Hindawi
Mobile Information Systems
Volume 2022, Article ID 7339647, 14 pages
https://doi.org/10.1155/2022/7339647

Research Article

BA-CNN: Bat Algorithm-Based Convolutional Neural Network Algorithm for Ambulance Vehicle Routing in Smart Cities

Taha Darwassh Hanawy Hussein,[1] **Mondher Frikha**,[2] **Sulayman Ahmed**,[3] **and Javad Rahebi**[4]

[1]*ENIS, Universite de Sfax, Sfax, Tunisia*
[2]*ENETCOM, Universite de Sfax, Sfax, Tunisia*
[3]*Kirkuk University, Kirkuk, Iraq*
[4]*Department of Software Engineering, Istanbul Topkapi University, Istanbul, Turkey*

Correspondence should be addressed to Javad Rahebi; javadrahebi@gmail.com

Received 26 April 2022; Revised 11 May 2022; Accepted 14 May 2022; Published 8 June 2022

Academic Editor: M. Praveen Kumar Reddy

This article proposes an ambulance vehicle routing approach in smart cities. The approach is based on the bat algorithm and convolutional neural network (BA-CNN). It aims to take transfer the patients confidentially, accurately, and quickly. The type of CNN used in this research is a residual network (ResNet). The node method is responsible for creating the city map. In the beginning, information about the accident place is received by the control station and forwarded to both the hospital and the ambulance. The driver feeds the data that contain the ambulance vehicle's node position and the accident location to the BA-CNN vehicle routing algorithm. The algorithm then obtains the shortest path to reach the location of the accident by the driver. When the vehicle arrives at the accident location, the driver updates the algorithm with hospital and accident positions. Then, the shortest path (which leads to the fast reach time) to the hospital is calculated. The bat algorithm provides offline data for a possible combination of different source and destination coordinates. The offline data are then trained by utilizing a neural network. The neural network is used for finding the shortest routes between source and destination. The performance evaluation of the BA-CNN algorithm is based on the following metrics: end-to-end delay (EED), throughput, and packet delivery fraction (PDF). This BA-CNN is compared with counterparts, including three different existing methods such as TBM, TVR, and SAODV. The experiments demonstrate that the PDF of our method is 0.90 for 10 malicious nodes, which is higher than in the TBM, TVR, and SAODV.

1. Introduction

Recently, scholars have devoted more attention to the Internet of Things (IoT) technology, which is implemented in different fields. In the IoT, the devices are intelligent, communicate, and exchange data among themselves through the connected network. The devices use sensors to collect the information and transmit it to the base stations. Very briefly, for communication and data collection, two types of sensors are available in smart devices. In the network layer, the devices in IoT exchange (collect and transmit) data among them [1]. Nowadays, the IoT is implemented in different places, such as smart homes [2] and smart cities [3]. In smart cities, IoT is utilized in different infrastructures, such as industry connections, health, and transportation systems.

Applications programs are used in managing and interconnection among heterogeneous devices. Accordingly, a vast amount of data (big data) is produced in these devices [4]. Distributed computing systems (e.g., cloud computing) are utilized in processing data of smart cities. These systems are used to analyse big data in real-time. Recently, edge computing has been used to process big data. In the calculations and processing, sensor layer capabilities and generating objects are used. This model is known as fog computing [5]. The high ability to process big data in smart cities can be provided by fog computing with the cloud-computing layer. Despite dealing with big data being a critical issue, significant challenges in data security face smart cities.

As mentioned earlier, smart cities manipulate a large amount of data exchange. To maintain the confidentiality of information, data transmission is required to be protected. Medical data security (including health-related data) is considered one of the crucial issues in smart cities. Over the network and its infrastructure, it is important to maintain the confidentially of health-related data transmission. Pieces of information may be leaked if the data are intercepted in the middle of the transmission. Therefore, during medical data transmission, there is a possibility of losing the data conditionality and the data manipulated. In some cases, disrupting treatment and even killing patients may be caused by manipulating medical data in smart cities. Accordingly, in smart cities, data encryption of medical information is essential. To other medical centers, the transmission of patient records on the blockchain and the IoT is a new treatment technique [6–8]. Human has a significant role in natural accidents and disasters. These disasters or accidents can impact people and infrastructures on a larger scale. To carry out in such circumstances, effective logistical operations are challenging. Therefore, medical help is impossible sometimes leading to further losses. Thus, the operations of rescue and assistance are needed to be managed efficiently after accidents or disasters [9].

In emergencies, it is important to take the injured persons to the nearest hospitals. The following details are needed to operate efficiently: (1) location in the number of wounded persons and (2) the availability of ambulances. These are the key factors input to determine solutions for the accidents. After collecting these inputs, there must be an emergency schedule to rescue the injuries as fast as possible [10]. Therefore, this issue is concerned with the capacitated ambulance routing problem (CARP).

The ambulance routing problem (ARP) is one of the CARP forms. It refers to injuries and their requests. This ARP aims to find the optimum (minimum) required distances between the ambulance station and accident places and between accident and hospital places to take the victims [11,12]. The ambulance collects the victims carefully from the accident location to the hospital. After that, the ambulances will return to their station. The CARP can propose different solutions: heuristic, metaheuristic, or exact solutions approaches.

So far, despite proposing approaches, they lack the feasibility of the accident's problematic situation. The best explanation is that the CARP is a nondeterministic polynomial problem (NP-hardness). Therefore, typically, heuristic and metaheuristic approaches are utilized to efficiently tackle CARP with an ambulance's optimal route to the accident. Generally, several algorithms have been used to solve CARP problems, such as particle swarm optimization, genetic algorithm, ant colony, and simulated annealing. The various metaheuristics solution qualities are not significantly different for these optimization problems [12]. The methodology of this article is based on the bat algorithm and convolution neural network.

Certain problem classes are currently best solved with evolutionary algorithms, many of which have several advantages, such as simplicity and flexibility [13,14]. In 2010, Yang proposed the bat algorithm (BA) [15] as an example of an evolutionary algorithm. It has been designed on the ability of the echolocation of bats. Specifically, the species Microchiroptera is used for a more extensive form of this ability to differentiate in the diverse dark types of insects. After the creation of the BA and verified its efficiency, several applications in several areas emerged, such as (1) image processing [16,17], (2) data mining [16,18], (3) solving the numerical optimization problem [19], and (4) optimization area [20]. They were developed, and their flexibility is evident [14,17]. The basic BA has several advantages over traditional global optimization methods [13].

The contribution of this study is using the bat algorithm and convolutional neural network (BA-CNN) to find the best route between the injured or patient person and the best and most suitable hospital. It aims to take transfer the patients confidentially, accurately, and quickly.

The structure of the following sections is as follows: the literature on ambulance routing algorithms is introduced in Section 2. Section 3 describes the methodology of the proposed algorithm. Section 4 discusses the performance evaluations of the algorithm. Section 5 presents the conclusion of this article.

2. Literature Review

The shortest path (between the accident spot and the hospital) algorithm was proposed for ambulance routing in work [9]. In the set of node locations, Dijkstra's algorithm was utilized for discovering the shortest path. The authors in reference [10] implemented effective communication metrologies using an ambulance to communicate traffic lights. To alter the traffic light signal, the authors utilized a vehicle to infrastructure communication technology. Via a communication protocol, the system first sends a short-wave message about the ambulance locations and routes to the core network. According to the position of the ambulance, the core network controlled the traffic light signals.

To calculate the shortest path between the hospital and the accident place, the authors in reference [12] utilized the star algorithm. Furthermore, the algorithm can transfer it to the ambulance route control system. The authors in reference [21] used the video image processing concept to control the traffic light signals for allowing emergency vehicles to pass (e.g., fire engines and ambulances). The image processing technique analyses vehicle density. Additionally, the vehicle's RFID signal is received by the system, and then accordingly, the traffic light is controlled, and then minimize the congestion of traffic. Similarly, for emergency vehicle routing applications, RFID-based traffic congestion control was suggested in reference [22]. The RFID system reads all vehicles' information via the roadside capturing unit. The details are then sent to the control unit of the traffic main. For safe passing, in case of passing the emergency vehicle of the roadside unit, the traffic signal is changed by the main traffic control unit.

For the smart traffic light system, the authors in reference [23] proposed a miniature model. The system consists of two units installed in the main control unit and emergency vehicle. The emergency condition of the emergency vehicle is sent to the vehicle's module to the main control unit. Then, the traffic signal is controlled by the main control unit to the corresponding direction and then allows the emergency vehicle. For smart city applications, a smart traffic signal control system was proposed in reference [24]. The system utilized the following modules: (1) public transport signal priority, (2) adaptive traffic signal control, (3) emergency vehicle signal pre-emption, (4) message broadcasting, and (5) roadside unit control. In the beginning, with the vehicle near the intersection point of the traffic light signal, the emergency vehicle's direction and position are shared. Then, the traffic near the intersection point is controlled to allow the vehicle. For emergency vehicles routing nearby traffic signals, a novel method using the Li-Fi protocol was proposed in reference [25]. With some fixed modulated frequency, the Li-Fi model is fixed in emergency and nonemergency vehicles and the systems of traffic light control. The emergency vehicle (in a stuck case) uses Li-Fi protocol to communicate with the nearest traffic light control systems to change the traffic signal.

Emergency vehicle clearance control system-based IoT was proposed in reference [26]. The system consists of (1) traffic signal RFID control, (2) vehicle RFID control, and (3) the main control server. The system operates as follows: first, the main control server receives the emergency vehicle position and location. Second, based on the route traffic, these details are sent to the emergency vehicle. Accordingly, the vehicle RFID control is implemented at nearby intersection points to alter the traffic signal. After the vehicle passes, the traffic signal schedule is changed to normal by traffic signal RFID. The authors in reference [27] suggested a new smart traffic light system. The system is based on (1) Google maps, (2) micro-controlled traffic lights, (3) MQTT protocol, and (4) Android apps. In the beginning, the driver utilizes a Google map to determine the shortest path to the hospital (minimum time spent at each traffic signal in the route) [28,29]. Via MQTT protocol, it is sent to the main traffic control center. Then, according to the ambulance near the traffic signal, the traffic signal is changed by the main traffic control center.

The authors in reference [30] proposed traffic light control utilizing Internet of Things (IoT) technology. For the emergency vehicle's clearance, the traffic signal is controlled by GPS and application servers. According to the vehicle position, the traffic light signal is controlled by the application server. For emergency vehicle clearance, a traffic management system-based fuzzy logic was proposed in reference [31]. First, the method sends the vehicle's emergency level to the traffic management system (TMS).

According to the emergency level, the TMS fetches the following information: (1) occupancy level of the road, (2) average vehicle speed of the corresponding road, and (3) road map. Then, fuzzy logic control receives the information and obtains the congestion level and congestion vehicle to the TMS. Next, for fast clearance to the destination, the emergency response route and plan are then generated by the TMS and transferred to the emergency vehicle. For the emergency vehicle, a routing technique-based genetic algorithm was proposed in reference [32]. In the beginning, the following details are received by the genetic algorithm: (1) origin and destination data, (2) the current position of the vehicle, and (3) traffic data. Accordingly, for the emergency vehicle, the shortest path and new feasible route are determined.

In reference [33], the emergency vehicle routing problem is solved using an improved genetic algorithm. The route node's mathematical model was developed using graph theory, and the genetic algorithm solves it by origin and destination details of the emergency vehicle. The author in reference [34] proposed mixed integer programming to tackle the ambulance routing issue. This algorithm receives details of the victims, the number of ambulances available near the accident place, and hospital details, and this details program solves the best route for the ambulance. In reference [35], distributed ant colony optimization was used for the navigation of the emergency vehicle. Optimal routing was developed for the ambulance to quickly reach the accident place using ant colony optimization.

In reference [36], an emergency vehicle routing schedule is generated using the Google map distance matrix API's real-time traffic data. In this method, mixed integer programming generates vehicle routing based on incident location, vehicle tracking, and Google map routes. In reference [37], the ambulance routing problem was solved using the petal algorithm with particle swarm optimization (PSO). The problem was framed using open vehicle routing or vehicle routing problem with pickup and delivery concept. In reference [38], a multiobjective model-based ambulance routing algorithm was developed. This method sends accident location and victim injury levels to the algorithm's input to generate the ambulance's optimal route schedule.

In reference [39], the ambulance routing techniques-based clustering concept is developed. In this method, the shortest distance between the patient and the hospital is generated based on the following algorithms: K-means algorithm, weighted K-means, and density-based cluster algorithm. In reference [40], the emergency vehicle's optimal routing is developed using critical healthcare HTH service and real-time critical healthcare HTH service. The IoT sensor is placed on the roadside, and it sends the data (i.e., traffic level and the number of vehicles on the road). Accordingly, the algorithm generates the possible and the shortest routes for the ambulance. In reference [41], GSM and cloud-based automatic control of traffic signals for emergency vehicle clearance are developed. The GSM module in the emergency vehicle sends the details to the cloud via an android mobile application. Based on this data, the control unit and the cloud server alter the traffic signal to clear the vehicle route.

In references [42,43], the deep learning-centered intrusion detection system (IDS) and hybrid machine learning are developed for detecting the malicious attacks in VANET.

The modified K-harmonic means-based clustering was developed in the deep learning-based IDS to reduce the propagation delay. However, these IDS systems were mainly concentrated on the classification; it is not concentrated on the routing between the vehicles.

The trust-based model (TBM) identified the rogue nodes in a vehicular network [44]. Next, the trust value of the node was estimated by TBM, and it was used to detect the rogue node. The developed TBM was used to provide secure data broadcasting with less overhead. However, this TBM was mainly dependent on the trust value of the nod during the relay node selection. In reference [45], the trust based on vehicles and roadside units (TVR) optimized the cluster head selection. The cluster head selected from the TVR was considered a highly trusted node used to stabilize their cluster. However, the increment in the transmission range affected the performance of TVR. Moreover, a secure ad-hoc on-demand distance vector (SAODV) method [46] was developed to identify the black hole nodes in the VANET. However, appropriate fitness function values such as trust and distance were not considered while generating the transmission path.

3. Modelling of the BA-CNN Algorithm

This section analyses the ambulance vehicle routing logic optimization methodology using the (BA-CNN) algorithm. Figure 1 shows the general scenario of the ambulance vehicle routing. The injuries or people nearby the accident are informed of the accident's location. The number of injured people and their injured level are sent to the control station for rescue. To convey all the accident details, the control station members contact the hospital near the accident location and the ambulance station to transfer all the accident details. The hospital is then prepared to deliver the necessary facilities to treat the injuries. The vacant ambulance near the accident location is then contacted via the ambulance station to transfer the details. Next, the vehicle moves to the accident location and carries the injuries to the nearest hospital. In the diagram, the road transport concept is indicated by dash lines. At the same time, in ambulance vehicle routing, the mobile communication concept is referred to by the dot-dash line. For ambulance vehicle routing, Figure 1 depicts the routing of an ambulance vehicle as follows: the road transport concept is illustrated with the dashed line, whereas the mobile communication concept is presented with the dot-dash line.

Figure 2 shows the process of ambulance routing in the station. The details of the accidents are sent to the ambulance station. For the accident, the details of the local route map and local node details are fetched by the ambulance station. Then, the bat algorithm is fed with the gathered details to determine the best routes from the ambulance location to the accident and then to the hospital. The route is generated by the BAT algorithm trained by neural networks for all possible combinations of sources and destinations. The neural networks provide the optimal routing plan from the trained network. Via mobile communication, the details are sent to the ambulance driver. Accordingly, the driver follows the route to take the injuries to the hospital.

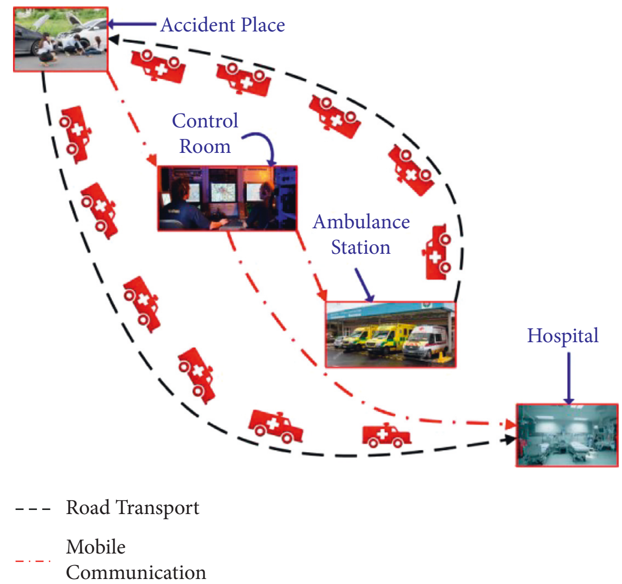


Figure 1: The scenario of the ambulance vehicle routing concept.

3.1. Mathematical Model. With minimum time and resources, the intelligence actions are utilized to construct the direction of the ambulance without following the unwanted routes in the available route map and meet the target. The vehicle must either follow the shortest route or accelerate over the trustworthy node to meet the target and satisfy the above requirements. Here, the node's trustworthiness is identified by calculating the trust value for the node. The shortest path means the minimum distance between the target and the source. The minimum distance can be determined as follows:

$$
\begin{aligned}
F_1 &= \mathrm{distSD}(i) \\
&= \sqrt{\left[\left(S_x(i) - D_x(i)^2\right)\right] + \left(S_y(i) - D_y(i)\right)^2},
\end{aligned} \tag{1}
$$

where F_1 denotes the fitness function. The distance between the source and destination locations is denoted by $\mathrm{distSD}(i)$, and i defines the current iteration. Additionally, $S_y(i)$ and $S_x(i)$ refer to the y and x source location coordinates, whereas $D_y(i)$ and $D_x(i)$ are the y and x destination location coordinates, respectively.

In monitoring the shortest direction, undesirable route positions must be paid attention to by the vehicle. Otherwise, the emergency vehicle consumes more time (delay) in the overall process by taking the unwanted routes. The vehicle must keep the maximum distance from the unwanted route locations to avoid these situations. The following equation is used to calculate the maximum distance:

$$
\begin{aligned}
F_2 &= \mathrm{distAU}(i) \\
&= \sqrt{\left[\left(A_x(i) - U_x(i))\right)^2 + \left(A_y\left(i - U_y(i)\right)\right)^2\right]},
\end{aligned} \tag{2}
$$

where F_2 denotes the fitness function. The distance between the ambulance vehicle and unwanted location is denoted by $\mathrm{distAU}(i)$, and i defines the current iteration. $A_y(i)$ and

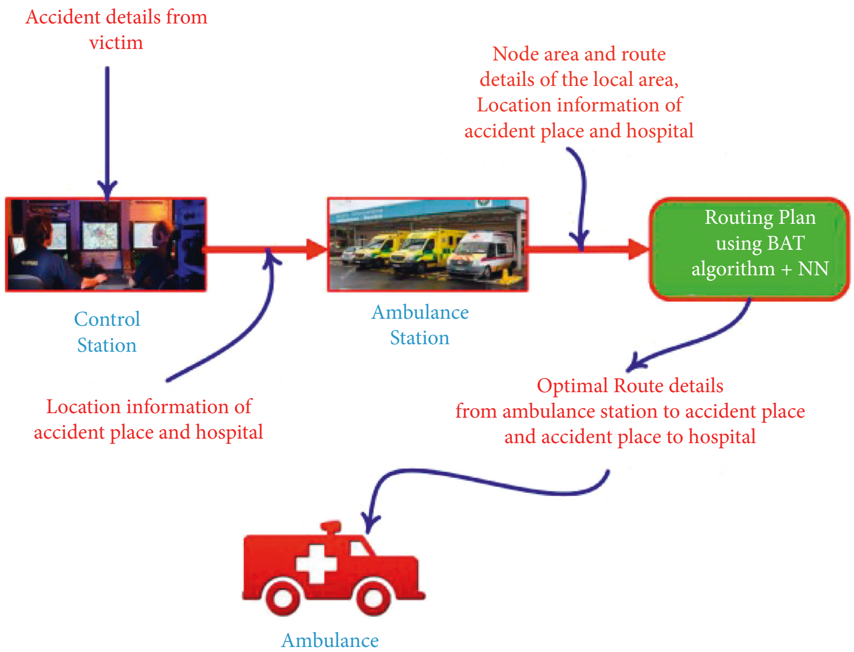


FIGURE 2: The process of the ambulance routing in the ambulance station.

$A_x(i)$ are the ambulance vehicle location coordinates, and $U_y(i)$ and $U_x(i)$ are the unwanted route location coordinates.

In this network topology, the trust value of each node is the combination of direct and indirect trust values. The direct trust (DT) is also referred to as a local trust value, which is generally identified based on the packet sending behavior, as expressed in the equation (3). Moreover, the indirect trust (IT) is calculated based on the experience of the remaining nodes that exist in the network, which is used in the decision-making process. Equation (4) shows the indirect trust and combination of trust is expressed in equation (5). The trust value in the objective function is to provide robustness against the malicious nodes:

$$\mathrm{DT}_{i,j}(t) = \frac{B_{i,j}(t)}{C_{i,j}(t)}, \tag{3}$$

$$\mathrm{IT}_{i,j}(t) = \frac{1}{q}\sum_{l=0}^{q} \mathrm{DT}_{l,j}, \tag{4}$$

where the number of packets correctly transmitted and received between node i and j in time t is represented as $B_{i,j}(t)$ and $C_{i,j}(t)$, respectively; the neighbors of the i th node are denoted as q:

$$F_3 = \mathrm{DT}_{i,j}(t) + \mathrm{IT}_{i,j}(t). \tag{5}$$

For the ambulance vehicle routing, the fitness function or objective function is calculated using the following equation:

$$F = F_1 + \frac{1}{1+F_2} + F_3. \tag{6}$$

The objective function (6) determines the best local and the best points of the global path. Therefore, according to the best global direction point, the new route point for emergency vehicles is obtained. This loop is repeated until the ambulance reaches its target.

3.2. BAT Algorithm. The echolocation activity of bats is the main idea of the bat algorithm. The activity comprises two steps: first, echolocation, a very intense sound pulse, is emitted. Then, they listen to the echo that can get back from the around objects. The amplitudes of signals are varied depending on their nature. Each sound signal is characterized by a specific emission rate of loudness, duration, and pitch. Specifically, bats travel at random routes and speeds, with variable pulse emission rate, loudness, and length [47,48].

The majority of bats use fixed frequency signals, whereas some use tuning frequency signals. Usually, the frequency range of bats varies from 150 kHz and 25 kHz. Therefore, BAT algorithms concepts are as follows: in the beginning, bats use echolocation. Next, they separate the distinction between obstacle and victim. Figure 3 illustrates the flow diagram of the bat algorithmic rule.

The procedure can be clarified as follows:

Step 1. Bat position $(X_i = X_1, X_2, X_3 \ldots X_d)$ and velocity $(V_i = V_1, V_2, V_3 \ldots V_d)$ are initialized. The number of optimized parameters of the system is symbolized with d. For the initial BAT position (X_i), the following process $F(i)$ discovers the objective function value. The calculations are performed as follows:

$$X(i) = \mathrm{rand}(n,d); \quad n \text{ is the number BAT population}, \tag{7}$$

$$V(i) = \mathrm{rand}(n,d); \quad n \text{ is the number BAT population}, \tag{8}$$

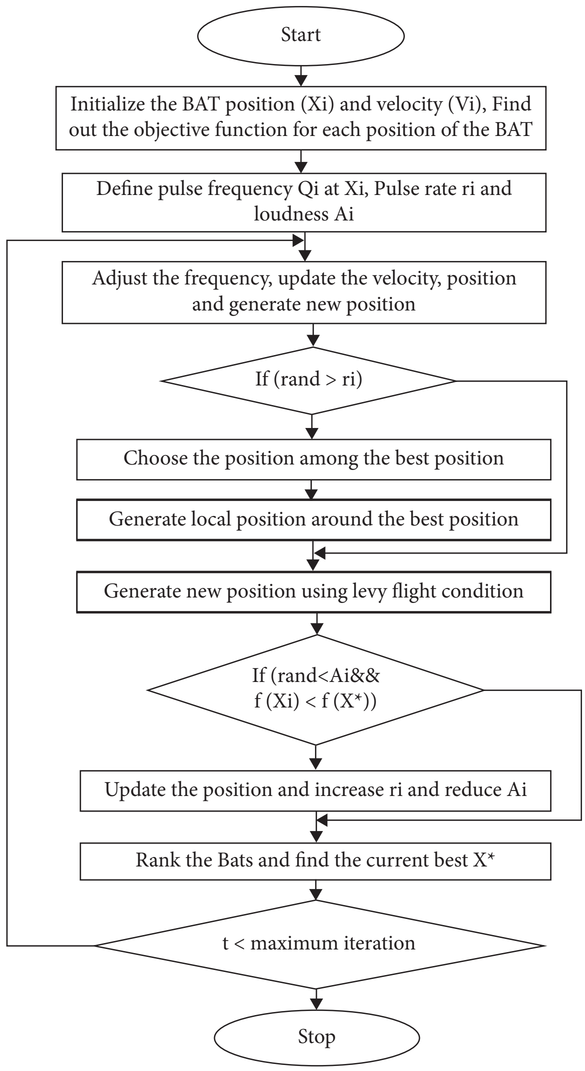


Figure 3: Flowchart for the bat algorithm.

$$F(i) = \text{Fitness value calculated based on.} \tag{9}$$

Lastly, to manipulate the bat algorithm further, store the objective function value and the corresponding position of the bat. equation (6).

Step 2. In this step, we determine the following bat's values: loudness factor (A_i), pulse rate (r_i), and pulse frequency (Q_i).

Step 3. In this step, start bat algorithm's essential loop and beginning of iteration from 1. Here, we fix the maximum iteration for the main loop.

Step 4. We change bats' frequency parameters using maximum and minimum frequency values. The following formula can perform this process:

$$Q_i = Q_{\min} + \left(Q_{\max} - Q_{\min}\right) * \text{rand}(n, d), \tag{10}$$

where $Q_{\max}$ and $Q_{\min}$ represent the maximum and minimum bat frequency, respectively. Additionally, rand (n, d) represents the rand number between one to zero.

Alter the bat's speed to determine the best frequency and position. It can be performed as follows:

$$V(t) = V(t-1) + \left(X(t-1) - X^*\right)Q_i, \tag{11}$$

where Q_i refers to the bats' frequency, and the best position among all bat positions is defined by X^*. Additionally, $X(t-1)$ is the last bat iteration position. Lastly, the current and previous bat iteration velocities are denoted by $V(t)$ and. $V(t-1)$

From the following equation, we update bats' position by using bats' updated velocity:

$$X(t) = X(t-1) + V(t). \tag{12}$$

As formulated below, adding the bats' old position with their loudness factor determines their optimal position:

$$X(\text{new}) = X(t) + \text{rand} * A_i(t). \tag{13}$$

Step 5. In this phase, a comparison is carried out between the pulse rate of the bats and the random pulse rate. When the pulse rate is less than a random pulse, the algorithm performs the following functions:

(1) Determine a solution between the best solution to the problem.
(2) The local solution is created around the determined best solution.
(3) Update the bats' loudness factor and pulse rate by the following equation:

$$\begin{aligned} r_i(t) &= \delta * r_i(t-1), \\ A_i(t) &= A_i^0\left(1 - e^{(-\varnothing t)}\right), \end{aligned} \tag{14}$$

where $\varnothing$ and δ are random numbers by considering them as $(0 < \delta < 1$ and $\varnothing > 0)$ and considering iteration reach infinite. Accordingly, the loudness factor and pulse rate are calculated as follows:

$$\begin{aligned} t &\longrightarrow \infty, \\ r_i(\infty) &\longrightarrow 0, \\ A_i(\infty) &= A_i^0 \text{ as.} \end{aligned} \tag{15}$$

Last, levy flight conditions are utilized to create a new solution.

Step 6. If (rand $< Ai$ & F $(Xi) < F(X^*)$)

Here, a comparison is carried out between the loudness factor and the random loudness factor. Additionally, it is also compared to the current solution with the best solution. Consequently, by satisfying both conditions, the bats' loudness factor and pulse rate are changed (increased or reduced) according to formula (13).

End if

The solution of the bats is sorted, and calculate the current best solution is X^*.

End while

Loop end

Step 7. The bat algorithm presents the best solution.

3.3. Convolutional Neural Network. The CNN used in this research is ResNet, combined with a bat algorithm to identify the best and most secure relay nodes to communicate the VANET. The residual block is utilized by the ResNet [49], which is used to solve gradient disappearance and degradation issues in the general CNNs. The residual block extends the network depth and enhances the network performance.

Figure 4 shows the architecture of the residual block, where the residual is $F(x)$ and the residual block has input is x. In the first layer, after accomplishing the linear transformation and the activation, the $F(x)$ is the gained output. Next, the second layer's input is added to the $F(x)$ after performing the second layer's linear transformation. The ReLU is used to activate it to acquire the output. Next, x is inserted with the output's second layer. Then, ResNet is utilized to activate it. This path is referred to as a shortcut connection.

$$F = W_2\sigma(W_2 x), \tag{16}$$

$$y = F(x, \{W_i\}) + x, \tag{17}$$

$$y = F(x, \{W_i\}) + W_s x. \tag{18}$$

Equations (16)–(18) show the operation of the residual block, where σ in equation (14) denotes the nonlinear function ReLU. In equation (15),y specifies the general output of the shortcut and second ReLU. Several channels and a linear transformation W_s are accomplished over x using the shortcut operation when the dimensions of input and output are needed to alter in the ResNet.

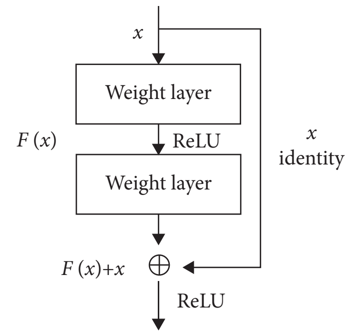


Figure 4: Block of ResNet.

Table 1: BAT algorithm specification.

Parameter	Value
Maximum number of iterations	100
Bat population	100
Minimum frequency	0
Maximum frequency	1
Loudness factor	0.5
Pulse rate	0.5

Table 2: The details of the coordinates.

Coordinates		Values
x coordinate		(−10)–(+10)
y coordinate		(−10)–(+10)
Unwanted route location points	Diameter	0.2, 0.3, 0.2, 0.2, 0.4, 0.2, 0.2, 0.2, 0.2, 0.5, and 0.2.
	x coordinate	1.5, 4.0, 1.2, 8.0, 1.2, 4.8, 4.9, 5.1, 2.8, 5.6, and 7.3,
	y coordinate	4.5, 3.0, 1.5, 9.0, 5.6, 4.8, 6.3, 6.5, 8.9, 1.4, and 7.6

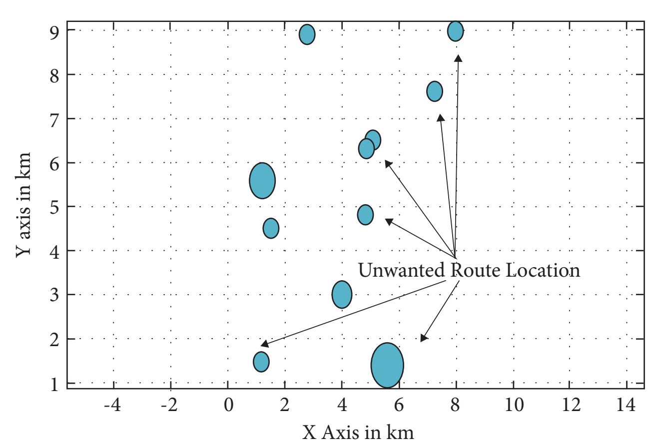


Figure 5: Unwanted node details of the considered area.

4. Performance Evaluation

This section conducts simulation experiments of an ambulance vehicle routing via MATLAB R2017a (Core i3 Intel processor and a 1.7 GHz PC). Table 1 summarizes the assumptions assigned to the algorithm:

4.1. Evaluation of the Routing Process. The coordinate details of the coordinates are listed in Table 2. Additionally, in the considered area, the unwanted node details are shown in Figure 5.

Figure 5 revealed 11 unwanted nodes, showing us the unwanted route location. These nodes are selected randomly. Two cases are used to test the algorithm between the coordinates: (1) 0.7–4.2 and (2) 4.2–7.1. Figures 6(a) and 6(c) show the corresponding results of route variation at every iteration. Figures 6(b) and 6(d) show the convergence graph of both cases. Accordingly, the total distance between the coordinates of Case 1 (0.7–4.2) during the iteration start is around 15.5 km. The figure revealed that the distance is reduced by increasing the number of iterations. Therefore, the distance became 7.5 km at the end of the iterations (i.e., 100). The algorithm needed 34 sec only to find the optimum route.

Figure 6(d) illustrates the total distance between the coordinates (4, 2) to (7, 1) is around 11.3 km from the iteration's starting. Additionally, the distance is decreased at the end of the iterations (i.e., 100) to around 3.7 km. Accordingly, the optimal route is calculated and found within 36 seconds. Besides, the algorithm is evaluated using the following sample coordinates of accidents, hospital, and ambulance locations (Table 3).

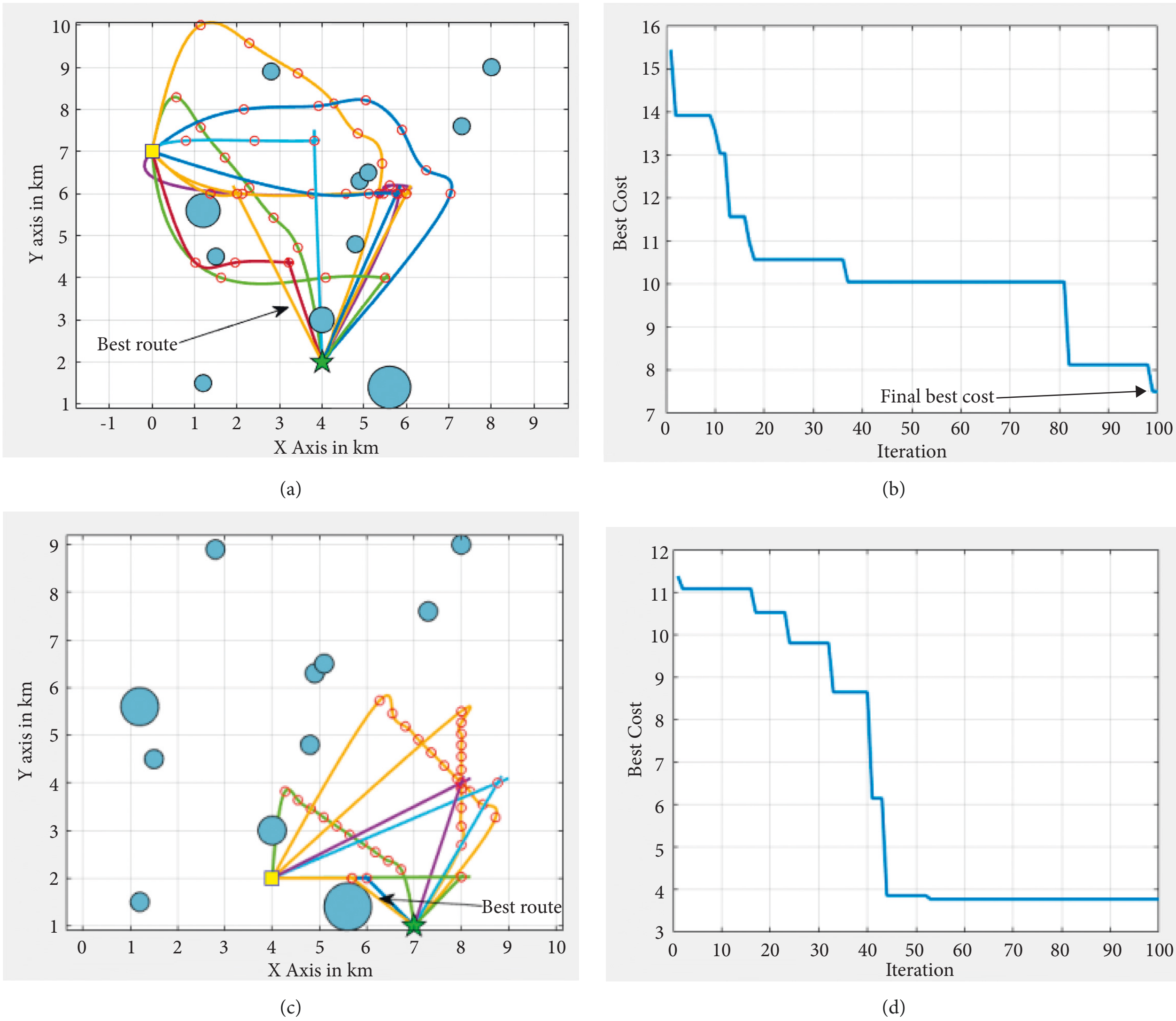


FIGURE 6: The outcomes of two test conditions. (a) Best route between (0, 7) to (4, 2). (b) Convergence graph. (c) Best route between (4, 2) to (7, 1). (d) Convergence graph.

TABLE 3: Sample coordinates of accidents, hospital, and ambulance locations.

			Cases		
			1	2	3
Location	Accident	X	4	0	9
		Y	2	6	9
	Hospital	X	7	3	6
		Y	1	4	7
	Ambulance	X	0	8	7
		Y	7	5	9

Figure 7 shows the outcomes of the simulation of Case 1. The findings show that the optimum distance between the accident and the ambulance station is 8.2 km. Furthermore, the distance between the hospital and the accident place is optimized at 8.2 km. The algorithm calculated the optimum route in 63 secs.

Figure 8 illustrates the simulation outcomes of Case 2. The findings show that the ambulance station's distance to the accident is optimized at 9.1 km. Furthermore, the optimum distance between the accident location to the hospital is 4.6 km. The algorithm calculated the optimum route in 62 secs. Similar to Figure 8, but for Case 3, the same parameters are simulated and depicted in Figure 9. The findings showed that the distance between the ambulance location to the accident is optimized to 2.4 km. The optimum distance between the accident location to the hospital is 3.6 km. the algorithm calculated the optimum route in 40 secs.

The following process proposed is to train the ResNet by collecting data from the bat algorithm. The route coordinates from this bat algorithm (the bat algorithm's outputs) for all possible sources and destination points are taken as the input for the ResNet. Figures 10(a) and 10(b) illustrate the relation between input source-destination coordinates with output x and y coordinates from the bat algorithm, respectively. These input and output coordinates are taken as training data for the ResNet.

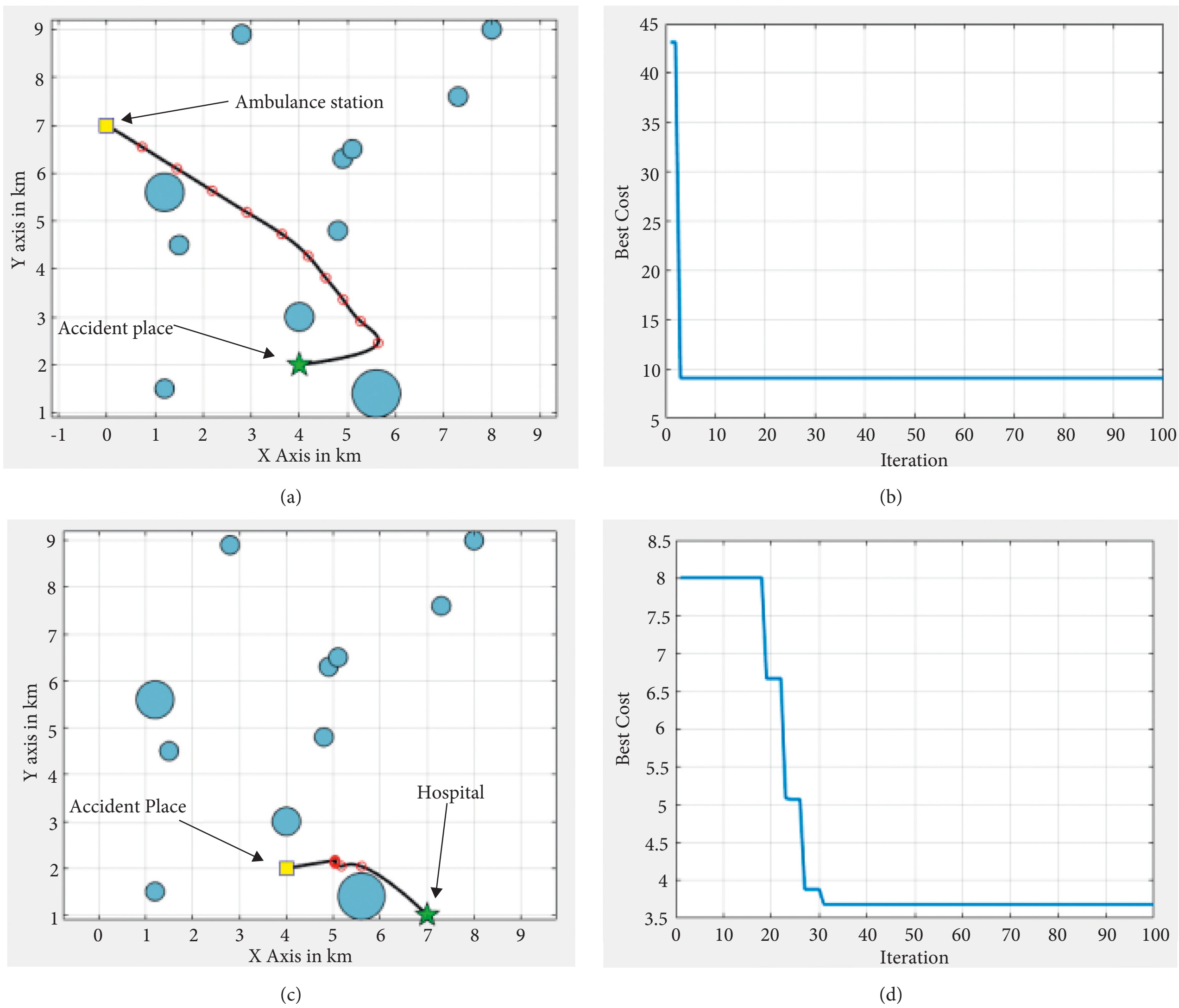


FIGURE 7: The outcomes of the simulation of Case 1. (a) Route from ambulance location to the accident. (b) Convergence graph. (c) Route from accident location to the hospital. (d) Convergence graph.

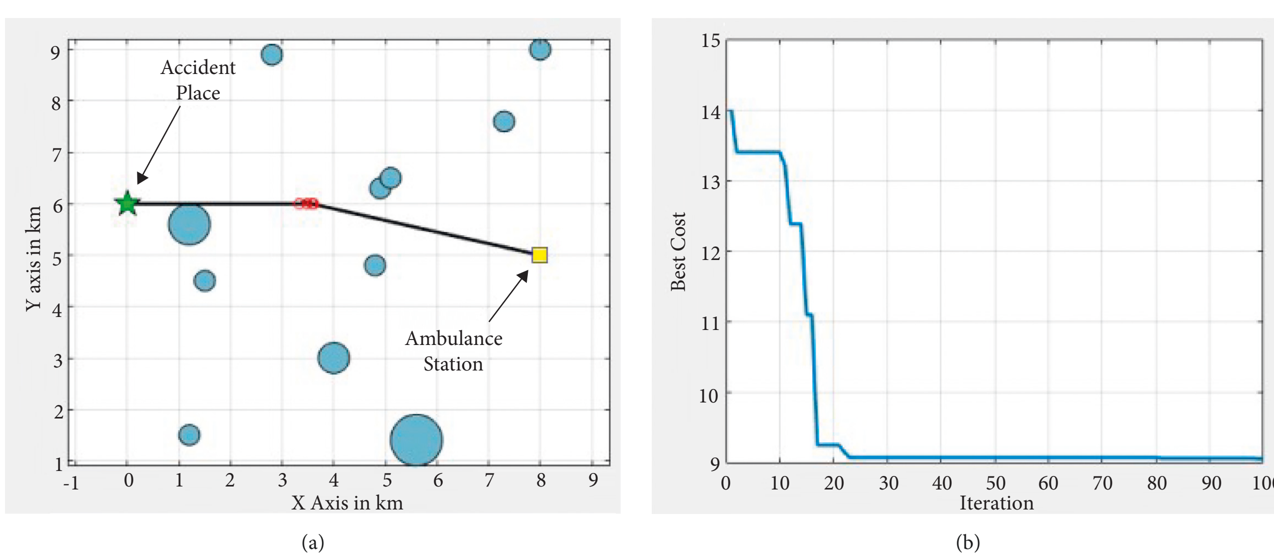
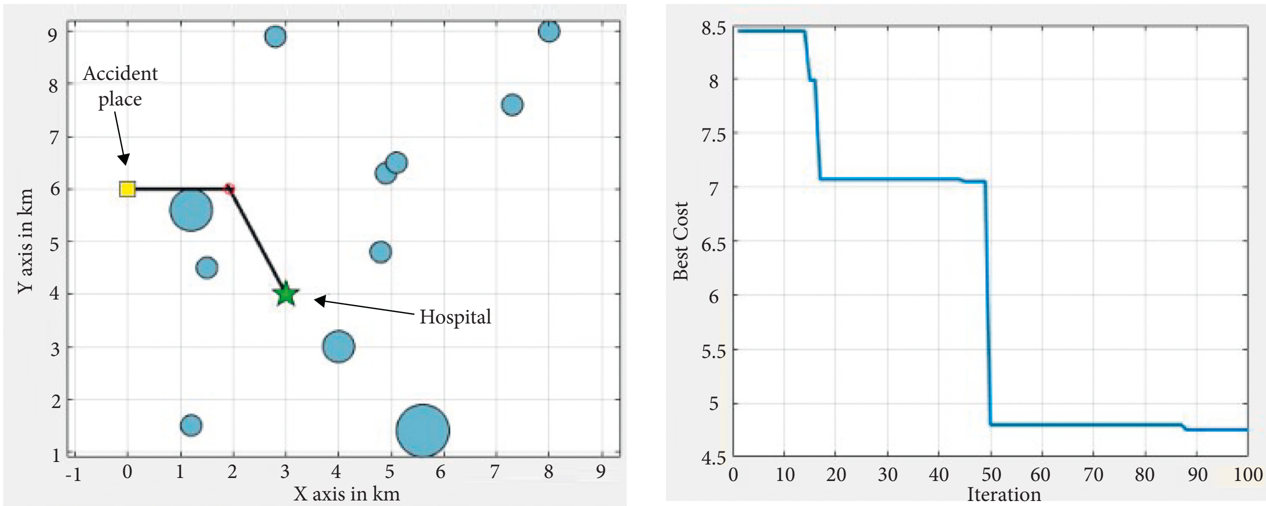
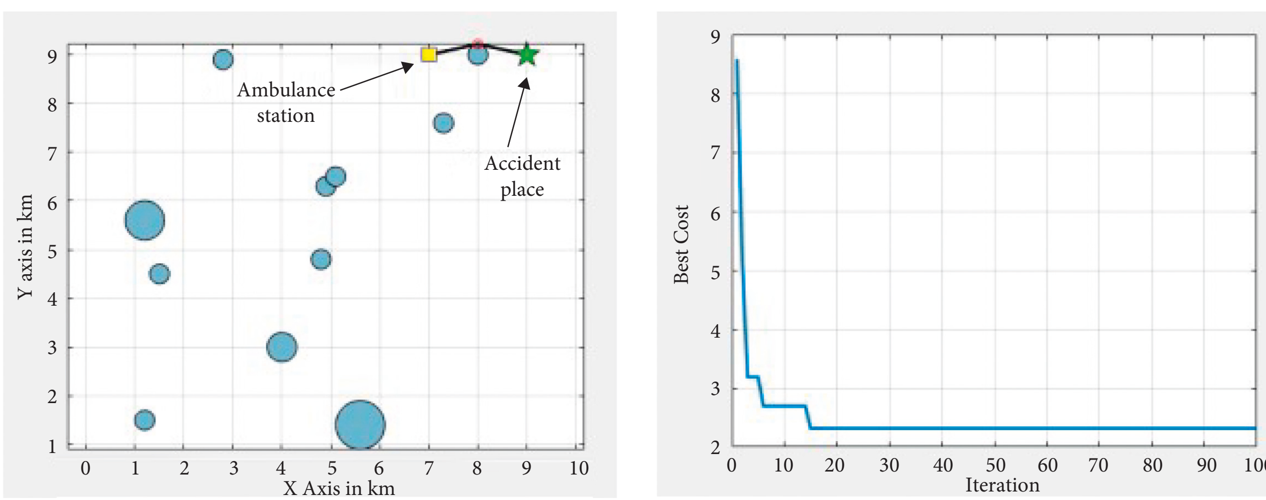


FIGURE 8: Continued.

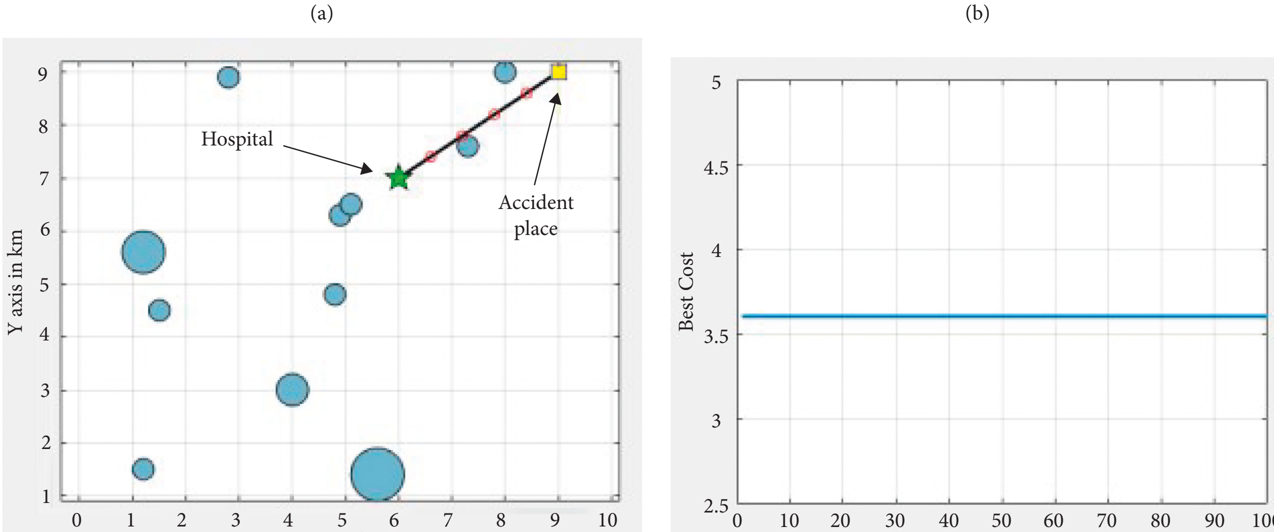


(c) (d)

Figure 8: The outcomes of the simulation of Case 2. (a) Route from ambulance location to the accident. (b) Convergence graph. (c) Route from accident location to the hospital. (d) Convergence graph.



(a) (b)



(c) (d)

Figure 9: The outcomes of the simulation of Case 3. (a) Route from ambulance location to the accident. (b) Convergence graph. (c) Route from accident location to the hospital. (d) Convergence graph.

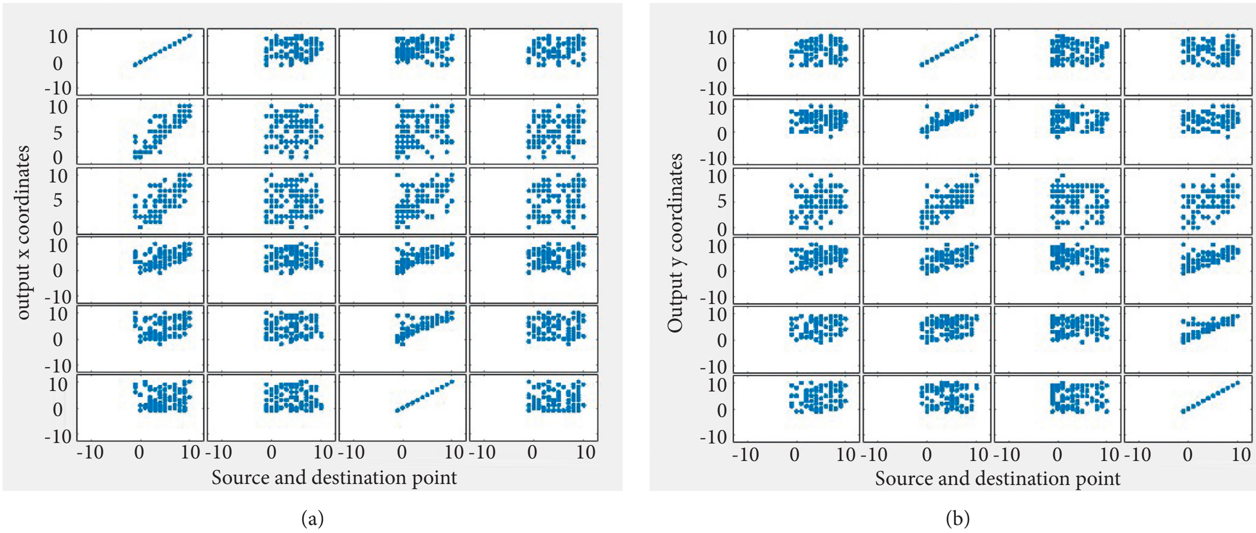


Figure 10: Training data for the ResNet from bat algorithm. (a) Input coordinates vs. output x coordinates. (b) Input coordinates vs. output y coordinates.

Table 4: Comparative analysis of BA-CNN.

Performances	Methods	Number of malicious nodes 5	10	15	20	25
PDF	TBM [44]	0.61	0.46	0.41	0.35	0.26
	TVR [45]	0.83	0.79	0.73	0.72	0.63
	SAODV [46]	0.80	0.68	0.62	0.59	0.55
	BA-CNN	0.96	0.90	0.89	0.83	0.76
Throughput (kbps)	TBM [44]	148	159	150	146	122
	TVR [45]	153	163	168	157	148
	SAODV [46]	149	161	161	150	139
	BA-CNN	161	173	175	162	158
Routing load	TBM [44]	0.1	0.28	0.78	1.19	1.05
	TVR [45]	0.08	0.15	0.62	1.01	0.9
	SAODV [46]	0.1	0.21	0.71	1.15	1.00
	BA-CNN	0.05	0.10	0.34	0.87	0.95
EED (s)	TBM [44]	0.110	0.110	0.135	0.151	0.180
	TVR [45]	0.090	0.100	0.124	0.147	0.150
	SAODV [46]	0.090	0.130	0.130	0.150	0.162
	BA-CNN	0.005	0.019	0.100	0.109	0.110

4.2. Performance Analysis. The performance of the BA-CNN is analysed with the 100 vehicles with varying malicious nodes 5, 10, 15, 20, and 25. Here, the network is analysed in the network size of 500 m × 500 m with a mobility speed of 10–40 m/s. The efficiency of the BA-CNN is analysed by comparing it with three existing methods as TBM [44], TVR [45], and SAODV [46]. The metrics used in performance evaluation are an end-to-end delay, routing load, packet delivery fraction, and throughput.

The comparative analysis of the BA-CNN with existing methods such as TBM [44], TVR [45], and SAODV [46] are shown in Table 4 and Figure 11. Here, the comparison is made by varying the malicious nodes into 5, 10, 15, 20, and 25. The analysis determined that the BA-CNN outperforms well than the TBM, TVR, and SAODV [46]. The main reason for BA-CNN with the better performance is the better relay node selection using the integration of BA and CNN (i.e., ResNet). The packet delivery of the BA-CNN is improved by avoiding the malicious nodes while transmitting the data packets over the VANET. Accordingly, the successful data transmission resulted in higher throughput for BA-CNN. Since the BA-CNN uses the optimal fitness functions during the generation of the transmission path, it does not require a high amount of control packets for generating the transmission path. The fitness functions used in the BA-CNN are the distance between source and destination, the distance between ambulance vehicle and unwanted location, and trust. Therefore, this BA-CNN achieves less routing load than the TBM [44], TVR [45], and SAODV [46]. Moreover, the shortest path generation

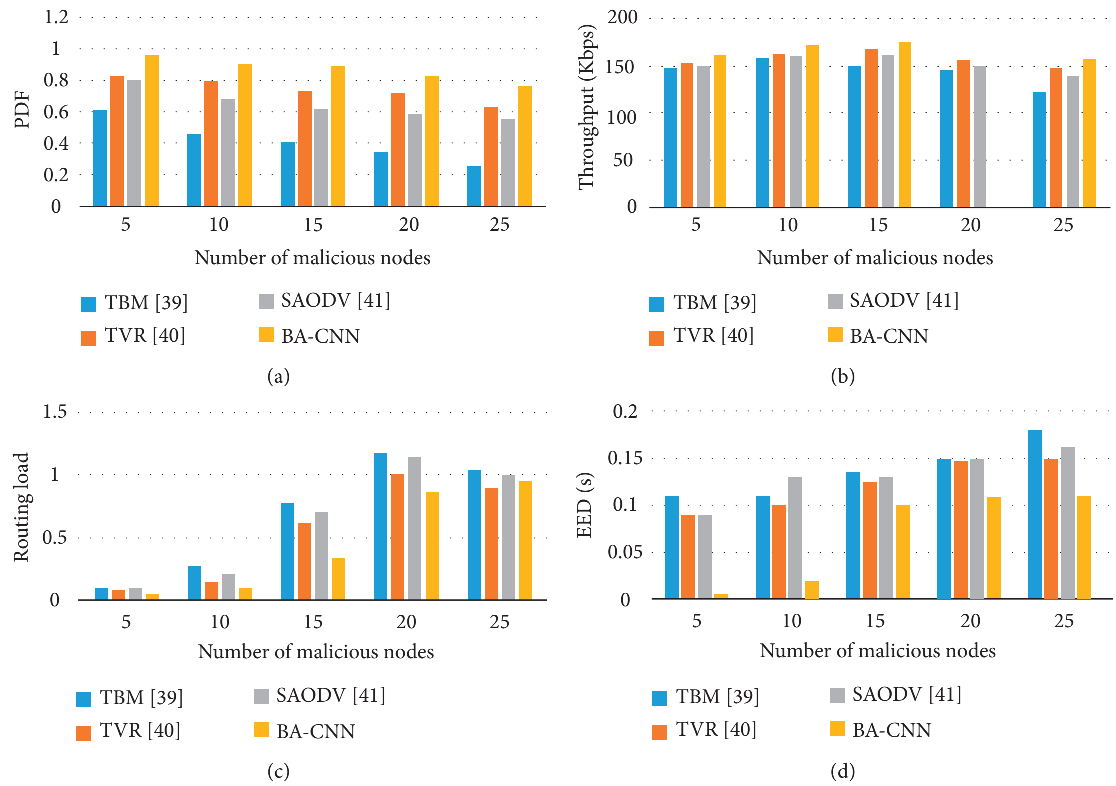


Figure 11: Performance comparison of BA-CNN. (a) PDF. (b) Throughput. (c) Routing load. (d) EED.

using BA-CNN is used to obtain less EED than the TBM [44], TVR [45], and SAODV [46].

5. Conclusion

Nowadays, management systems have become a substantial concern in different fields, for example, ambulance applications, military, banks, healthcare industry, enterprises, and educational centers. As we have seen, ambulance vehicle routing is a vital challenge in health and processing in smart cities. This work proposed the BA-CNN routing algorithm of ambulances following an accident. We observed that the proposed (BA-CNN) could achieve better solutions through extensive simulation experiments. In all scenarios, the system provided shorter total tour distances quickly. Accordingly, it can be argued that BA-CNN is resilient against the changes in the number of locations and the underlying topologies. Therefore, it can be seen that the bat algorithm continuously generates better results despite the different significant problem parameters. The experiments demonstrated the superiority of the BA-CNN algorithm over its counterparts (TBM, TVR, and SAODV). Furthermore, for ten malicious nodes, the PDF of the algorithm was 0.90, which is higher than the counterparts. For future works, the authors recommend using other new metaheuristic methods to improve the performance of the system.

Data Availability

No data were used to support this study.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. Latif, Z. Idrees, J. Ahmad, L. Zheng, and Z. Zou, "A blockchain-based architecture for secure and trustworthy operations in the industrial Internet of Things," *Journal of Industrial Information Integration*, vol. 21, Article ID 100190, 2021.

[2] M. J. Baucas, S. A. Gadsden, and P. Spachos, "IoT-based smart home device monitor using private blockchain technology and localization," *IEEE Networking Letters*, vol. 3, no. 2, pp. 52–55, 2021.

[3] H. Mora, J. C. Mendoza-Tello, E. G. Varela-Guzmán, and J. Szymanski, "Blockchain technologies to address smart city and society challenges," *Computers in Human Behavior*, vol. 122, Article ID 106854, 2021.

[4] X. Zhang and Y. Wang, "Research on intelligent medical big data system based on Hadoop and blockchain," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–21, 2021.

[5] B. Adanur, B. Bakir-Güngör, and A. Soran, "Blockchain-based fog computing applications in healthcare," in *Proceedings of the 2020 28th Signal Processing and Communications*

Applications Conference (SIU), pp. 1–4, Gaziantep, Turkey, January 2020.

[6] M. J. Gul, B. Subramanian, A. Paul, and J. Kim, "Blockchain for public health care in smart society," *Microprocessors and Microsystems*, vol. 80, Article ID 103524, 2021.

[7] W. Ahmad, A. Rasool, A. R. Javed, T. Baker, and Z. Jalil, "Cyber security in iot-based cloud computing: a comprehensive survey," *Electronics*, vol. 11, no. 1, p. 16, 2021.

[8] M. Majid, "Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: a systematic literature review," *Sensors*, vol. 22, no. 6, p. 2087, 2022.

[9] P. Arunmozhi and P. J. William, "Automatic ambulance rescue system using shortest path finding algorithm," *International Journal of Science and Research*, Article ID 20131836, 2014.

[10] C. Suthaputchakun and Y. Cao, "Ambulance-to-Traffic light controller communications for rescue mission enhancement: a Thailand use case," *IEEE Communications Magazine*, vol. 57, no. 12, pp. 91–97, 2019.

[11] T. D. H. Hussein, M. Frikha, S. Ahmed, and J. Rahebi, "Ambulance vehicle routing using BAT algorithm," in *Proceedings of the International Conference on Electrical, Communication, and Computer Engineering*, pp. 1–5, ICECCE), Kuala Lumpur, Malaysia, August 2021.

[12] T. Michael and D. Xavier, "Intelligent ambulance management system with A∗ algorithm," in *Proceedings of the 3rd IEEE International Conference on Recent Trends in Electronics*, pp. 949–952, Information & Communication Technology (RTEICT), Bangalore, India, May 2018.

[13] J. Xie, Y. Zhou, and H. Chen, "A novel bat algorithm based on differential operator and Lévy flights trajectory," *Computational Intelligence and Neuroscience*, vol. 2013, Article ID 453812, 4 pages, 2013.

[14] X.-S. Yang, *Nature-inspired Optimization Algorithms*, Academic Press, Cambridge, MA, USA, 2014.

[15] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 65–74, Springer, Berlin, Germany, 2010.

[16] K. E. Heraguemi, N. Kamel, and H. Drias, "Multi-objective bat algorithm for mining numerical association rules," *International Journal of Bio-Inspired Computation*, vol. 11, no. 4, pp. 239–248, 2018.

[17] A. S. Abdullah, Y. E. Özok, and J. Rahebi, "A novel method for retinal optic disc detection using bat meta-heuristic algorithm," *Medical, & Biological Engineering & Computing*, vol. 56, no. 11, pp. 2015–2024, 2018.

[18] X.-S. Yang and X. He, "Bat algorithm: literature review and applications," *International Journal of Bio-Inspired Computation*, vol. 5, no. 3, pp. 141–149, 2013.

[19] P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai, and V. Istanda, "Bat algorithm inspired algorithm for solving numerical optimization problems," *Applied Mechanics and Materials*, vol. 148, pp. 134–137, 2012.

[20] A. H. Gandomi, X.-S. Yang, A. H. Alavi, and S. Talatahari, "Bat algorithm for constrained optimization tasks," *Neural Computing & Applications*, vol. 22, no. 6, pp. 1239–1255, 2013.

[21] S. Umar, L. U. Sadiku, and D. A. Tonga, "Intelligent-Based Control System for Effective Road Traffic Management in Nigeria: A Proposed Model," *International Journal of Latest Engineering Science (IJLES)*, vol. 2, no. 6, 2019.

[22] S. Priyadarshi, R. Mehrotra, and S. Shekhar, "Self control & monitoring traffic management system," in *Proceedings of the 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC)*, pp. 366–370, Greater Noida, India, October 2019.

[23] B. Ghazal, K. ElKhatib, K. Chahine, and M. Kherfan, "Smart traffic light control system," in *Proceedings of the 2016 third International Conference on Electrical, Electronics, Computer Engineering and Their Applications (EECEA)*, pp. 140–145, Beirut, Lebanon, May 2016.

[24] W.-H. Lee and C.-Y. Chiu, "Design and implementation of a smart traffic signal control system for smart city applications," *Sensors*, vol. 20, no. 2, p. 508, 2020.

[25] R. Shanmughasundaram, S. P. Vadanan, and V. Dharmarajan, "Li-fi based automatic traffic signal control for emergency vehicles," *Computer Communications*, in *Proceedings of the 2018 Second International Conference on Advances in Electronics*, pp. 1–5, Bangalore, India, October 2018.

[26] K. Kaur and A. Sharma, *Information and Communication Technology for Competitive Strategies*, vol. 190, Springer, Singapore, 2019.

[27] K. M. Almuraykhi and M. Akhlaq, "STLS: smart traffic lights system for emergency response vehicles," in *Proceedings of the 2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1–6, IEEE, Sakaka Saudi Arabia, May 2019.

[28] F. Sajid, A. R. Javed, A. Basharat, N. Kryvinska, A. Afzal, and M. Rizwan, "An efficient deep learning framework for distracted driver detection," *IEEE Access*, vol. 9, Article ID 169270, 2021.

[29] A. R. Javed, M. Usman, S. U. Rehman, M. U. Khan, and M. S. Haghighi, "Anomaly detection in automated vehicles using multistage attention-based convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4291–4300, 2020.

[30] S. Deshmukh and S. B. Vanjale, "IOT based traffic signal control for reducing time delay of an emergency vehicle using GPS," in *Proceedings of the 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pp. 1–3, IEEE, Pune, India, August 2018.

[31] S. Djahel, N. Smith, S. Wang, and J. Murphy, "Reducing emergency services response time in smart cities: an advanced adaptive and fuzzy approach," in *Proceedings of the 2015 IEEE First International Smart Cities Conference (ISC2)*, pp. 1–8, Guadalajara, Mexico, October 2015.

[32] V. Constantinescu and M. Patrascu, "Route encoding in evolutionary control systems for emergency vehicles," in *Proceedings of the 2017 15th International Conference on ITS Telecommunications (ITST)*, pp. 1–5, IEEE, Warsaw, Poland, May2017.

[33] M. Chen, "Improved genetic algorithm for emergency logistics distribution vehicle routing problems," in *Proceedings of the 2014 IEEE International Conference on Security*, pp. 385–388, Pattern Analysis, and Cybernetics (SPAC), Wuhan, China, December 2014.

[34] A. El Fallahi and I. Sefrioui, "A linear programming model and memetic algorithm for the Emergency Vehicle Routing," in *Proceedings of the 2019 4th World Conference on Complex Systems (WCCS)*, pp. 1–5, IEEE, Ouarzazate, Morocco, April 2019.

[35] E. Mouhcine, Y. Karouani, K. Mansouri, and Y. Mohamed, "Toward a distributed strategy for emergency ambulance routing problem," in *Proceedings of the 2018 4th International Conference on Optimization and Applications (ICOA)*, pp. 1–4, IEEE, Mohammedia Morocco, June 2018.

[36] N. Rathore, P. K. Jain, and M. Parida, "A routing model for emergency vehicles using the real time traffic data," in

Proceedings of the 2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), pp. 175–179, IEE, Singapore, August 2018.

[37] T. Tlili, M. Harzi, and S. Krichen, "Swarm-based approach for solving the ambulance routing problem," *Procedia Computer Science*, vol. 112, pp. 350–357, 2017.

[38] R. Tavakkoli-Moghaddam, P. Memari, and E. Talebi, "A bi-objective location-allocation problem of temporary emergency stations and ambulance routing in a disaster situation," in *Proceedings of the 2018 4th International Conference on Optimization and Applications (ICOA)*, pp. 1–4, IEE, Mohammedia, Morocco, June 2018.

[39] C. R. Kamireddy and B. N. Keshavamurthy, "Efficient routing of 108 ambulances using clustering techniques," in *Proceedings of the 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–6, IEE, Chennai, India, May 2016.

[40] A. Sharma and R. Kumar, "An optimal routing scheme for critical healthcare HTH services—an IOT perspective," in *Proceedings of the 2017 Fourth International Conference on Image Information Processing (ICIIP)*, pp. 1–5, IEEE, Shimla, India, March 2017.

[41] M. K. Madisa and M. K. Joseph, "Android and cloud based traffic control system," in *Proceedings of the 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, pp. 1–4, IEEE, Durban, South Africa, September 2018.

[42] N. C. Velayudhan, A. Anitha, and M. Madanan, "Sybil attack detection and secure data transmission in VANET using CMEHA-DNN and MD5-ECC," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–13, 2021.

[43] H. Bangui, M. Ge, and B. Buhnova, "A hybrid machine learning model for intrusion detection in VANET," *Computing*, vol. 104, no. 3, pp. 503–531, 2022.

[44] K. N. Tripathi and S. C. Sharma, "A trust based model (TBM) to detect rogue nodes in vehicular ad-hoc networks (VANETS)," *International Journal of Systems Assurance Engineering and Management*, vol. 11, no. 2, pp. 426–440, 2020.

[45] F. Mirsadeghi, M. K. Rafsanjani, and B. B. Gupta, "A trust infrastructure based authentication method for clustered vehicular ad hoc networks," *Peer-to-Peer Netw. Appl.*vol. 14, no. 4, pp. 2537–2553, 2021.

[46] A. Kumar, V. Varadarajan, A. Kumar et al., "Black hole attack detection in vehicular ad-hoc network using secure AODV routing algorithm," *Microprocessors and Microsystems*, vol. 80, Article ID 103352, 2021.

[47] K. Premkumar and B. V Manikandan, "Speed control of Brushless DC motor using bat algorithm optimized Adaptive Neuro-Fuzzy Inference System," *Applied Soft Computing*, vol. 32, pp. 403–419, 2015.

[48] K. Premkumar and B. V Manikandan, "Bat algorithm optimized fuzzy PD based speed controller for brushless direct current motor," *Eng. Sci. Technol. an Int. J.*vol. 19, no. 2, pp. 818–840, 2016.

[49] A. Wang, M. Wang, H. Wu, K. Jiang, and Y. Iwahori, "A novel LiDAR data classification algorithm combined capsnet with resnet," *Sensors*, vol. 20, no. 4, p. 1151, 2020.