China Petroleum Processing and Petrochemical Technology

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

EFFICIENT RESOURCE ALLOCATION IN VEHICULAR FOG COMPUTING: A MULTI-OBJECTIVE METAHEURISTIC OPTIMIZATION APPROACH FOR ENERGY OPTIMIZATION

V. S. Nici Marx

Research Scholar, Dept. of Computer Science and Engineering, Annamalai University

Annamalainagar – 608002, Tamilnadu, India, Email: nicimarx4896@gmail.com

Dr. G. Ramachandran

Associate Professor, Dept. of Computer Science and Engineering, Annamalai University

Annamalainagar – 608002, Tamilnadu, India, Email: gmrama1975@gmail.com

A. Kanthimathinathan

Associate Professor, Dept. of Computer Science and Engineering, Annamalai University

Annamalainagar – 608002, Email:kanthi_88@yahoo.co.in

Abstract

Vehicular Fog Computing (VFC) represents a novel paradigm that aims to improve the performance of vehicular applications by bringing computing and storage resources in closer proximity to vehicles. In VFC, vehicles can share data with each other and with Road Side Units (RSUs) to accomplish tasks such as traffic congestion detection, accident prediction, and infotainment. This study presents a multi-objective optimization model designed to identify the most suitable facility locations within VFC networks. The VFC infrastructure is initially set up, and relevant vehicle information is extracted as features. After that the essential features are selected by using the hybrid Jellyfish Egret Swarm Optimization (JESO) algorithm. The JESO algorithm combines the Jellyfish Search Optimization (JSO) algorithm with the Egret Swarm Algorithm (ESA) to identify the best optimal solution among the fog nodes during their communication, creating a hybrid approach. Based on the selected attributes, vehicles are grouped into clusters using the K-means algorithm, followed by the extraction of resource information for the Virtual Machines (VMs) on the cloud server. Finally, the Coati Optimization Algorithm (COA) performs optimal resource allocation within the VFC based on the clustered vehicle features and VM attributes. This proposed technique is implemented in the MATLAB platform, and its performance is assessed using various performance metrics. In the experimental evaluation, the performance of this research is compared with existing approaches such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), showing better performance.

Keywords: Road Side Units (RSUs), Vehicular Fog Computing (VFC), Jellyfish Egret Swarm Optimization (JESO) algorithm, K-means algorithm, cloud server, Coati Optimization Algorithm (COA),

1. Introduction

A new paradigm known as Vehicular Cloud Computing (VCC) blends the ideas of cloud computing with vehicular networks while utilizing the strength of Vehicular Ad hoc Networks (VANETs) for increased functionality and efficiency [1]. VANETs allow mobile, dynamic networks of cars to connect with one another and with roadside infrastructure in a decentralized, self-organizing fashion. The Vehicle Cloud (VC) in VCC systems, which consists of computer resources for several vehicles, aids the Remote Cloud (RC), a centralized cloud computing infrastructure that is external to the cars and enables real-time decision-making to improve traffic safety and driving comfort [2]. VCC can support various applications, such as traffic management, emergency response, data storage, and edge computing but still it is a nascent and evolving technology [3]. It offers a promising way to create a dynamic and efficient computing environment for vehicles, it creates a path to make use of underutilised vehicle resources like processing power, storage capacity, and internet connectivity [4, 5] and also enables the sharing of resources among vehicles or their rental to external entities, catering to various applications, including those within the automotive network [6]. This strategy has the potential to fulfil the rising demand for resources in the automotive network, but it needs careful analysis and creative problem-solving to be put into practice. Additionally, a number of computing paradigms, including Mobile Cloud Computing (MCC), multi-accessor Mobile Edge Computing (MEC), Vehicular Edge Computing (VEC), and Vehicular Fog Computing (VFC), have recently been developed to handle the delay-sensitive applications of modern intelligent vehicles [7, 8].

The Internet of Things (IoT) and cloud computing have been integrated into vehicular networks, allowing for the collection of real-time data on a variety of topics, such as traffic patterns, weather and vehicle performance. For in-depth research and decision-making, the acquired data is subsequently effortlessly transported to cloud servers. Platforms like IoT and cloud computing are crucial for giving customers the resources they need [9, 10]. They provide a workable remedy for the issues with vehicle networks. [11,12], leading to the evolution of the cloud vehicle as an extension of mobile cloud computing [13]. The traditional method of sending sense-actuate data directly from automobiles to the cloud has become unfeasible as vehicular applications increasingly demand low latency and energy-efficient solutions, posing a significant issue for the appropriate distribution of resources in VCC [14,15]. The inherent latency, limited bandwidth, high energy consumption, and network connectivity challenges associated with cloud communication have hindered the seamless operation of vehicular systems, especially in scenarios requiring real-time decision-making and critical safety applications.

Many research studies have been undertaken to address energy-aware resource allocation in VCC, employing a diverse range of algorithms namely Cuckoo Search Algorithm (CSA) and the Rat Swarm Optimizer (RSO), along with various techniques, all aimed at achieving optimized and efficient resource allocation [16-18]. These algorithms aim to improve the overall efficiency and energy consumption of VCC by intelligently distributing computational tasks, storage, and networking resources among connected vehicles. These intelligent algorithms leverage historical data and real-time feedback to make informed decisions, optimizing resource utilization and adapting to changing traffic patterns. Moreover, privacy and security concerns have prompted investigations into secure and privacy-preserving resource allocation mechanisms to safeguard sensitive data exchanged within VCC. As researchers continue to investigate and refine these algorithms, VCC's potential for enhancing vehicular communication and services becomes increasingly achievable. It represents a novel and evolving field of research, where cloud and vehicular networking intersect [19, 20].

The main contributions in this paper are:

- An innovative metaheuristic optimization approach is suggested for effective resource allocation in VCC that takes energy conservation into consideration.
- The suggested metaheuristic optimization method and models could provide trustworthy resource management and communication analysis, which is especially important in order to fulfil the rising needs for communication and computation in new networking applications.
- To determine the vehicle's energy, sound, speed, and delay, a hybrid Jellyfish Egret Swarm Optimization (JESO) method was used. Comparing the JESO approach to other well-known metaheuristic algorithms, it showed the most impressive feature selection and parameter optimization capabilities.
- In the K-means clustering process, k points are chosen as the cluster centroids from the data space.
- The Coati optimization technique is presented to determine the best resource distribution to vehicle cloud servers and enhance their functionality and effectiveness.
- The effectiveness of the suggested technique is assessed using a variety of performance indicators, such as energy usage, throughput, and packet delivery ratio. Utilizing MATLAB software, this strategy is put into practice.

The paper is structured in the following way: in section 2, the existing work pertaining to the proposed vehicular fog/cloud computing systems based on various metaheuristic approach is explicated; In section 3 provides an explanation of the proposed methodology, while Section 4 delves into the results and discussion of the proposed approach; and in section 5, the paper is concluded with future enhancement.

2. Related Survey

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

A highly efficient fog/cloud dumping system for vehicles certain challenges that face computing systems may be less significant with cloud-based solutions. An optimum offloading for the current network design could soon change into the least effective offloading because of the dynamic nature of VCC systems, which is mostly brought on by the movement of automobiles.

Qun et al. [21] presented an energy-aware technique for load balancing in fog-based VANETs. They employed the ACO-ABC approach, which combines artificial bee colonies with ant colony optimization. The results of the Network Simulator 2 simulation indicated that the VANET's energy usage increased as the number of nodes increased. Furthermore, when the number of jobs rose, the suggested method improved load balancing.

Gai et al. [22] examined the Swarm Optimized Non-Dominated Sorting Genetic Algorithm (SONG) for optimizing delay and energy-aware facility placement in vehicular fog networks. SONG combines the Non-dominated Sorting Genetic Algorithm (NSGA-II) and the Speed-constrained Multi-Particle Swarm Optimization (SMPSO), two popular Evolutionary Multi-Objective (EMO) techniques. The bounds of the delay-energy solutions and the related layout design were initially demonstrated by solving an example problem using the SONG approach. The performance of the SONG algorithm's development was then assessed using real vehicle traces and three quality indicators: Inverted Generational Distance (IGD), Hyper-Volume (HV), and CPU delay gap.

In accordance with the Service Level Agreements (SLA) of the applications, Materwala *et al.*'s [23] offloading program uses Evolutionary Genetic Algorithm (EGA) to optimize energy for edge-cloud integrated computing systems. Through the use of an adaptive penalty function integrated into EGA, the proposed approach incorporates optimization restrictions. The proposed method was evaluated utilizing comparative analysis and numerical testing against random and genetic algorithm-based offloading as well as a baseline with no offloading. Results reveal that, on average, the proposed method saves 2.97 times more energy than random offloading and 1.37 times more energy than no offloading.

In order to maximize the effectiveness of IoT task processing, Hameed *et al.* [24] presented a dynamic cluster-enabled capacity-based load-balancing technique for energy and performance-aware vehicular fog distributed computing. This method creates clusters that act as pools of computer resources based on the position, speed, and direction of the vehicles. By detecting departure timings from clusters, the article also suggests a method for predicting a vehicle's future position inside the dynamic network.

3. Proposed Methodology

The VFC optimization model is a multi-objective optimization model that aims to minimize service delay and energy consumption in VFC networks. The model takes into account several factors, including the location and capacity of fog nodes, traffic demand between fog nodes and

PDs (Presumably, PDs refer to some devices), communication costs between fog nodes, and the energy consumption of fog nodes. The model selects features using the adaptive Jellyfish Egret Swarm Optimization (JESO) algorithm, an adaptive swarm intelligence algorithm that combines ocean and fisher score information to identify the best path and reduce travel time. The K-Means algorithm is used to cluster vehicles into groups based on their residual energy and signal strength. This grouping helps connect vehicles with similar energy levels and communication capabilities, ultimately reducing service delay and energy consumption. Finally, the optimal facility locations in VFC networks are determined using the Coati optimization algorithm. Figure 1 displays the block diagram for the proposed technique.

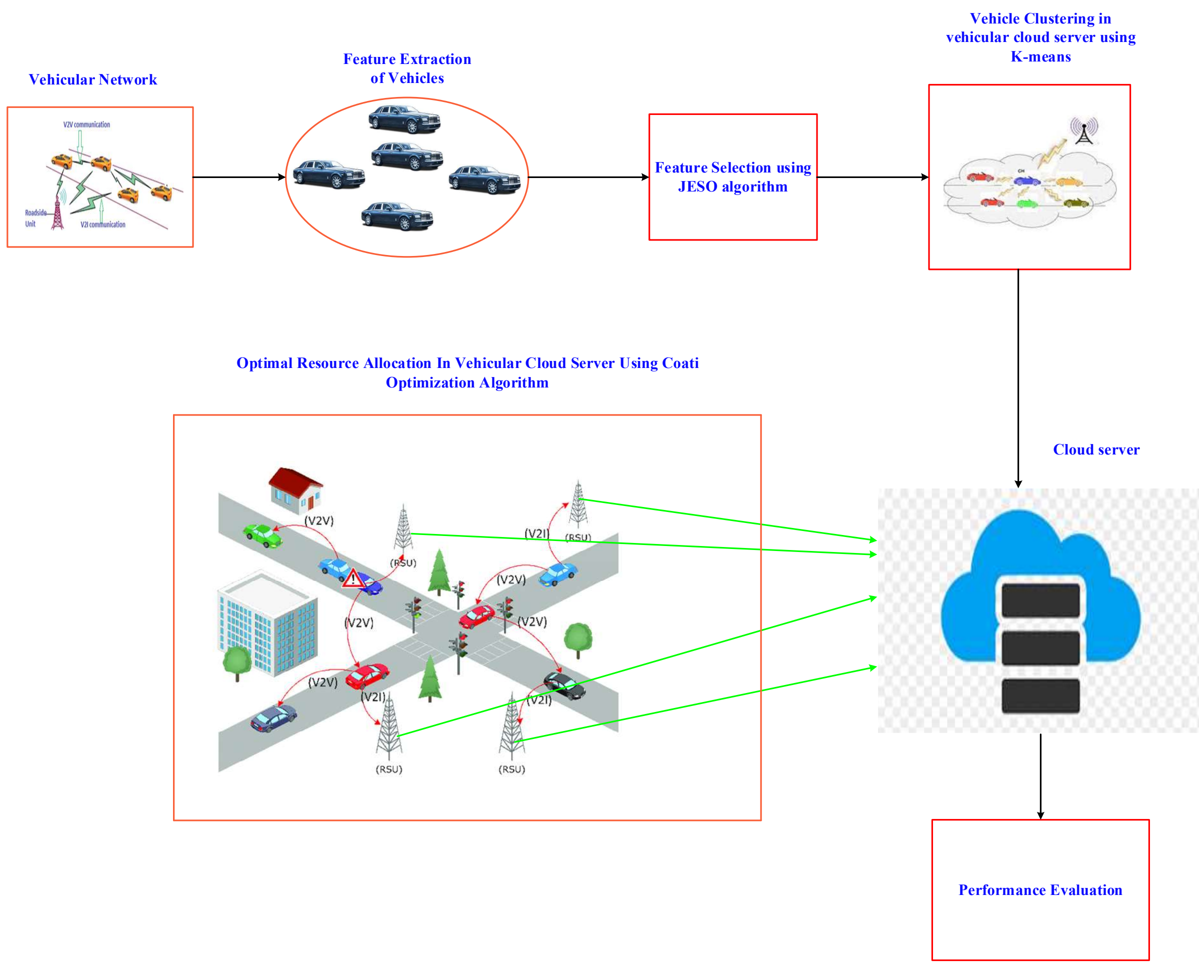


Figure 1 Block diagram for the proposed research methodology

The proposed procedure is explained as follows,

3.1. Vehicular Network Initialization

This section detailed explained about the vehicular network model. Considered as $R\{U(.), F(\chi)\}$ a linked undirected graph is the VANET situation. Set's $U(.)$ vertices consist of a cloud server (K), $\{u_1, u_2 ....., u_U\}$ a set of cars, a set of fog nodes $S = \{s_1, s_2 ....., s_S\}$. Highway traffic is a USK system that communicates unifiedly to a higher fog-cloud layer through BS or RSU. The collection of edges with weights $\chi = \{\chi_1, \chi_2 .....\}$ is represented by the function $F(\chi)$. In this case, χ_p each weight indicates the NUD (Network Usage Descriptor) that exists among every connect of vertices $U(.)$. In this study, proposed model χ_p by a triple $(e_{a2a}, \chi_{a2a}, \varsigma_{a2a})$ where e_{a2a} represents the end-to-end latency, χ_{a2a} indicates the connection bandwidth between two nodes, and ς_{a2a} denotes the transmission rate between two nodes. For every passenger, each vehicle has ϑ a fixed capacity of one DD, with a random range of 0 to D_ϑ. So, the total number of DDs in a USK system (cluster) might be specified as B $W = \sum_{\vartheta=1}^{U} D_\vartheta$.

A vehicle with a suitable arrival and departure rate that enters or exits the USK system using a Poisson process is called λ_u *and* μ_u. To send data and requests via one-hop connection, the cars employed the 802.11p communication standard. By shaking hands with one another, each vehicle may communicate the available fog nodes. The task offloading module decides which fog nodes to transfer tasks onto when the DD has a work to offload. The host vehicle will assess the available resources, such as RAM, vCPU, VM, bandwidth, etc., and decide whether to accept the job with a probability or offload it to a nearby fog node $\coprod_u$ or a distant CDC.

For the various facility location situations in the VFC, we developed a model for the typical reaction time ς_{sys}^{avg} and energy consumption h_{sys}^{avg} for vehicular applications based on the aforementioned hypotheses. Figure 2 provides the broad structure.

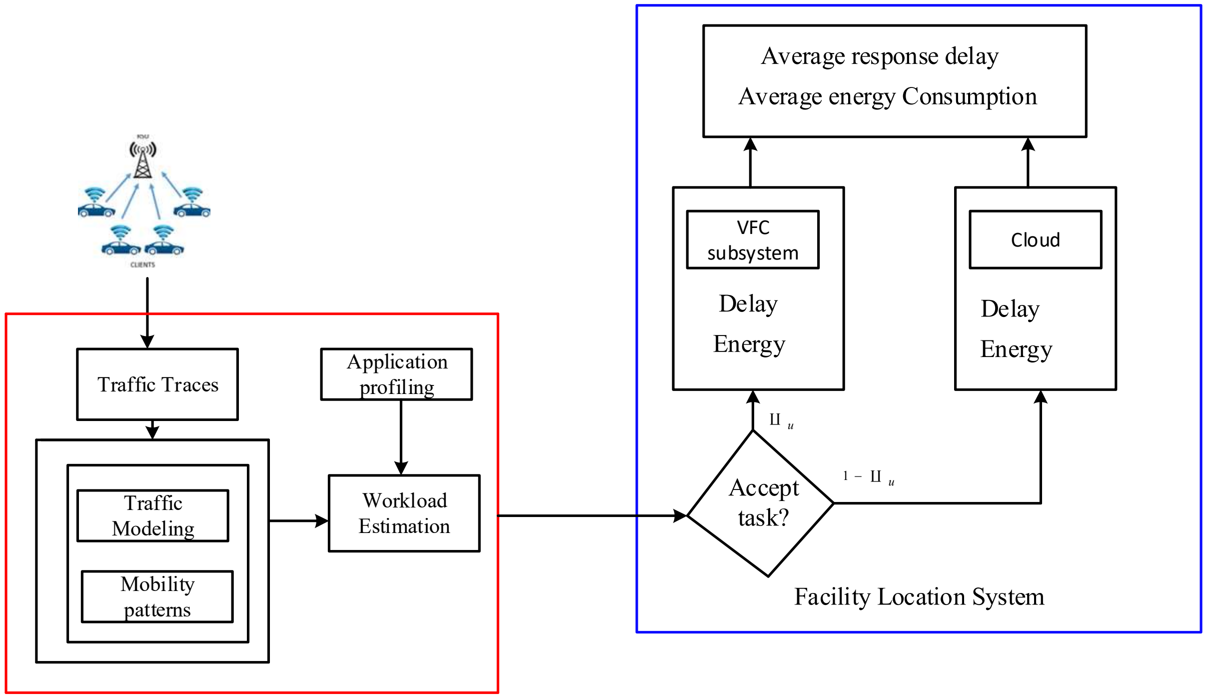


Figure 2 power consumption-delay trade-off in a VFC system

Decision-making factors

- ❖ The likelihood that the host vehicle u would approve the DD request is represented by $\coprod_u$ a probabilistic variable.
- ❖ A binary variable $TU_{u,e}$ indicating the location of the cloud-fog offloading choice, i.e.

$$TU_{u,e} = \begin{cases} 1 & \text{task e rom vehicle u is of e loaded to fog node e} \\ 0 & \text{to cloud} \end{cases} \quad (1)$$

- ❖ Both RSUs and taxis and buses have the ability to serve as fog nodes. A portion of these nodes will be chosen for a period of time to handle the workloads for passenger devices (PD). Whether a taxi or bus is chosen as a fog node is indicated by $TU_{e,s}$ the binary variable.

$$TU_{e,s} = \begin{cases} 1 & \text{i f RSU or taxi/bus s is selected as fog node e} \\ 0 & \text{not selected} \end{cases} \quad (2)$$

3.2. Feature Extraction of vehicles

In vehicular fog computing, feature extraction is the process of extracting useful information from the data collected by vehicles and the vehicular network. This information can be used to perform a variety of tasks, such as vehicle tracking, collision detection, traffic management, safety applications, and entertainment and infotainment [25]. The features that can be extracted from vehicles include their identification, location, speed, acceleration, heading, state, and behavior. The features that can be extracted from vehicles include:

Identification: The vehicle's unique identifier, such as its license plate number or vehicle identification number (VIN).

Location: The vehicle's current location, such as its GPS coordinates.

Speed: The vehicle's current speed, such as its odometer reading or radar measurement.

Acceleration: The vehicle's rate of change of speed, such as its accelerometer reading.

Heading: The direction in which the vehicle is facing, such as its compass reading.

State: The vehicle's current state, such as its engine RPM, brake status, or airbag deployment status.

Behavior: The vehicle's current driving behavior, such as its braking, acceleration, and steering patterns.

3.3. Feature Selection using JESO Algorithm

The process of selecting the most pertinent characteristics from a dataset is known as feature selection. This may be done to boost the performance of the adaptive Jellyfish Egret Swarm Optimization (JESO) method, for example, by lowering data noise or simplifying the models. The Jellyfish Search Algorithm (JSA) is prone to local optima and convergence problems. To address these issues, a proposed optimal resource allocation method calculates the egret swarm score for the initialized population and uses this value instead of randomly generated points. This helps the algorithm to avoid local optima and find global solutions. The adaptive JESO algorithm is then used to select features in vehicular fog networks. The algorithm first initializes a population of jellyfish. Each jellyfish represents a possible solution to the feature selection problem. The jellyfish then move through the search space, evaluating different features and updating their positions based on the egret swarm information. The jellyfish are more likely to move towards features with a high egret swarm score. The JSA provides a detailed description of the main steps, configuration parameters, and Pseudo code for a particular process.

An artificial optimization algorithm's population is often initialized at random. Due to low population variety, this method's drawbacks include sluggish convergence and a propensity to get stuck at local optimum [26]. Equation (3) shows that JSO uses a chaotic map known as the logistic map to broaden the starting population. This map generates more diverse beginning populations than doe's random selection and has a reduced likelihood of premature convergence.

$$P_{x+1} = \omega P_x \left(1 - P_x\right), 0 \le P_x \le 1 \tag{3}$$

To create the initial population of jellyfish, ω is set to 4.0 and the value of P_x is used. Here, P_{x+1} symbolizes the location of the xth jellyfish, and P_x stands for the logistic chaotic value related to that place. Because it is full of nutrients, an ocean circulation $\left(\overline{Trend}\right)$ attracts jellyfish. The direction of the ocean current may be determined by calculating the average of all the jellyfish in the population that have vectors pointing in the direction of the jellyfish that is now in the best position. Equation (4) may be used to model this ocean circulation.

$$\left(\overline{Trend}\right) = P^* - \mu * R\left(0,1\right) \tag{4}$$

Where P^* is the jellyfish that is now at the swarm's optimum position, μ is an average jellyfish location, and $\mu > 0$ is a distribution coefficient that is related to the strength of the ocean current $\left(\overline{Trend}\right)$. Thus, the new positions of each jellyfish are given by equations (5) and (6).

$$P_x\left(i+1\right) = P_x\left(i\right) + R\left(0,1\right) * \overline{Trend} \tag{5}$$

$$P_x\left(i+1\right) = P_x\left(i\right) + R\left(0,1\right) * P^* - \mu + R\left(0,1\right) * \omega \tag{6}$$

Where $P_x\left(i+1\right)$ is the jellyfish's new position and $P_x\left(i\right)$ is its existing location. Jellyfish move in both passive (type A) and active (type B) ways after. Most jellyfish first display type A motion when a swarm has just begun to assemble. They gradually start to move more in a type B manner. Jellyfish move in a type A pattern around their own areas. Equation (7) provides the revised position of each jellyfish in correspondence.

$$P_x\left(i+1\right) = P_x\left(i\right) + \gamma * R\left(0,1\right) * \left(U_b - L_b\right) \tag{7}$$

Where $\gamma > 0$ is a motion coefficient that is proportional to the length of the motion around each jellyfish's position, U_b and L_b are the upper bound and lower bound on the search space, respectively.

Every jellyfish in a swarm therefore moves about in search of a more advantageous location to obtain nourishment. The motion and most recent location of a jellyfish can be observed by Equations (8), (9) and (10), respectively. This action is seen as an efficient use of the local search space.

$$\overline{St} = R\left(0,1\right) * \overline{D} \tag{8}$$

$$\overline{D} = \begin{cases} P_y(i) - P_x(i) \; if \;\; f(P_x) \geq f(P_j) \\ P_x(i) - P_y(i) \; if \;\; f(P_x) < f(P_j) \end{cases} \tag{9}$$

Hence,

$$P_y(i+1) = P_y(i) + \overline{St} \tag{10}$$

Where f is a specific function of the location P.

If the objective function's value is improved, the egret swarm optimization algorithm's new location is approved. The egrets have taken the place of the jellyfish that was in its prior location. High energy consumption in exchange for perhaps higher rewards is a characteristic of Great Egret with Aggressive Strategy. ESO algorithm defines methods to better simulate egrets' learning habits. $D_{a,x}$ and $D_{b,x}$. $D_{a,x}$ indicate the direction change based on the same group of egrets' ideal position. $D_{b,x}$ signifies a direction shift accomplished in accordance with all egrets' ideal positions [27].

$$D_{b,x} = \frac{P_x^{best} - P_x}{\left|P_x^{best} - P_x\right|} * \frac{f_x^{best} - f_x}{\left|f_x^{best} - f_x\right|} + D_x^{best} \tag{11}$$

$$D_{g,x} = \frac{P_g^{best} - P_x}{\left|P_g^{best} - P_x\right|} * \frac{f_g^{best} - f_x}{\left|f_g^{best} - f_x\right|} + D_g^{best} \tag{12}$$

Where P_g^{best} and P_x^{best} stand for the respective global historical ideal position and individual historical optimal position. The goal function is f. D_x^{best} indicates the historical change in the ideal direction depending on the position of the same group of egrets. D_g^{best} indicates the historical change in the ideal direction based on the ideal location of all egrets. Then, because the ocean current is rich in nutrient-rich food, jellyfish are drawn to it. As additional jellyfish assemble, a swarm eventually forms. The random integer used for the time control function fluctuates from 0 to 1 over time. When its value reaches, equation (13)'s temporal control function causes the jellyfish to float together with the ocean current.

$$Ctl(i) = \left|\left(1 - \frac{i}{Max_I}\right) * (2 * R(0,1) - 1)\right| \tag{13}$$

Where $Ctl(i)$ is the time control function, and Max_I signifies the maximum number of iterations, and i is the time indicated by the iteration number.

Input: Obtained features of vehicles

Output: Selected best path

Begin

Initialize population, fitness, food, iteration I and maximum iteration Max_I

Compute fitness

Set iteration $rr_t = 1$

While $\left(I \leq Max_I\right)$ **do**

Calculates time control by, $Ctl(i) = \left|\left(1 - \frac{i}{Max_I}\right) * \left(2 * R(0,1) - 1\right)\right|$

Calculate the ocean current $\left(\overline{Trend}\right) = P^* - \mu * R(0,1)$

Update the jellyfish's new location

$$P_x(i+1) = P_x(i) + R(0,1) * P^* - \mu + R(0,1) * \omega$$

{

If

Jellyfish move in a type A manner. $P_x(i+1) = P_x(i) + \gamma * R(0,1) * \left(U_b - L_b\right)$

Else

Jellyfish move in a type B manner $\overline{D} = \begin{cases} P_y(i) - P_x(i) \; if \;\; f(P_x) \geq f(P_j) \\ P_x(i) - P_y(i) \; if \;\; f(P_x) < f(P_j) \end{cases}$

End if

Get each **egret's** history ideal location as well as the overall historical best position using,

$$D_{b,x} = \frac{P_x^{best} - P_x}{\left|P_x^{best} - P_x\right|} * \frac{f_x^{best} - f_x}{\left|f_x^{best} - f_x\right|} + D_x^{best}$$

$$D_{g,x} = \frac{P_g^{best} - P_x}{\left|P_g^{best} - P_x\right|} * \frac{f_g^{best} - f_x}{\left|f_g^{best} - f_x\right|} + D_g^{best}$$

Calculate fitness

Set $I = I + 1$

End while

Return features of vehicles

End

3.4. Vehicle clustering vehicular cloud using K-means technique

K-means clustering is a machine learning algorithm that can be used to group vehicles in a vehicular cloud into a predefined number of clusters. The algorithm works by first randomly selecting k points in the data space as the cluster centroids. The nearest cluster's centroid is then chosen to house each car. The centroids are then revised to represent the average of the cars in each cluster. You keep doing this until the centroids stop moving. There are several ways that vehicle clustering may be utilized to enhance the performance of vehicular cloud applications. For example, it can be used to optimize the allocation of resources, such as bandwidth and storage, to different clusters [28]. It can also be used to improve the reliability of communication between vehicles in the same cluster. Additionally, it can be used to facilitate the sharing of data and services between vehicles. The data control unit gathered the messages, filtered them to remove any duplicates, and then used a K-means clustering algorithm to divide the messages into four groups. The following are the definitions of the terms:

$$Sim_M = \frac{\sum_{n=1}^{N} M_{S*A*D}}{N} \tag{14}$$

The neighbors of the target node are displayed in this image N along with how closely their speeds, accelerations, and orientations reflect those of the target node represents M_{S*A*D}. The mobility similarity identifies how similar each of the target node's neighboring nodes' motion states are to one another. The bigger the value of the mobile similarity, the more similar the node's mobility state is to the motion states of surrounding automobiles. The cluster heads that are chosen are determined by the vehicle's position and the typical separation between it and its neighbors. As a result, average adjacency must be considered in addition to location. The placement element describes where the automobile is on the street, for example, if it is in the middle of the road. As

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

the cluster head approaches the road's center, its usage rate increases, allowing it to accept additional cluster member nodes [29]. The following are examples of location factors:

$$L = 1 - \frac{|w_i - w_o| * |v_i - v_o|}{|w_o| * |v_o|} \tag{15}$$

Here, $(v_0, w_o)$and (v_0, w_o)signify the coordinates of the road center and the vehicle node, respectively. The cluster structure becomes more compact and communication will be more successful if the cluster head is close to the close neighbors. It clarifies the usual adjacency.

$$D = 1 - \frac{\sum D_{ij}}{N} \tag{16}$$

When N, the total number of neighbors of the target node, is taken into account, the D_{ij} normalized distance between the target node i and the neighbor node j is significant.

$$D_{ij} = \frac{\sqrt{(v_i - v_j)^2 + (w_i - w_j)^2}}{T_R} \tag{17}$$

Here, T_R is the communication radius, $(v_j, w_j)$is the neighbor nodes' coordinates, and $(v_i, w_i)$is the destination node's coordinates.

The following Ψ is the definition of the cluster head selection factor:

$$\Psi = \lambda_1 * S_M + \lambda_2 * L + \lambda_3 * D \tag{18}$$

It is beneficial to utilize public transportation as much as possible as cluster leaders for local communications. Because of this, it is anticipated that the cars will be evenly spaced out along the whole route and that the cluster head may be selected from one of the cars during the research's cluster head selection process. Once the values for each vehicle have been established, the cluster head with the greatest cluster head selection Ψ factor is selected as the cluster head. After the data is stored in the storage cloud server, the vehicles can be clustered using a variety of k-means clustering. The choice of k-means clustering algorithm depends on the specific application and the features that are being used to cluster the vehicles.

3.5. Optimal Resource Allocation in Vehicular Cloud Server Using Coati Optimization Algorithm

The Coati optimization algorithm is a bio-inspired algorithm that simulates the foraging behavior of coatis to find the optimal solution to an optimization problem. In this case, the Coati

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

optimization algorithm is used to find the optimal resource allocation in a vehicular cloud server. The vehicular cloud server is a network of vehicles that are connected to each other and to the Internet. The vehicles can share their resources, such as computing power, storage, and bandwidth, with each other. Due to the ability to offload compute positions to other vehicles, the performance of the tasks can be enhanced, and the energy consumption of the vehicles can be decreased. In this situation, the Coati optimization method evaluates resource allocations while taking into consideration VM properties. Each coati in the population is a potential solution to the problem of resource allocation when factors like VM attributes are taken into factor. The coatis then wander across the search area, assessing various resource allocations and revising their placements in accordance with how the coatis forage [30]. When taking into account the distinct characteristics of VMs, the coatis are more inclined to gravitate toward resource allocations that are more efficient.

Procedure of COA algorithm

Step 1: Initialization

Equation (19) is used to initialize the coatis position in the search space at random at the start of the COA implementation.

$$X_i^{P1} : x_{ij}^{P1} = x_{ij} + r.(Iguana_j - I.x_{ij}), \\ for\, i = 1,2,..., \left[\frac{N}{2}\right] and\ j = 1,2,...,m. \tag{19}$$

Step 2: The iguana collapses on the ground

The iguana is dropped to the ground and then positioned at random somewhere inside the search area. Equations (20) and (21), which represent the search space, are then used to demonstrate how coatis on the ground move in response to this random position.

$$X = \frac{1}{4}\log\left(i + \frac{1}{i_{\max}}\right)b \tag{20}$$

$$for\, i = \left[\frac{N}{2}\right] + 1, \left[\frac{N}{2}\right] + 2,..., N\ and\ j = 1,2,....,m. \tag{21}$$

Step 3: Fitness function calculation

This article's major objective is to suggest a resource allocation strategy that is effective and minimizes VFC's overall energy consumption, including the power consumed by the RSU.

$$OF = Max\sum_{x\in y} M_x.T_y \tag{22}$$

Where T_y represents the maximum number of time units that vehicles in category y can travel, and M_x denotes the Million Instructions Per Second (MIPS) rating for the CPU of vehicle x.

Step 4: Update function

If the revised location for each coati results in an increase in the value of the objective function, the update process is considered acceptable; otherwise, the coati remains in its previous position. Equation (23), applied for i = 1, 2, ..., N, defines this update criterion.

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} < F_i, \\ X_i, & else. \end{cases} \tag{23}$$

The term 'iguana' signifies the location of a coati in the search space, representing the position of the best member. The new position is X_i^{P1} calculated for coati 'i' when its objective function value is 'i', with 'r' denoting a randomly selected integer within a specified interval.

Step 5: Generated near the position

In order to mimic this behavior, a random position is generated near to where each coati is placed using equations (24) and (25).

$$Ib_j^{local} = \frac{Ib_j}{t}, ub_j^{local} = \frac{ub_j}{t},\ t = 1,2,...T. \tag{24}$$

$$\begin{aligned} &X_i^{P2} : x_{ij}^{P2} = x_{ij} + r.(1-2r).\left(Ib_j^{local} + r.(ub_j^{local} - Ib_j^{local})\right), \\ &for\, i = 1,2,...,N,\ j = 1,2,...,m. \end{aligned} \tag{25}$$

Step 6: Best position update

This condition, which is repeated by equation (26), indicates that the newly calculated position is acceptable if it raises the value of the goal function.

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} < F_i, \\ X_i, & else. \end{cases} \tag{26}$$

Where, 'i' has been calculated in X_i^{P2} its updated position. 'Iguana' signifies the location of the iguana within the search space, representing the position of the best member, while 'coati'

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

represents the value of the objective function. 'r' is a randomly chosen real number within the specified interval. Finally, the coati optimization technique is employed to enhance the resource allocation in mobile cloud servers. The effectiveness of the suggested research approach is evaluated in the next section.

4. Performance Evaluation

The proposed VANET system was simulated using MATLAB simulation tools. Using MATLAB software (2022 b) and an Intel Core i5-2450M CPU 2.50GHz laptop with 6GB Memory, the proposed technique is used to authenticate the attendance of the predictable and proposed VFC. Performance metrics like throughput, delay, energy, SNR and Throughput can be used to examine the proposed method. The two bases of arrival rate-based results and time-based outcomes, respectively, are used to validate the proposed approach. Table 1 lists the implementation variables for the suggested approach.

Table 1: Simulation parameters

S. No	Description	Parameters
1	Radio Range	300m
2	Road Side unit	10
3	Short time period	5 seconds
4	Road length	5 km
5	Simulation Area	1000m *1000m
6	Number of Access point	8
7	Signal to Noise ratio	20dB
8	Channel model	Nakagami m fading model
9	Road configuration	1 lane in each direction
10	Traffic constant	0.25
11	Packet Size	512 Bytes
12	Data generation	Poisson distribution
13	Vehicle speed	12-30 m/s

14	Number of vehicles	150
15	Mobility model	Random waypoint

4.1. Performance Metrics

In order to assess the performance of the proposed system and make comparisons with other methods, various parameters including delay, delivery ratio, energy consumption, Signal-to-Noise Ratio (SNR), and throughput were employed. This section provides explanations for each of these parameters and presents a comparative analysis of all three optimization algorithms: Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and the proposed algorithm.

Packet delivery ratio (PDR)

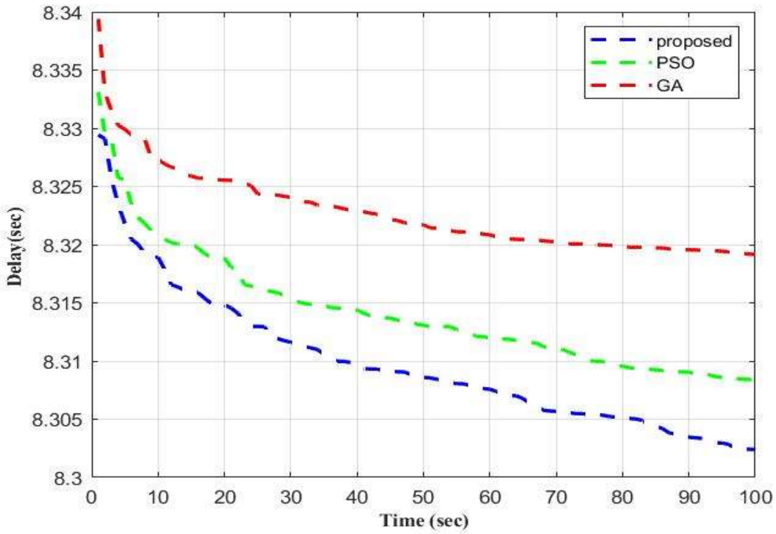
$$PDR = \frac{Number\ of\ packets\ received\ by\ the\ destination}{Number\ of\ packets\ sent\ by\ the\ source} \tag{27}$$

Delay

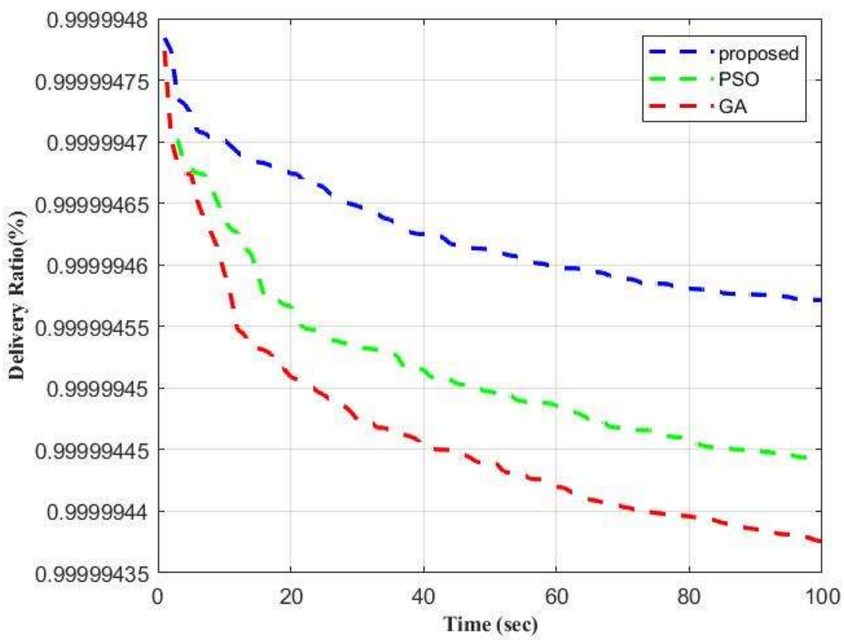
$$D = \frac{\sum_{x=0}^{q}\left(\left(time\ of\ receiving\ the\ xth\ packet\right) - \left(time\ of\ sending\ the\ xth\ packet\right)\right)}{total\ number\ of\ packets\ received\ by\ the\ destination} \tag{28}$$

Throughput

$$T = \frac{\sum_{x=0}^{q}\left(packets\ received\right)}{t} \tag{29}$$



(a)



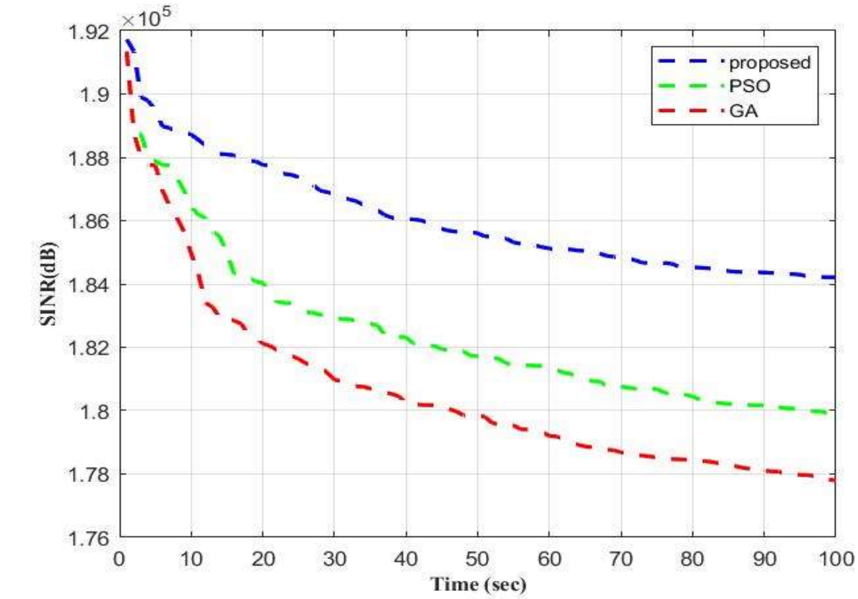
(b)

Figure 3 Performance analysis of (a) Delay and (b) Delivery Ratio

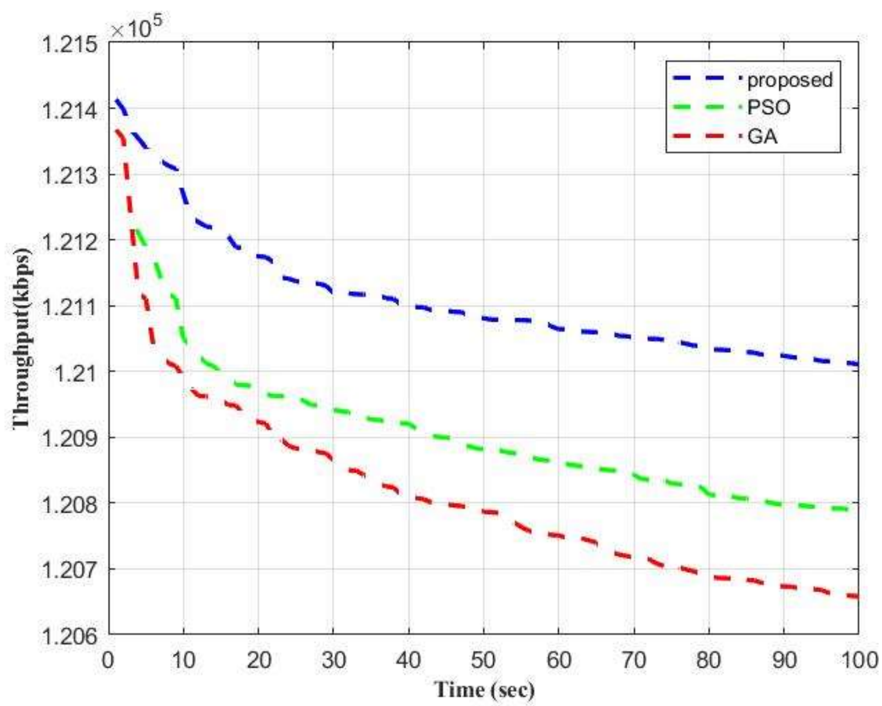
The delay and delivery ratio performance comparison are shown in Figure 3. The proposed approach is contrasted with traditional approaches like PSO and GA. The delay of the proposed method is 8.303*10^{-6}. The PSO and GA both have delay values of 8.309*10^{-6} and 8.324*10^{-6}, at 100 sec respectively. The delivery ratio of the proposed method is achieved the 0.99999463. The

Catalyst Research Volume 23, Issue 2, December 2023 Pp. 4134-4159

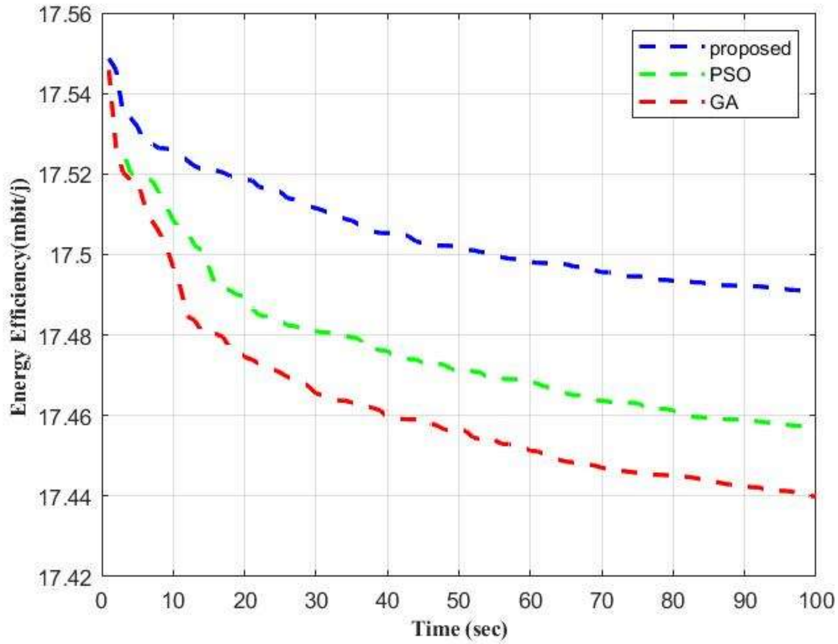
conventional technique of PSO and GA is attained the 0.99999445 and 0.99999437. Based on the analysis, the proposed technique is achieved efficient incomes of delivery ratio.



(a)



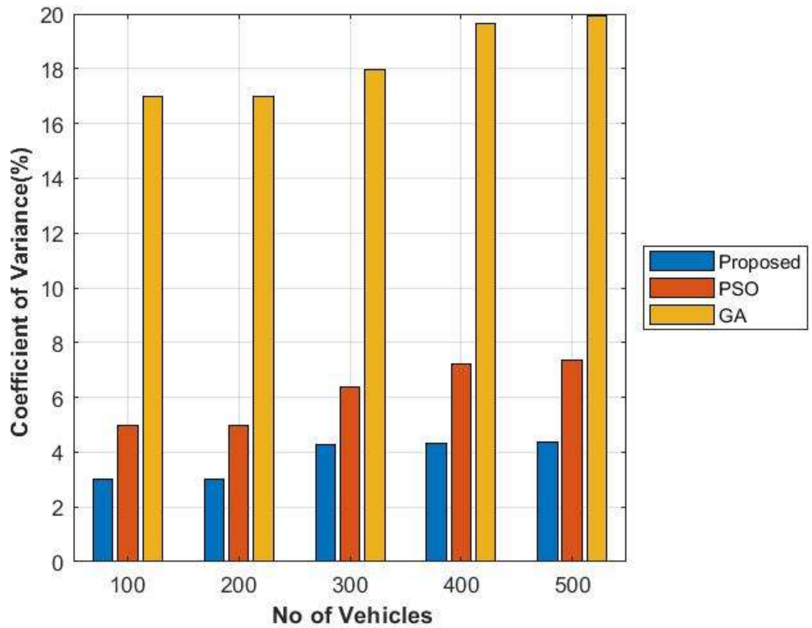
(b)



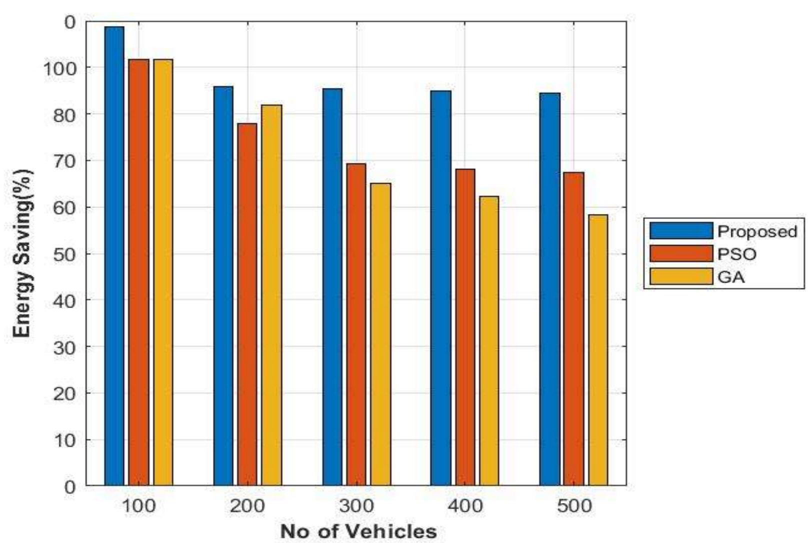
(c)

Figure 4 Performance analysis of (a) SNR (b) Throughput and (c) Energy efficiency

Figure 4(a) presents an analysis of the SNR output in both the proposed and existing methods within VFC. In this context, the proposed approach, which assigns weights to each surrounding vehicle's coordinates based on SNR values and distances, represents an advancement in the concept of VFN centroid localization. Because, the proposed research methodology uses the better optimal location function with the VFC. The throughput of the proposed and existing technique is illustrated in figure 4(b). The proposed technique is achieved the 1.21 at 100 seconds. The existing technique of PSO and GA is attained the 1.208 and 1.205 seconds. Based on the analysis, the proposed technique is achieved efficient incomes of throughput. The energy efficiency for the proposed technique is compared to that of the PSO and GA in Fig.4(c). The proposed method has the highest energy efficiency at 17.49, followed by the PSO method at 17.456 and the GA method at 17.44. The results of the proposed model show the performance regarding delay, throughput and packet delivery ratio.

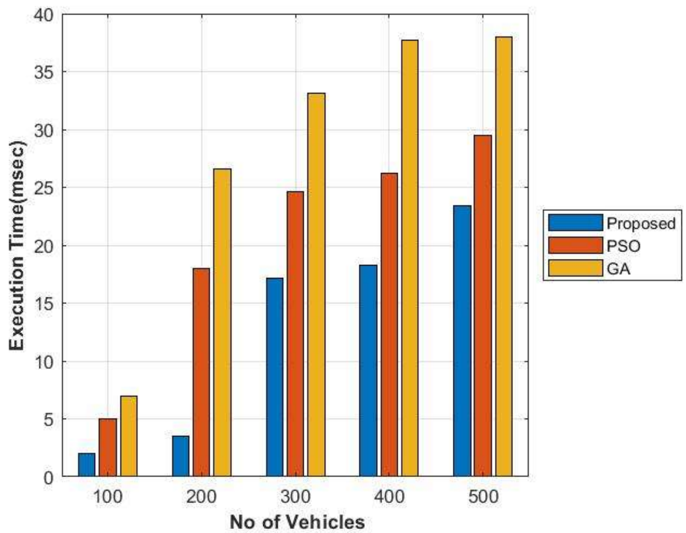


(a)

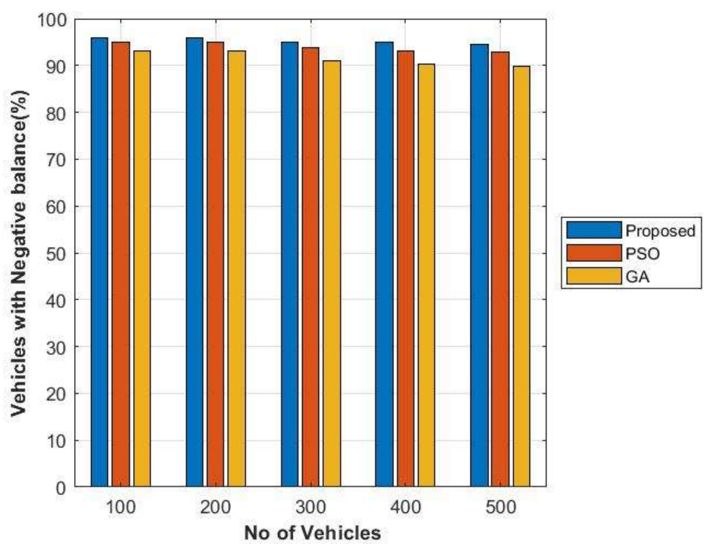


(b)

Figure 5 The impact of the number of vehicles on (a) the Coefficient of Variance (CV) of energy balance and (b) energy savings is examined in the study



(a)



(b)

Figure 6 The study investigates the influence of the number of vehicles on (a) the algorithm's execution time and (b) the percentage of vehicles with a negative balance

Figure 5(a) shows that the suggested approach delivers a fair distribution of energy savings, resulting in a lowering Coefficient of Variance (CV) value, due to the restricted number of vehicles that may carry out their responsibilities. In contrast, both PSO and GA exhibit an uneven distribution, leading to an increasing CV value. This trend translates to a consistent 50% energy savings with the proposed technique, while PSO and GA experience diminishing savings as the number of cars increases, as depicted in Figure 5(b). Both the proposed and existing systems display polynomially rising execution times as the number of automobiles rises, as seen in Figure 6(a). It is acceptable since the execution time of these systems regularly stays below 10% of the duration of the energy-manager period. Conversely, Figure 6(b) highlights that the proposed strategy enables every vehicle to achieve energy savings within 10 cycles, whereas PSO and GA struggle to do so due to resource limitations when the number of vehicles in the edge-cell exceeds 300. These tests demonstrate that the suggested metaheuristic optimization technique enables vehicles to save energy regardless of the number of vehicles under management. Energy efficiency was increased by the network's low communication costs as well as the cloud server's segmentation of messages into safe and unsafe types to be sent on processing machine fog nodes and cloud infrastructure.

5. Conclusion

In this paper the energy aware cluster based ideal facility locations for VFC is proposed. Here, by extending the routing durations in the suggested technique with the adaptive JESO methodology, the number of feature residual nodes was enhanced. The vehicle clustering formation in vehicular cloud server is done by the K-means clustering algorithm and communication between the cluster head selection. Then, best resource allocation for the cloud server by using the Coati optimization algorithm. In contrast with existing research techniques, the performance of the proposed method has been assessed. The energy usage inside the vehicular fog-based VANET develops as the number of nodes increases, according to simulation findings performed in the MATLAB environment. In optimal resource allocation in vehicular cloud server the proposed approach is compared with the existing PSO and GA approach based on delay, delivery ratio, throughput, energy efficiency and SNR. Based on all the metrics the proposed attains better performance than the existing proposed attains highest efficiency. Future research will focus on IoT-based vehicle mobility and security from the point of origin to the final destination.

References

[1] Soni, Y., & Jangirala, S. (2022). Survey on vehicular cloud computing and big data. *EasyChair Preprint*

[2] Jones, R., Seidelin, C. F., Neang, A. B., & Lee, C. P. (2023). Lessons Learned from a Comparative Study of Long-Term Action Research with Community Design of Infrastructural Systems. *Proceedings of the ACM on Human-Computer Interaction*, *7*(CSCW1), 1-35.

[3] Sarkar, S., Wankar, R., Srirama, S. N., & Suryadevara, N. K. (2019). Serverless management of sensing systems for fog computing framework. *IEEE Sensors Journal*, *20*(3), 1564-1572.

[4] Li, C., Zuo, X., & Mohammed, A. S. (2022). A new fuzzy-based method for energy-aware resource allocation in vehicular cloud computing using a nature-inspired algorithm. *Sustainable Computing: Informatics and Systems*, *36*, 100806.

[5] Kouser, R. R., & Manikandan, T. (2023). A hybrid Square Shaped Adaptive Neuro-Fuzzy Interference with M-PeSOA model for optimal path selection. *Journal of Intelligent & Fuzzy Systems*, (Preprint), 1-13.

[6] Shaik, S. (2019). Resource and Service Management for Fog Infrastructure as a Service. *Ph. D. Dissertation*.

[7] Bi, C., Li, J., Feng, Q., Lin, C. C., & Su, W. C. (2023). Optimal deployment of vehicular cloud computing systems with remote microclouds. *Wireless Networks*, 1-13.

[8] Binwal, D. C., & Kapoor, M. (2022). A survey on architecture, applications, and challenges in vehicular fog computing. *International Journal of Sensors Wireless Communications and Control*, *12*(3), 194-211.

[9] Xie, Y., Sheng, Y., Qiu, M., & Gui, F. (2022). An adaptive decoding biased random key genetic algorithm for cloud workflow scheduling. *Engineering applications of artificial intelligence*, *112*, 104879.

[10] Braun, T. D., Siegel, H. J., Beck, N., Bölöni, L. L., Maheswaran, M., Reuther, A. I., ... & Freund, R. F. (2001). A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems. Journal of Parallel and Distributed computing, 61(6), 810-837.

[11] Sui, T., Marelli, D., Sun, X., & Fu, M. (2020). Multi-sensor state estimation over lossy channels using coded measurements. *Automatica*, *111*, 108561.

[12] Lv, Z., Qiao, L., & You, I. (2020). 6G-enabled network in box for internet of connected vehicles. *IEEE transactions on intelligent transportation systems*, *22*(8), 5275-5282.

[13] Iqbal, M. A., Olaleye, O. G., & Bayoumi, M. A. (2016). A review on internet of things (IoT): security and privacy requirements and the solution approaches. *Global Journal of Computer Science and Technolog*, *16*(7), 1-10.

[14] Elhoseny, M., Thilakarathne, N. N., Alghamdi, M. I., Mahendran, R. K., Gardezi, A. A., Weerasinghe, H., & Welhenge, A. (2021). Security and privacy issues in medical internet of things: overview, countermeasures, challenges and future directions. *Sustainability*, *13*(21), 11645.

[15] Stoyanova, M., Nikoloudakis, Y., Panagiotakis, S., Pallis, E., & Markakis, E. K. (2020). A survey on the internet of things (IoT) forensics: challenges, approaches, and open issues. *IEEE Communications Surveys & Tutorials*, *22*(2), 1191-1221.

[16] Midya, S., Roy, A., Majumder, K., & Phadikar, S. (2018). Multi-objective optimization technique for resource allocation and task scheduling in vehicular cloud architecture: A hybrid adaptive nature inspired approach. *Journal of Network and Computer Applications*, *103*, 58-84.

[17] Tariq, J., Javed, M., Ayub, B., Armghan, A., Ijaz, A., Alenezi, F., ... & Zulfiqar, A. (2023). Nature inspired algorithm based fast intra mode decision in HEVC. *Multimedia Tools and Applications*, 1-16.

[18] Ahmed, B., Malik, A. W., Hafeez, T., & Ahmed, N. (2019). Services and simulation frameworks for vehicular cloud computing: a contemporary survey. *EURASIP Journal on Wireless Communications and Networking*, *2019*, 1-21.

[19] Ma, K., Li, Z., Liu, P., Yang, J., Geng, Y., Yang, B., & Guan, X. (2021). Reliability-constrained throughput optimization of industrial wireless sensor networks with energy harvesting relay. *IEEE Internet of Things Journal*, *8*(17), 13343-13354.

[20] Hussain, Md Muzakkir, and MM Sufyan Beg. "CODE-V: Multi-hop computation offloading in Vehicular Fog Computing." Future Generation Computer Systems 116 (2021): 86-102.

[21] Qun, R., & 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), 1665-1676.

[22] Gai, K., Qin, X., & Zhu, L. (2020). An energy-aware high performance task allocation strategy in heterogeneous fog computing environments. IEEE Transactions on Computers, 70(4), 626-639.

[23] Materwala, H., Ismail, L., Shubair, R. M., & Buyya, R. (2022). Energy-SLA-aware genetic algorithm for edge–cloud integrated computation offloading in vehicular networks. Future Generation Computer Systems, 135, 205-222.

[24] Hameed, A. R., ul Islam, S., Ahmad, I., & Munir, K. (2021). Energy-and performance-aware load-balancing in vehicular fog computing. Sustainable Computing: Informatics and Systems, 30, 100454.

[25] Al-Wesabi, Fahd N., et al. "Energy aware resource optimization using unified metaheuristic optimization algorithm allocation for cloud computing environment." Sustainable Computing: Informatics and Systems 35 (2022): 100686.

[26] Chou, Jui-Sheng, and Dinh-Nhat Truong. "A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean." Applied Mathematics and Computation 389 (2021): 125535.

[27] Chen, Zuyan, et al. "Egret swarm optimization algorithm: an evolutionary computation approach for model free optimization." Biomimetics 7.4 (2022): 144.

[28] Li, Luyao, Yang Qiu, and Jing Xu. "A K-means clustered routing algorithm with location and energy awareness for underwater wireless sensor networks." Photonics. Vol. 9. No. 5. MDPI, 2022.

[29] Al Maruf, Md, et al. "Faster fog computing based over-the-air vehicular updates: A transfer learning approach." IEEE Transactions on Services Computing (2021).

[30] Dehghani, Mohammad, et al. "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems." Knowledge-Based Systems 259 (2023): 110011.