



International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Energy Aware-Adaptive Walrus Optimization Algorithm (EA-AWaOA) Based Routing And Deep Q Learning Based Load Balancing In Vehicular Fog Computing

R.Sunitha¹, Dr.B.Sumathi²

¹Ph.D Scholar and Assistant Professor, Department of Artificial Intelligence & Machine Learning, CMS College of Science and Commerce, Coimbatore, Tamilnadu, India.

²Head & Associate Professor, Department of Artificial Intelligence & Machine Learning, CMS College of Science and Commerce, Coimbatore, Tamilnadu, India.

ABSTRACT

The emerging vehicular applications demand for a lot more computing and communication capacity to excel in their compute-intensive and latency-sensitive tasks. Fog computing, which focuses on moving computing resources to the edge of networks, complements cloud computing by solving the latency constraints and reducing ingress traffic to the cloud. However, efficient load management in vehicular fog computing is challenging due to the dynamic nature of the Vehicular Ad-Hoc Network (VANET). In this paper introduced the Energy Aware-Adaptive Walrus Optimization Algorithm (EA- AWaOA) for optimal selection of cluster heads and routes by the residual energy, inter-vehicle distance, closeness and vehicle velocity in fog-based VANET. Initially, it performs clustering by AWaOA while considering the mobility of fog devices. Secondly, a capacity-based load-distribution mechanism by Deep Q Learning (DQL) efficiently balances the load among the vehicular nodes. DQL is introduced for estimating the capacity load through the queue status obtained from each ground vehicular node, a load balancing scheme based on DQL and a reward control function for rapid convergence of Q-learning. This strategy minimizes the correlation among consecutive samples and improves learning stability. It proposes a mechanism for identifying a vehicle's departure time from the cluster, which allows predicting the future position of the vehicle within the dynamic network. The simulation results in the Network Simulator 2 (NS2) environment showed that as the balanced network energy consumption, reduced network delay, and improved network utilization in the vehicular fog computing.

Keywords: Network Simulator 2 (NS2), Vehicular Ad-Hoc Network (VANET), Fog Computing (FC), Deep Q Learning (DQL), Adaptive Walrus Optimization Algorithm (AWaOA), Internet of Things (IoT), and Load-Balancing (LB).

INTRODUCTION

With the advent of the Internet, there has been a myriad of hardware and software technologies added to the market with the most prevalent in the domain of communication being Internet of Things (IoT) enabled technology. The set of devices or the "Things" which share a common resemblance to directly or indirectly connect to the Internet, operate within the confines of their functionality and exchange, analyze, process and deliver data in the common language, these set of devices working in tandem are defined as IoT. IoT, which has witnessed a dramatic surge in the recent past, has had an immense impact on every aspect of human lives, ranging from wearable gears to sensors monitoring ecological changes in remote locations to regulating physical metrics in manufacturing processes [1]. A myriad of applications, ranging from intelligent transportation to smart building architecture, has led to the advent of IoT technology in smart city design realization. IoT is one of the spotlight innovations which have the potential to provide unlimited benefits to society. The development of the IoT is about to reach a stage at which many of the objects around us will have the ability to connect to the Internet to communicate with each other without human intervention. Originally, the IoT was intended to reduce human data entry efforts and use different types of sensors to collect data from the environment and permit automatic storage and processing of all these data [2].

IoT Real-Time (RT) applications such as smart buildings, smart transportation, smart healthcare, and military

surveillance need high Bandwidth (BW) and enough processing power to operate. As a result, managing and storing a lot of data is quite difficult, particularly when it rises locally and momentarily. As a result, it could demand for processing power and storage space from outside sources. In addition to delivering information quickly, the data gives end users "tools of wisdom" by offering wise choices, judgements, and analytical predictions that enhance their day-to-day functioning [3].

As the IoT is characterized by limited computations in terms of processing power and storage, it suffers from many issues such as performance, security, privacy and reliability. A significant amount of processing is needed to enable these operations. IoT devices can't manage this because they frequently have low computing power. For this reason, Cloud Computing (CC) services are used by IoT applications to carry out computation and storage tasks. The integration of the IoT with the cloud, known as the Cloud of Things (CoT), is the right way to overcome most of these issues. The CoT simplifies the flow of IoT data gathering and processing and provides quick, low-cost installation and integration for complex data processing and deployment. However, as there are a large number of IoT devices with heterogeneous platforms, the development of new IoT applications is a difficult task. This is because IoT applications generate huge amounts of data from sensors and other devices. These big data are subsequently analyzed to determine decisions regarding various actions. Sending all these data to the cloud requires excessively high network bandwidth [4].

Depending on the needs of the user, the processed data is then returned to the nearest IoT device next to the use. The centralized CC model, though having virtually unlimited resource pooling capabilities, ceases to welcome its proponents because of its failure in building common and multipurpose platform that can provide feasible solutions to the mission critical requirements of an IoT-aided SG. To overcome these issues, Fog Computing (FC) comes into play. FC also uses IoT resources, such as smart hubs, switches, routers, sensors, and cameras, which can meet the needs of RT-critical applications [5].

The term FC was coined by Cisco. It is a new technology that provides many benefits to different fields, especially the IoT. Similar to the cloud, fog computing provides services to IoT users such as data processing and storage. FC is based on providing data processing capabilities and storage locally to fog devices instead of sending them to the cloud. Both the cloud and fog provide storage, computing and networking resources. FC is a distributed computing framework installed on the intermediary network switches, devices sensors and machines to eliminate the connectivity, bandwidth, and latency issues prevalent in CC. CC and FC platforms complement each other to form a mutually beneficial and interdependent service continuum between the Cloud and the endpoints to make computing, storage, control, and communication possible anywhere along the continuum. A robust FC-based Smart Grid (SG) topology will allow dynamic augmentation of associated Fog Computing Nodes (FCN), thus rejuvenating the elasticity and scalability profiles of mission critical SG requirements. The objective of FC is to standardize and ensure secure communication between all devices, SG components, old and new, across all vendors, so that customization and service costs are kept to a minimum [6].

The purpose of FC in the IoT is to improve efficiency, performance and reduce the amount of data transferred to the cloud for processing, analysis and storage. Therefore, the data collected by sensors will be sent to network edge devices for processing and temporary storage, instead of sending them into the cloud, thus reducing network traffic and latency. In the fog computing, various edge nodes can cooperate to share computing, caching and communication resources so as to complete some computation tasks locally without interacting with the CC center via the fronthaul links. These Fog Nodes (FNs) include but are not limited to the base stations (BSs), Wi-Fi access points (APs), relays and routers. The computation tasks proposed in the user terminals can be offloaded to the FNs when necessary. FCNs exhibit diverse cost profiles, it is necessary to have an optimal user-to-FCN association such that the corresponding upload latency is minimized. Also, the number of successful connections is limited by the number of Bandwidth Units (BU) or Computational Resource Blocks (CRBs) available at each FCN [7].

Similarly, another crucial part of this middle layer that distinguishes between apps that must be performed on fog or cloud is the Fog Gateway (FGW). Furthermore, because of the advancements in the FC model, some new uses are being proposed and achieved, like face detection, RT processing, distributed edge computing (EC), and live streaming. The number of vehicles on the roads is increasing which is leading to traffic-related issues such as accident, wastage of time, and pollution. To address the traffic-related problems and to create an efficient collaboration among vehicles, Vehicular Ad-Hoc Networks (VANETs) are being introduced. This network is a form of Mobile Ad-hoc Network (MANET) that provides a communication interface for Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure interaction (V2I) [8].

One possible option for implementing an intelligent transport system in a city is VANET. Any vehicle with this modem may act as a repeater for other VANET members due to VANET technology. Vehicles are the nodes of a VANET. VANET is a type of mobile network. The purpose of VANETs is to enable communication between vehicles. These nodes must so implement communication radio interfaces, and a specific range spectrum needs to be assigned for VANET Data Transmission (DT). The low latency requirement for safety and control messages for V2V communication is currently met by Dedicated Short-Range Communications (DSRC)/Wireless Access in Vehicular Environments (WAVE), and Long-Term Evolution (LTE) is utilised for V2I communication. An On-Board Unit (OBU) is utilised in every vehicle for communicating with the other vehicles during V2V communication, and each vehicle functions as a mobile router. The hidden terminal

problem, broadcast storm problem, excessive latency for safety message transmission, message security, packet routing, congestion control, Quality of Service (QoS), and Resource Management (RM) are some of the other problems that affect VANET. A hierarchical structure can be utilized to address these problems. Clustering is the process by which two or more close vehicles that share certain characteristics come together to form a group in a hierarchical structure. Data Mining (DM) and Machine Learning (ML) both make extensive use of the clustering concept [9].

In VANET clustering, a crucial component in the cluster creation process is Cluster Head (CH). There are several approaches to establish a cluster, and it is based on input metrics. A CM vehicle is known as a Cluster Member (CM) vehicle. Some algorithms, known as Cluster Gateways (CG), employ two CM in addition to CH and CM to interact with other clusters on behalf of the CH. Every member of a cluster is referred to as CM until designated as CG. Among the participating vehicles, CH is chosen from the ones that are the most stable. As CM, the remaining vehicles enter the cluster. Consequently, there is no need to offer distinct CM selection criteria because CH selection (CHS) is a component of the cluster formation process. Several hierarchical Routing Protocols (RP) and Multi-Hop (MH) are suggested to address the VANET's restricted range communication issue. In order to process and store data during vehicle interaction, the user also requires a platform. Using a Vehicular Cloud (VC) model, the vehicles are linked to form a cloudlet [10].

Most research on various routing protocols in IoT and VANET recently focuses on mechanisms to create energy-efficient routes. Energy plays a significant role in performing significant operations in VANET networks. As the node energy decreases, the network connection decreases as well. Node failure due to power limitation causes system failure; hence, it terminates the internet connection. To create a more effective hybrid VANET routing protocol for Data Transmission (DT) and Load Balancing (LB) objectives. Since this problem is an NP-Hard problem, the meta-heuristic and nature-inspired algorithms are the best choices for solving it [11].

This paper introduced a new energy-aware method DQL for load balance managing in the fog-based VANET using an optimization algorithm named Adaptive Walrus Optimization Algorithm (AWaOA). Several bio-inspired algorithms show their efficacy in LB systems, including an Ant Colony Optimization (ACO) and Artificial Bee Colony (ABC). LB approaches are thought to be an effective way to mitigate the aforementioned issues in order to prevent the bottleneck problem and make efficient use of the available BW. However, it is very difficult to apply scheduling and LB approaches in a dynamic, mobile network such as VANET.

The following is a summary of this paper's contributions:

- Vehicular fog distributed computing architecture based on AWaOA is presented to handle IoT requests efficiently.
- A Data collection method is provided to determine when a vehicle leaves the cluster. In a dynamic context, this feature facilitates the creation of clusters.
- A capacity-based Load-Distribution (LD) method is developed to achieve inter-cluster global LB and local LB between clusters.
- Examine real automotive network scenarios and put them into practice in a simulated setting to demonstrate the work's effectiveness.

LITERATURE REVIEW

Haider et al., [12] proposed a Server-Based Network Congestion (SBNC) for load balancing in VANET. The proposed approach clusters Road Side Units (RSU) within specified ranges and incorporates dedicated load balancing and network scheduler RSU to manage route selection and request-response scheduling, thereby balancing RSU loads. Proposed work introduce three key algorithms: optimal placement of dedicated RSU, a scheduling policy for packets/data/requests/responses, and a congestion control algorithm for load balancing. Using the VanetMobiSim library of Network Simulator-2 (NS-2), evaluate proposed approach based on Residual Energy consumption (RE), End-To-End Delay (E2E), Packet Delivery Ratio (PDR), and Packet Overhead (PO). Results indicate substantial improvements in load balancing through proposed server-based approach.

Qun et al., [13] presented a suitable method based on EA for load balancing in fog-based VANETs using a hybrid optimization algorithms are Ant Colony Optimization and Artificial Bee Colony (ACO-ABC). The simulation results in the NS2 environment showed that as the number of nodes increased, the consumed amount of energy in the VANET is increased. Also, with the increasing number of tasks, load balancing by the proposed method has been improved. Finally, the simulation experiments showed that the proposed hybrid algorithm outperforms all other algorithms. To evaluate proposed approach based on Total Energy Consumption (EC), Residual Energy (RE), and Imbalance degree. Results indicate substantial improvements in load balancing through proposed server-based approach.

Roh et al., [14] adopted a Q-learning based Load Balancing Routing (Q-LBR) in VANET. Q-LBR through a combination of three key techniques, namely, a low-overhead technique for estimating the network load through the queue status obtained from each ground vehicular node by the Unmanned aerial vehicle Relay Node (URN), a load balancing scheme based on Q-learning and a reward control function for rapid convergence of Q-learning. Through diverse simulations, demonstrate that Q-LBR improves the PDR, Network Utilization (NU) and latency by more than lesser respectively, compared to the existing protocol.

Mehta and Mahato [15] formulated an Ant Colony Optimization (ACO) and Artificial Bee Colony (ABC) for

load balancing in VANET fog servers. The inter-RSU scheduling approach is developed for traditional requests and a load-balancing mechanism for Smart Requests to enhance the network performance of Vehicular Fog Network (VFN). The proposed approach uses a scheduling algorithm to schedule Traditional requests among RSU so that energy consumption at the RSUs is uniform and minimal. A Hybrid algorithm is used that integrates the benefits of ACO and ABC optimization-based techniques to achieve effective load balancing in VANET fog servers. Load balancing reduces the queue waiting time in fog servers. The simulation results display that proposed energy-efficient scheduling algorithm reduces energy consumption as compared to other scheduling algorithms and the Hybrid load-balancing algorithm reduces the queue waiting time as compared to other load-balancing techniques. The performance evaluation metrics are Packet Delivery Ratio (PDR), E2E, Throughput, and Packet Loss Ratio (PLR).

Sethi et al., [16] examined an online energy-efficient Inter-RSU Scheduling Algorithm (EE-IRSA) and a distributed Ant Colony Optimization-based Load balancing technique (d-ACOL) for optimizing the request fulfillment of traditional and smart application requests, respectively. The proposed EE-IRSA ensures minimum and uniform energy consumption across RSUs while d-ACOL ensures minimum queue waiting time of requests across fog servers. In addition, a second-price auction game-based relay vehicle selection technique is proposed which further minimizes the energy consumption of RSUs. Simulation results show that EE-IRSA with relay vehicle selection reduces the energy consumption by less, and d-ACOL reduces the queue waiting time by an average of lesser as compared to other load balancing techniques. The performance evaluation metrics are Energy Consumption (EC), E2E, Throughput, and Network Overhead.

Wu et al., [17] developed an Approximation Algorithm (AA) and Deep Reinforcement Learning (DRL) for load balance in a VANET. Proposed approach with the objective of minimizing the maximum load under transmits power, storage capacity, per task completion time and energy consumption constraints. The formulated problem is proved to be Nondeterministic Polynomial-hard (NP-hard), then it is investigated by decomposing it into two subproblems, i.e., how to offload tasks for the case of fixed transmit power and how to adjust transmit power for the given offloading decision. For the first subproblem, an approximation algorithm is proposed by offloading the tasks in the vehicle with the maximum load to the vehicle with minimum load. For the second subproblem, an adjustment strategy for transmit power is customized to further reduce the computing load. The algorithms are evaluated on an integrated simulation platform with open street map, Simulation of Urban Mobility (SUMO), NS-3 and dataset of Google cluster-usage traces. Simulation results show that, the proposed algorithms outperform three state-of-the-art works for most cases, in terms of the maximum load. Besides, the load can be further reduced by the proposed adjustment strategy.

Madasamy and Balasubramaniam [18] designed the Genetic Algorithm (GA) and Dynamic Programming (DP) is used in designing load-balanced clusters. The proposed Angular Zone Augmented Elitism-Based Im-migrants GA (AZEIGA) used elitism-based immigrants GA to deal with the population and DP to store the out-come of old environments. AZEIGA ensures clustering of load-balanced nodes, which prolongs the network lifetime. Experimental results show that AZEIGA works appreciably well in homogeneous resource class VANET. The simulation proves that AZEIGA gave better performance in PDR, Network Lifetime (NL), average delay, routing, and clustering overhead.

Giridhar et al., [19] suggested the Adaptive Neural Fuzzy Clustering and Quantum Glowworm Swarm Optimization-Based Routing (ANFC-QGSOR) for load balancing. ANFC-QGSOR technique initially allows the vehicles to communicate with one another. For effective CHS and cluster assembly, the ANFC technique is used with three input parameters: RE, distance, and node degree. In addition, by developing a fitness function, the QGSOR approach is used to select the best routes to the destination. NS is used to simulate the proposed ANFC-QGSOR method (NS3 tool). The experimental results demonstrated that the ANFC-QGSOR technique surpassed earlier state-of-the-art technologies in a variety of evaluation variables.

Elhoseny et al., [20] implemented an Intelligent Energy-Aware Oppositional Chaos Game Optimization-Based Clustering (IEAOCGO-C) for VANET. IEAOCGO-C technique aims to select CH in the network proficiently. The proposed IEAOCGO-C model constructs clusters based on Oppositional-Based Learning (OBL) with the CGO algorithm to improve efficiency. Besides, it computes a fitness function involving five parameters, namely Throughput (THRPT), PDR, Network Lifetime (NLT), End To End Delay (ETED) and Energy Consumption (ECM). The experimental validation of the proposed model is accomplished, and the outcomes are studied in numerous aspects with existing models under several vehicles and measures. The simulation outcomes reported the enhanced performance of the proposed approach over the recent technologies. As a result, it has resulted in higher NLT, THRPT, PDR, lower ETED, and ECM, as lesser average values over the other methods under all vehicle numbers.

Abbas et al., [21] improved an Intelligent Cluster Head (ICH) algorithm for load balance in VANET. ICH is introduced to increase cluster stability, and at the same time reduce the number of uplink connection at the Base Station (BS). In this study, the same mobile BS was used if there was no need for establishment of new BS for VANET. The proposed method produced stable cluster by electing CH based on the most important parameters that had the highest impact on the cluster stability. The simulation was done by using MATrix LABoratory (MATLAB) through a real data. The results showed that the proposed method provided less number of uplink connections than other previous studies and enhancing the stability of the system. The method resulted in less

number of CH by lesser from previous methods. The performance assessments are PDR, E2E, Throughput, and PLR.

PROPOSED METHODOLOGY

Adaptive Walrus Optimization Algorithm (AWaOA), an appropriate EA approach for LB in fog-based VANET is presented in this paper. It operates in two stages: It first clusters via AWaOA while taking Fog Device (FD) mobility into account. Then, the load is effectively distributed across the vehicle nodes using a capacity-based load-distribution mechanism by DQL efficiently balances the load among the vehicular nodes. DQL for estimating the capacity load through the queue status obtained from each ground vehicular node, a load balancing scheme based on DQL and a reward control function for rapid convergence of Q-learning. Fog serves as a mediator between cloud and IoT devices in the suggested method. To meet the needs of RT-critical applications at the network's edge, the Fog Layer (FL) has FD. When vehicles serve as fog devices, the suggested method takes VFC into account. However, compared to cloud data inputs, vehicle FD has limited communication, computation, and storage capabilities. Data collection technique considers the direction, speed, and position of vehicles to create clusters that serve as a computational resource pool. A method for determining a vehicle's departure time from the cluster is suggested. Future location of the vehicle in the dynamic network was enabled by this mechanism.

NETWORK STRUCTURE

Network architecture includes of the cloud layer, the intermediate vehicle FL, and the bottom layer IoT devices. The FGW node receives data signals from IoT devices in the network on a regular basis. There is a classification process in FGW. The task is effectively categorized as either time-critical or delay-tolerable using this classification process. The accessibility and ability of FD are examined prior to allocating the FL time-sensitive tasks. The FL completes the necessary work and provides the solution to IoT devices after verification. The data is sent to the cloud if it needs to be stored. When offloading tasks to the FL, vehicle mobility and capacity limitations are taken into account.

ROUTING MECHANISM

Vehicle fog devices are supposed to receive tasks from IoT devices that are linked to the upper layer. Many different kinds of sensors are installed in modern vehicles. Each vehicle's radio range encompasses the whole road's width. Every vehicle in the network transmits beaconing messages to allow the vehicles to connect. The time stamp, location coordinates, road condition, and vehicle identification are all included in this message. Using the beaconing message, vehicles establish a connection with neighbour nodes that fall within their communication range and calculate the approximate distance between them. The term "clusters" is used to describe these collections of linked vehicles. In order to represent actual mobile scenarios, such as the position, speed, and acceleration of moving devices, the Random Waypoint Model (RWP) [22]. CHS is linked along with nearby nodes to build clusters. Since the stable vehicle's mobility, the system model turns into vibrant generally. Hence, CM and CHs are changed in particular interval time. Since, this process, the beaconing message's overhead is amplified. Therefore, forecast the upcoming location of vehicles to reduce the beaconing message's overhead.

Adaptive Walrus Optimization Algorithm (AWaOA) based cluster head selection (CHS)

Adaptive Walrus Optimization Algorithm (AWaOA) is developed for optimal selection of CH, and routes in the routing [23]. Of these intelligent behaviors, three are the most obvious: (i) Guiding individuals to feed under the guidance of a member with the longest tusks. (ii) Migration of walruses to rocky beaches (iii) Fight or escape from predators. At the beginning of WaOA implementation, populations of walruses are randomly initialized. The first CH is determined at this stage by using the lower and upper bounds of the problem space for the energy, distance, proximity, and speed of vehicles. The dimension of the CHS is denoted by D. AWaOA population matrix is determined using equation (1),

$$X = \begin{bmatrix} X_{1,D} \\ \vdots \\ X_{i,D} \\ \vdots \\ X_{N,D} \end{bmatrix}_{N \times m} = \begin{bmatrix} X_{1,D,1} & \dots & X_{1,D,j} & \dots & X_{1,D,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{i,D} & \dots & X_{i,D,j} & \dots & X_{i,D,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{N,D,1} & \dots & X_{N,D,j} & \dots & X_{N,D,m} \end{bmatrix}_{N,D \times m} \quad (1)$$

where X is the walruses' population, $X_{i,D}$ is the i^{th} walrus (candidate solution), X_{i,D_j} is the value of the j^{th} decision variable suggested by the i^{th} walrus, N is the number of walruses, and m is the number of decision variables. As mentioned, each walrus is a candidate solution to the problem, and based on its suggested values for the decision variables, the objective function of the problem can be evaluated.

The estimated values for the objective function obtained from walrus are specified in equation (2),

$$F = \alpha \left(\frac{RE}{RE_{\max}} \right) + \beta \left(\frac{CC}{CC_{\max}} \right) + \gamma \left(\frac{1}{IVD} \right) + \delta \left(\frac{1}{VV} \right) \quad (2)$$

where F is the objective function vector, $\alpha, \beta, \gamma, \delta$ is the weighting coefficients $\alpha + \beta + \gamma + \delta = 1$. The objective is to maximize the fitness value while selecting the most stable and energy-efficient cluster heads. Objective function values are the best measure of the quality of candidate solutions. The candidate solution that results in the evaluation of the best value for the objective function is known as the best member. On the other hand, the candidate solution that results in the worst value for the objective function is called the worst member. According to the update of the values of the objective function in each iteration, the best and worst members are also updated for optimal CH selection.

Residual Energy (RE): Vehicles with higher remaining energy are preferred as cluster heads to prevent early node failure and prolong network lifetime.

Inter-Vehicle Distance (IVD): Neighboring vehicles with shorter communication distances are selected to reduce transmission energy consumption and packet loss.

Closeness Centrality (CC): Vehicles having stronger connectivity with neighboring nodes are prioritized to improve cluster stability and communication reliability.

Vehicle Velocity (VV): Vehicles with relatively stable speed are selected to minimize frequent cluster head changes caused by high mobility.

The process of updating the position of walrus in the AWAOA is modeled in three different phases based on the natural behaviors of this animal.

Phase 1: Feeding Strategy (exploration)

Walrus have a varied diet, feeding on more than sixty species of marine organisms, such as sea cucumbers, tunicates, soft corals, tube worms, shrimp, and various mollusks. However, walrus prefers benthic bivalve mollusks, particularly clams, for which it forages by grazing around the sea floor, seeking and detecting food with its energetic flipper motions and susceptible vibrissae. In this search process, the strongest walrus with the tallest tusks guides the other walrus in the group to find food. The length of the tusks in the walrus is similar to the quality of the objective function values of the candidate solutions.

Therefore, the best candidate solution with the best value for the objective function is considered the strongest walrus in the group. This search behavior of the walrus leads to different scanning areas of the search space, which improves the exploration power of the AWAOA in the global search. The process of updating the position of walrus is mathematically modeled based on the feeding mechanism under the guidance of the most vital member of the group, using equation (3) and (4). In this process, a new position for walrus is first generated according to equation (3).

This new position replaces the previous position if it improves the objective function's value; this concept is modeled in equation (4).

$$x_{ij}^{P_1} = x_{ij} + \text{rand}_{ij} \cdot (SW_j - I_{ij} \cdot x_{ij}), \quad (3)$$

$$X_i = \begin{cases} X_i^{P_1}, & F_i^{P_1} < F_i \\ X_i, & \text{else,} \end{cases} \quad (4)$$

Where $X_i^{P_1}$ is the new generated position for the i^{th} walrus based on the 1st phase, $X_{ij}^{P_2}$ is its j^{th} dimension, $F_i^{P_1}$ is its objective function value, rand_{ij} are random numbers from the interval $[0, 1]$, SW is the strongest walrus with the best value of the objective function, and I_{ij} are integers selected randomly between 1 or 2.

Phase 2: Migration

One of the natural behaviors of walrus for is their migration to outcrops or rocky beaches due to the warming of the air in late summer. This migration process is employed in the AWAOA to guide the walrus in the search space to discover suitable areas in the search space. This behavioral mechanism is mathematically modeled using equation (5) and equation (6). This modeling assumes that each walrus migrates to another walrus (randomly selected) position in another area of the search space. Therefore, the proposed new position is first generated based on equation (5). Then according to equation (6), if this new position improves the value of the objective function, it replaces the previous position of walrus.

$$x_{ij}^{P_2} = \begin{cases} x_{ij} + \text{rand}_{ij} \cdot (x_{kj} - I_{ij} \cdot x_{ij}), & F_k < F_i \\ x_{ij} + \text{rand}_{ij} \cdot (x_{ij} - x_{kj}), & \text{else,} \end{cases} \quad (5)$$

$$X_i = \begin{cases} X_i^{P_2}, F_i^{P_2} < F_i \\ X_i, \text{ else} \end{cases} \quad (6)$$

Where $x_i^{P_2}$ is the new generated position for the i^{th} walrus based on the 2nd phase, $x_{i,j}^{P_2}$ is its j^{th} dimension, $F_i^{P_2}$ is its objective function value, X_k , $k \in \{1, 2, \dots, N\}$ and $k \neq i$ is the location of selected walrus to migrate the i^{th} walrus towards it, $x_{k,j}$ is its j^{th} dimension, and F_k is its objective function value.

Phase 3: Escaping and Fighting Against Predators (Exploitation)

Walruses are always exposed to attacks by the polar bear and the killer whale. The strategy of escaping and fighting these predators leads to a change in the position of the walruses in the vicinity of the position in which they are located. Simulating this natural behavior of walruses improves the AWaOA exploitation power in the local search in problem-solving space around candidate solutions [24]. For simulation of this phenomenon in AWaOA, a neighborhood is assumed around each walrus, which first is generated a new position randomly in this neighborhood using equation (7). Then if the value of the objective function is improved, this new position replaces the previous position according to equation (8).

$$x_{i,j}^{P_3} = x_{i,j} + (lb_{local,j}^t + (ub_{local,j}^t - rand \cdot lb_{local,j}^t)) \quad (7)$$

$$X_i = \begin{cases} X_i^{P_3}, F_i^{P_3} < F_i \\ X_i, \text{ else} \end{cases} \quad (8)$$

where $X_i^{P_3}$ is the new generated position for the i^{th} walrus based on the 3rd phase, $x_{i,j}^{P_3}$ is its j^{th} dimension, $F_i^{P_3}$ is its objective function value, t is the iteration contour, lb_j and ub_j are the lower and upper bounds of the j^{th} variable, respectively, $lb_{local,j}^t$ and $ub_{local,j}^t$ are local lower and local upper bounds allowable for the j^{th} variable, respectively, to simulate local search in the neighborhood of the candidate solutions.

To further improves the search capability of WaOA by introducing an adaptive exploration–exploitation mechanism. During the early iterations, the algorithm emphasizes global exploration to avoid premature convergence and discover promising candidate solutions. As the optimization progresses, the search gradually shifts toward local exploitation around the best-performing solutions to accelerate convergence toward the global optimum.

Instead of using a fixed control parameter, introduce an adaptive coefficient to equation (9),

$$A(t) = 2 \left(1 - \left(\frac{t}{T} \right)^\lambda \right) \quad (9)$$

where t is the current iteration, T is the maximum iterations, λ is adaptive convergence factor (typically 1.5–3). The parameter behaves as beginning of $A(t) \approx 2 \rightarrow$ large search steps (global exploration), Middle of balanced exploration/exploitation, at the end of $A(t) \approx 0 \rightarrow$ fine local search. After updating the walruses' position based on the implementation of the first, second, and third phases, the first AWaOA iteration is completed, and new values are calculated for the position of the walruses and the objective functions. Update and improve candidate solutions is repeated based on the AWaOA steps according to Equations (3) to (9) until the final iteration. Upon completion of the algorithm execution, AWaOA introduces the best candidate solution found during execution as the solution to the given problem. The AWaOA implementation flowchart is presented in Figure 1, and its pseudocode is specified in Algorithm 1.

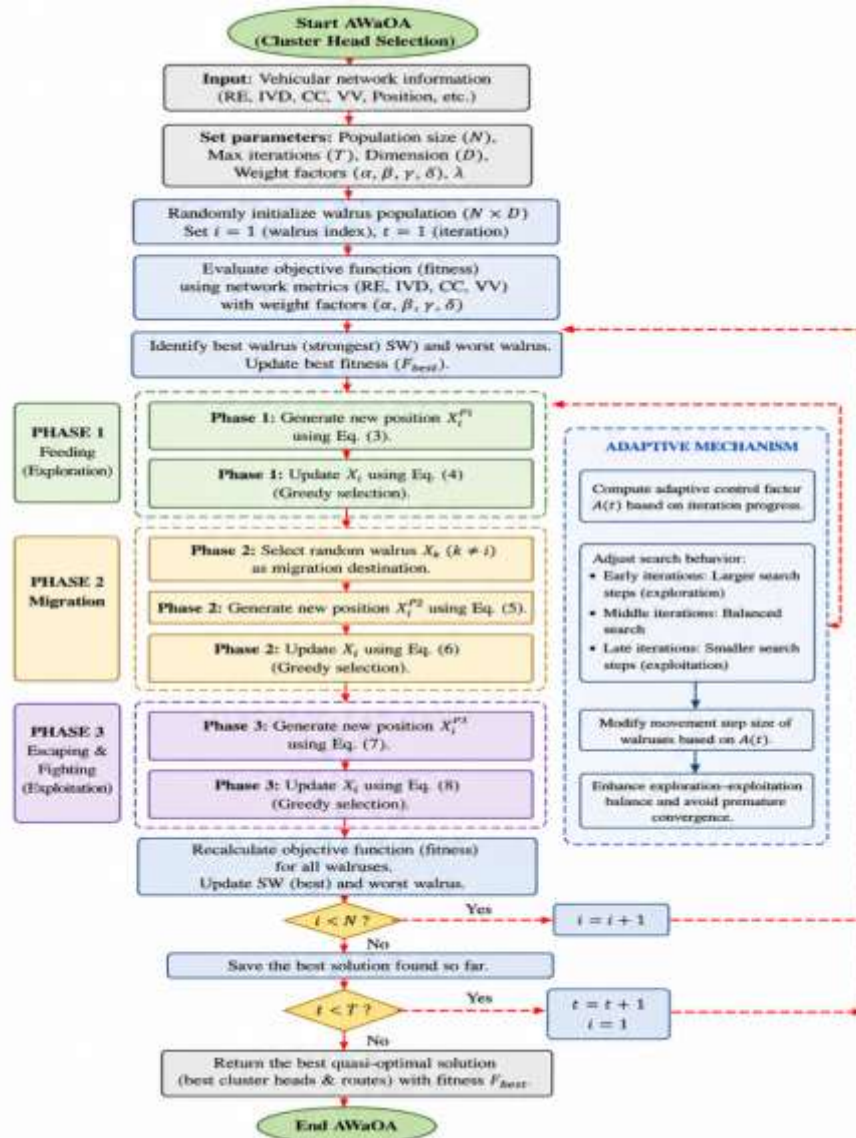


FIGURE 1. FLOWCHART OF AWaOA

Algorithm 1. Pseudocode of AWaOA

Start

Inputs: Number of walruses (N) and the total number of iterations (T)

Outputs: The optimal CH solutions

Input all optimization problem information.

Set the number of CH(N) and b having dimension D, the total number of iterations (T).

Initialization process of walruses' locations.

For t = 1: T

Update strongest walrus based on objective function value criterion.

For i= 1: N

Phase1: Feeding strategy (exploration)

Calculate new CH location of the jth walrus using equation (3).

Update the ith walrus location using equation (4).

Phase 2: Migration

Choose an immigration destination for the ith walrus.

Calculate new CH location of the jth walrus using equation (5).

Update the ith walrus location using equation (6).

Phase 3: Escaping and fighting against predators (exploitation)

Calculate a novel CH solution in the neighborhood of the ith walrus using equation (7).

Update the ith walrus location based on CH location using equation (8).

End for i

Introduce an adaptive coefficient to equation (9),

Save the best candidate CH solution so far.
end for t
Output the best quasi-optimal solution obtained by AWaOA for given problem.
End.

COOPERATIVE OFFLOADING

IoT requests the necessary services from the fog layer following the route setup stage. As a result, IoT devices within its communication range send requests to the FGW. The CH fog devices receive the tasks from FGW. The task distribution between the clusters and inside the vehicle cluster is the responsibility of this CH fog device. FGW uses beaconing and prediction algorithms to choose the appropriate node as a CH. FGW gathers data on neighboring mobile nodes in this way. The gateway node chooses CH based on the mobile node's velocity being less than or equal to a computed threshold and proximity (close to the FGW). The task's deadline is taken into account by the threshold value. A list of nearby nodes that are CM is obtained by the elected CH. The node count and the background of the network may determine the creation of many clusters.

However, the Data Packets (DP) is regarded as dropped if CH leaves the FGW's range before accepting the results. Task replication guarantees Task Completion (TC) with a slight increase in TC time, recovers this condition. FGW transfers the responsibility to CH following cluster formation. Other CHs that are outside of FGW's communication range are connected to the elected CH. Based on node capacity, the CH divides tasks between clusters and within a cluster. IoT devices transmit sensed data to the FGW during the DT phase. The suggested clustering process is used by the FGW to elect CH and identify essential data. A cluster's remaining computing capacity can be sent to the FGW node by the CH. CH is then given IoT tasks by the gateway node (GWN) according to its computational capacity.

CAPACITY-BASED LOAD-DISTRIBUTION USING DEEP Q LEARNING (DQL)

Achieving the maximum Packet Delivery Ratio (PDR) is a result of capacity-based LD within the flow nodes. Based on the vehicles' vicinity, speed, and path, the DC of mobile FN chooses CH. Additionally, to improve network utilisation, the capacity-based load-distribution mechanism effectively distributes the load both inside and between clusters using Deep Q Learning (DQL) algorithm. The following constitutes a formalised Objective Function (OF) based on the above listed factors,

$$\text{Max} \sum_{i_t=1}^{i_{t_{\max}}} P_{td}(i_t) \forall i_t \in T \quad (10)$$

$$P_{td}(it) = \sum_{P_{td}=1}^{P_{td}(\max)} C \times N_p \quad \forall N_p \in P_{td} \quad (11)$$

$$C_1: E_i \leq E_{\text{initial}} \forall i \in T \quad (12)$$

$$C_2: F_{pr}(i) \geq T_{s_{pr}}(i) \forall i \in T \quad (13)$$

$$C_3: RSE_F \geq E_{tr} \quad (14)$$

$$C_4: RSE_F \geq E_{rc} \quad (15)$$

$$C_5: S_{i, \text{queue}} \leq T_{\text{queue}} \forall i \in T \quad (16)$$

$$C_6: \text{Min} \sum_{i=1}^{i_{\max}} E \forall i \in T \quad (17)$$

Equation (10) can be used to maximise the network's PDR value. Additionally, the type of DP is explained by equations (11–12). Here, if C is true, N_p is the counter of DP that is increased. The energy of a FN at any given moment is $<$ or $=$ its starting energy, as stated in equation (13). Furthermore, restrictions in equations (14–17) need to be verified for correctly transfer information to the end point. Prior to giving IoT tasks to a FN for full calculation, the suggested work confirms the limitations C2 and C5. Furthermore, the C6 constraint demonstrates a clear correlation between the higher network resource utilisation and the minimum EC. The following equation (18) can be used to get the transmit energy E_{tr} ,

$$E_{tr} = \frac{T \times p_{\text{size}} \times T_p \times S}{D_{\text{rate}}} \quad (18)$$

Here the packet size is denoted by p_{size} . Transmission power is indicated by T_p . Vehicle speed is indicated by S . The network's data rate is denoted by D_{rate} . Additionally, the following equation can be used to calculate reception energy E_{rc} :

$$E_{rc} = \frac{T \times p_{size} \times R_p \times S}{D_{rate}} \quad (19)$$

The LD in the network gets more stable as the simulation time grows, which is significant. The network's PDR rises over time as a result of effective network utilization. Q-learning is a model-free learning method and one of the classical Reinforcement Learning (RL) algorithms. The action a_t should be chosen based on the current situation s_t and policy. Then the corresponding reward r_t and the next state s_{t+1} can be obtained. Next, the action value $Q(t) = Q(s_t, a_t)$ should be updated using the Bellman equation (20),

$$Q(t) \leftarrow Q(t) + \beta (r_t + \gamma_{\max, t+1} Q(t+1) - Q(t)) \quad (20)$$

Due to the large complexity of the environment considered such a Q-table will be of unacceptable size. So, a Q-learning algorithm based on Neural Networks (NN) is used instead. In DQL, the learning process uses two neural networks: the trained network and the target network. These networks have the same architecture but different weights. The parameters of the target DQL are updated according to the evaluation of trained DQL at a certain rate where the parameters of the trained DQL are backpropagation using Adam optimizer algorithm. It is important to emphasize that the parameters of the target network are not trained but synchronized periodically with the parameters of the main Q-network [25]. Using the Q-values of the target network to train the main Q-network increases the stability of the training. For a state value given as input to the NN, Q-values corresponding to more than one action will be obtained as output.

From these outputs, the action maximizing the Q-value is selected and the target action specified as equations (21-22),

$$\gamma_i = r_i + \gamma_{\max, t+1} Q(t+1), \quad (21)$$

$$L = \frac{1}{N} \sum_{i=0}^{N-1} (Q_i - \gamma_i)^2 \quad (22)$$

All other possible outcomes are then used to approximate this target action by checking the difference between the Q-values and the target values. By calculating the error between the Q-values and the target values the loss function calculation is also performed.

The three main components of the proposed DQL are defined as follows:

State: The state vector $s \in S$ contains information about the RSU will process at the time when the requests generated for the $x \in X$ service are collected.

Action: The action vector consists of the elements $|X|S|$. Any element in this vector represents an action that can be taken by the vehicle. The vehicle can either process the request if it has the processing and application capabilities to do so or it can initiate its routing to the nearest available Neighbor vehicle that can respond to this request.

Rewards: The rewards and the Q-values are determined by the proposed DQL algorithm 2. Since the reward depends on the connection speed and vehicle data capacity, the most suitable servers and backhaul connections at that time are evaluated to ensure optimal data transfer. The vehicle path selection for the from server to server depends on the type of vehicle speed and velocity that the server can process, the current load of connections between servers and the data processing speed. The reward equation (23) the parameter ξ is a weighting coefficient.

$$R_Q = \begin{cases} v_e c_m \xi \text{ if } m_i \text{ feasible for } x_i \\ -100 & \text{if } m_i \text{ not feasible for } x_i \end{cases} \quad (23)$$

Algorithm 2. Proposed DQL Algorithm

Initialize: Pre-process (vehicle quality, speed, pack size, vehicle speed and density, application type)

Initialize: Routing environment, Main Q-network Q, Target Q-network Q', Experience Replay Memory D.

Initialize learning rate α , discount factor γ , exploration rate ϵ , minibatch size N, target update interval C, and maximum episodes.

While packets remain to be processed do:

Observe current vehicle state s.

If Vehicle_i can run application type and message are not processed:

Realize Action (Process packet/application)

Update reward R_Q (vehicle processing capacity, queue length, vehicle speed, transmission delay).

Update routing path.

Else if

Offload remaining packets to the next available vehicle.
 End If
 Observe next state s' .
 Return R_Q
 Initialize $r = R_Q$
 If convergence criteria is not met
 Select action a using the ϵ -greedy policy
 Agent a realizes the action, observes the reward r and new state s' .
 Stores the transition sequence $(s, a, r, s', done)$ into experience replay memory D .
 If D has enough experience:
 If replay memory contains at least N samples
 Randomly sample a minibatch of size N .
 For each $(s_i, a_i, r_i, s'_i, done_i)$ sequence in Minibatch:
 If $done_i$ is True:

$$y_i = r_i$$

 Else if

$$y_i = r_i + \gamma \max_{a' \in A} Q'(s'_i, a')$$

 End If
 Compute Loss function $L = \frac{1}{N} \sum_{i=1}^N (Q(s_i, a_i) - y_i)^2$
 Update Main Q-network using the Adam optimizer.
 For each C step
 Duplicate the weights of Q network into Q' network.
 Update exploration rate $\epsilon = \epsilon \times \epsilon_{decay}$.
 Until $\epsilon \geq \epsilon_{min}$
 End for C
 Output
 Return Optimal routing path, Load-balanced vehicle selection, Trained Q-network.

RESULTS AND DISCUSSION

NS2 simulator was used to simulate routing methods. Two network simulation scenarios were examined with respect to the network's load: one with a balanced load and one without. In addition, three traffic scenarios: parked vehicles, vehicles in an urban setting, and vehicles in general, have been considered for simulation. Additionally, the network's simulation time (T) and transmission range (S) are varied during these tests. If the DP is received after the deadline, there is no chance that the packet will reach its destination; if it does, the probability is 1. Residual Energy (RE), End-To-End Delay (E2ED), Packet Loss Ratio (PLR), Packet Delivery Ratio (PDR) in the network has been used to assess the performance of the routing methods. Assume a 1000*1000 m² network with 10 IoT devices and 31 fog nodes. The system and simulation parameters utilised in the experiments are summarised in Table 1. Five runs of the simulation are averaged to determine the results.

Table 1. Simulation Parameter And System Parameters

Simulation Parameter	Value (Units)
Channel	Wireless Channel
Propagation Model	TwoRayGround
MAC Layer	MAC/802-11
Queue Type	Queue/ Drop Tail/ PriQueue
Mobility Model	Random Waypoint Model
Queue Length	20
VFN count	31
IoT node	10
FGW	1
Interval of VFN in clusters	4-6
Interval of vehicular fog speed (Km/h)	50-90
RP	AODV
Transport Layer Protocol	TCP
Simulation Time (Seconds)	500
Transmission Range(m)	50-300
Initial Energy of mobile FN(j)	50
Receiving Power(W)	1.2

Information Rate (Kbits/seconds)	250
Transmission Power (W)	1.4
Time taken for processing(Seconds)	Object Dimension

RE: The network's residual VFN energy is referred to as RE. Joules (J) are used to measure the energy dissipation.

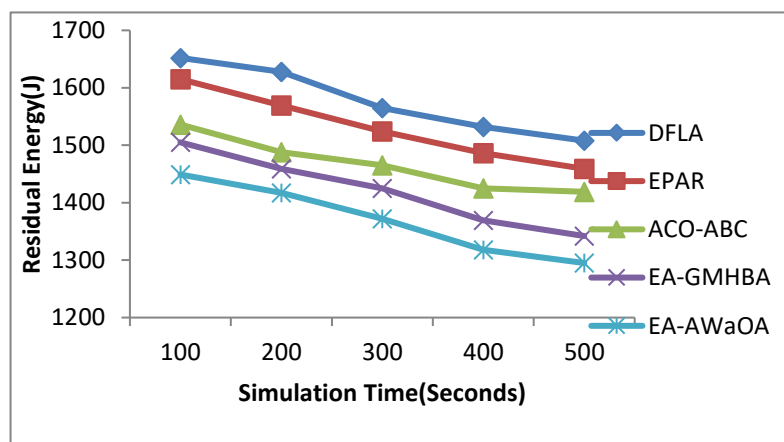
E2ED: The time it takes for an IoT device to transmit data to the vehicular fog network is known as E2ED. Then get the resulting information. It involves a number of delays, including processing, queuing, and communication. Milliseconds (ms) are the unit of measurement.

Packet Delivery Ratio (PDR): In a vehicular fog network, the percentage of successfully delivered IoT tasks to FD, relative to all tasks produced by IoT devices, is represented by PDR.

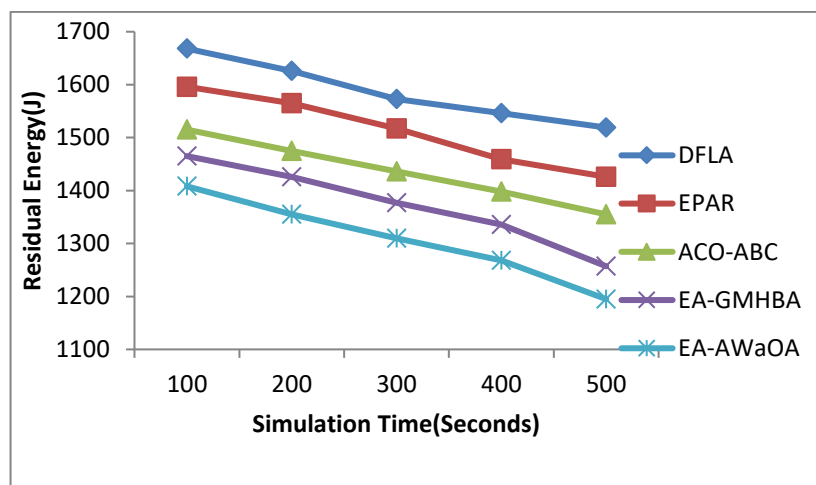
Packet Loss Ratio (PLR): It indicates the proportion of tasks from IoT devices in the vehicular fog network that are ineffectively sent to FD.

Parked vehicles: In certain cases, the initial RE of both LB and unbalanced systems is the same when it comes to parked vehicles. This is because the FN is originally used in close proximity to IoT devices. After a while, the FN nearer the FGW becomes heavily loaded. Thus, EC was raised.

Vehicles in urban area scenario: Vehicles travel between 50 and 60 km/h on average in urban areas. Additionally, these vehicles become caught in traffic and stop at traffic signals. The goal of the suggested method is to use these cars as fog devices while taking mobility into account. Because closer FN is used, the EC of the two methods is initially equal. However, as time passes in Figure 1a, the EC in the suggested method decreases because of the capacity-based LB's effective load distribution. As a result, the network resources are used effectively, optimizing the vehicular fog network's EC. When contrasting the imbalanced load strategy with the LB technique, the RE of the fog node is higher. The NL is raised as a result. Notably, one of the most important aspects of Electric Vehicles (EVs) is battery life.



(a) Re Comparison For Routing Protocols (Parked Vehicles)



(b) Re Comparison For Routing Protocols (Vehicles In Urban Area)

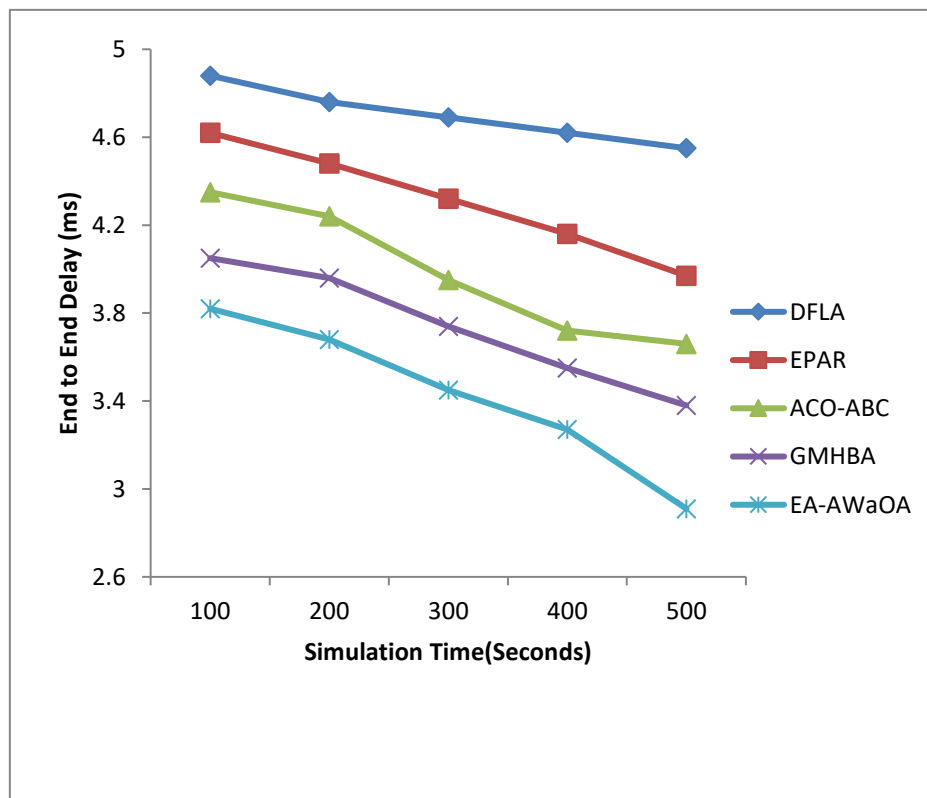
Figure 2. Residual Energy (Re) Vs. Routing Protocols

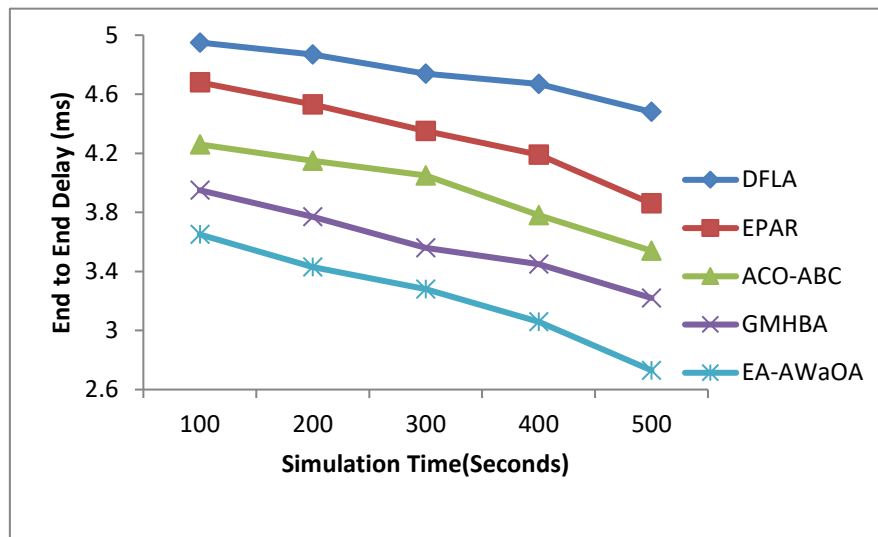
Figure 2(a) and Figure 2(b), together with Table 2 present the comparison of residual energy (RE) for different routing protocols in the parked vehicle and urban scenarios, respectively. The proposed EA- AAWAOA scheme

effectively balances the routing load among fog nodes (FNs), resulting in efficient utilization of network resources and improved energy conservation throughout the simulation period. Figure 2(a), EA-AWaOA consistently achieves the lowest energy consumption, consuming the residual energy values of 1449 J, 1417 J, 1372 J, 1318 J, and 1295 J at simulation times of 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. In comparison, the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA retain higher residual energy values of 1508 J, 1459 J, 1419 J, and 1342 J, respectively, at 500 s, indicating comparatively higher energy consumption. Figure 2(b) in the urban area scenario, the proposed EA-AWaOA demonstrates superior energy efficiency by achieving residual energy values of 1408 J, 1355 J, 1310 J, 1268 J, and 1195 J at 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. At 500 s, the existing approaches DFLA, EPAR, ACO-ABC, and EA-GMHBA record higher residual energy values of 1519 J, 1426 J, 1355 J, and 1257 J, respectively. These results confirm that the proposed routing protocol effectively minimizes energy consumption through adaptive load balancing and energy-aware cluster management, thereby extending the operational lifetime and enhancing the overall energy efficiency of the vehicular fog computing network.

Table 2. Residual Energy (Re) Comparison For Parked Vehicle And Urban Scenarios

Simulation time (seconds)	Residual Energy (RE) (Joules)- Parked Vehicles				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaOA
100	1652	1615	1536	1505	1449
200	1628	1569	1488	1459	1417
300	1565	1524	1465	1425	1372
400	1532	1486	1425	1369	1318
500	1508	1459	1419	1342	1295
Simulation time (seconds)	Residual Energy (RE) (Joules)- Vehicles in urban area				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaOA
100	1668	1596	1515	1465	1408
200	1626	1565	1475	1426	1355
300	1573	1517	1436	1377	1310
400	1546	1459	1398	1336	1268
500	1519	1426	1355	1257	1195


(a) E2ED Comparison For Routing Protocols (Parked Vehicles)



(b) E2ED Comparison For Routing Protocols (Vehicles In Urban Area)

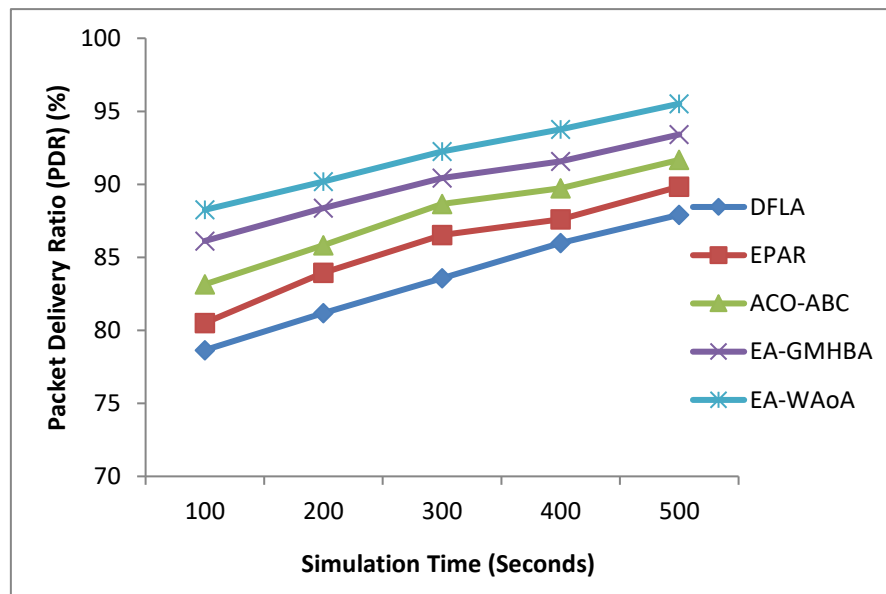
Figure 3. End To End Delay (E2ED) Vs. Routing Protocols

Figure 3(a) and Figure 3(b), together with Table 3, compare the end-to-end delay (E2ED) of the proposed EA-AWAOA routing protocol with the existing methods under parked vehicle and urban scenarios. The proposed approach achieves the lowest E2ED by employing a capacity-based load balancing mechanism that distributes computational tasks among fog nodes (FNs) according to their processing capabilities. This efficient task allocation minimizes network congestion, reduces packet queuing time, and accelerates packet delivery. In the parked vehicle scenario, the E2ED decreases progressively with simulation time for all routing protocols.

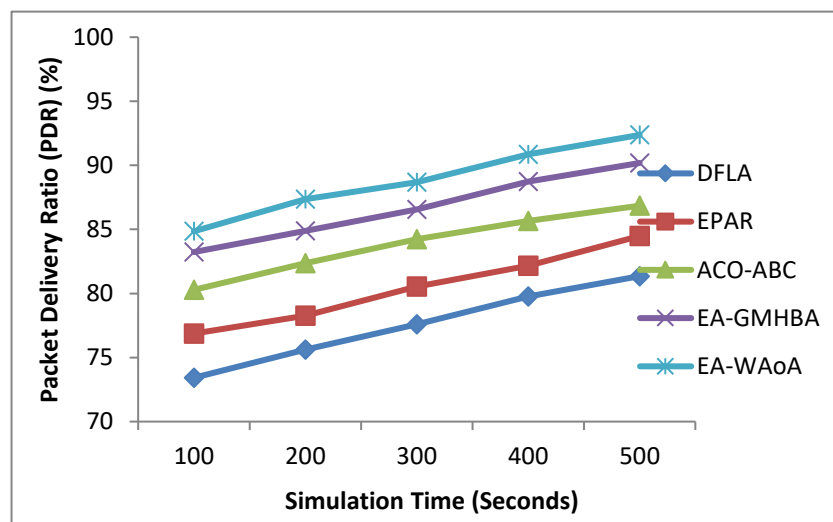
Figure 3(a), EA-AWAOA consistently outperforms the existing methods by achieving E2ED values of 3.82 ms, 3.68 ms, 3.45 ms, 3.27 ms, and 2.91 ms at simulation times of 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. At 500 s, the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA record higher E2ED values of 4.55 ms, 3.97 ms, 3.66 ms, and 3.38 ms, respectively. Similarly, in the urban scenario, the proposed EA-AWAOA demonstrates superior performance with E2ED values of 3.65 ms, 3.43 ms, 3.28 ms, 3.06 ms, and 3.2.73 routing protocols DFLA, EPAR, ACO-ABC, and EA-GMHBA achieve higher E2ED values of 4.48 ms, 3.86 ms, 3.54 ms, and 3.22 ms, respectively, at 500 s. These results demonstrate that the proposed EA-AWAOA effectively reduces communication latency by optimizing routing decisions and balancing the processing load across fog nodes, thereby enhancing the quality of service (QoS) and overall network performance in vehicular fog computing environments.

Table 3. End To End Delay (E2ED) Comparison For Parked Vehicle And Urban Scenarios

Simulation time (seconds)	End To End Delay (E2ED) (ms)- Parked Vehicles				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWAOA
100	4.88	4.62	4.35	4.05	3.82
200	4.76	4.48	4.24	3.96	3.68
300	4.69	4.32	3.95	3.74	3.45
400	4.62	4.16	3.72	3.55	3.27
500	4.55	3.97	3.66	3.38	2.91
Simulation time (seconds)	End To End Delay (E2ED) (ms)- Vehicles in urban area				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWAOA
100	4.95	4.68	4.26	3.95	3.65
200	4.87	4.53	4.15	3.77	3.43
300	4.74	4.35	4.05	3.56	3.28
400	4.67	4.19	3.78	3.45	3.06
500	4.48	3.86	3.54	3.22	2.73



(a) Packet Delivery Ratio (PDR) For Routing Protocols (Parked Vehicles)

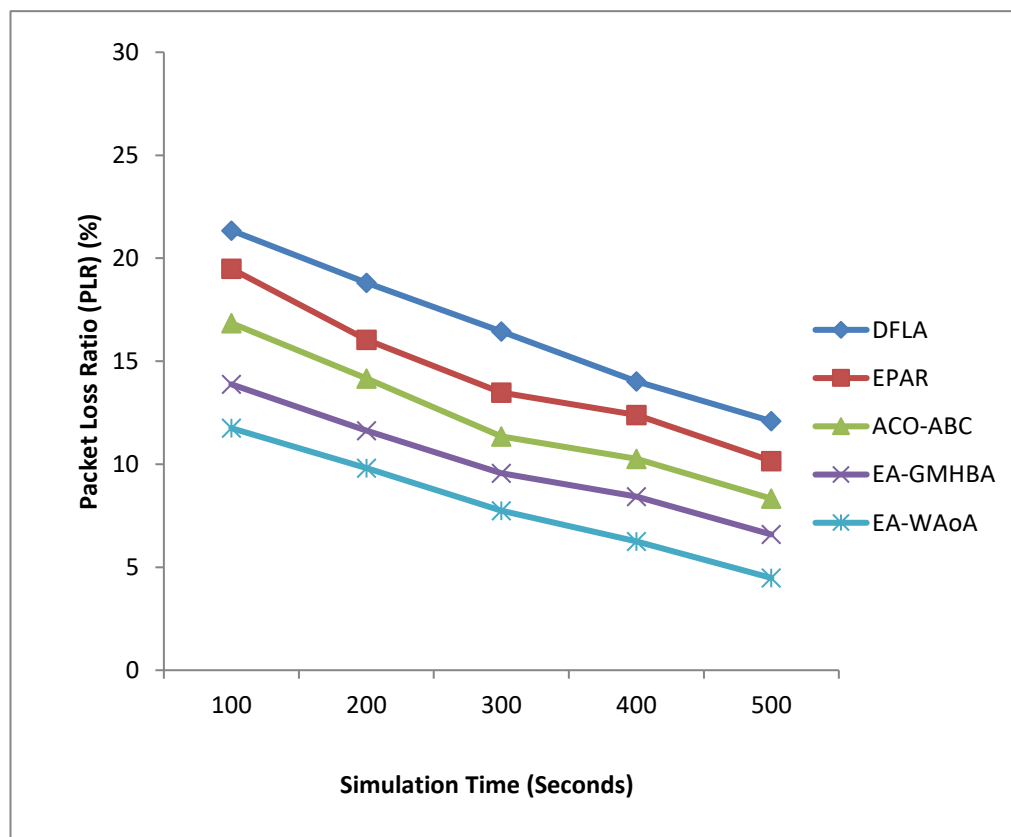


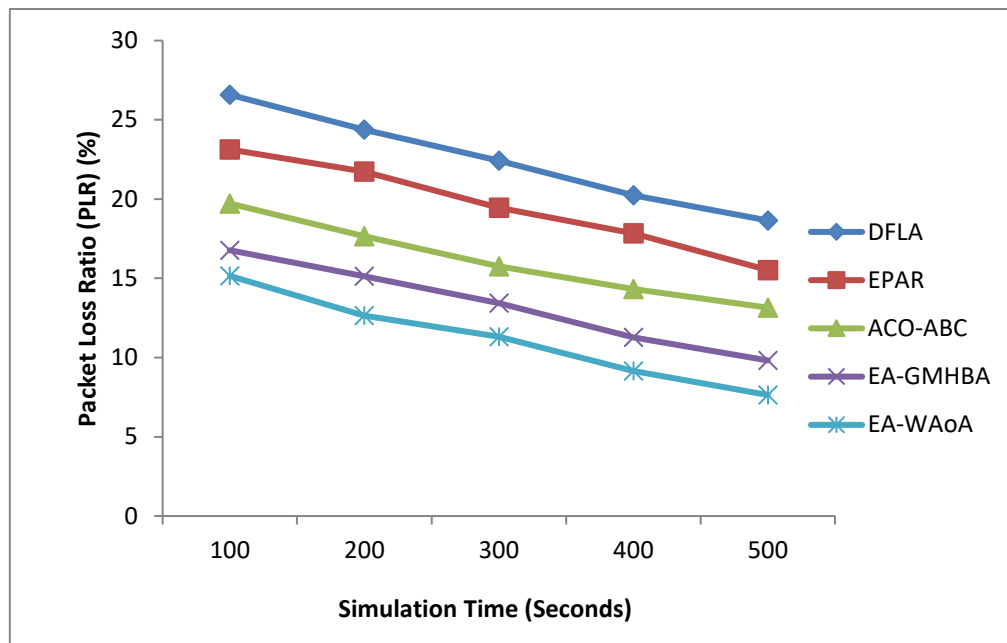
(b) Packet Delivery Ratio (PDR) For Routing Protocols (Vehicles In Urban Area)
Figure 4. Packet Delivery Ratio (PDR) Vs. Routing Protocols

Figure 4(a) and Figure 4(b), together with Table 4, illustrate the packet delivery ratio (PDR) achieved by the proposed EA-AWaoA routing protocol and the existing methods in parked vehicle and urban scenarios, respectively. The proposed routing protocol consistently achieves a higher PDR owing to its energy-aware routing strategy and capacity-based load balancing mechanism, which effectively distribute network traffic among fog nodes (FNs). This balanced traffic allocation minimizes packet losses caused by congestion and improves the reliability of data transmission. In the parked vehicle scenario, the PDR increases steadily with simulation time for all routing protocols. Figure 4(a), EA-AWaoA achieves PDR values of 88.25%, 90.18%, 92.25%, 93.76%, and 95.52% at simulation times of 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. At 500 s, the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA attain lower PDR values of 87.91%, 89.84%, 91.67%, and 93.41%, respectively. The superior PDR of the proposed method indicates that a larger proportion of transmitted packets successfully reach their intended destinations, demonstrating more reliable routing performance. Similarly, in the urban scenario, the proposed EA-AWaoA maintains the highest PDR throughout the simulation, achieving 84.85%, 87.36%, 88.69%, 90.85%, and 92.37% at 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. Figure 4(b), the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA achieve lower PDR values of 81.35%, 84.48%, 86.85%, and 90.18%, respectively, at 500 s. These results demonstrate that the proposed routing protocol effectively balances the communication load among clusters, both within and beyond the Fog Gateway (FGW) communication range, thereby reducing packet losses and enhancing data delivery reliability in highly dynamic vehicular fog computing environments.

Table 4. Packet Delivery Ratio (PDR) Comparison For Parked Vehicle And Urban Scenarios

Simulation time (seconds)	Packet Delivery Ratio (PDR) (%) - Parked Vehicles				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaoA
100	78.65	80.51	83.15	86.12	88.25
200	81.19	83.95	85.83	88.37	90.18
300	83.56	86.52	88.65	90.44	92.25
400	85.98	87.61	89.73	91.57	93.76
500	87.91	89.84	91.67	93.41	95.52
Simulation time (seconds)	Packet Delivery Ratio (PDR) (%) - Vehicles in urban area				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaoA
100	73.42	76.87	80.28	83.23	84.85
200	75.63	78.26	82.35	84.88	87.36
300	77.58	80.54	84.24	86.57	88.69
400	79.76	82.17	85.67	88.72	90.85
500	81.35	84.48	86.85	90.18	92.37


Packet Loss Ratio (PLR) For Routing Protocols (Parked Vehicles)



(b) Packet Loss Ratio (PLR) For Routing Protocols (Vehicles In Urban Area)

Figure 5. Packet Loss Ratio (PLR) Vs. Routing Protocols

Figure 5(a) and Figure 5(b), together with Table 5, present the comparison of the packet loss ratio (PLR) achieved by the proposed EA-AWaoA routing protocol and the existing methods in parked vehicle and urban scenarios, respectively. The proposed routing protocol consistently exhibits the lowest PLR due to its energy-aware routing strategy and capacity-based load balancing mechanism, which effectively distribute network traffic among fog nodes (FNs). This balanced traffic allocation alleviates network congestion, minimizes packet collisions, and reduces packet retransmissions, thereby improving communication reliability. In the parked vehicle scenario, the PLR decreases progressively with simulation time for all routing protocols. Figure 5(a), EA-AWaoA achieves the lowest PLR values of 11.75%, 9.82%, 7.75%, 6.24%, and 4.48% at simulation times of 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. At 500 s, the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA record higher PLR values of 12.09%, 10.16%, 8.33%, and 6.59%, respectively. These results indicate that the proposed routing protocol significantly reduces packet losses by selecting stable communication paths and efficiently balancing the routing load among fog nodes. Similarly, in the urban scenario, the proposed EA-AWaoA consistently outperforms the existing approaches, achieving PLR values of 15.15%, 12.64%, 11.31%, 9.15%, and 7.63% at 100 s, 200 s, 300 s, 400 s, and 500 s, respectively. Figure 5(b), the existing methods DFLA, EPAR, ACO-ABC, and EA-GMHBA achieve higher PLR values of 18.65%, 15.52%, 13.15%, and 9.82%, respectively, at 500 s. The lower packet loss ratio obtained by the proposed EA-AWaoA demonstrates its ability to maintain reliable packet forwarding and stable communication links under highly dynamic vehicular conditions. Consequently, the proposed routing protocol enhances network reliability, improves communication efficiency, and contributes to better overall performance in vehicular fog computing environments.

Table 5. Packet Loss Ratio (Plr) Comparison For Parked Vehicle And Urban Scenarios

Simulation time (seconds)	Packet Loss Ratio (PLR) (%) - Parked Vehicles				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaoA
100	21.35	19.49	16.85	13.88	11.75
200	18.81	16.05	14.17	11.63	9.82
300	16.44	13.48	11.35	9.56	7.75
400	14.02	12.39	10.27	8.43	6.24
500	12.09	10.16	8.33	6.59	4.48
Simulation time (seconds)	Packet Loss Ratio (PLR) (%) - Vehicles in urban area				
	DFLA	EPAR	ACO-ABC	EA-GMHBA	EA-AWaoA
100	26.58	23.13	19.72	16.77	15.15
200	24.37	21.74	17.65	15.12	12.64
300	22.42	19.46	15.76	13.43	11.31

400	20.24	17.83	14.33	11.28	9.15
500	18.65	15.52	13.15	9.82	7.63

CONCLUSION AND FUTURE WORK

In this paper, Energy Aware- Walrus Optimization Algorithm (EA-AWAO) is combined with Deep Q-learning (DQL), was used to enhance efficient routing and load balancing techniques for vehicular fog computing. This framework solved the problems according to the mobility of vehicles, uneven workload distribution, limited availability of resources in fog nodes and energy constraints using metaheuristic optimization and reinforcement learning. AWAO algorithm, optimum cluster heads and optimal routes are selected by the residual energy, inter-vehicle distance, closeness and vehicle velocity. Then, the load balancing scheme was performed through DQL regarding allocation of computation tasks according to the fog nodes' capabilities and buffer status to avoid congestion. Furthermore, DQL provided the adaptation of routing strategies considering the network dynamics which increased the reliability of the communication and QoS. The efficiency of the proposed routing protocol was analyzed under parked vehicles and urban scenarios. The results of simulations indicated that the proposed EA-AWAO outperformed the existing techniques in all metrics. In particular, it decreased the residual energy consumption and end-to-end delay and increased the packet delivery ratio while decreasing the packet loss ratio has been attained by proposed model. The proposed model can be used as an effective solution in terms of intelligent routing and load balancing in the vehicular fog computing environment. Possible research directions for future research include the use of multi-objective optimization techniques, trust-aware routing, blockchain-based security solutions, and deep reinforcement learning with federated or multi-agent approach in order to increase scalability, security and adaptability of the Internet of Vehicles (IoV) and next generation 6G enabled vehicular networks. In order to improve energy consumption (EC) and overall performance of the network, possible directions for future research include the application of the deep learning based IoT task scheduling framework along with hybrid optimization algorithms in the vehicular fog computing environment.

REFERENCES

1. Stergiou, C., Psannis, K.E., Kim, B.G. and Gupta, B., 2018. Secure integration of IoT and cloud computing.Future generation computer systems, 78, pp.964-975.
2. Humayun, M., 2020. Role of emerging IoT big data and cloud computing for real time application.International Journal of Advanced Computer Science and Applications, 11(4), pp.1-14.
3. Xu, X., Liu, Q., Luo, Y., Peng, K., Zhang, X., Meng, S. and Qi, L., 2019. A computation offloading method over big data for IoT-enabled cloud-edge computing. Future Generation Computer Systems, 95, pp.522-533.
4. Bazgir, E., Haque, E., Sharif, N.B. and Ahmed, M.F., 2023. Security aspects in IoT based cloud computing.World Journal of Advanced Research and Reviews, 20(3), pp.540-551.
5. Oma, R., Nakamura, S., Duolikun, D., Enokido, T. and Takizawa, M., 2018. An energy-efficient model for fog computing in the internet of things (IoT). Internet of Things, 1, pp.14-26.
6. Aazam, M., Zeadally, S. and Harras, K.A., 2018. Deploying fog computing in industrial internet of things and industry 4.0. IEEE Transactions on Industrial Informatics, 14(10), pp.4674-4682.
7. Shukla, S., Hassan, M.F., Khan, M.K., Jung, L.T. and Awang, A., 2019. An analytical model to minimize the latency in healthcare internet-of-things in fog computing environment. PloS one, 14(11), pp.1-31.
8. Butun, I., Sari, A. and Österberg, P., 2019. Security implications of fog computing on the internet of things. In 2019 IEEE international conference on consumer electronics (ICCE), pp.1-6.
9. Shukla, S., Thakur, S., Hussain, S., Breslin, J.G. and Jameel, S.M., 2021. Identification and authentication in healthcare internet-of-things using integrated fog computing based blockchain model. Internet of Things,15, pp.1-30.
10. Ngabo, D., Wang, D., Iwendi, C., Anajemba, J.H., Ajao, L.A. and Biamba, C., 2021. Blockchain-based security mechanism for the medical data at fog computing architecture of internet of things.Electronics,10(17), pp.1-17.
11. Goyal, A., Kaushik, S. and Khan, R., 2021. IoT based cloud network for smart health care using optimization algorithm. Informatics in Medicine Unlocked, 27, pp.1-8.
12. Haider, S.E., Khan, M.F. and Saeed, Y., 2024. Adaptive load balancing approach to mitigate network congestion in VANETS. Computers, 13(8), pp.1-17.
13. Qun, R. and Arefzadeh, S.M., 2021. A new energy-aware method for load balance managing in the fog-based vehicular ad hoc networks (VANET) using a hybrid optimization algorithm. IET Communications,15(13), pp.1665-1676.
14. Roh, B.S., Han, M.H., Ham, J.H. and Kim, K.I., 2020. Q-LBR: Q-learning based load balancing routing for UAV-assisted VANET. Sensors, 20(19), pp.1-18.
15. Mehta, T. and Mahato, D.P., 2023. Effective scheduling and nature inspired hybrid load balancing in VANETs. In IET Conference Proceedings CP832, 2023(5), pp. 266-273.

16. Sethi, V., Pal, S., Vyas, A., Jain, S. and Naik, K., 2022. Energy-and-delay-aware scheduling and load balancing in vehicular fog networks. *Telecommunication Systems*, 81(3), pp.373-387.
17. Wu, Y., Wu, J., Chen, L., Yan, J. and Han, Y., 2021. Load balance guaranteed vehicle-to-vehicle computation offloading for min-max fairness in VANETs. *IEEE Transactions on Intelligent Transportation Systems*, 23(8), pp.11994-12013.
18. Madasamy, B. and Balasubramaniam, P., 2022. Enhanced load balanced clustering technique for VANET using location aware genetic algorithm. *Promet-Traffic&Transportation*, 34(1), pp.39-52.
19. Giridhar, K., Anbuananth, C. and Krishnaraj, N., 2023. Energy efficient clustering with Heuristic optimization based Routing protocol for VANETs. *Measurement: Sensors*, 27, pp.1-8.
20. Elhoseny, M., El-Hasnony, I.M. and Tarek, Z., 2023. Intelligent energy aware optimization protocol for vehicular adhoc networks. *Scientific Reports*, 13(1), pp.1-14.
21. Abbas, A.H., Audah, L. and Alduais, N.A.M., 2018. An efficient load balance algorithm for vehicular ad-hoc network. In *2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, pp.207-212.
22. Silva, R.T., Colletti, R.R., Pimentel, C. and de Moraes, R.M., 2016. BETA random waypoint mobility model for wireless network simulation. *Ad Hoc Networks*, 48, pp.93-100.
23. Trojovský, P. and Dehghani, M., 2023. A new bio-inspired metaheuristic algorithm for solving optimization problems based on walruses behavior. *Scientific Reports*, 13(1), pp.1-32.
24. Hasanien, H.M., Alsaleh, I., Ullah, Z. and Alassaf, A., 2024. Probabilistic optimal power flow in power systems with renewable energy integration using enhanced walrus optimization algorithm. *Ain Shams Engineering Journal*, 15(3), pp.1-14.
25. Yıldız, Ö. and Sokullu, R.I., 2023. Deep Q-Learning based resource allocation and load balancing in a mobile edge system serving different types of user requests. *Journal of Electrical Engineering*, 74(1), pp.48-56.