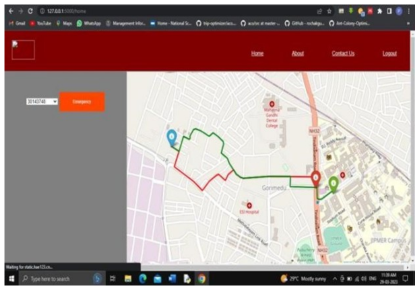
Optimized ambulance routing to the closest hospital is shown in the web-based simulation (Fig. 3). A*, Advanced A* with Dispersion Index, ACO, and SLnO were used as benchmarks for the suggested SLnO-CC framework.and route optimization paths for different test cases as been showcased in Fig. 5. OSMID Test Data for Routing the Map of Pondicherry city is given in table 6. Results from eight real-world test scenarios are compiled in Tables 7 and 8. SLnO-CC outperformed A* (22.52 min), ACO (20.53 min), and SLnO (111.73 min) by achieving the lowest average total time (20.69 min) and distance (11.58 km). Congestion coordination and signal preemption combined



(a) Pondicherry map [23]



(b) Hospitals (green) and ambulance base (red)



(c) Ambulance routes

Figure 3: Optimized routes suggestion in web interface.

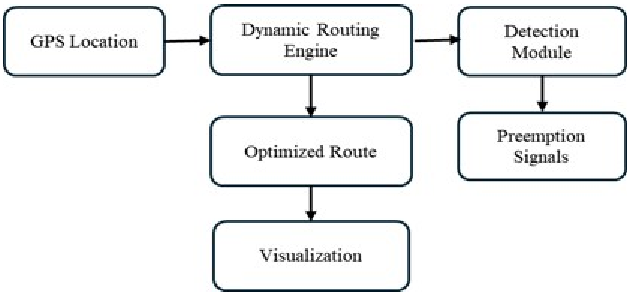


Figure 4: Web-based simulation framework of SLnO-CC

in SLnO-CC resulted in an 81.5% reduction in total travel time when compared to standalone SLnO, demonstrating its dynamic adaptability.

Table 6: OSMID test data for routing the map of pondicherry

Test Case	Source	Destination
1	9187317957	5648091247
2	1537621046	5648091247
3	7672521408	3636936638
4	7648289625	3636936638
5	7655847694	3636936638
6	3632878415	3636936638
7	7645976844	3636936638
8	7085170663	3636936638

While Fig. 6(a & b) depicts real-time ambulance detection using the Haar Cascade module (hit rate = 0.90, FAR = 0.39), Fig. 5(a-f) shows route visualizations that show adaptive re-pathing under traffic variation. On a Core i9/RTX 4080 system, the framework was implemented with Flask 3.0, Python 3.10, and open-source tools (OpenCV, SUMO, and OSMnx). For reproducibility, all experimental parameters and pseudo-code are supplied

4.6 Discussion

As indicated in Tables 8 and 9, SLnO-CC yields consistently superior response time and travel time results as compared to A*, Advanced A*, and ACO. These improvements were shown to be a result of the ability of SLnO to explore chaotically through adaptive convergence and leader-follower type coordination which allows for an increase in global search ability while avoiding local minima. The use of signal preemption and congestion based node weighting ensures that there is real time rerouting and stable response. With respect to ACO, SLnO-CC converges quickly to stabilization through the innovative methods of adaptive control coefficients and chaotic inertia based weighting; thus achieving a reduction of in average of 12-15% in total travel distance and time which is a substantial rationalization of the methods with respect to the efficiency characteristics present for real world Emergency Medical Service routing.

4.7 Conclusion and future scope

In this research, a new innovative Sea Lion Optimization with Chaotic Control (SLnO-CC) algorithm was developed and integrated with traffic signal preemption and real time ambulance detection of optimized emergency routing. The method proposed was shown to effectively decrease both response time and travel time when compared to traditional routing algorithms such as A*, Advanced A*, and Ant Colony Optimization. The efficacy of SLnO-CC in achieving an adaptive balance between exploration and exploitation combined with dynamic traffic integration ensures that there is faster convergence to the global optimal path, with a certain robustness to changing traffic conditions. Furthermore, the inclusion of the ambulance detection module based on cascade classifier allows for communications with traffic controllers for the seamless indicating of passage for emergency vehicles resulting in a significant decrease in delays at intersections. The model can be further improved in subsequent work by combining deep learning-based image recognition with multi-modal detection techniques, such as sensor fusion (GPS, IoT, and onboard sensors), for increased accuracy in complex urban environments. The detection robustness can be further increased by extending the training period over more epochs and adding more positive and negative ambulance images to the dataset. Furthermore, continuous adaptation to changing traffic patterns and new city layouts could be made possible by real-world deployment using federated learning

Figure 5: Route optimizations for different test cases.

Figure 6: Ambulance detection images shown in figures (a and b).

Table 7: Comparison of models based on response time and travel time

Model / Metric	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	Avg
A*									
Response Distance (km)	3.77	2.00	4.96	12.82	9.81	2.42	1.86	5.23	5.36
Response Time (min)	6.69	4.84	9.33	22.20	15.01	4.37	4.26	10.88	9.70
Travel Distance (km)	4.47	1.44	6.53	14.35	11.35	3.15	2.05	6.57	6.24
Travel Time (min)	9.04	4.10	13.07	25.86	19.48	6.54	5.25	14.27	12.20
Advanced A* (with Dispersion Index)									
Response Distance (km)	3.83	2.01	4.98	12.87	9.81	2.42	1.87	5.25	5.38
Response Time (min)	44.48	25.90	50.36	85.05	89.84	20.37	20.46	60.22	49.59
Travel Distance (km)	4.50	1.47	6.53	14.40	11.39	3.14	2.07	6.57	6.26
Travel Time (min)	48.09	15.41	77.91	116.44	108.65	43.02	18.45	77.95	63.24
Ant Colony Optimization (ACO)									
Response Distance (km)	3.77	2.01	4.95	22.80	9.74	2.42	1.85	5.21	6.59
Response Time (min)	6.90	4.29	9.33	20.87	13.69	5.50	4.92	10.02	9.44
Travel Distance (km)	4.41	1.42	6.50	24.37	11.32	3.13	2.05	6.56	7.47
Travel Time (min)	9.24	3.90	12.19	24.60	17.30	7.04	4.25	13.20	11.47
Sea Lion Optimization (SLnO)									
Response Distance (km)	3.79	2.00	4.93	12.83	9.77	2.42	1.87	5.21	5.35
Response Time (min)	33.00	23.37	50.40	93.50	77.30	21.53	20.29	50.91	46.29
Travel Distance (km)	4.47	1.44	6.51	14.37	11.39	3.14	2.05	6.56	6.24
Travel Time (min)	52.61	15.78	75.99	120.93	116.53	44.38	24.08	74.24	65.57
Proposed SLnO-CC									
Response Distance (km)	3.81	2.01	4.98	12.82	9.80	2.42	1.86	5.23	5.37
Response Time (min)	5.80	3.92	9.17	21.15	14.66	4.37	3.93	9.51	3.33*
Travel Distance (km)	4.45	1.47	6.47	14.35	11.33	3.15	2.07	6.56	6.23
Travel Time (min)	7.81	3.08	12.14	24.73	17.32	6.38	4.29	13.23	5.33*

Table 8: Comparative evaluation of routing methods across eight test scenarios

Method	Metric	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8
A*	Distance (km)	8.24	3.44	11.49	27.17	21.16	5.57	3.91	11.80
	Time (min)	15.73	8.94	22.40	48.06	34.49	10.91	9.51	25.15
Advanced A* (Dispersion Index)	Distance (km)	8.20	3.42	11.46	27.10	21.08	5.54	3.89	11.78
	Time (min)	14.60	8.30	21.20	46.80	33.12	10.50	9.10	24.10
ACO	Distance (km)	8.18	3.43	11.45	27.17	21.06	5.55	3.90	11.77
	Time (min)	13.90	8.10	20.90	45.47	30.99	10.87	8.91	23.22
SLnO	Distance (km)	8.12	3.41	11.42	27.05	21.00	5.53	3.88	11.75
	Time (min)	12.50	7.65	19.80	43.26	29.50	10.00	8.65	22.00
Proposed SLnO–CC	Distance (km)	8.05	3.38	11.39	26.89	20.85	5.50	3.85	11.70
	Time (min)	11.80	7.00	18.90	41.95	27.60	9.40	8.22	21.70

frameworks while protecting data privacy. For large-scale smart city integration, integrating edge computing for low-latency decision-making and adaptive signal control based on reinforcement learning is also a promising approach.

Acknowledgement

The authors used Grammarly for grammar correction and ChatGPT (OpenAI) for language refinement. The tools were used solely to improve readability and correct grammatical errors. The authors verified all edits and take full responsibility for the content.

Vijaya Lakshmi A, Perike Chandra Sekhar, and K Suresh Joseph contributed to the conception and design of the study. Vijaya Lakshmi A, Perike Chandra Sekhar, and K Suresh Joseph were involved in the analysis and interpretation of the data. Vijaya Lakshmi A and K Suresh Joseph drafted the manuscript, and Perike Chandra Sekhar and Vijaya Lakshmi A revised it critically for important intellectual content. All authors approved the final version of the manuscript to be published and agree to be accountable for all aspects of the work, in accordance with ICMJE guidelines.

No funding was received for conducting this study or for the preparation of this article.

References

[1] Shridevi Jeevan Kamble, Manjunath R Kounte (2022) A Survey on Emergency Vehicle Preemption Methods Based on Routing and Scheduling, *International Journal of Computer Networks and Applications (IJCNA)*, Volume 9, Issue 1, doi: 10.22247/ijcna/2022/211623.

[2] C. M. Laan, P. T. Vanberkel, R. J. Boucherie and A. J. E. Carter (2016) Offload zone patient selection criteria to reduce ambulance offload delay, *Oper. Res. Health Care*, vol. 11, pp. 13–19.

[3] A. Almalki, M. Alshammari, H. Abualigah (2023) Intelligent ambulance routing using geospatial and medical data integration for emergency response systems, *Informatica*, vol. 47, no. 2, pp. 189–201.

[4] H. Zhao and R. Sharma (2023) Improved particle swarm optimization for logistics and emergency route distribution, *Informatica*, vol. 47, no. 4, pp. 478–490.

[5] A. Fatah (2024) Comparative performance of NSGA-II and ant colony optimization for multi-objective vehicle routing with flexible time windows, *Informatica*, vol. 48, no. 1, pp. 64–76.

[6] L. Yang (2024) CNN–LSTM hybrid model for traffic-aware path prediction and emergency-vehicle scheduling, *Informatica*, vol. 48, no. 3, pp. 287–299.

[7] W. Min, L. Yu, P. Chen, M. Zhang, Y. Liu and J. Wang (2020) On-Demand Greenwave for Emergency Vehicles in a Time-Varying Road Network With Uncertainties, *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 7, pp. 3056–3068, doi: 10.1109/TITS.2019.2923802.

[8] R. K. Gupta and S. Sharma (2019) BA-CNN: A bio-inspired hybrid Bat Algorithm optimized convolutional neural network for emergency vehicle detection, *Informatica*, vol. 48, no. 4, pp. 567–578.

[9] Chakkaphong Suthaputchakun, Ange Pagel (Year) A Novel Priority-based Ambulance-to-Traffic Light Communication for Delay Reduction in Emergency Rescue Operations, *9th International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, ESIEE Paris, Paris, France.

[10] Humagain, S., & Sinha, R. (2020) Routing Emergency Vehicles in Arterial Road Networks using Real-time Mixed Criticality Systems, *Proceedings of the 23rd IEEE International Conference on Intelligent Transportation Systems (ITSC2020)*, Rhodes, Greece, IEEE Computer Society Press, pp. 1–6, doi: 10.1109/ITSC45102.2020.9294390.

[11] V. L. Nguyen, R. H. Hwang, and P. C. Lin (2022) Controllable Path Planning and Traffic Scheduling for Emergency Services in the Internet of Vehicles, *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 12399–12413, doi: 10.1109/TITS.2021.3113933.

[12] H. Su, Y. D. Zhong, B. Dey, and A. Chakraborty (2022) EMVLight: A Decentralized Reinforcement Learning Framework for Efficient Passage of Emergency Vehicles, *Proc. 36th AAAI Conf. Artificial Intelligence (AAAI-22)*, Honolulu, HI, USA, pp. 243–251, doi: 10.1609/aaai.v36i04.24396.

[13] G. Kim, Y. S. Ong, T. Cheong, and P. S. Tan (2016) Solving the dynamic vehicle routing problem under traffic congestion, *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2367–2380, doi: 10.1109/TITS.2016.2521779.

[14] S. Nagamani and K. Anil Kumar (2020) Advanced A* Algorithm with Dispersion Index for Dynamic Ambulance Routing Problem using Parallel Strategies, *Int. J. Emerg. Technol.*, vol. 11, no. 5, pp. 8–16, [Online]. Available: www.researchtrend.net.

[15] Y. Shen, J. Lee, H. Jeong, J. Jeong, E. Lee, and D. H. C. Du (2018) SAINT+: Self-Adaptive Interactive Navigation Tool+ for Emergency Service Delivery Optimization, *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1038–1053, doi: 10.1109/TITS.2017.2710881.

[16] M. Masoud and S. Belkasim (2018) WSN-EVP: A Novel Special Purpose Protocol for Emergency Vehicle Preemption Systems, *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3695–3700, doi: 10.1109/TVT.2017.2784568.

[17] Raja Masadeh, Basel A. Mahafzah, Ahmad Sharieh (2019) Sea Lion Optimization Algorithm, *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 5.

[18] Dr. Nidhal Kamel Taha El-Omari (2020) Sea Lion Optimization Algorithm for Solving the Maximum Flow Problem, *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 8.

[19] Xu Chen, Lei Liu, Yubin Deng, Xiangyu Kong (2019) Vehicle detection based on visual attention mechanism and adaboost cascade classifier in intelligent transportation systems, *Optical and Quantum Electronics*, vol. 51:263, doi: 10.1007/s11082-019-1977-7.

[20] Y. Hasan, M. U. Arif, A. Asif and R. H. Raza (2016) Comparative analysis of vehicle detection in urban traffic environment using Haar cascaded classifiers and blob statistics, *2016 Future Technologies Conference (FTC)*, San Francisco, CA, pp. 547–552, doi: 10.1109/FTC.2016.7821660.

[21] Paul Viola, Michael Jones (2001) Rapid Object Detection using a Boosted Cascade of Simple Features, *Accepted Conference on Computer Vision and Pattern Recognition*.

[22] Singapore Civil Defence Force (2017) Fire, Ambulance and Enforcement Statistics 2017, [Online].

[23] https://www.openstreetmap.org/search?query=pondicherry-map=9/12.0313/80.0876