Link stability - based optimal routing path for efficient data communication in MANET

Renisha Pulinchuvallil Salim [a,1,*], Rajesh Ramachandran [b,2]

[a] Research Scholar, Bharathiyar University. Assistant Professor, Providence College of Engineering, India
[b] Associate Professor CHRIST (Deemed to be university) Bangalore,India
[1] reni82@gmail.com; [2] rajeshchristuniv@outlook.com
* corresponding author

ARTICLE INFO

Article history
Received April 18, 2024
Revised May 13, 2024
Accepted May 17, 2024
Available online August 31, 2024

Keywords
Mobile Ad hoc network (MANET)
Optimal routing path (OP)
C - K Means
Pearson correlation coefficient based SWIFFT (PCC-SWIFFT)
Digital signature algorithm (DSA)
Entropy-based gannet optimization algorithm (E-GOA)
Data communication (DC) and link stability (LS)

ABSTRACT

The paper delves into the complexities of Mobile Ad hoc Networks (MANETs), which consist of a diverse array of wireless nodes. In such networks, routing packets poses a significant challenge due to their dynamic nature. Despite the variety of techniques available for optimizing routing in MANETs, persistent issues like packet loss, routing overhead, and End-to-End Delay (EED) remain prevalent. In response to these challenges, the paper proposes a novel approach for efficient Data Communication (DC) by introducing a Link Stability (LS)-based optimal routing path. This approach leverages several advanced techniques, including Pearson Correlation Coefficient SWIFFT (PCC-SWIFFT), Galois-based Digital Signature Algorithm (G-DSA), and Entropy-based Gannet Optimization Algorithm (E-GOA). The proposed methodology involves a systematic process. Initially, the nodes in the MANET are initialized to establish the network infrastructure. Subsequently, the Canberra-based K Means (C-K Means) algorithm is employed to identify Neighboring Nodes (NNs), which are pivotal for creating communication links within the network. To ensure secure communication, secret keys (SK) are generated for both the Sender Node (SN) and the Receiver Node (RN) using Galois Theory. Following this, PCC-SWIFFT methodologies are utilized to generate hash codes, serving as unique identifiers for data packets or routing information. Signatures are created and verified at the SN and RN using the G-DSA. Verified nodes are subsequently added to the routing entry table, facilitating the establishment of multiple paths within the network. The Optimal Path (OP) is selected using the E-GOA, considering factors such as link stability and network congestion. Finally, Data Communication (DC) is initiated, continuously monitoring LS to ensure optimal routing performance. Comparative analysis with existing methodologies demonstrates the superior performance of the proposed model. In summary, the proposed approach offers a comprehensive solution to enhance routing efficiency in MANETs by addressing critical issues and leveraging advanced algorithms for key generation, signature verification, and path optimization.

This is an open access article under the CC–BY-SA license.

1. Introduction

Mobile Ad hoc Networks (MANETs) have emerged as a dynamic and rapidly advancing field in wireless communication, also referred to as the wireless mobile multi-hop or mobile packet radio network [1], [2]. MANETs consist of Mobile Nodes (MNs) interconnected via wireless connections, forming an independent system [3], [4]. These nodes move freely in any direction at random speeds,

leading to a constantly changing network topology. Each node operates within its own transmission range and communicates with others via radio frequencies [5], often necessitating communication through multiple hops due to the restricted transmission range.

In a mobile area network (MANNET), routing is the process of creating efficient routes from a source to a destination using a number of mobile intermediate nodes [6]. However, node mobility, scarce wireless resources, variability of node transmission power, and wireless channel loss characteristics make it difficult to achieve efficient routing [7]. To address these challenges and discover optimal routes, effective Routing Protocols (RPs) are necessary [8]. Reactive and Proactive protocols are two main classes of RPs. Reactive protocols like Ad hoc On-demand Distance Vector (AODV), Multicast AODV (MAODV), Dynamic Source Routing (DSR), and Ant Colony Optimization (ACO) initiate route discovery only when necessary [9]. Routing information is kept current by proactive protocols like Destination Sequenced Distance Vector (DSDV) and Optimised Link State Routing (OLSR) via recurring exchanges [10], [11].

However, maintaining valid routes is challenging due to node failures and the dynamic nature of the radio channel, which can render established routes invalid [12]. Therefore, gauging link quality for route selection is essential to achieve greater performance in terms of Quality of Service (QoS) values [13]. Approaches such as the QoS-Support Measure (QoS-SM) and Fuzzy Logic have been employed to develop optimal RPs that consider link quality alongside other metrics [14].

Moreover, MANETs are vulnerable to various threats, including attacks from external adversaries or node misbehavior within the network [15], [16]. Misbehaving nodes can disrupt routing by delaying or dropping packets, deviating from the original routing. This can lead to an increase in traffic during Data Transmission (DT), affecting MANET performance. To address this issue, novel solutions such as the G-DSA digital signature-based hashing algorithm and the E-GOA optimal route selection algorithm have been proposed to ensure data integrity and secure route selection.

In conclusion, ongoing research in MANETs continues to explore innovative approaches to improve routing efficiency, QoS, and security, leveraging newer technologies and methodologies to address the evolving challenges in wireless mobile communication networks.

1.1. Problem Statement

There are still certain limitations that are required to be improved even though many route selection algorithms are constructed for the optimal RPs in MANETs. The following are a few of the limitations discovered in the existing works.

- Generally, by utilizing the distance betwixt the Mobile Nodes, Link Stability is gauged. However, it is unsuccessful in choosing the optimal node selection for Data Transmisiion devoid of the consideration of mobility, speed, along with the direction of mobility.
- At times, removing malicious nodes could enter the other clustering group as a new entry.
- End to End Delay, routing overhead, along with packet loss takes place in the existing system owing to the single path routing.
- In most cases, the approaches are not suitable since signal stability, can be affected by a few of the environmental conditions, which is the existing method's drawback.

On scrutinizing these problems, designing an optimal Reactive Protocols by considering Link Stability, is the aim of the proposed system.

The proposed model's structure is organized as follows; a few works associated with the proposed approach are examined in section 2. The steps entailed in the proposed routing model are explicated in section 3. The experimental outcomes in contrast to the prevailing methods are scrutinized in section 4. The paper with suggestions for future enhancement is deduced in section 5.

2. Related Works

Sirmollo & Bitew (2021) introduce the Mobility Aware Routing Algorithm (MARA) for MANETs [17], emphasizing node mobility and energy considerations to reduce link breakage. While MARA outperforms existing protocols in minimizing link breakage, it neglects factors like Link Quality (LQ), leading to a decline in Network Lifetime (NL). In contrast, the LS-based optimal routing path proposed in the first paper integrates techniques like PCC-SWIFFT, G-DSA, and E-GOA to address various routing challenges including packet loss, routing overhead, and EED. By combining MARA's mobility awareness with the LS-based approach, a more comprehensive and efficient routing solution for MANETs could be achieved, considering node mobility, energy, and link stability.

Tran et al. (2021) propose the Deep Q-Network (DQN) [18] Design for QoS Multicast Routing [19] (DQMR) protocol for Cognitive Radio MANETs (CR-MANETs), aiming to establish Effective QoS Multicast (EQM) trees. Although DQMR demonstrates superiority over MAODV in routing delay, control overhead, and Packet Delivery Ratio (PDR), it suffers from the drawback of relying on repeated node visits for route optimization, thus diminishing its time reliability. In contrast, the approach outlined, focuses on addressing various routing challenges in MANETs through a Link Stability (LS)-based optimal routing path, leveraging advanced techniques like PCC-SWIFFT, G-DSA, and E-GOA. Despite the success in QoS metrics, its time-reliability remains questionable. Integrating the QoS optimization aspects of DQMR with the robust routing framework proposed could potentially offer a more comprehensive solution for MANETs, addressing both efficient routing and QoS requirements.

Chen et al. (2020) propose the TA-AOMDV routing protocol [20] for stable path selection in MANETs, considering factors like residual energy, available bandwidth, queue length, and link stability. While their protocol shows improvements in PDR and TP, it remains vulnerable to attacks during Dynamic Topology (DT). Conversely, it presents a comprehensive approach to MANET routing, leveraging LS-based optimal routing with advanced techniques like PCC-SWIFFT, G-DSA, and E-GOA. The approach here emphasizes stability and performance, it lacks robustness against attacks. Integrating TA-AOMDV's stable path selection with the security-focused framework in the previous study, could enhance both stability and security in MANETs

Velusamy et al. (2021) propose a node disjoint Multipath Routing (MPR) technique for MANETs [21] that prioritizes energy efficiency and Network Lifetime (NL) improvement. While the outcomes show promising results with reduced energy consumption and enhanced Neteork Lifetime, the approach encounters challenges due to increased routing time for establishing multiple node-disjoint paths. The paper focuses on addressing various MANET routing challenges by introducing a Link Stability (LS)-based optimal routing path. It integrates advanced techniques like PCC-SWIFFT, G-DSA, and E-GOA to optimize routing paths based on link stability and network congestion. The study, emphasizes energy efficiency and Network Lifetime enhancement, and it could benefit from incorporating the comprehensive routing framework. By integrating the energy-efficient aspects of MPR with the robust

routing framework proposed earlier, a more balanced and efficient solution for MANETs could be achieved, considering both energy consumption and routing performance.

It is found that this comprehensive approach to address the challenges of routing optimization in MANETs by introducing a novel Link Stability (LS)-based optimal routing path. It employs advanced techniques such as Pearson Correlation Coefficient SWIFFT (PCC-SWIFFT), Galois-based Digital Signature Algorithm (G-DSA), and Entropy-based Gannet Optimization Algorithm (E-GOA) to enhance routing efficiency and security. In contrast, [17] proposes a mobility-aware quality enhanced cluster-centered routing protocol for MANETs, utilizing the Improved Animal Migration Optimization (IAMO) algorithm for clustering and Improved Ant Colony Optimization (IACO) algorithm for routing path computation. While both papers contribute to the field, the new proposal provides a more comprehensive solution by addressing critical issues like packet loss, routing overhead, and End-to-End Delay (EED) and offering a systematic methodology with empirical validation, whereas [17] lacks detailed discussion on security aspects and empirical evaluation, limiting its applicability in real-world scenarios.

[22] discusses a routing selection policy for Mobile Ad-Hoc Networks (MANETs) that incorporates a trust-based mechanism within the Ad-hoc On-Demand Distance Vector (AODV) routing protocol. Trust-based mechanisms aim to enhance network security and reliability by considering the trustworthiness of nodes in routing decisions.

A comprehensive solution to enhance routing efficiency in MANETs by introducing a novel Link Stability (LS)-based optimal routing path and leveraging advanced techniques such as PCC-SWIFFT and E-GOA. In contrast, [23] recommends a hybridized version of the DSR protocol, MET-MFODSR, utilizing Minimum Execution Time scheduling and Moth Flame Optimization (MET-MFO) for route optimization. It is aimed to improve routing mechanisms, which offers a more systematic approach with detailed methodologies for key generation, signature verification, and path optimization. However [23], while demonstrating improved performance in simulation findings, faces criticism for the deficiency of population diversity in MET-MFO, potentially impacting the source routing process and overall efficiency.

[24] discuss the use of Artificial Neural Networks (ANN) for optimizing energy consumption and routing in Mobile Ad Hoc Networks (MANETs). ANN is a machine learning technique inspired by the biological neural networks of the human brain and is known for its ability to learn complex patterns and make predictions based on data.

3. Method

3.1. Proposed Link Stability-Based Efficient Data Communication System In Manet

By using PCC-SWIFFT, G-DSA, and E-GOA techniques, LS-based efficient DC in MANET is proposed in this work. By utilizing PCC-SWIFFT techniques, which are wielded for the node's authentication, the hash code was generated in the proposed system. Then, the signature was created by using G-DSA, which was wielded for signature creation on the sender side and signature verification on the receiver side. Lastly, since E-GOA possesses a pre-eminent processing capability for huge-dimensional issues, it was wielded to select the optimal routing path. Fig. 1 portrays the proposed technique's block diagram.

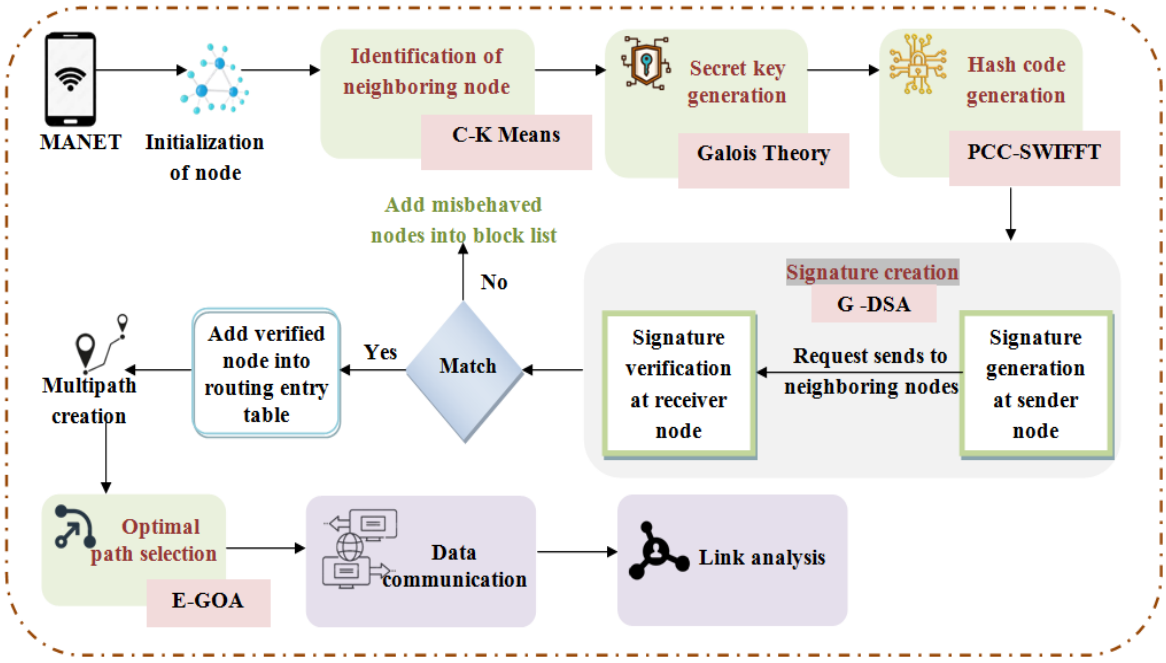

Fig. 1. Block diagram of the proposed method

MANET Initialization of node, this is the starting point where a mobile node is initialized within the MANET. Identification of neighboring node, using the C-K Means algorithm, the node identifies its neighboring nodes. Secret key generation, the node generates a secret key based on Galois Theory. Hash code generation, a hash code is generated using the PCC-SWIFFT method. Signature creation, the node creates a digital signature using the G-DSA (presumably a variant of the Digital Signature Algorithm). Signature generation at sender node, the sender node generates a signature to be sent along with the data. Request sends to neighboring nodes, the sender node sends a request or data to neighboring nodes, which includes the signature. Signature verification at receiver node, the receiver node verifies the signature to ensure the data's integrity and authenticity. Match, there is a decision point where if the signature matches (is verified), the process moves to adding the verified node into the routing entry table. If not, the misbehaved nodes are added to a block list. Add verified node into the routing entry table, verified nodes are added to the routing table for future communication. Multipath creation, multiple paths are created for data transmission to provide redundancy and reliability. Optimal path selection, using the E-GOA algorithm, the optimal path for data transmission is selected. Data communication, data is communicated over the selected optimal path. Link analysis, the link is analyzed, which could be for performance, security, or reliability purposes

3.1.1. Initialization of Node

A set of nodes is encompassed in the MANET. In the network, the entire node behaves as a router to transmit data. At first, the nodes in the MANET are initialized randomly for efficient data communication in this work. It is mathematically expressed as,

$$N = N_1, N_2, N_3, \dots, N_n \tag{1}$$

Where, the nodes in the MANET are expressed by N and the n^{th} number of the node is expressed by N_n.

Algorithm 1: Initialization of Node Algorithm

Input: Number of nodes in MANET (N)

Output: Initialized nodes for efficient data communication

Begin

for each node i from 1 to N:

Initialize node i randomly

return Initialized nodes

End

3.1.2. Identification of Neighboring Node

The neighboring nodes that are connected to a certain distance from the particular node were identified by utilizing the C-K Means algorithm after initialization [25]. An unsupervised learning algorithm that groups nodes together into a cluster centered on the distance betwixt nodes is termed K means algorithm [26]. There is a difficulty in clustering data in the existing K-means algorithm, that is, clusters are of varying sizes and densities [27]. So, to calculate the node's distance, a Camberra distance technique is proposed, which ameliorates the model's performance. Therefore, the proposed scheme is named C-K Means. The following are the steps of C-K Means.

Systematically evaluating different values of K using a combination of quantitative metrics, visual inspection, and validation, you can determine the best. K value for the C-K Means algorithm that optimally partitions your data into meaningful clusters.

- Let K be the number of clusters to be formed. Initially, the cluster center (C_N) is randomly initialized. It is articulated as,

  $$C_N = [C_{N1,}C_{N2,}C_{N3,\ldots,}C_{Na} \quad (2)$$

  Where, the a^{th} number of the cluster center is expressed by C_{Na}

- The distance of nodes is calculated from these cluster centers and it is analogized to the neighbor nodes. Here, to calculate the distance of nodes, camberra distance is wielded. N represents the neighbor node. Thus, the distance (D_N) is computed as,

  $$D_N = \sum \frac{C_N - N}{C_N + N} \quad (3)$$

- The nodes with the least difference in the distance are clustered together after comparing the distance among nodes. The centroids' position is recalculated when all nodes are assigned. Until there is no alteration in the centroid, the steps for all iterations are repeated. Finally, the NNs (A) are identified, which is given by,

  $$A = [A_1, A_{2,}A_{3,\ldots,}A_o] \quad (4)$$

  Where, the o^{th} number of the NN is articulated by A_o.

Next, before the request from the SNs was sent to the neighbour nodes, processes like SK generation, hash code generation, and signature formation were completed for efficient transmission.

Algorithm 2: Identification of Neighboring Node Algorithm

Input: Initialized nodes, Number of clusters (K)

Output: Neighboring nodes identified using C-K Means algorithm

Begin

Initialize cluster centers randomly:

for each cluster center j from 1 to K:

Initialize cluster center j randomly

Calculate distance of nodes from cluster centers using Canberra distance:

for each node i:

for each cluster center j:

Calculate distance using $D_N = \sum \frac{C_N - N}{C_N + N}$

Cluster nodes based on least distance:

Repeat until convergence:

Assign nodes to clusters based on least distance

Recalculate centroids

Identify Neighboring Nodes (NNs):

for each node i:

Identify NNs based on clusters

return Neighboring Nodes (NNs)

End

3.1.3. Secret Key Generation

To facilitate maximum security, both the SN and the RN have to generate an SK in a MANET during communication [28]. Thus, by combining the public keys of the SN and RNs, SK is obtained utilizing Galois Theory. Since Galois Theory's arithmetic [29] properties allow it to be wielded for the encryption along with decryption of information, it is utilized for cryptography.

By deploying the DSA algorithm, the public key of the SNs along with RNs is generated. By performing point multiplication with the base point (W), the public key (P_{key}) is computed. It is symbolized by,

$$P_{key} = W * \rho \tag{5}$$

Where, the private key, which is generated randomly, is expressed by ρ. Therefore the SK ($\overline{S_{key}}$) obtained utilizing Galois Theory is given by,

$$S_N(S_{key}) = \sqrt{S_N(p_{key})} + \sqrt{R_N(p_{key})} \quad (6)$$

$$R_N(S_{key}) = \sqrt{S_N(p_{key})} + \sqrt{R_N(p_{key})} \quad (7)$$

$$(\overline{S_{key}}) = \{S_N(S_{key}), R_N(S_{key})\} \quad (8)$$

Where, the SN is represented by S_N, the RN is expressed by R_N, the SK obtained in the SN is articulated by $S_N(S_{key})$, the SK obtained in the RN is symbolized by $R_N(S_{key})$, and the public key is denoted by (p_{key}). The same values of the SK are exhibited by both the SN and RN.

3.1.4. Hash Code Generation

Deploying the SKs of the SN following the production of the SKs produced the hash code. The process of transforming any given key into another value is termed Hash code generation [30]. Here, by employing PCC-SWIFFT techniques, the hash code is generated. A compilation of provably secure hash functions is called SWIFFT. It is centered on the notion of FFT. Owing to memory requirements, the existing SWIFFT hashing algorithm is computationally intractable. Hence, PCC techniques are proposed to calculate a linear combination to conquer this problem. Thus, the proposed model is named PCC-SWIFFT. The steps of PCC-SWIFFT are elucidated below,

- Step 1: At first, the SK $(\overline{S_{key}})$ is fed into the PCC-SWIFFT technique. Afterward, it is processed by multiplying the $c-th$ row by$\varepsilon^{c-1}, c = 1,2,3, \dots x$,. Here, the appropriate fixed element is symbolized by ε and the x^{th} number of rows is represented by x.
- Step 2: Next, on each column (*d = 1,2,3,...y*), the FFT is computed, which is simple to invert as well as is carried out to accomplish diffusion. It is given by,

$$(\beta_{1,d}, \dots, \beta_{x,d}) = FFT(\varepsilon^0 \overline{S_{key_{1,d}}}, \dots, \varepsilon^{x-1} \overline{S_{key_{x,d}}} \quad (9)$$

  Where, the FFT values are denoted by β and the y^{th} number of the column is indicated by *y*.

- Step 3: Then, a linear combination is computed across each row. Since it compresses the input, it is performed to achieve confusion. Here, to generate a hash code by using the PCC, a linear combination is computed. PCC is the most common way of measuring a linear combination. To gauge the strength along with the direction of the relationship betwixt '2' variables, PCC is wielded. Finally, the hash code $S_N(\delta)$ is obtained by,

$$S_N(\delta) = \frac{\sum(\beta_{c,d}-\bar{\beta})(\lambda_{c,d}-\bar{\lambda})}{\sqrt{\sum(\beta_{c,d}-\bar{\beta})^2 \sum(\lambda_{c,d}-\bar{\lambda})^2}} \quad (10)$$

Where, the co-efficient function is denoted by λ.

3.1.5. Signature Creation

The signature was created by utilizing G-DSA to validate the authenticity as well as the integrity of the data in the SN after the generation of the hash code . DSA relies on the algebraic characteristics of discrete logarithmic functions and modular exponentiations to create a digital signature consisting of "2" 160-bit numbers obtained from the message digests along with private keys. The private key generator is compromised with a random value in the existing DSA. Therefore, the SK that is derived using (7)

will be incorporated in the encryption and decryption process in order to reduce the complexity of the model. As a result, G-DSA is the proposed model's name. Fig. 2 portrays the architecture of G-DSA.

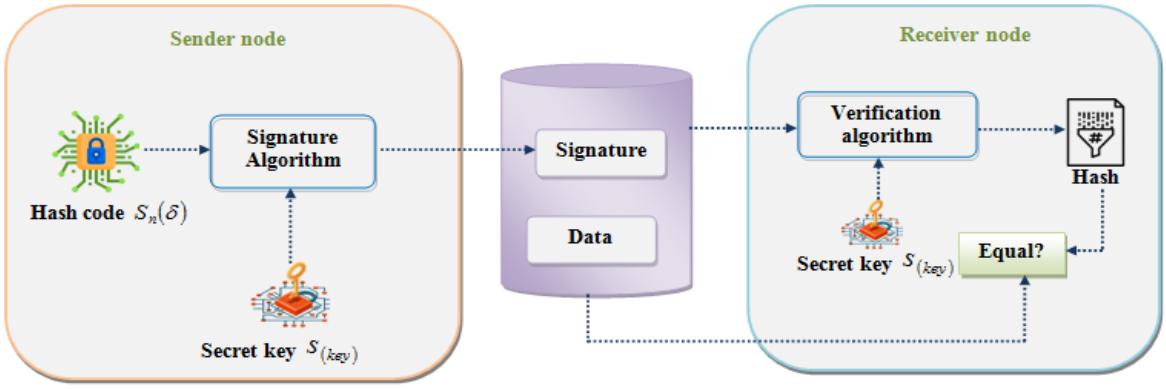

Fig. 2. Architecture of G-DSA

Signature creation and signature verification are the '2' phases. It is elucidated below:

- **Signature generation**: At first, $\{u, v\}$ is computed to generate the message signature. Two integer values are u and v in a digital signature. By wielding the random number (r) and the base point (W), the value for is computed. It is specified as,

$$u = (r * W \ mod \ q) mod \ b \tag{11}$$

Where, the prime numbers are q, b. A new random number (r) should be chosen and the value of u is computed again if the value of u is zero. Then, to obtain the value of v, , $S_N(\delta)$, r and $\overline{S_{key}}$ are considered. It is computed as,

$$v = \left(r^{-1} S_N(\delta) + \overline{S_{key}} + u\right) mod \ b \tag{12}$$

- **Signature verification**: Once the RNs received the request from the SN, validation of the signature was done. At first, to verify the message signature, ω is computed, which is the modular multiplicative inverse of v mod b.

$$\omega = v^{-1} mod \ b \tag{13}$$

For digital signal verification, the mathematical formula is given by,

$$V = \left(\left(h_1 * W + h_2 - \overline{S_{key}}\right)\right) mod \ b \tag{14}$$

Where, the variables are h_1 and h_2 . It is articulated as,

$$h_1 = S_N(\delta) * \omega \ mod \ b \tag{15}$$

$$h_2 = u * \omega \ mod \ b \tag{16}$$

The condition to check the digital signature's validity is given as,

$$\begin{cases} V = u, valid \\ V \neq u, invalid \end{cases} \tag{17}$$

The received message is accepted by the RN if the signature verification is validated successfully.

Algorithm 3: Galois-based Digital Signature Algorithm

Input: hash code $S_N(\delta)$ and secret key $(\overline{S_{ky}})$

Output: Digital signature verification

Begin

Initialize hash code $S_N(\delta)$ and secret key $(\overline{S_{key}})$

For all training steps do \\ Signature creation

Compute u and v using (10) and (11)

Obtain digital signature $\{u, v\}$ \\ Signature verification

Compute ω

$$\omega = v^{-1} mod\ b$$

Verify signature

$$V = ((h_1 * W + h_2 - \overline{S_{key}})) \text{ mod b}$$

If $(V = u)$

Valid signature

Else

Invalid signature

End If

End For

End

The RN decrypts the message together with generates the hash code once the receiver accepts the request from the SN. The node is verified and it will be appended to the routing entry table if the hash code generated by the RN and the SN match each other or else it will be appended to a blocked list as a misbehaved node.

3.1.6. Optimal Path Selection using E-GOA

From the routing entry table, multiple paths are established as of the source to the destination node. After that, by utilizing the E-GOA approach, the OP is chosen from . A meta-heuristic optimization algorithm that is inspired by the gannet's predation behavior is GOA. GOA encompasses '3' phases namely: 1) Initialization phase; 2) Exploration phase; 3) Exploitation phase.

The gannet's flock will construct a straight line or a semi-circular array to catch the prey if it discovers prey in this algorithm. (i) U-shaped dive mode, (ii) V-shaped dive mode, (iii) abrupt turning, along with (iv) random wandering are the '4' diverse behaviors taken by it for predation. Devoid of consuming too much energy, the prey is easily caught by the gannet centered on this behavior. Centered on the average position of individuals, the population's position will be calculated, which might not accurately reflect the most recent trends in the existing GOA. Hence, the entropy technique is proposed to calculate the population's position, which ameliorates the model's performance to deal with this problem. Therefore, the proposed system is named the E-GOA approach. The '3' phases of E-GOA are explained below,

3.1.6.1. Initialization Phase

At first, gannets are initialized as (multipath). At this initialization phase, the gannet's position is initialized as,

$$L_M = \begin{bmatrix} l_{1,1} & \cdots & l_{1,e} & \cdots & l_{1,\kappa-1} & l_{1,\kappa} \\ l_{2,1} & \cdots & l_{2,e} & \cdots & l_{2,\kappa-1} & l_{2,\kappa} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & l_{e,f} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ l_{P-1,1} & \cdots & l_{P-1,e} & \cdots & l_{P-1,\kappa-1} & l_{P-1,\kappa} \\ l_{P,1} & \cdots & l_{P,e} & \cdots & l_{P,\kappa-1} & l_{P,\kappa} \end{bmatrix} \quad (18)$$

Where, the position of individual is denoted by $\mathrm{l_e}$. Each $l_{e,\mathrm{f}}$ in the matrix is calculated by using the below equation,

$$l_{e,f} = \emptyset_1 \times (\mathrm{up_f} - \mathrm{lb_f}) + \mathrm{lb} \ \text{ where, } \mathrm{e} = 1,2,3,\ldots,\mathrm{P} \text{ and } \mathrm{f} = 1,2,3,\ldots,\mathrm{k} \quad (19)$$

Where, the upper and lower bounds are $\mathrm{up_f}$ and $\mathrm{lb_f}$, the random number in the range (0, 1) is articulated by $\emptyset_1$, the number of individuals in the populace is , and the dimension size is symbolized by k. The values of the L matrix are assigned to the memory matrix ($\mathrm{L_{M(m)}}$) by the initialization phase. The change in the gannet individual's position will be recorded in $\mathrm{L_{M(m)}}$ and the fitness value is calculated for every possible solution in every iteration of the assessment. Here, centered on the low energy with high TP, the fitness value is computed. After fitness evaluation, $\mathrm{L_{M(m)}}$ is wielded if the memory matrix individual executes superior to the individual of the present solution, or else the solutions in the $\mathrm{L_M}$ matrix continue to be wielded.

3.1.6.2. Exploration Phase

Gannets begin to probe for prey in the exploration phase. They modify their dive pattern in line with the prey dive's depth once they discover the prey. U-shaped dive mode and V-shaped dive mode (μ) are the dive pattern. It is mathematically articulated as.

$$\mathrm{v} = 2 * \cos(2 * \pi * \phi_2) * \mathrm{T} \quad (20)$$

$$\pi = 2 * \tau(2 * \pi * \phi_2) * \mathrm{T} \quad (21)$$

$$\text{Where, } \mathrm{T} = 1 - \frac{\mathrm{T_z}}{\mathrm{T_{max}}} \quad (22)$$

$$\tau = \begin{cases} -\frac{1}{\pi} * \mathrm{l} + 1, \mathrm{l} \in (0, \pi) \\ \frac{1}{\pi}\mathrm{l} - 1, \mathrm{l} \in (\pi, 2\pi) \end{cases} \quad (23)$$

Where, the iteration is symbolized by T, the number of current iterations is denoted by $\mathrm{T_z}$ and the maximum number of iterations is articulated by $\mathrm{T_{max}}$. The gannet's position has been updated by utilizing these '2' strategies. It is mathematically represented as,

$$\mathrm{L_{M(mz)}}(\mathrm{T}+1) = \begin{cases} \mathrm{L_{M(z)}}(\mathrm{T}) + \theta_1 + \theta_2, & \sigma \geq 0.5 \\ L_{M(z)}(T) + \vartheta_1 + \vartheta_2, & \sigma \geq 0.5 \end{cases} \quad (24)$$

Where, the gannet's updated position is $L_{M(mz)}(T+1)$, the random number that is randomly wielded to choose two dive strategies is expressed by σ , and random numbers in the range $(-v, v)$ and are $(-\mu, \mu)$ are θ and ϑ , respectively. It is expressed as,

$$\theta = I * (L_{M(z)}(T) - L_{M(\emptyset)}(T)) \tag{25}$$

$$\vartheta = J * (L_{M(z)}(T) - L_{M(g)}(T)) \tag{26}$$

Where, the variables are I and J. It is expressed as,

$$I = (2 * \emptyset_4 - 1) * v \tag{27}$$

$$J = (2 * \emptyset_5 - 1) * \mu \tag{28}$$

Where, the arbitrarily chosen individual in the current populace is represented by $L_{M(\emptyset)}(T)$ and the individual's position in the present population is $L_{M(g)}(T)$. To gauge the population's position, the entropy technique will be wielded. It is symbolized by,

$$L_{M(g)}(T) = -\sum_{z=1}^{P} L_{M(z)}(T) \log L_{M(z)}(T) \tag{29}$$

3.1.6.3. Exploitation Phase

It will start to capture the prey when the gannet has high capture capacity in the exploitation phase. The capture capacity (Z) is calculated by,

$$Z = \frac{1}{\left(\frac{W_{gt} * \xi^2}{0.2 + (2 - 0.2) * \phi_6}\right)} \tag{30}$$

$$T_2 = 1 + \frac{T_z}{T_{max}} \tag{31}$$

Where, the gannet's weight is represented by W_{gt} and the gannet's speed is expressed by ξ. The position is updated with an abrupt turning if the gannet's catching ability is within the gamut of catchable prey or else the gannet is incapable of catching the prey and carries out a Levy movement to seek out the next target at random. The gannet's updated position is presented by,

$$L_{M(mz)}(T+1) = \begin{cases} T * delta * \left(L_{M(z)}(T) - L_{M(best)}(T)\right) + L_{M(z)}(T), & Z \geq \varsigma \\ L_{M(best)}(T) - \left(L_{M(z)}(T) - L_{M(best)}(T)\right) * E * T, & Z < \varsigma \end{cases} \tag{32}$$

$$delta = Z * L_{M(z)}(T) - L_{M(best)}(T) \tag{33}$$

$$E = L_{evy}(K) \tag{34}$$

Where, the constant is symbolized by , the best-performing individual in the present populace is signified by $L_{M(best)}(T)$, and the Levy flight function is $L_{evy}(\blacksquare)$. It is specified by,

$$L_{evy}(K) = 0.01 \times \frac{F \times G}{|\psi|^{\frac{1}{w}}} \tag{35}$$

$$G = \left(\frac{\Gamma(1+w) \times \sin\left(\frac{\pi w}{2}\right)}{\Gamma\left(\frac{1+w}{2}\right) \times w \times 2^{\left(\frac{w-1}{2}\right)}}\right)^{\frac{1}{w}} \tag{36}$$

Where, the random values in the range (0, 1) are F and G. A pre-determined constant is $w = 1.5$.

Algorithm 4: Entropy-based Gannet Optimization Algorithm (E-GOA)

Input: multiple paths (M)

Output: optimal path (O_{path})

Begin

Initialize gannet (M) , Population size (P) and maximum number of iteration (T_{Max})

\\ Initialization Phase

Initialize position of gannet L_M

Generate memory matrix ($L_{M(m)}$)

Evaluate fitness value

While ($T_z \leq T_{Max}$)

If ($\emptyset > 0.5$) then

\\ Exploration Phase

For $L_{M(m)}$ **do**

If ($\sigma \geq 0.5$)

Update position using (23)

Else ($\sigma < 0.5$)

Update position using (23)

End If

End For

Else

\\ Exploitation Phase

For $L_{M(m)}$ **do**

If ($\varsigma \geq 0.2$)

Update position using (31)

Else ($\varsigma < 0.2$)

Update position using (31)

End If

End For

End If

End While

Return Optimal Path

End

The proposed model selects the OP (O_{path}) utilizing E-GOA from the multipath for DC in the same way as attacking the prey based on certain strategies. Lastly, by appraising the distance amongst the nodes, the DC is commenced along with the LS is scrutinized.

4. Results and Discussion

Here, the proposed system's performance is examined by comparing its outcomes with other prevailing models. The suggested method is used in the Python working platform. Performance study was conducted for the '4' sections: NN identification, hash code generation, signature production, and OP selection.

4.1. Performance Analysis of Neighboring Node Identification

Regarding NN identification time, the performance investigation of the proposed C-K Means is authenticated. After that, obtained outcomes are analogized to the prevailing models, namely (i) K Means, (ii) Partition Around Medoids (PAM), (iii) Clustering Large Application (CLARA), together with (iv) Fuzzy C-Means (FCM).

The NN identification time of the C-K Means model and the prevailing models is depicted in Fig.3. For 100 nodes, when analogized to the prevailing models, the C-K Means model identified the NN at 38457ms, which is lower, while the prevailing models took 43652ms (K Means), 48754ms (PAM), 53457ms (CLARA), and 58476ms (FCM) to recognize the NN. Likewise, when analogized to the prevailing models, the C-K Means model took a lower time to identify the NN for varying numbers of nodes. Hence, it is deduced that the C-K Means model is more efficient in the identification of NNs.

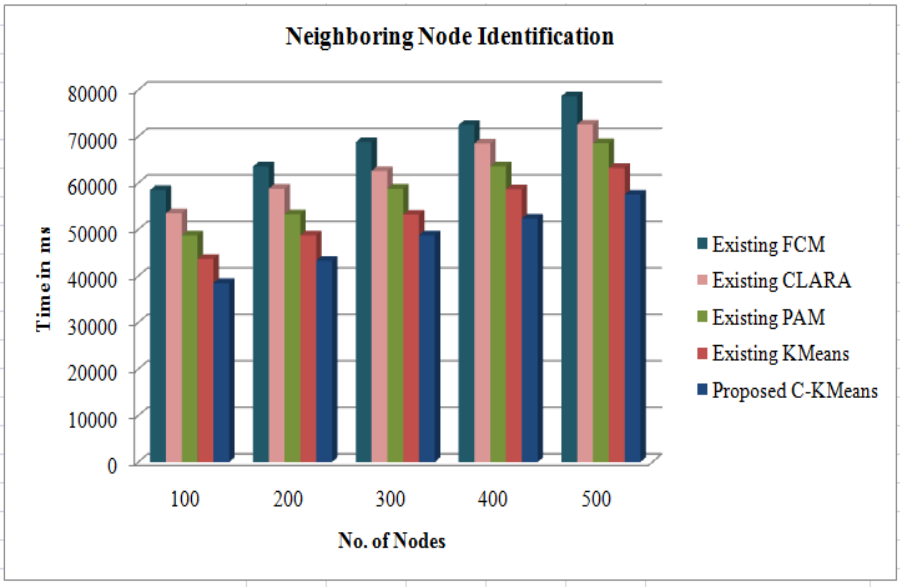

Fig. 3. Neighboring node identification time of proposed model and existing models

4.2. Performance Analysis of Hash Code Generation

Regarding hash code Generation Time (GT), the proposed PCC-SWIFFT's performance analysis is validated. The obtained outcomes are analogized to the prevailing models, namely SWIFFT, Tiger, Secure Hashing Algorithm 512 (SHA 512), and Message Digest Algorithm 5.

The performance investigation of the PCC-SWIFFT model along with the prevailing models regarding hash code GT is exhibited in Table 1. The PCC-SWIFFT system took 728ms to generate

hash code, whilst the existing models took 832ms, 874ms, 924ms, and 975ms for SWIFFT, tiger, SHA 512, and MD 5, respectively. A lower GT is displayed by the PCC-SWIFFT model when weighed against the prevailing models. Thus, it is deduced that the PCC-SWIFFT model shows superior performance in the generation of hash code.

Table 1. Hash code generation time of proposed model and existing models

Techniques	Hash code generation Time (ms)
MD5	975
SHA512	924
Tiger	874
SWIFFT	832
Proposed PCC-SWIFFT	728

4.3. Performance Analysis of Signature Creation

Regarding key GT, signature creation time, along with signature verification time, the proposed G-DSA is validated.

Fig.4 exhibits the proposed system's performance analysis. To generate the key, create the signature, and verify the signature, the time taken by the proposed model is 458ms, 386ms, and 379ms, respectively. It is inferred from these outcomes that the proposed model is highly effective in signature creation and verification.

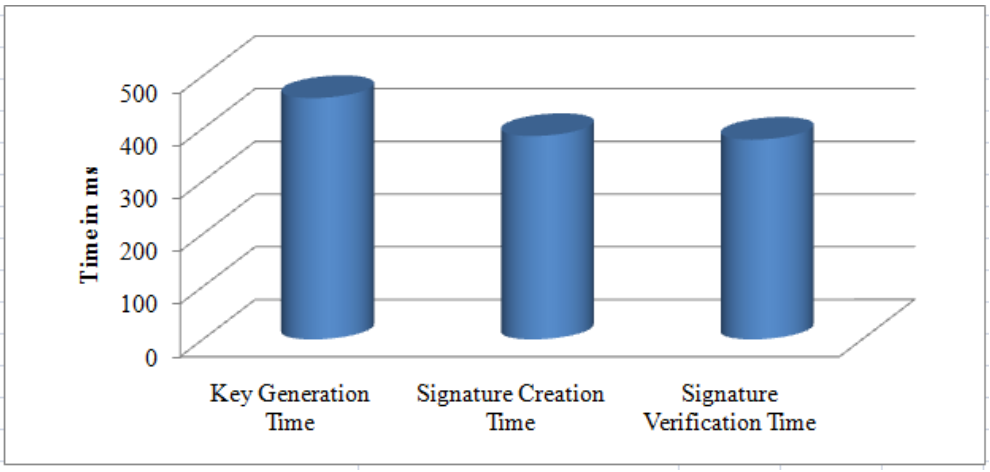

Fig. 4. Performance Analysis of the proposed model

4.4. Performance Analysis of Optimal Path Selection

Regarding TP, delay, PDR, code rate, and the total number of control packets, the performance investigation of the proposed E-GOA is confirmed. Next, the acquired outcomes are analogized to the prevailing models, namely GOA, Whale Optimization Algorithm (WOA), Lion Optimization Algorithm (LOA), and Sparrow Search Optimization (SSO).

Table 2 evinces the comparative investigation of the E-GOA model together with the prevailing models regarding TP and delay. TP is delineated as the number of data packets that are directed over the total duration of the simulation. At 5s, a TP of 402mbps is accomplished by the E-GOA model, which is higher analogized to the prevailing models, whilst a TP of 389mbps (GOA), 356mbps (WOA), 335mbps (LOA), and 312mbps (SSO) is attained by the prevailing models. Likewise, the E-GOA model

attains higher TP than the existing models for varying simulation times. Similarly, for a simulation time of 5s, a delay of 102s is attained by the E-GOA model, which is higher analogized to the existing models. The E-GOA model displays a higher delay than the existing models for diverse simulation times. Hence, it is deduced that the E-GOA model exhibits better performance than existing models

Table 2. Table 2. Comparative analysis of the proposed model and existing models in terms of throughput and delay

Metrics	Simulation time (s)	Proposed	E-GOA	GOA	WOA	LOA	SSO
Throughput (mbps)	5	402		389	356	335	312
	10	426		398	387	377	356
	15	460		412	403	392	378
	20	486		447	436	415	398
	25	510		467	444	432	401
Delay (s)	5	102		97	80	76	64
	10	267		167	135	122	88
	15	345		265	246	234	112
	20	467		365	342	312	245
	25	587		467	423	411	365

The performance investigation of the E-GOA and existing models regarding PDR is displayed in Fig 5. The ratios of the sum of the packets that are obtained by the destination to the sum of packets produced are termed PDR. When weighed against the prevailing models, namely GOA, WOA, LOA, and SSO, the E-GOA model has a higher PDR of 25 for a simulation time of 5s. A higher PDR is evinced by the E-GOA model than the prevailing models for varying simulation times. Hence, it is inferred that the E-GOA model exhibits superior performance to the prevailing systems.

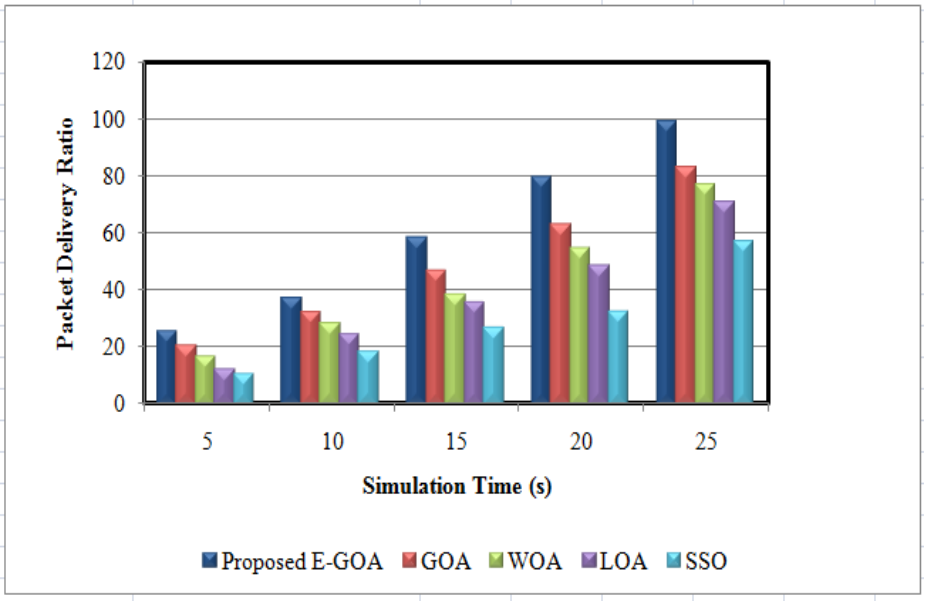

Fig. 5. Performance analysis of proposed and existing models in terms of packet delivery ratio

The comparative investigation of the E-GOA along with the prevailing models regarding code rate and the total number of control packets is depicted in Table 3. When weighed against the prevailing models, the E-GOA model attains a higher code rate of 22 for the simulation time of 4s, while the

prevailing models attain the code rate of 18, 15, 11, and 9 for GOA, WOA, LOA, and SSO, respectively. Likewise, the E-GOA model achieves a higher code rate than the existing models for varying simulation times. When analogized to the existing models, the total number of control packets obtained by the E-GOA model is 12, which is higher for a simulation time of 8s. Likewise, the total number of the E-GOA model is higher analogized to the prevailing models for varying simulation times. Hence, it is deduced from the outcomes that the E-GOA model is more efficient in selecting the OP.

Table 3. Comparative analysis of the proposed model and existing models in terms of code rate and the total number of control packets

Metrics	Simulation time (s)	Proposed E-GOA	GOA	WOA	LOA	SSO
Code rate	5	22	18	15	11	9
	10	35	28	22	19	15
	15	57	40	37	28	22
	20	78	66	54	49	36
	25	102	87	76	70	64
Total number of control packets	5	12	10	8	6	5
	10	21	18	15	12	10
	15	34	25	21	19	15
	20	56	41	32	28	24
	25	82	67	50	44	39

5. Conclusion

An LS-based efficient DC in MANET has been proposed in the work by utilizing PCC-SWIFFT, G-DSA, and E-GOA techniques. By utilizing PCC-SWIFFT and G-DSA algorithms, the hash code and the signature were created in the proposed system. Lastly, the OP was chosen by employing the E-GOA technique. Then, the experimental analysis is carried out in which the proposed model's performances are scrutinized regarding the diverse metrics. The final outcomes exhibit that for a simulation time of 5s, the hash code GT of 728ms, signature creation and GT of 386ms and 379ms, respectively, and TP of 402mbps is accomplished by the proposed system. Likewise, for all other metrics, a higher performance is exhibited by the proposed system. Thus, it is concluded from the simulation findings that the proposed system is more efficient in DC in MANET. DC is done with a little more energy efficiency and response time. The work may introduce an efficient optimal cluster head-centered clustering technique in the future to solve this issue.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. None of the authors have received any funding or grants from any institution or funding body for the research.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

References

[1] A. Srivastava, S. K. Gupta, M. Najim, N. Sahu, G. Aggarwal, and B. D. Mazumdar, "DSSAM: digitally signed secure acknowledgement method for mobile ad hoc network," *EURASIP J. Wirel. Commun. Netw.*, vol. 2021, no. 1, p. 12, Dec. 2021, doi: 10.1186/s13638-021-01894-7.

[2] P. Theerthagiri, "FUCEM: futuristic cooperation evaluation model using Markov process for evaluating node reliability and link stability in mobile ad hoc network," *Wirel. Networks*, vol. 26, no. 6, pp. 4173–4188, Aug. 2020, doi: 10.1007/s11276-020-02326-y.

[3] U. Srilakshmi, N. Veeraiah, Y. Alotaibi, S. A. Alghamdi, O. I. Khalaf, and B. V. Subbayamma, "An Improved Hybrid Secure Multipath Routing Protocol for MANET," *IEEE Access*, vol. 9, pp. 163043–163053, 2021, doi: 10.1109/ACCESS.2021.3133882.

[4] D. E. M. Ahmed and O. O. Khalifa, "An overview of MANETs: Applications, characteristics, challenges and recent issues," *Int. J. Eng. Adv. Technol.*, vol. 6, no. 4, pp. 128–133, 2017, [Online]. Available at: https://csit.ust.edu.sd/files/2018/06/D4927046417-1.pdf.

[5] P. K. Pattnaik, B. K. Panda, and M. Sain, "Design of Novel Mobility and Obstacle-Aware Algorithm for Optimal MANET Routing," *IEEE Access*, vol. 9, pp. 110648–110657, 2021, doi: 10.1109/ACCESS.2021.3101850.

[6] V. Tilwari *et al.*, "MCLMR: A Multicriteria Based Multipath Routing in the Mobile Ad Hoc Networks," *Wirel. Pers. Commun.*, vol. 112, no. 4, pp. 2461–2483, Jun. 2020, doi: 10.1007/s11277-020-07159-8.

[7] A. Alshehri, A.-H. A. Badawy, and H. Huang, "FQ-AGO: Fuzzy Logic Q-Learning Based Asymmetric Link Aware and Geographic Opportunistic Routing Scheme for MANETs," *Electronics*, vol. 9, no. 4, p. 576, Mar. 2020, doi: 10.3390/electronics9040576.

[8] S. Patel and H. Pathak, "A mathematical framework for link failure time estimation in MANETs," *Eng. Sci. Technol. an Int. J.*, vol. 25, p. 100984, Jan. 2022, doi: 10.1016/j.jestch.2021.04.003.

[9] M. B. Dsouza and D. H. Manjaiah, "Energy and Congestion Aware Simple Ant Routing Algorithm for MANET," in *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Nov. 2020, pp. 744–748, doi: 10.1109/ICECA49313.2020.9297470.

[10] B. K. Tripathy, S. K. Jena, P. Bera, and S. Das, "An Adaptive Secure and Efficient Routing Protocol for Mobile Ad Hoc Networks," *Wirel. Pers. Commun.*, vol. 114, no. 2, pp. 1339–1370, Sep. 2020, doi: 10.1007/s11277-020-07423-x.

[11] P. K. Shrivastava and L. K. Vishwamitra, "Comparative analysis of proactive and reactive routing protocols in VANET environment," *Meas. Sensors*, vol. 16, p. 100051, Aug. 2021, doi: 10.1016/j.measen.2021.100051.

[12] N. Veeraiah and B. T. Krishna, "An approach for optimal-secure multi-path routing and intrusion detection in MANET," *Evol. Intell.*, vol. 15, no. 2, pp. 1313–1327, Jun. 2022, doi: 10.1007/s12065-020-00388-7.

[13] S. Sankar Ganesh and G. Ravi, "Retracted Article: Real time link quality based route selection and transmission in industrial Manet for improved QoS," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 7, pp. 6873–6883, Jul. 2021, doi: 10.1007/s12652-020-02331-1.

[14] S. R and A. H, "Adaptive fuzzy logic inspired path longevity factor-based forecasting model reliable routing in MANETs," *Sensors Int.*, vol. 3, p. 100201, Jan. 2022, doi: 10.1016/j.sintl.2022.100201.

[15] B. U. I. Khan *et al.*, "Exploring Manet Security Aspects: Analysis Of Attacks And Node Misbehaviour Issues," *Malaysian J. Comput. Sci.*, vol. 35, no. 4, pp. 307–338, Oct. 2022, doi: 10.22452/mjcs.vol35no4.2.

[16] U. Srilakshmi, S. A. Alghamdi, V. A. Vuyyuru, N. Veeraiah, and Y. Alotaibi, "A Secure Optimization Routing Algorithm for Mobile Ad Hoc Networks," *IEEE Access*, vol. 10, pp. 14260–14269, 2022, doi: 10.1109/ACCESS.2022.3144679.

[17] K. Karthick and R. Asokan, "Mobility Aware Quality Enhanced Cluster Based Routing Protocol for Mobile Ad-Hoc Networks Using Hybrid Optimization Algorithm," *Wirel. Pers. Commun.*, vol. 119, no. 4, pp. 3063–3087, Aug. 2021, doi: 10.1007/s11277-021-08387-2.

[18] T.-N. Tran, T.-V. Nguyen, K. Shim, D. B. Da Costa, and B. An, "A New Deep Q-Network Design for QoS Multicast Routing in Cognitive Radio MANETs," *IEEE Access*, vol. 9, pp. 152841–152856, 2021, doi: 10.1109/ACCESS.2021.3126844.

[19] N. Fareena and S. Sharmila Kumari, "Retracted Article: A distributed fuzzy multicast routing protocol (DFMCRP) for maximizing the network lifetime in mobile ad-hoc networks," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 5, pp. 4967–4978, May 2021, doi: 10.1007/s12652-020-01936-w.

[20] Z. Chen, W. Zhou, S. Wu, and L. Cheng, "An Adaptive on-Demand Multipath Routing Protocol With QoS Support for High-Speed MANET," *IEEE Access*, vol. 8, pp. 44760–44773, 2020, doi: 10.1109/ACCESS.2020.2978582.

[21] B. Velusamy, K. Karunanithy, D. Sauveron, R. N. Akram, and J. Cho, "Multi-Objective Function-Based Node-Disjoint Multipath Routing for Mobile Ad Hoc Networks," *Electronics*, vol. 10, no. 15, p. 1781, Jul. 2021, doi: 10.3390/electronics10151781.

[22] V. Matre and P. A. Vikhar, "Routing Selection Policy on Mobile Ad-Hoc Network using Trust based Mechanism Through AODV Routing Protocol," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 8s, pp. 683–694, Dec. 2023. [Online]. Available at: https://ijisae.org/index.php/IJISAE/article/view/4261.

[23] S. A. Almazok and B. Bilgehan, "A novel dynamic source routing (DSR) protocol based on minimum execution time scheduling and moth flame optimization (MET-MFO)," *EURASIP J. Wirel. Commun. Netw.*, vol. 2020, no. 1, p. 219, Dec. 2020, doi: 10.1186/s13638-020-01802-5.

[24] J. Y. Hande and R. Sadiwala, "Optimization of energy consumption and routing in MANET using Artificial Neural Network," *J. Integr. Sci. Technol.*, vol. 12, no. 1, pp. 718–718, 2024. [Online]. Available at: https://pubs.thesciencein.org/journal/index.php/jist/article/view/a718.

[25] K. Kandali, L. Bennis, and H. Bennis, "A New Hybrid Routing Protocol Using a Modified K-Means Clustering Algorithm and Continuous Hopfield Network for VANET," *IEEE Access*, vol. 9, pp. 47169–47183, 2021, doi: 10.1109/ACCESS.2021.3068074.

[26] M. Usama *et al.*, "Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges," *IEEE Access*, vol. 7, pp. 65579–65615, 2019, doi: 10.1109/ACCESS.2019.2916648.

[27] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Inf. Sci. (Ny).*, vol. 622, pp. 178–210, Apr. 2023, doi: 10.1016/j.ins.2022.11.139.

[28] S. J. Ahmad, I. Unissa, M. S. Ali, and A. Kumar, "Enhanced security to MANETs using digital codes," *J. Inf. Secur. Appl.*, vol. 66, p. 103147, May 2022, doi: 10.1016/j.jisa.2022.103147.

[29] I. Stewart, "Galois Theory, Fifth Edition," *Galois Theory, Fifth Ed.*, pp. 1–351, Jan. 2022, doi: 10.1201/9781003213949-1.

[30] R. P. Salim and R. R, "A Framework for Integrating the Distributed Hash Table (DHT) with an Enhanced Bloom's Filter in Manet," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 2, pp. 440–448, 2022, doi: 10.14569/IJACSA.2022.0130252.