

Optimized Cluster-Based Routing Protocol for Improved Bandwidth and Reduced Delay in Vehicular Ad-Hoc Networks

G. Karthikeyan¹, Dr. S. Rizwana²

¹Assistant Professor, Department Computer Science (Software Systems) PSG College of Arts & Science, Coimbatore.

²Assistant Professor & Head, Department Computer Science Erode Arts and Science College, Erode.

ARTICLE INFO

Received: 12 Oct 2024

Revised: 11 Dec 2024

Accepted: 24 Dec 2024

ABSTRACT

Vehicular Ad Hoc Network (VANET) is a cutting-edge technology that enables communication between vehicles and the surrounding road infrastructure, paving the way for intelligent transportation systems. Ensuring stable connections in VANETs requires a robust and reliable routing protocol, as these networks lack central coordination, exhibit high node mobility, and possess highly dynamic topologies, making routing a significant challenge. Many existing mobility-based routing protocols fall short in addressing Quality of Service (QoS) requirements, as their performance is heavily influenced by vehicle speed and driving conditions. On the other hand, QoS-based approaches often fail to account for the challenges posed by high-speed mobility, leading to frequent connection failures due to the increasing mobility of nodes within a given area. To overcome these challenges, this paper introduces a novel Cluster-Based Congestion Control Routing (CBCCR) protocol designed to enhance overall network efficiency by improving route throughput, optimizing bandwidth utilization, and minimizing end-to-end latency. The CBCCR protocol addresses the limitations of repetitive route detection and frequent Cluster Head (CH) reelection processes, thereby enhancing route stability and network performance. The proposed approach involves several key steps. First, the network is divided into stationary clusters. Next, a new distributed CH selection method is introduced, leveraging specific parameters to optimize the selection process. To further improve efficiency, an Enhanced BAT Algorithm (EBA) is employed for CH selection, and a novel routing method is developed to identify the most suitable CH based on the destination's position and the locations of neighboring CHs. Simulation results highlight the effectiveness of the CBCCR protocol, showcasing significant improvements in bandwidth utilization, increased throughput, and reduced transmission delays. These findings demonstrate that the CBCCR protocol offers a robust and efficient solution to the routing challenges in VANETs, making it a promising approach for advancing intelligent transportation systems.

Keywords: Routing, Cluster-Based Congestion Control Routing Protocol, Enhanced Bat Algorithm, Vehicular Adhoc Network.

INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) represent a groundbreaking technology that facilitates communication between vehicles and surrounding road infrastructure, enabling advanced intelligent transportation systems. Establishing stable connections in VANETs requires a robust routing protocol, as these networks are inherently decentralized, feature highly mobile nodes, and exhibit dynamic topologies, making routing a significant challenge. Existing mobility-based routing protocols often fail to address Quality of Service (QoS) requirements due to their dependency on vehicle speed and driving conditions. Conversely, QoS-based methods overlook the challenges posed by high-speed mobility. As a result, frequent connection failures occur in these networks due to the rapid movement of nodes within a given region.

This paper presents a novel Cluster-Based Congestion Control Routing (CBCCR) protocol designed to enhance overall network efficiency by improving route throughput, optimizing bandwidth utilization, and minimizing end-to-end latency. The CBCCR protocol eliminates repetitive route detection

processes, thereby reducing the frequency of Cluster Head (CH) reelection. In dynamic scenarios, such as those spanning a two-minute duration, the CBCCR protocol enhances route stability and ensures efficient bandwidth utilization while minimizing delays.

The proposed approach involves several key steps. First, the network is divided into multiple stationary clusters. A new distributed CH selection mechanism is then introduced, which selects CHs based on specific parameters. The research also proposes an Enhanced BAT Algorithm (EBA) to optimize the CH selection process. Finally, a novel routing strategy is developed to identify the most suitable candidate CH based on the destination's location and the positions of neighboring CHs.

Simulation results validate the effectiveness of the CBCCR protocol, demonstrating significant improvements in bandwidth utilization, increased throughput, and reduced transmission delays. This innovative protocol provides an efficient solution to the challenges of routing in highly dynamic VANET environments.

II. LITERATURE REVIEW

The most appropriate communication method for delivering safety-related messages in VANET is a routing-based protocol. Various current VANET routing techniques have been developed to improve performance. Abuashour et al. introduced protocols such as Cluster-Based Lifetime Routing (CBLTR), Intersection Dynamic VANET Routing (IDVR), and a control overhead reduction method (CORA). The CBLTR protocol enhances route stability and average throughput in bidirectional segment scenarios by selecting cluster heads (CHs) based on vehicle lifespan. The IDVR protocol improves route stability and reduces end-to-end delay in grid architecture, leveraging software-defined networking to select intersection CHs and determine optimal routes. The CORA algorithm optimizes control overhead messages among cluster members and CHs to reduce message traffic. These protocols, tested using SUMO and MATLAB simulations, performed better than many existing methods in terms of throughput and overhead reduction.

Luo et al. developed the Cluster-Based Routing (CBR) protocol, which reduces average routing overhead and end-to-end delay jitter as vehicle density increases. It meets real-time application requirements by ensuring steady data transmission delays. Taleb et al. proposed the Velocity-Heading Based Routing Protocol (VHRP), implemented over the DSDV protocol, which improves routing performance by reducing link breakages and enhancing end-to-end throughput through directional awareness. Muniyandi et al. introduced Rectangle-Aided Location-Aided Routing (RALAR), utilizing a moving rectangular zone based on node mobility. RALAR, further optimized with Genetic Algorithms (GA), outperformed existing protocols like KALAR and LAR in Packet Delivery Ratio (PDR), end-to-end delay, routing overhead, and energy usage.

Rana et al. proposed the Mobility Aware Zone-Based Ant Colony Optimization Routing (MAZACORNET) protocol, a hybrid multipath ACO-based method incorporating zones for scalability and a proactive-reactive approach to reduce broadcasting and congestion. MAZACORNET demonstrated improved delivery ratio and scalability in dense networks. Zhang et al. developed a Micro Artificial Bee Colony (MABC) algorithm for multicast routing, optimizing QoS requirements like network lifetime and delay cost. Hernafi et al. introduced an intelligent clustering model using bio-inspired mobility measurements to maintain stable inter-vehicular communication paths and reroute traffic effectively, resulting in reduced fuel consumption and emissions.

Moridi et al. presented the Reliable Multi-Level Routing System (RMRPTS), which uses fuzzy logic and Tabu search to ensure robust clustering and path maintenance in dynamic topologies. The protocol demonstrated higher PDR, reduced delay, and fewer packet losses compared to traditional methods. Fatemidokht et al. proposed a fuzzy logic-enhanced ACO model, which achieved superior data packet delivery and reduced end-to-end delay. Mohammed Nasr et al. introduced a novel clustering and Cluster Head Election (CHE) method suitable for desert environments, ensuring efficient and reliable communication. Malathi et al. proposed a clustering approach based on vehicle distance and velocity, enhancing cluster stability through a super cluster-head selection process.

The discussed methods employ diverse strategies for CH selection, considering factors like vehicle ID, speed, and direction. Reactive routing protocols generally reduce control overhead compared to

proactive protocols, but CBR-based methods still face challenges with control overhead. Overall, the research focuses on improving route stability, reducing end-to-end delay, and minimizing control overhead in VANETs for various topology scenarios.

III. PROPOSED RESEARCH METHODOLOGY

Introduced Cluster-Based Congestion Control Routing (CBCCR) protocol seeks to decrease number of re-election processes for fresh CHs by eliminating the route discovery procedure. In a bidirectional segment situation, the CBCCR protocol focus to improve route stability and effective bandwidth usage while reducing delay. The following procedures are taken in this research work: initially, the segment is split into several stationary clusters. After which, to choose a CH depending on specified factors, a unique distributed CH election method is presented. Finally, depending on CH's neighbors with destination location, novel routing technique is suggested to choose the most suited candidate CH. The clustering approach, which transforms a large network into tiny grouped networks termed as clusters, is the most common option for reducing control overhead messages. For every cluster, one among cluster members (CMs) has to be selected as the Cluster Head, who is accountable for all local cluster communication (CH). The Improved Control Overhead Message Reduction Algorithm (ICORA) will considerably minimize control overhead since it limits communication between every CM and CH rather than sharing control overhead messages amongst entire CMs inside the cluster.

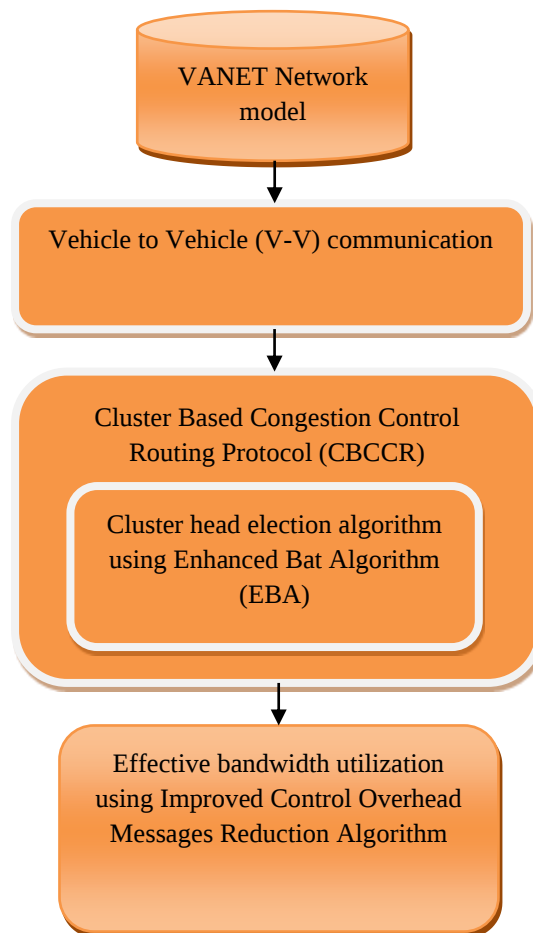


Figure 1. The overall process of the proposed methodology

3.1. Diving of Cluster

segment is a bidirectional road; every segment is separates as numerous clusters that equal half of a normal vehicle's transmission range. consider that entire vehicles has a predetermined understanding of cluster coordination and identification in this scenario. At each unit of time, each vehicle must be assigned to single cluster depending on its position, with exclusive ID for every vehicle and cluster. Figure 2 depicts a section containing 2 clusters, as well as the cluster edges that connect

them. When a vehicle reaches any cluster zone (reaches cluster edge lines among clusters) at any point in time, it forms a member of that cluster as well as should transmit a HELLO message to cluster's Cluster Head.

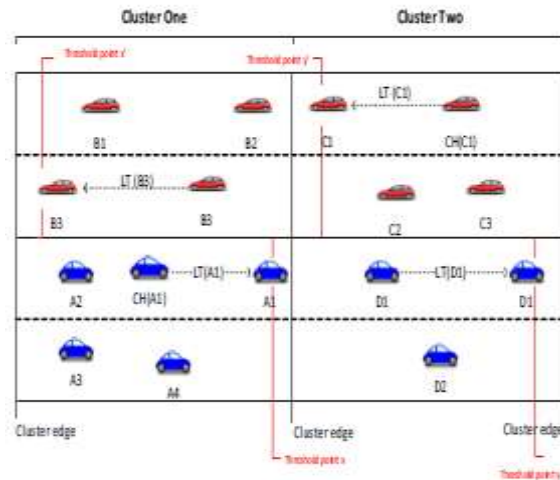


Figure 2. Segment with two clusters

3.2. Election of Cluster Head (CH)

Every vehicle entering a predefined stationary cluster zone periodically calculates a value known as the Life-Time Cost (LT). This LT is determined based on the vehicle's current velocity and its distance to the directional edge of the cluster, calculated using the Euclidean distance formula. The vehicle with the highest LT is selected as the Cluster Head (CH) and retains this role until it reaches a designated directional threshold point. During this period, no additional elections are conducted.

The directional threshold point is a location within the cluster, positioned at a specific distance from the cluster's directional edge. This distance is calculated based on the current velocity of the CH and the time required to complete the re-election process. The gap between the directional threshold point and the cluster's directional edge must be sufficient to ensure a seamless handover of the CH role to another vehicle, avoiding any loss of communication. This setup ensures that vehicles in the cluster can complete the re-election process without interruptions.

The threshold distance (D_{th}) is dynamically calculated, factoring in the current velocity of the CH, such that the handover time—comprising the re-election period and the time required to transmit CH information to the new CH—is precisely 0.2 seconds. The formula for calculating the threshold distance for each cluster is provided and illustrated in Equation 1. $D_{th}(CID) = V_{CH}(CID) \times HOT$ (1)

here:

$D_{th}(CID)$: specific cluster's threshold distance

V_{CH} : particular cluster's CH velocity

HOT: Hand Over Time

CID: Cluster Identification

Present a basic procedure for electing CH at a particular period in Figure 2. Vehicles A1 and B3 are going in opposing direction in cluster 1 also have greatest LT in their respective directions, however LT of A1 vehicle is larger than B3 vehicle's LT, hence A1 vehicle is chosen to be cluster 1 Cluster head. In 2nd cluster, identical election procedure will take place, and since c1 vehicle's LT is larger than D1 vehicle's LT, C1 vehicle will be chosen as Cluster Head in 2nd cluster. Every chosen cluster head (A1 in addition to C2) maintains its status as Cluster Head till it reaches its respective threshold point (x in addition to y'). A new election process shall begin when any Cluster Head reaches its associated threshold point. Every vehicle that joins a cluster turn out to be a member of that cluster, which classifies the vehicles in real time based on their location. Then, at any given moment, the LT is computed for entire vehicles inside their linked clusters. LT is computed depending on how long every vehicle will

stay in the cluster, which is mostly determined by the distance to the cluster's forthcoming directional edge, and also velocity. vehicle with highest LT at a given period is chosen as CH and will remain until it reaches the directional threshold point. Fresh election has to be held at this period, and a fresh CH should be chosen. goal of not constantly updating the CHs is to decrease control overhead's messages created via the re-election process, or to increase LT for CH.

• **Cluster Head Election using Enhanced Bat Algorithm (EBA)**

The echolocation behaviour of bats inspired the bat algorithm. Bats will lessen loudness as well as raise the produced ultrasonic sound's frequency while chasing preys [23]. The BA was created using these real-life bat features. The following are the basic steps of BA as explained mathematically. Each bat in the BA has 3 vectors: frequency, velocity, location vector, which are formed as (2), (3), and (4) at time step t :

$$V_i(t+1) = V_i(t) + (X_i(t) - Gbest)F_i \quad (2)$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \quad (3)$$

here $Gbest$ indicates greatest position acquired to this point and F_i indicates frequency of i th bat that is revised as:

$$F_i = F_{min} + (F_{max} - F_{min})\beta, \quad (4)$$

here β is in between $[0, 1]$ is an arbitrary vector drawn from a uniform distribution. by (3) and (4), it is clear that varied frequencies encourage the ability of exploration of bats to best result.

All the equations, to some degree, can assure the BA ability of exploitation. nevertheless, to carry out better intensification, an arbitrary walk process is also used as:

$$X_{new} = X_{old} + \varepsilon At, \quad (5)$$

here X_{old} means one solution chosen arbitrarily between present best solutions, ε is a arbitrarily chosen value between $[-1, 1]$, A specifies entire bats' average loudness at current period. Here rand is a arbitrary value uniformly distributed between $[0, 1]$. For a level, BA can be measured as a balanced mixture of global and intensive local search. pulse emission rate (r) and loudness (A) manage the balancing between 2 search approaches. When A increased, artificial bats tries to do a diversification instead of intensification. The parameters are revised as:

$$A_i(t+1) = \alpha A_i(t) \quad (6)$$

$$r_i(t+1) = r_i(0)[1 - \exp(-\gamma t)], \quad (7)$$

here α and γ are constants, α has similar sense of cooling factor. For guarantying that artificial bats are moving toward best solutions, loudness along with emission rate are modified while good results are identified. However, formula's first as well as second item impact the approach as a result that it does global and local search, in that order. first item of (4) might decrease convergence rate quickly as well as second item of (4) might produce premature convergence issue. For resolving this issue, Fuzzy systems is effectively similar as original BA.

Here, a fuzzy system is used to improve the bat algorithm with the goal of dynamically setting parts of the algorithm's parameters. objective is to improve the algorithm's performance in optimization situations by evaluating it against other metaheuristics using benchmark mathematical functions. Similar work may be found in domain of fuzzy logic for adjusting parameters in metaheuristics: Changing the parameters of the bat algorithm, which include wavelength 3, loudness (volume) A_0 , low frequency and high frequency, is usually done by trial and error. The current work presents an operation of a fuzzy system that is accountable for dynamically adjusting any of these parameters to increase the algorithm's performance and achieve higher effectiveness. After making the changes to the bat algorithm, experiments were run with standard mathematical functions to see how successful it was.

As it is more frequent in this form of fuzzy control, the fuzzy system presented is of the Mamdani type, and defuzzification method was the centroid. In inputs and outputs, membership functions are triangular. In addition, the triangle membership functions were selected depending on previous experience with this form of fuzzy control. There are nine rules in the fuzzy system. EBA differs from

the original Bat method in that it uses a new fuzzy controller to dynamically control explorations as well as exploitations' degree through creating scaling factor (SF) values while developing new candidate solutions by mutation on current candidate solutions. In function optimization, phrases 'exploitation' and 'exploration' were employed casually, with a general meaning as follows. The mutation success rate and scaling factor, that serve as input and output, correspondingly, of suggested fuzzy controller, are also formally defined here.

- **Exploitation:** generate a new candidate solution x_i' which is in or around neighborhood of the parent solution x_i .
- **Exploration:** generate a new candidate solution x_i' which is NOT in neighborhood of the parent candidate solution x_i .
- **Mutation Success Rate (MSR):** A mutation is termed as successful when mutated candidate solution x_i' is higher than parent solution x_i , $f(x_i') < f(x_i)$. MSR is calculated by dividing number of successful mutations by total mutations in present generation. total mutations is equivalent to population size because each candidate solution is alone modified one time each generation.
- **Scaling Factor (SF):** A scalar value generated by suggested fuzzy controller as well as utilized in eq. (7) to generate new candidate solutions during mutation of current candidate solutions.

$$x_i^{t+1} = x_i^t + v_i^{t+1} * SF^{t+1} \quad (8)$$

The suggested EBA enhancement is based on the creation and incorporating unique fuzzy controller with Bat method. The reader must have a fundamental knowledge of fuzzy logic, fuzzy set theory, and related words to achieve comprehend the proposal. following equation (8), that uses SF values provided by fuzzy controller, substitutes original mutation procedure in EBA. The scaling factor value for next period (t+1) created by fuzzy controller after observing the current period's MSR is represented by the SF^{t+1} term in (8). Eq. (8) can either enable added explorations (with $SF^{t+1} > 1.0$) or added exploitations (while $SF^{t+1} < 1.0$) or have no important explorative or exploitative impact on mutation step size (with $SF^{t+1} \approx 1.0$). The MSR is the input to suggested fuzzy controller. EBA calculates current MSR at every period t and provides it to fuzzy controller as an input. MSR is determined by dividing the number of better candidate solutions by total current solutions (that is, number of good x_i t+1 data, divided by size of the population). EBA uses a fuzzy controller method, which acquire current MSR as input and outputs an SF (scaling factor) data, which EBA utilizes in (8) to generate new results by mutating the present candidate solutions.

Pulse emission rate is also progressively adjusted by EBA depending on present level of exploration and exploitation. It is accomplished by multiplying r_i by SF^t current value after each $m = 50$ generations to gradually alter the pulse emission rate for explorations or exploitations. Simultaneously, based on the current values of MSR and SF^t , the step size value, which is kept individually for every candidate solution, is either increased/decreased. EBA maintains of different step sizes for different candidate solutions x_i , which helps it adjust the number of exploitations and explorations for each candidate solution.

3.3. Segment Routing Procedure

The CBLTR protocol is designed to distribute packets within a segment via selected Cluster Heads (CHs). Each CH maintains its own routing table, which stores the Cluster Head Identification (CHID) along with their respective locations, Life-Time Cost (LT), and expiry times. The contents of the CH routing table are illustrated in Figure 3 and include the following fields: CHID, location, LT, and expiry time. The expiry time ensures that the routing table remains up-to-date.

When a packet reaches a local CH, the CH consults its routing table to identify candidate CHs near the destination, regardless of their direction. The packet is then forwarded to the candidate CH with the highest LT. If two candidate CHs have identical LT values, the CH moving in the same direction as the local CH is selected.

If no suitable relaying CHs are available near the destination, the local CH initiates a store-and-forward recovery process. In this process, the packet is temporarily stored in a designated buffer while the CH continues to move until it identifies another relaying CH for packet forwarding.

Routing Table			
CHID	Location	LT	Expiry time

Figure 3. CH Routing table

When the vehicle obtains a packet at any point during the simulation, it initially verifies its Cluster Head routing table, after that chooses the Cluster Heads which are nearest to destination from other table is termed as the candidate Cluster Head table (that has similar structure as Cluster Head routing table in Figure 3). As next forwarder vehicle, the CH with the highest LT in the candidate CH table is chosen. If the LT data are equivalent, the one nearest to the destination, irrespective of direction, is chosen. Ultimately, if Cluster Head routing table is empty, present cluster head will do a store-and-forward operation.

3.4. Improved Control Overhead Messages Reduction Algorithm (ICORA)

Suggest a better version of the CORA (ECORA) method in this part. Remove CMHELLO message's periodical update from CORA [24] algorithm since its CMHELLO message is only created three times: when CM enters and departs cluster zone, when CH reaches at the preset threshold point. As a result, number of CMHELLO messages is mostly independent of period. CMHELLO message numbers was considerably decreased due to CORA. CH, conversely, continues to rely on time to broadcast CHADS messages. The CH should update its status on a regular basis to keep any new vehicles in the cluster informed.

As a result, the CHLT divided by CH update period (τ secs) equals CHADS messages number created by CH. CORA method mostly solves the issue of often forwarded CMHELLO messages, although CHADS messages are still time-dependent. here, I suggest an ICORA. The major goal of ICORA is to decrease CHADS messages number. For addressing the issue, developed a CHADS prediction method which allows CHs to transmit CHADS during a particular time period rather than over the entire CHLT. As a result, every CH should predict the time it will take for each CM to enter its cluster zone. primary difference between VANET and MANET is that vehicles travel in a predictable and constrained mobility pattern. That is, if we recognize a vehicle's starting position and speed, then can forecast where it will be in the future. In other words, if the CH can anticipate the period when any Cluster Member would reach its cluster zone, it can only transmit CHADS message during that time. Explaining a new CHADS prediction method, mathematical computation of control overhead messages, cluster head transmit period forecast in the subsections that follow.

- **CHADS prediction method**

When a Cluster Head gets a CMHELLO message (tagged with LEAVE) from one of its CMs, it indicates that CMID connected to the message plans to leave cluster zone within CMLT. CH then notify its Cluster Head neighbours of expected arrival time of new CM/CMs in its cluster zone. adjacent CHs compute a period of time to transmit CHADS message once they receive the CH notification. When CM does not obtain CHADS message within τ secs of entering cluster zone, the CMHELLO messages are transmitted after seconds. Suggest a CHADS prediction method in ICORA that allows Cluster Head to forecast period for transmitting CHADS messages rather than broadcasting throughout this CHLT.

The fundamental principle of the CHADS prediction algorithm is explained in phases in Figure 4. initially, Showed a picture of a clustered highway setting. Suppose that these clusters are placed in the centre of the highway topology, rather than at the highway edges (in the borders). Cluster Heads of clusters ID $n-1$, n , and $n+1$, correspondingly, are vehicles within A2, B5, B8 circles. remainder of the vehicles are Cluster Members as well. vehicles within the rectangle are Cluster Members on their way out of their clusters' zone. when any of Cluster Head notices a new CMHELLO message from Cluster Members planning to LEAVE their cluster zones, it must notify the neighbouring CHs. This notification

alerts the CHs in the area of new vehicles arriving within a certain time frame. CH that gets these notifications has to prepare for new vehicle arrivals within specified time frame. When A2 along with B8's Cluster Heads obtain CMHELLO messages from A3 as well as B6 vehicles, correspondingly, every CH sends a notification message to its neighbour CHs (like B5 vehicle in the instance) so that they may prepare for any of new arriving vehicles. B5 begins sending CHADS messages inside the specified set period of time once it obtains any notifying message from any of adjacent Cluster Heads, etc. We may optimize CHADS number to be proportionate to notification messages number obtained by Cluster Head rather than $CHLT \times \tau$ as this approach only transmits the CHADS message when cluster head gets any of notifying message from cluster head's neighbour. Figure 2 shows transition diagram for CH, that includes four modes: sleep, active, notification, and broadcast. The following is an explanation of each mode: Initially, there's CH sleep mode, which is used by all other modes when no CMHELLO messages are received. The CH remains idle in this mode, and no CHADS messages are transmitted.

Algorithm 1: Pseudo-code for CHADS prediction

```

1: for t = anytime to t <= simulation time do
2:   CH mode = sleep
3:   if CH obtains LEAVE message then
4:     CH mode = notification
5:     CH forward notification message
6:     [CM, time]
7:   end if
8:   if CM obtains CH notification message then
9:     CH mode = broadcast
10:    transmit CHADS at [time, time +  $\tau$ ]
11:  end if
12: end for

```

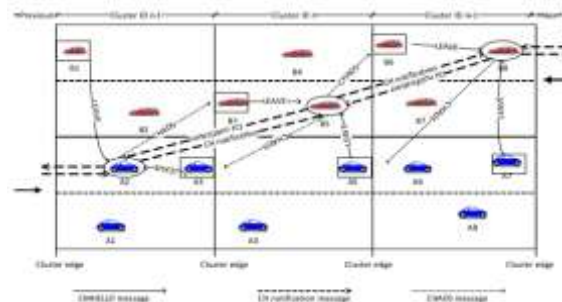


Figure 4. Snapshot of clustered highway scenario

The Cluster Head (CH) operates in different modes depending on the received messages and cluster conditions. In the first scenario, the CH switches to active mode upon receiving a CMHELLO message. In the second scenario, it transitions to broadcast mode when it receives a notification message from any nearby Cluster Head. The CH enters active mode under three conditions: when it receives a CMHELLO message indicating the arrival of a new vehicle, it remains active for a specified time period (t); if no additional messages are received, it reverts to sleep mode. If the CH receives a CMHELLO message tagged with LEAVE, it switches to notification mode to notify neighboring Cluster Heads of the vehicle's departure. In the third case, when the CH receives a notification from another Cluster Head, it transitions to broadcast mode to disseminate information.

In notification mode, the CH compiles and shares a list of Cluster Members (CMs) departing its zone, including their IDs and departure times, with neighboring Cluster Heads. In broadcast mode, the CH broadcasts Cluster Head Advertisement Messages (CHADS) for a set duration (τ seconds) before returning to sleep mode, its default silent state. If the CH receives a CMHELLO message tagged with LEAVE, it switches to notification mode, forwarding the necessary messages to its neighbors.

The number of CHADS messages broadcast by a CH is proportional to the number of vehicles leaving the cluster zone within a specific period. As vehicles continuously move, those entering the cluster eventually depart, making the total number of CHADS messages dependent on the incoming and

outgoing traffic, as described in Algorithm 2. The ECORA method calculates control overhead based on the number of CHADS messages broadcast, while the ICORA method enhances this by considering both the CHADS messages proportional to the number of vehicles in the cluster and the notification messages, which are equal to the number of Cluster Members at any given time. This ensures efficient control message handling within the cluster.

$$Total\ Adsmessag_k = Total\ CMHELLO_k + Total\ CHAds_k + Total\ CH\ notification_k \quad (9)$$

• CH broadcast time prediction

On highways, vehicles go at predetermined maximum and minimum speeds. Imagine that the majority of vehicles are travelling at a steady speed and staying within predetermined range limitations. highway is separated into stationary cluster zones in highway scenario. Every cluster zone is half the length of a standard vehicle's transmission range. As a result, any CH may interact with its neighbours at any moment without experiencing a communication failure. Only in the event that no vehicles are present within cluster zone have communication failure.

A significant issue with clustered topology is the frequent broadcasting of status messages by Cluster Heads (CHs), which leads to a substantial amount of control overhead. To mitigate this, a Cluster Head Broadcast Time Prediction Mechanism can be introduced, enabling CHs to predict the time it will take for new Cluster Members (CMs) to arrive. With this mechanism, CHs broadcast their status only during the predicted time frame, significantly reducing the number of Cluster Head Advertisement Messages (CHADS) transmitted.

In highway scenarios, most vehicles travel at constant speeds. When a vehicle is about to leave a cluster zone or enter at its designated threshold point, it sends a CMHELLO message. The CH uses this message to estimate the predicted departure time of the CM from its cluster zone. Additionally, the CH notifies neighboring Cluster Heads about the upcoming CMs and their estimated arrival times in the respective clusters. This approach assumes that all vehicles in the clustered topology have synchronized clocks, ensuring precise timing for communication.

The threshold point is a critical safety parameter for vehicles in a cluster zone. It represents the point at which a vehicle must leave the cluster zone without losing communication with the CH. At this point, the vehicle must send a CMHELLO message to inform the CH that it is departing the cluster. The time taken for a vehicle to send the CMHELLO message is denoted as T. The threshold point for vehicles entering the cluster zone can be computed using Equation 10, ensuring seamless communication and efficient clustering.

$$D_{ith} = V_i \times T \quad (10)$$

here:

Dith : Absolute distance of ith vehicle among its threshold point and cluster edge.

Vi : ith vehicle Velocity

T: Time to send CMHELLO message to Cluster Head.

absolute distance among threshold point and vehicle i's cluster edge is provided through |Dith|, its velocity is indicated as Vi. vehicle's

Algorithm 2 : CHADS Message Overall Number

```

1: overall CH ads = 0
2: for i = 1 to Max number of CMs do
3:   for CID = 1 to Max number of clusters do
4:     if CM location = CID location then
5:       overall CH ads = total of CH ads + 1
6:     else
7:       overall CH ads = overall CH ads
8:     end if
9:   end for
10: end for
11: return total of CH ads

```

lifespan to go away from present cluster and go into next is forecasted to be as follows:

$$\text{Leave time}_i = \text{current time}_i + \frac{|D_{ith}|}{V_i} \quad (11)$$

When a vehicle reaches the predetermined threshold, it must transmit a CMHELLO message to CH. This notification should provide the expected vehicle departure time. while CH gets message, it sends a notification to neighbours' CHs, notifying them of the arrival of new CMs. The CMID and the predicted arrival time to neighbour cluster zones (derived in Equation 11) are included in the notification message. An example numerical result of determining the threshold distance for any vehicle while considering vehicle velocity and time for delivering CMHELLO message is presented.

IV. RESULTS AND DISCUSSION

The empirical analysis and discussion of the suggested CBCCR technique are discussed in this section. This section gives details on how to evaluate the findings. In NS-2, the experiment is examined, and a simulation with 50 and 100 nodes is created for examination. The network's delay, energy, and throughput were among the measures used in the research. The network energy refers to the energy left in the nodes after transmission has ended, and it should be set to a maximum value to extend the network's lifetime. The total data rates transferred through the network in a given period is the network's throughput, and the delay is the period it takes for the information to be sent. The most efficient approach delivers the maximum energy and throughput while causing minimum delay.

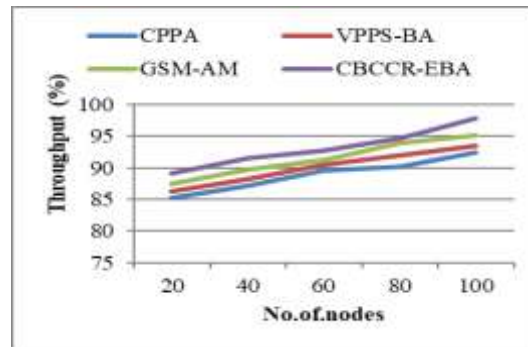


Figure 5. Throughput Comparison among suggested CBCCR-EBA and existing methods

In the Figure 5 illustrates throughput of suggested CBCCR-EBA is better performance than the existing methods. The proposed CBCCR-EBA produce higher throughput. It concludes that suggested technique produces higher throughput when compared to current routing protocol approaches.

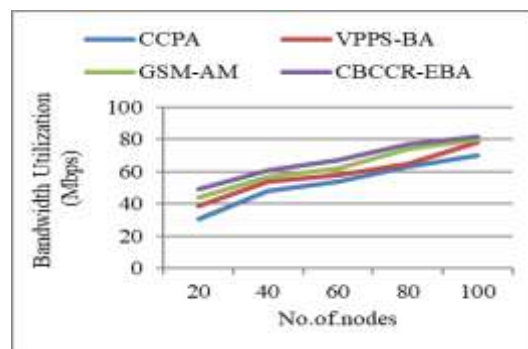


Figure 6. bandwidth utilization contrast between the proposed CBCCR-EBA and existing methods

In the Figure 6. shows the Consumed Energy of the proposed CBCCR-EBA is better performance than the existing methods. The proposed CBCCR-EBA consume less energy. It concludes that suggested approach consumes less energy when compared to present routing protocols.

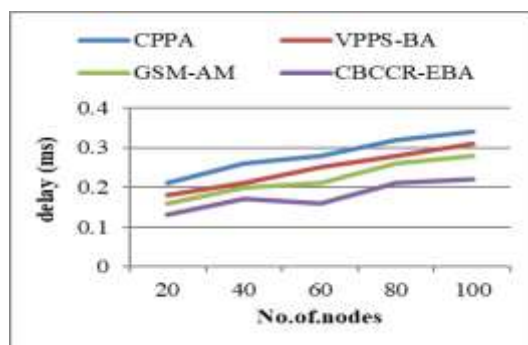


Figure 7. Comparison of delay between the proposed CBCCR-EBA and existing methods

In the Figure 7. illustrates delay comparison among proposed CBCCR-EBA is better performance than the existing methods. The proposed CBCCR-EBA has less delay. It concludes that suggested approach gives less delay when compared to present routing protocols.

V. CONCLUSION

This study presented a unique approach for improving CBR protocol performance in any VANET environment. This work's objective is to offer a broadcast method that destructively decreases rebroadcasting vehicles' number, resulting in bandwidth saving. In a segment topology, a novel Cluster-Based Congestion Control Routing (CBCCR) protocol is first proposed. The CHs are chosen depending on their highest LT, re-election procedure is alone necessary when CHs arrive at their respective threshold point. CBCCR protocol indicates a substantial increase in average throughput based on simulation findings. A new technique for selecting new CHs has been added to the CBCCR protocol. the chosen CHs have a longer LT span, procedure is highly stable. Then, the Enhanced Bat Algorithm is presented for the cluster head election process, which is implemented to enhance network efficiency and decrease end-to-end delay. Finally, a novel technique for computing best time for updating or interchanging control messages among CMs and Cluster Head was presented, with goal of reducing control overhead messages in clusters. The HELLO messages are propagated by CORA in 3 scenarios: first, when CM get into cluster zone, then when CM leaves cluster zone, finally, when a new Cluster Head declares itself. CORA considerably reduced HELLO messages' number in every cluster as well as in segment with many clusters generally, based on simulation findings. In the future, a research on a machine learning-based multi-path routing method will be conducted, which will provide a competitive advantage over existing routing measures.

REFERENCES

- [1]. Hussain, R., Hussain, F., & Zeadally, S. (2019). Integration of VANET and 5G Security: A review of design and implementation issues. *Future Generation Computer Systems*, 101, 843-864.
- [2]. Kumar, R., & Dave, M. (2012). A review of various vanet data dissemination protocols. *International Journal of u-and e-Service, Science and Technology*, 5(3), 27-44.
- [3]. Singh, G. D., Tomar, R., Sastry, H. G., & Prateek, M. (2018). A review on VANET routing protocols and wireless standards. In *Smart computing and informatics* (pp. 329-340). Springer, Singapore.
- [4]. Agarwal, P. (2017). Technical review on different applications, challenges and security in VANET. *Journal of Multimedia Technology & Recent Advancements*, 4(3), 21-30.
- [5]. Bayad, K., Bourhim, E. H., Rziza, M., & Oumsis, M. (2016, May). Comparative study of topology-based routing protocols in vehicular ad hoc network using IEEE802. 11p. In *2016 International Conference on Electrical and Information Technologies (ICEIT)* (pp. 526-530). IEEE.

-
- [6]. Erritali, M., & El Ouahidi, B. (2013, July). Performance evaluation of ad hoc routing protocols in VANETs. In *Third International Symposium on Automatic Amazigh processing* (Vol. 3, No. 2, pp. 33-40).
 - [7]. Cha, S. H., & Lee, K. W. (2011, September). Mobility information and road topology based inter-vehicle routing protocol in urban. In *International Conference on Hybrid Information Technology* (pp. 271-277). Springer, Berlin, Heidelberg.
 - [8]. Wang, T., Liang, Y., Mei, Y., Arif, M., & Zhu, C. (2018). High-accuracy localization for indoor group users based on extended Kalman filter. *International Journal of Distributed Sensor Networks*, 14(11), 1550147718812722.
 - [9]. Hezam, M. A., Junaid, A., Syed, A. A., Nazri, M., Warip, M., Fazira, K. N., ... & Nurul Hidayah, R. (2018). Classification of security attacks in VANET: A review of requirements and perspectives.
 - [10]. Moniruzzaman, A. B. M., & Rahman, M. (2014). Analysis of topology based routing protocols for Vehicular Ad-hoc network (VANET). *arXiv preprint arXiv:1411.7662*.
 - [11]. Chai, R., Yang, B., Li, L., Sun, X., & Chen, Q. (2013, October). Clustering-based data transmission algorithms for VANET. In *2013 international conference on wireless communications and signal processing* (pp. 1-6). IEEE.
 - [12]. Abuashour, A., & Kadoch, M. (2017). Performance improvement of cluster-based routing protocol in VANET. *Ieee access*, 5, 15354-15371.
 - [13]. Luo, Y., Zhang, W., & Hu, Y. (2010, April). A new cluster based routing protocol for VANET. In *2010 second international conference on networks security, wireless communications and trusted computing* (Vol. 1, pp. 176-180). IEEE.
 - [14]. Taleb, T., Ochi, M., Jamalipour, A., Kato, N., & Nemoto, Y. (2006, April). An efficient vehicle-heading based routing protocol for VANET networks. In *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006*. (Vol. 4, pp. 2199-2204). IEEE.
 - [15]. Muniyandi, R. C., Qamar, F., & Jasim, A. N. (2020). Genetic optimized location aided routing protocol for VANET based on rectangular estimation of position. *Applied Sciences*, 10(17), 5759.
 - [16]. Rana, H., Thulasiraman, P., & Thulasiram, R. K. (2013, June). MAZACORNET: Mobility aware zone based ant colony optimization routing for VANET. In *2013 IEEE congress on evolutionary computation* (pp. 2948-2955). IEEE.
 - [17]. Zhang, X., Zhang, X., & Gu, C. (2017). A micro-artificial bee colony based multicast routing in vehicular ad hoc networks. *Ad Hoc Networks*, 58, 213-221.
 - [18]. Hernafi, Y., Ahmed, M. B., & Bouhorma, M. (2017). ACO and PSO algorithms for developing a new communication model for VANET applications in smart cities. *Wireless Personal Communications*, 96(2), 2039-2075.
 - [19]. Moridi, E., & Barati, H. (2017). RMRPTS: a reliable multi-level routing protocol with tabu search in VANET. *Telecommunication Systems*, 65(1), 127-137.
 - [20]. Fatemidokht, H., & Rafsanjani, M. K. (2018). F-Ant: an effective routing protocol for ant colony optimization based on fuzzy logic in vehicular ad hoc networks. *Neural Computing and Applications*, 29(11), 1127-1137.
 - [21]. Mohammed Nasr, M. M., Abdelgader, A. M. S., Wang, Z. G., & Shen, L. F. (2016). VANET clustering based routing protocol suitable for deserts. *Sensors*, 16(4), 478.

- [22]. Malathi, A., & Sreenath, N. (2017). An efficient clustering algorithm for VANET. *International Journal of Applied Engineering Research*, 12(9), 2000-2005.
- [23]. Yang, X. S. (2011). Bat algorithm for multi-objective optimisation. *International Journal of Bio-Inspired Computation*, 3(5), 267-274.
- [24]. Abuashour, A., & Kadoch, M. (2018). Control overhead reduction in cluster-based VANET routing protocol. In *Ad hoc networks* (pp. 106-115). Springer, Cham.

Author Biography



G. Karthikeyan was born on 13/01/1982 in Salem. He has completed his UG Degree B.Sc. (Computer Science) in Kongu Arts & Science College, Erode in 2002, completed his PG Degree MCA in Kongu Arts & Science College, Erode in 2005, Completed his M.Phil Degree in Periyar University in 2008. He is pursuing his PhD in computer science in Bharathiar University. Presently he is working as Assistant Professor in Department of Computer Science, PSG College of Arts and Science, Coimbatore. His research interests include Networking and VANET. He has presented and published many papers.



Dr. S. Rizwaana was born on 08/09/1979 in Erode. She has completed her UG Degree B.Sc. (Computer Science) in 2000. She completed her PG Degree MCA in 2003. She has completed her M.Phil degree in Periyar University in year 2007 and Ph.D. in Computer Science from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India in the year 2014. Currently she is working as Assistant Professor and Head in Department of Computer Science, Erode Arts and Science college, Erode. Her area of interest include Data Structures, Computer Applications, Operating Systems, Computer Networks.