International Journal of Innovative Research in Sciences and Engineering Studies (IJIRSES)

www.ijirses.com *ISSN: 2583-1658 (Online) | Volume: 3 Issue: 4 | 2023*

A Mobile Ad-Hoc Network with IoT Environment Routing Protocol Based on Energy Saving Optimization Method

Devendra Singh Chauhan[1]*, Nagesh Salimath[1]

[1]*Department of Computer Science, Madhyanchal Professional University, Bhopal, India*

Abstract: Many minimum energy (energy-efficient) routing protocols have been proposed in recent years. However, very limited effort has been made in studying routing overhead, route setup time, and route maintenance issues associated with these protocols. Without a careful design, an energy-efficient routing protocol can perform much worse than a normal routing protocol. In this paper, we first show that the minimum energy routing schemes in the literature could fail without considering the routing overhead involved and node mobility. We then propose a more accurate analytical model to track the energy consumptions due to various factors, and a simple energy-efficient routing scheme PEER to improve the performance during path discovery and in mobility scenarios. Our simulation results indicate that compared to a conventional energy-efficient routing protocol, PEER protocol can reduce up to 2/3 path discovery overhead and delay, and 50 percent transmission energy consumption.

Keywords: Energy-efficient routing protool, overhead, MAC.

Introduction

Mobile computer devices with batteries that can connect wirelessly make up most wireless ad hoc networks. The battery technology lags far behind while computer devices' processing power and memory grow at a very rapid rate. In order to extend the equipment and system operating duration, it is crucial to develop energy saving strategies. The range between a transmitter and a receiver, or r, determines the rate of attenuation of the broadcast signal in wireless networks. Path loss exponent p, whose value varies based on the operating environment, has a value between 2 and 6 Jamali et al. [1]. With power control, a sender can modify the transmission power in accordance with d rather than continuously utilising the maximum transmission power. Unfortunately, from a source to destnication, link level power regulation cannot guarantee the lowest possible for end-to-end energy usage. The main difference between Maximum Network Lifetime routing techniques and Minimum energy routing techniques is that the former aims to provide every node looking for an energy efficient path with the leftover battery power in a balanced way. While the later Minimum Energy routing protocols tries to restrict energy usage in the beginning itself. According to the different types of connection costs, Minimum Energy routing methods is divided into three categories: Minimum Total TransCeiving Power (MTTCP), Minimum Total Transmission Power (MTTP) and Minimum Total Reliable Transmission Power (MTRTP) Devi et al. [2]. The transmission power is used by MTTP protocols as the link cost measure while looking for the path between the source and the destination that has the least amount of overall transmission power. In order to determine the path with the least amount of transmission power, Dijkstra's Shortest Path method has also been updated. Although it aims to minimise energy usage between any two nearby nodes, PARO uses transmission power as the connection cost. To reduce energy usage between two nodes, data packets are sent by multiple intermediate nodes. The link cost in MTTCP protocols is calculated using the transmission power plus the receiving power. Numerous studies have also employed the Bellman-Ford method to determine the minimal overall transceiving power route r Debnath et al. [3]. The energy consumption resulting from data packet retransmissions, however, was not taken into account by the MTTP and MTTCP protocols that were presented in the literature. Instead, they suggested a MTRTP protocol to account for packet retransmissions' energy expenditure. The data packet transferred from one node to another uses a certain transmission power which is helps in finding the connection cost.

The conventional shortest path routing protocols, – for example Ad hoc On Demand Distance Vector (AODV) as well as Dynamic Source Routing (DSR) protocols, are enhanced to look for the path with the lowest cost whenever a new connection cost has been determined Hu, Z et al. [4]. However, even a simple adjustment might cause a number of issues. First, a great deal of energy is used during the route finding phase due to the high routing overhead, which also results in a substantial path setup delay. Furthermore, sustaining an energy-efficient path in a mobile context is impossible with the route management strategy employed in standard shortest path routing protocol. This study suggests the Progressive Energy-Efficient Routing (PEER) protocol for quicker path establishment and more effective path upkeep Khokhlachova et al. [5]. PEER searches for the most energy-efficient approach gradually and maintains the route constantly, in contrast to standard energy-efficient routing protocols that attempt to determine the best path throughout the route discovery phase and ensure the route's flexibility. Particularly, the most energy-effecient path approach is used to build a connection between the source and the destination instantly Ray, S et al. [6]. At the same time, the path for transmission also becomes the one with with least overhead. This helps in saving energy in all the transmissions that take place. Results of the trials show that PEER may dramatically minimise routing overhead and path setup latency and spend far less time in both static and mobile settings as compared to conventional minimal energy protocols. If a node needs to locate a way to a destination, it will initiate a route discovery process in on-demand routing protocols like AODV Jayalakshmi et al. [7]. It transmits the packet containing the route request and watches for the destination's response. When nearby nodes get a route request packet, retransmission takes place and the process goes on. However, this may result in high routing overhead, therefore only the initially received route request is broadcasted again and not the others. Additionally, only the first route request packet receives a response from the destination node. For instance, in Fig. 1, S and D both have surrounding nodes called A and B, and S requires a way to reach D.

Fig 1

As a result, A and B will both get the route request packet as soon as S broadcats it. In the event that A rebroadcasts the packet, nodes S, B, and D will receive it. As they have got the same route request from S, nodes S and B in a traditional on-demand protocol will delete the rebroadcast route request packet. As a result, SAD is the last route to be found. It is clear that these on-demand routing methods have an overhead of O(n), where n is the number of network nodes. However, route finding in energy-efficient routing methods differs significantly Jesudurai et al. [8]. Given that these packets may originate from more energy-efficient pathways, the intermediary nodes could no longer simply reject the redundant route request packets. This means that if the duplicate route request packets originate from a more energy-efficient channel, the intermediate nodes must process and rebroadcast them. It may be necessary for the nodes to transmit the same route request packet again as a result. In comparison to SB, if SAB is more energy-effecient, node B may be required to broadcast both the packets from S and A for the identical example in Fig.1. Based on the Bellman-Ford method, we may determine that the current routing overhead is O(n), and that this overhead rises sharply as the number of network nodes (n) grows Rahman et al. [9]. Similar discoveries were found after our first research in, and the identified issue is known as Flooding Waves.

Energy consumption model

A cost-based energy-efficient routing protocol is called PEER. The total of all connections contained in between the source and destination node is chosen as fulfilling specified criteria for a cost-based routing system. The best route is determined by taking out a suitable link cost for energy-efficient cost-based routing protocols Hou, Y.T. et al. [10]. This is doen because of the importance of link cost for the routing protocol. In this part, we'll calculate the connection cost and demonstrate how to determine the variables needed to do so.

IEEE 802.11 Distributed Coordination Function (DCF) and Point Coordination Function both specify two MAC methods (PCF). Since PCF is a centralised protocol, this research has primarily focused on DCF at the MAC layer because it is utilised in conjunction with PEER. Carrier Sensing Multiple Access with Collision Avoidance (CSMA/CA) is the foundation of DCF Alam et al. [11]. It comprises the physical carrier sensing technique and the virtual carrier sensing scheme. Network Allocation Vector is used to implement the virtual carrier sensing technique (NAV). A node will refresh NAV with the timeframe included in any packets it receives, such as RTS, CTS, and DATA packets. The NAV value predicts the conclusion of the current transmission session. This DCF protocol has been used to construct a link cost model. Use the notation $p_{l,a,b}$, $p_{m,b,a}$, $p_{a,b}$, and $p_{n,b,a}$ to signify the packet error rates for RTS, CTS, DATA, and ACK packets sent between nodes a and b. Additionally, for a variable i indicate 1-i by i and i to represent the mean value of i. Then, in Fig. 2, where t_0 represents the beginning state, is the state diagram for transferring a data packet via node a to one of its nearby nodes, node b. Depending on whether node b properly receives the RTS packet after node a sends it, the state will either move into t_1 with probability $p_{l,a,b}$ or stay in t_0 with probability $p_{l,a,b}$. Node b will send the CTS packet if it gets the RTS packet. CTS will be received by node a with probability p m,b,a, and the state will switch from t1 to t2; The state will return to t0 with probability $p_{m,b,a}$. Node x will send the data packet after receiving the CTS packet. The data packet will be collected by node b with probability $p_{a,b}$, and the state will switch from t2 to t3; The state will eventually return to t0 with probability pa, b. Node j will recognize the data packet after it has been received Niu Z et al. [12]. ACK will be accepted by node a with probability $p_{n,b,a}$, and the state will shift from t3 to t4, where the process terminates; The state returns to t0 with probability $p_{n,b,a}$. According to the state diagram, node a typically has to send $1{=}p_{l,a,b}$ RTS packets before node b may successfully receive one (from state t0 to state t1). In a similar manner, nodes b and i must send $1{=}p_{m,b,a}$ CTS packets from state t1 to state t2, node i must send $1{=}p_{a,b}$ data packets from state t2 to state t3, and node j must send $1{=}p_{n,b,a}$ ACK packets (from state t3 to t4). As a result, during the whole operation, the average numbers of RTS, CTS, data, and ACK transmissions are as follows: RTS: $1/(p^*_{a,b}\ p^*_{n,b,a})$, CTS: $1/(p^*_{m,b,a}\ p^*_{a,b}\ p^*_{n,b,a})$, data: $1/(p^*_{a,b}\ p^*_{n,b,a})$, ACK: $1/(p_{n,b,a})$. In a power control system, RTS and CTS packets are communicated at the highest possible power level Pm to minimise hidden terminal issues, while DATA and ACK packets are delivered between nodes n and m at the lowest possible transmission power level Pa,b to save energy. The nodes fix their NAVs to the Extended InterFrame Space (EIFS) time if they can detect the signal but are unable to accurately interpret it in order to minimise collisions caused by the use of asymmetric power in control and data packet transfers. With the exception of the transmission powers (Pa,b and Pb,a) and the packet error rates, the majority of link cost model parameters may be easily retrieved (pn,a,b, pc,b,a, pa,b, and pm,b,a). We'll demonstrate how to evaluate these parameters in this section. The following presumptions are made for parameter estimate purposes: 1) The route loss among two nodes is equal in both directions; and 2) The power level of the packet received and the average interference level (such as an RTS/CTS packet) related with the MAC layer can be determined using the physical layer. Both energy-efficient routing protocols and several power control systems make use of these basic assumptions. The received power level (P_r) at the receiver is proportional to $P_t\ /d^n$, where P_t is the transmission power level, since the wireless signal attenuates at a rate of $1\ /d^n$ (where d is the distance and n is the path loss exponent).

$$K = P_t/d^n$$

where K is an environment-dependent component. Using this method, the needed transmission power for additional packets can be calculated. To implement this method, a packet can be sent by the node which depends on the packet's received power level as well as the planned receiving power. For instance, if node A recognises that node B sent packet Pe (e.g., RTS, CTS, and broadcast packets) at per bit power level P_r and also that maximum power per bit transmission (P_m) was used to send the packet. Node A can use these two formula to determine the required per bit transmission power node B would need to use to send other packets to A.

$$P_r = K \times P_m/d^n$$

$$P_r^{th} = K \times P_t(B, A)/d^n$$

P is the bare minimum required received power level. P(B, A) is simple to calculate using:

$$P^t(B, A) = P_r^{th} \times P_m / P_r$$

Pt(A, B) for a packet travelling from A to B will be the same as Pt since it is expected that route loss will be equal in both directions (B, A). In other words, node A can determine the required transmission power it needs to transmit a packet to itself in addition to the needed transmission power node B needs to utilise. The major causes of packet error are collision, interference, and noise. Here, we use the carrier sensing zone to differentiate between the concepts of collision and interference. We refer to the mistake as collision if the network of carrier sensing zones are to blame, and interference otherwise. Since each node may check the interruption and noise level when the channel is open, getting these values is simple. The bit error rate (BER), according to the received power level and modulation method, may be determined when interference and noise levels are assessed. In the absence of an error correction mechanism, the packet error rate (PER) brought on by the interference and noise can also be calculated if the BER is provided. PER is expressed as $1 - (1 - \text{BER})^L$, where L is the packet's bit count. The majority of accidents in 802.11 occur during RTS transmission. As a result, we just need to take into account the packet error rate brought on by RTS packet collision. By keeping track of busy/idle slots, authors have also proposed a straightforward method for estimating the collision risk. where C_{t-i} with i=0 … n-1 are the final n slot samples and $p_c(t)$ is the predicted collision probability at any time t. If the i^{th} slot has vacancy or if the slot receives a transmission successfully, C_i is equal to 0, otherwise C_i equals 1. As a result, the strength of interference and noise, reception power, and packet size are used to determine the packet error rates for CTS, DATA, and ACK packets Narayandas et al. [13]. While for RTS packets, we must consider the packet error rate brought on by both collision and interference. Provide the p_{int}, p_c, and RTS packet error rates owing to noise, interference, and collision. Also indicate the RTS packet error rate. Using all these, the packet error rate will be:

$$p_{r,i,j} = p_{int} + pc - p_{int} * p_c$$

PEER control

As was previously mentioned, the path setup time is rather long and there is frequently, a significant overhead introduced during path finding in the current minimal energy based routing systems. On the contrary, a routing strategy shouldn't immediately choose a random route. Neither should it be dependent on a route maintenance scheme. Doing so subsequently makes route an energy-efficient one, however this process muchmore time consuming and also results in a larger overhead Aroulanandam V et al. [14]. Moreover, doing so does not ensure that the path found is the one that saves maximum energy. Therefore, the right option is to employe a route that is close to the least energy path and uses a maintenance plan in order to alter the path to further reduce energy thus.

Route discovery process

The PEER route finding approach has been covered in this section. A shortest path routing system would be the shortest way to discover a route between two nodes. Between the source node and the destination node, there could be several shortest (shortest hops) pathways Amiri et al. [15]. Assuming, for instance, that all the intermediate nodes in Figure 3 (A, B, E, F, G, and H) are the nodes that both S and D are nearby both of them are not in the range of transmission, there are six quickest (2 hops) pathways (SAD, SBD, SED, SFD, SGD, SHD). Selecting the shortest path that uses the least amount of energy is preferred (also termed as minimum energy shortest path).

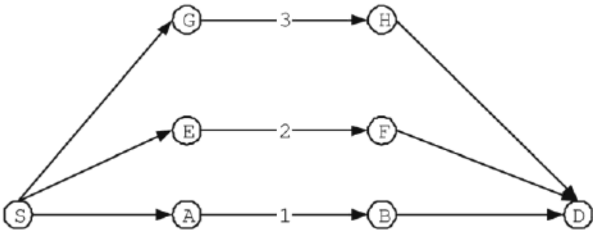

Fig 3

If you denote the set of pathways as P, the number of hops for path p as Np, and the amount of energy used by link i in path p as $E_{s,i}$, then the set of shortest paths would be Ps.

$$Ps = \arg\min(Np);\ p \in P$$

The collection of shortest pathways P_{ms} with the least amount of energy would be

$$P_{ms} = \arg\min X\ (\textstyle\sum E_{p,i}),\ p \in Ps$$

Even though there could be several minimal energy shortest paths in Pms, the routing protocol might select a particular one based on certain criteria, such as the arrival time of the route request packet Kathiroli et al. [16]. According to the definition mentioned before, the fundamental search algorithm would be: 1) Find all pathways with the fewest hops; 2) pick the shortest path(s) with the least amount of energy; (1). Two pieces of information—the hop count and the energy consumption—should be included in the route request packet in order to apply this technique. The route request packet is first broadcast by the source node with the hop count and energy consumption both set to 0. Then, it will only repeat the packet if one of the following circumstances is true:

1. The packet either originates from a shorter (fewer hops) path or the node hasn't already received one of those.

2. The packet originates from a path that has the smae number of hops as the last optimal path and also uses less energy.

The importance of these criterion are that they help in choosing the path which has the least hops and also uses the least amount of energy Maheswari et al. [17]. However, the path selection problems with this technique are comparable to those with alternative energy-efficient routing protocols. The destination node may thus get a large number of route request packets from several potential minimal energy shortest pathways. Even though the destination node has already received all of the route request packets, it may not be able to decide since it is unsure of how many route request packets it will receive. Assuming, for instance, that all six of the shortest routes (SAD, SBD, SED, SFD, SGD, and SHD) in Fig. 3 have identical energy usage and have been delivered to destination D, D could still be unable to choose the optimum path because it is unsure of the ideal time to decide. This problem can be resolved in a number of ways at the destination node. For every route request it gets, the destination may, for example, issue a route reply. This approach will waste energy since the destination might need to carry out several route reply messages Rajpoot et al. [18]. Additionally, before the optimum path is discovered, the source node may broadcast several data packets through a less energy-efficient path. Another possibility is that the destination starts a countdown after getting a packet with a routing request. It will restart the timer if a new route request comes in before the alarm sounds. If not, it will choose the best path discovered before the timeout expires and send a route reply packet back to the source. The optimal path will be chosen by the destination inside the time window if a time window is put up as the third alternative Li Y et al. [19]. The latter two techniques assist in reducing energy usage, but they could prolong the route setup process. We chose the second one in this study since it can adjust for the quantity of incoming route request packets. The destination may send back route reply packets fast to shorten the time required for route setup if just a small number of route request packets arrive at the destination. However, if multiple route request packets come in between two consecutive packets that too in inconvenient time gaps, it can result in inefficiency as the packets that come from energy-effecient ways have to wait before of the other packets Er-rouidi M et al. [20].

Route maintenance

Despite being less efficient than the least energy path, the route found during the path discovery phase may nonetheless result in a greater end-to-end energy consumption. Additionally, due to node moves and dynamic channel circumstances, the network environment might change drastically, and the formerly energy-efficient route may lose its efficiency over time. As a result, the route maintenance stage is essential for routing protocols that are energy-efficient.

The route maintenance system of PEER will not utilise multiple periodic messages since they will require more energy for signalling Vu Q.K. et al. [21]. An observing node will instead work with its neighbours to find a rather more energy-efficient path while continuously analysing data packets transferred in the area.

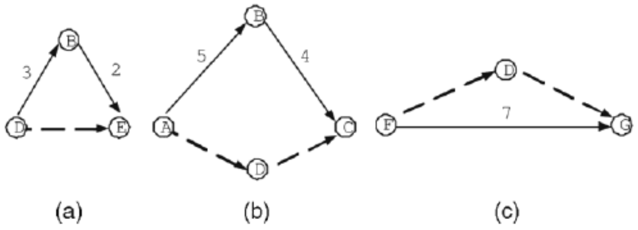

Fig 4

After receiving an RTS, CTS, or broadcast packet from a neighbouring node, each node can calculate the required transmission power as well as the link cost to that node, as explained in Section 3. In PEER, every forwarding node will add the link cost as an IP choice into the IP header of the packet kept for its next-hop receiver Bruzgiene R et al. [22]. As a result, each node keeps track of the data packets that are being transported nearby in order to stop and benefit on the connection costs associated with each packet to determine the cost of a route segment. Each time a node sends, receives, or overhears a data packet, it logs the following details into a link cost table.

1. Sender
2. receiver
3. Link cost between the sender and the receiver
4. Source
5. Destination
6. IP header ID
7. The current time

The MAC header may be used to retrieve (a) and (b) of these parameters, whereas the IP header can be used to obtain (c)-(g) of them. In order to maintain accuracy and save down on storage costs, link information is only going to be retained for a brief period of time.

A node can learn how a packet travels through its neighbourhood and the overall connection cost associated with it via the link cost table. For instance, Chart 1 displays the connection energy table for node D. Since a packet may be identified by its source, destination, and IP header ID, the table shows that node D keeps track of the path information for three packets: P1 (S1, D1, 1), P2 (S2, D2, 3), and P3 (S3, D3, 5). The first packet (P1) travels via a two-hop route segment in the vicinity of D (A, B, and C), with a total cost of 9 (5 + 4). A node can employ the Remove, Replace, and Insert actions to lower the cost of a segment of the local path, and also the cost of the end-to-end path created between the source and the destination, based on the data stored in its link cost table. The three procedures are shown in Fig. 4 operating around with a node D. We will go into depth about each of the three operations below:

Remove: Assume that in node X's link cost table there is a two-hop path segment X→ A→ B on the way to a destination Z, with a total cost of T. In order to update its routing database, X will change the next hop for the destination Z to B if it discovers that the link cost between X and B is less than the cost of the two-hop route segment. In Fig. 4a, node D is shown for a path and destination with the two-hop path information (D→ B→ E) from its link energy table, in addition to the total link cost (5). Using the RTS or CTS packets sent by node E, D may determine the link cost to E(P_T(D, E)) if node E is one of D's surrounding nodes. The next hop for destination D2 will be changed in D's routing table to E if P_T (D, E) < 5. The following packets for destination D2 will pass straight via E.

Replace: Assume that the link cost table of node X contains a two-hop path segment A→ B→ C on the way to a destination Z, with an overall cost of T. When updating its routing table, X will set C as the next hop for the destination Z if it discovers that the overall cost for the path segment A! X! C is less than the cost of the two-hop path segment A→ B→ C. Additionally, it will ask A to change Z's next hop from A to X. In Fig. 4b, Node D's link cost data for destination D1 contains details on the two-hop route segment (A→ B→ C), with a total cost of 9. D may calculate the connection costs to A and C if they are both one of his surrounding nodes (P_T (D, A), P_T (D, C)). The path A→ D→ C is more energy-efficient than A→ B→ C if P_T (D, A) + P_T (D, C) <9. As a result, node D ask node A to setup a hop to C for destination D1, bu tat the same time also update its next hop for destination D1 to D. If A agrees the request of D, D will receive the intended packets for D1 by A and from there it will forward the same packets to C. After a delay period, the routing data for destination D1 at node D will be deleted if A rejects D's request.

Insert: Assume that the link cost table of node X contains a one-hop path segment A→B on the way to a destination Z, with a total cost of T. When updating its routing database, X will assign B as the next hop for the destination Z if it discovers that the total cost of the path segment A→ X→ B is less than the expense of the one-hop path segment. Additionally, X will ask A to establish the destination Z as its next hop. Fig. 4c shows that Node D reports a link cost of 7 for the one hop route segment (F→ G) going to D3. D can calculate the connection costs to F and G if they are both one of his nearby nodes (PT (D, F), PT (D, G)). The route segment F! D! G is better at energy efficiency than F! G if PT (D, F) + PT (D, G) 7. Two changes will be made, first the next hop destination D3 will be changed to G and request will be sent to F to change the next hop for D3 thus resulting in change of the routing table for D.

It is important to note that a route segment with more than two hops can be subjected to both Replace and Remove procedures. Every node on the path segments that are being observed must be close neighbours of the monitoring node in order to estimate the connection cost without incurring additional signalling costs Chatterjee B et al. [23]. It is extremely unlikely that a route segment greater than two hops would have all of its nodes within the active monitoring range. Sometimes, several maintenance procedures wuth multiple hops which are planned beforehand are replaced with operations on a one-hop or two-hop path segment.

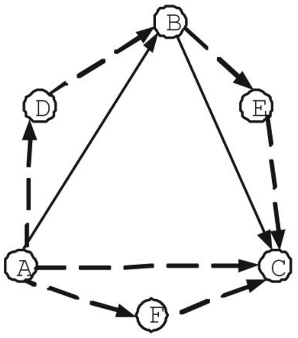

Fig. 5

A monitoring node only has to provide control messages during Replace and Insert operations in the suggested maintenance scheme to enable route modification. Since control messages are delivered only in case a faster path is found, The maintenance overhead is relatively minimal. This control message includes five things, namely, the Operation type, requester ID, destination, subsequent hop on the old path segment, and the overall cost for the new route segment S. Sugumaran et al. [24]. In the aforementioned example, D transmits A a control message for a Replace operation, which is [Replace, D, D1, B, the overall cost of new path segment ADC], and sends F a message for an Insert operation, which is [Insert, D, D3, G, the overall cost of new path segment FDG]. After receiving a control message, a node will analyse the routing table to gain all the routing information for the destination. The calculations of the previous path are not of any use anymore since the path has been altered, it will disregard the control message when the subsequent hop for such a destination differs from the one specified in the control message. Since Insert only must verify the one-hop transmission among these three processes, it may be easier to request than the other two. This might not be ideal, though.

An illustration. In fig 5, A data packet is sent from node A to node B. D knows that and energy efficient method would be to link nodes A and B by insert it into only one hop route segment. There are two other alternatives, though, and the optimum route section is AFC. The better option is to choose both Remove as well as Replace over Insert. In the case of PEER, a node will take more time to make a decision if it receives Remove or Insert requests. It will do the other action if it gets both an Insert request and a request for a Replace or Remove operation. If it receives requests for both Remove and Replace operations, it will choose the one that has the greatest potential for energy savings Kanthe A.M et al. [25]. For this particular example, node A will only carry out the Remove and Replace actions after receiving the Insert (by node D), Remove (by node C), and Replace (by node F) requests.

Performance evaluation

In order to implement the MTRTP protocol, we updated AODV to look for the least expensive way using the newly calculated link cost. 250 metres is the standard per hop transmission distance. Numerous tiny hops are used to conserve energy. All three protocols, including the standard AODV protocol, include power control, which allows a transmitter to change the broadcast strength depending on how close it is to the next-hop receiver. The simulation's network area is set to 1,200(m) + 1,200(m), and the network's nodes are dispersed at random. 1, 5, 10, 15, 20, 25, and 30 mW of transmission power are among the levels that are accessible. It is 35 mW on the Pm. The time period of this session follows an exponential distribution, whereas the session arrival rate follows a Poisson distribution.

Table 1

Parameter	Value	Parameter	value
Number of Nodes	60	Packet Size(byte)	512
Connection Arrival Rate	30	Connection Duration(min)	6
Max Speed(m/s)	10	Min Speed(m/s)	0.5

Constant Bit Rate (CBR) is the application protocol, whereby source and destination pairings are chosen at random. The mobility uses a 30-second stop interval and a modified random waypoint model. 50 packets are transmitted per second for each CBR session. Using the approach in, the route loss and collision rate are calculated. The remembering rate, referred to as filter memory, is set to 0.99. An outcome of the simulation was obtained by averaging over 20 runs with various seeds. Table 1 includes additional default setup options. We presume there isn't an energy saver mode for the nodes, thus regardless of whether it fails to receive a packet, a node will still use power to check the channel.

Additionally, a node requires energy when it hears packet communications. Hence, it is impossible to actively regulate the receiving power. Thus, we don't account for collecting power in the tests and simply compare transmission power S. Sugumaran et al. [26]. We first assess the suggested cost model's correctness, then we examine each protocol's route finding performance, and lastly we take energy use and RTS retransmissions into account in both stationary as well as mobile environments.

Accuracy of energy consumption model

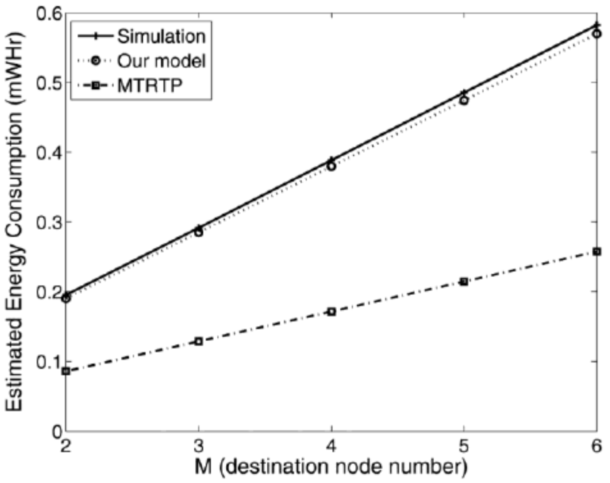

Fig 6

In this study, we compare the precision of our model to MTRTP. The transmission power level in this experiment is set at 1 mW for data packets and 5 mW for RTS and CTS packets. We employ static route and only take into account one path from the source (identified as node 0) to a destination node. This node is ually 2 to 6 hops distant along the path in order to eliminate the effect of identifying a route on the energy usage (numbered as nodes 2 to 6, respectively). We send 65,536 data packets using CBR. The rate of packet errors is fixed to 0.001. Figure 6 displays the outcomes of the simulations and the projected energy usage for each model. According to the findings, MTRTP significantly undervalues energy consumption, and the gap grows as the amount of intermediary nodes rises. In contrast, energy consumption depending on our cost model accurately reflects the results of the simulation.

Routing overhead and setup time

10,000 connection requests were replicated for every protocol in this research, and the total amount of routing packets, energy used, and setup time were recorded for each simulation. Figs. 7, 8, and 9 show the outcomes of the simulation. The findings show that the conventional on-demand routing protocol, followed by PEER and MTRTP, performs most effectively with respect to routing request, path setup time and energy consumption for routing overhead . MTRTP has significantly greater routing overhead and setup times than the on-demand routing protocol, and these times and overhead rise sharply with the number of nodes. This is due to Section 2's discussion of MTRTP's O(n^2)

(n = number of nodes) routing overhead. When a result, MTRTP was unable to expand effectively as the network's nodes increased.

Performance-wise, MTRTP appears to perform much worse than the PEER protocol. Most crucially, unlike MTRTP, PEER's efficiency statistics for routing overhead as well as route setup time grow only as a function of nodes in the network rather than rapidly Lorincz et al. [27]. As a result, PEER is more scalable as nodes increase. When there are 100 nodes, PEER can minimise the routing overhead by around 2/3, as well as the associated energy use and path setup time.

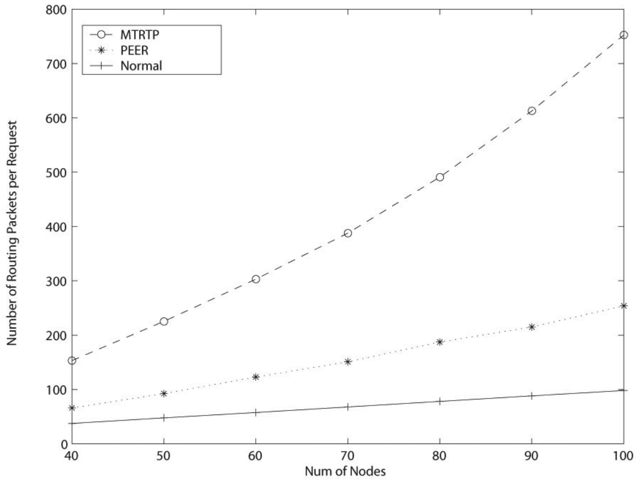

Fig 7

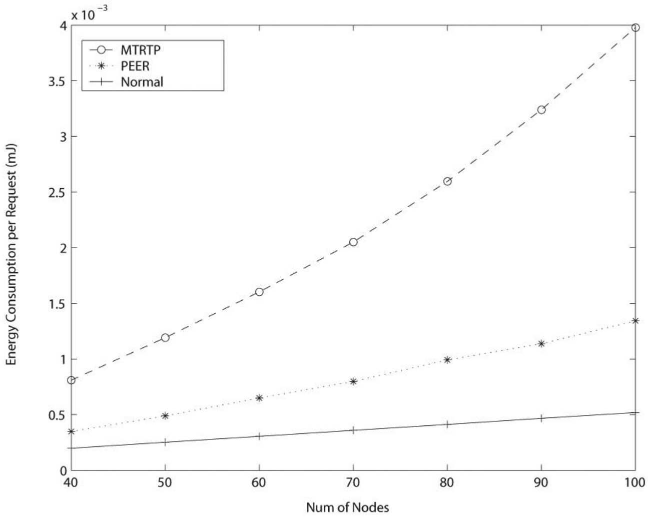

Fig 8

Performance in static scenario

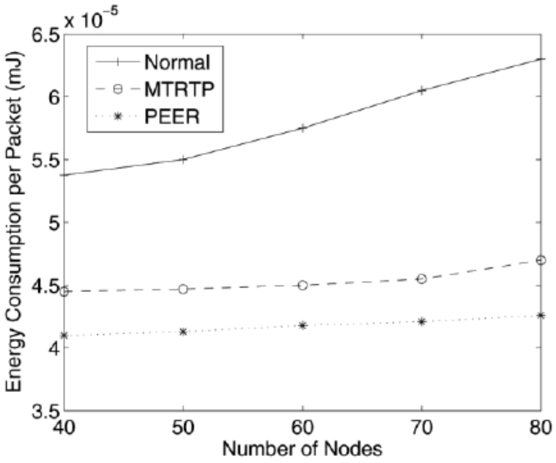

Fig 10

In a static situation, we varied the connection arrival rate, average packet size and node density to assess the energy consumption and overall amount of RTS retransmissions of the three protocols. Each protocol's simulation lasts for five hours. For each simulation, we kept track of the amount of energy used by all the packets that were acquired, the overall amount of packets that were obtained at all the destination nodes, as well as the total number of RTS retransmissions. The two measures we employed to assess the procedures are as follows:

- Consumption of energy per packet: The value of total energy used and the amount of packets received are taken and the former is divided by the later. This measure shows how effective each procedure is in using energy
- RTS Retransmissions per Data Packet on Average: It is calculated by dividing the total amount of packets received by the total number of RTS retransmissions. The RTS packet has a very short packet size and is broadcast at maximum power. Collisions, which can involve both RTS messages and data packet collisions, are the main cause of RTS retransmissions. This statistic can therefore represent the collision rate for each protocol. More power will be used, there will be a longer end-to-end latency, and less throughput as a result of a greater collision rate.

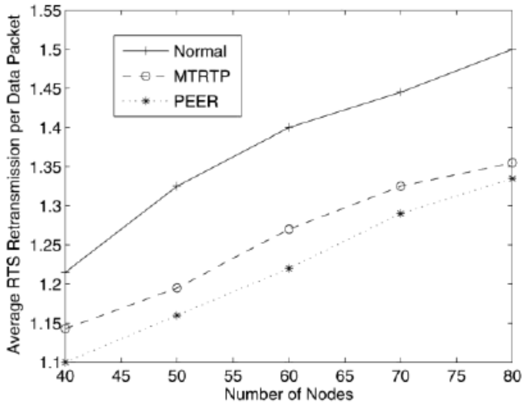

Fig 11

Figs. 10, 11, 12, 13, 14, and 15 display the model findings. The PEER approach, which is accompanied by the MTRTP approach and the standard protocol, performs most effectively for all three sets of tests in regards to Energy Consumption per Packet and Average RTS Retransmission per Data Packet.

As opposed to the shortest route in the typical AODV protocol, the PEER and MTRTP protocols look for the most energy-efficient way, which allows them to perform significantly more in regard to energy usage Venkatasubramanian et al. [28]. PEER outperforms MTRTP in regard to energy usage for a number of reasons. First, the PEER protocol may look for a more energy-efficient way and employ a more precise link cost. Secondly, The MTRTP has a significant routing overhead therefore, there is a very high probaibly that node in between the source and destination do not pick the path search request. The increased energy use is partly a result of the greater routing overhead. Thirdly, PEER protocol can sustain a power saving path better and swiftly adjust the path to environmental changes. Hidden terminal issues can be avoided if the transmission power for RTS/CTS are used to its maximum capacity. Utlisling the same helps in letting the branches present in transmission zones of the sender as well as the receiver to pinpoint NAV values. However, Since all three protocols employ power control, a node A's reduced transmitting energy for information or ACK packets also limits the sensing area for all other nodes to identify the node's transmission. When broadcasting at a greater range (above the maximum transmission range), the nodes that are unable to detect node A's broadcast may turn into hidden nodes and clash with the reception at A. The range of data transfers for the standard protocol might range from extremely tiny up to the transmission limit.

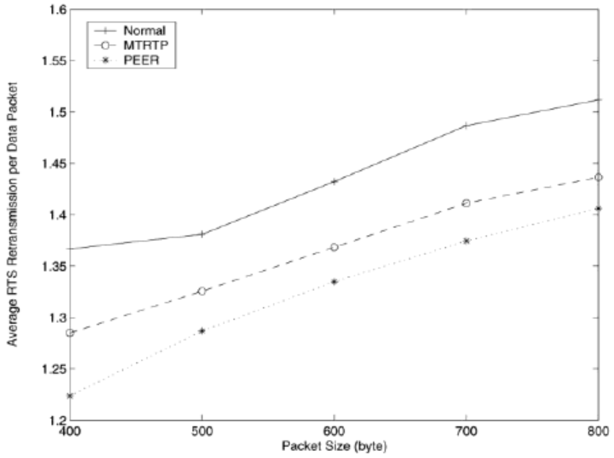

Fig 13

Significant hidden node issues and subsequent collisions will result from the wide disparity in transmission ranges. There will be fewer collisions when the two energy-efficient routing techniques attempt to leverage certain shorter distance links caused by a hidden node delivering data packets.

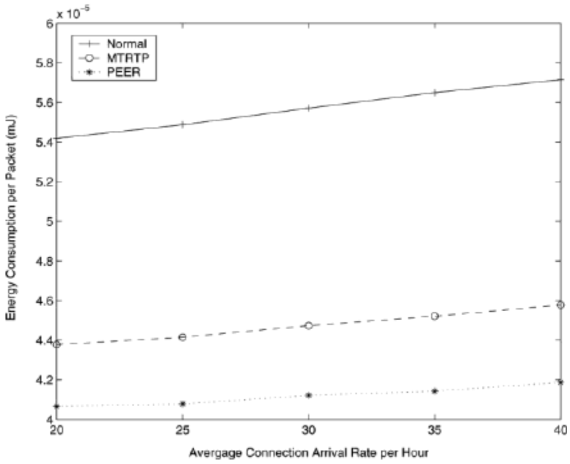

Fig 14

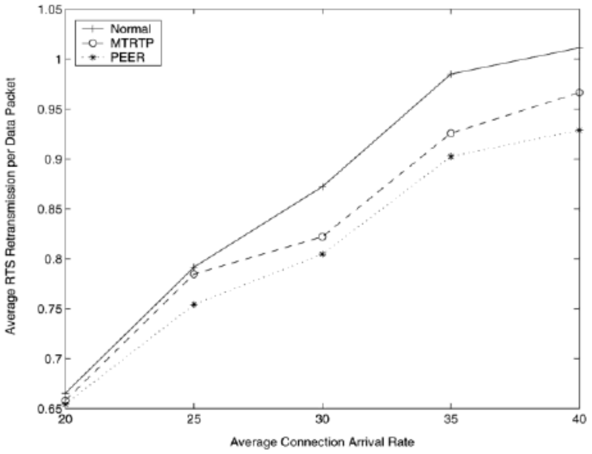

Fig 15

RTS/CTS packets, which are typically considerably smaller than data packets, might potentially clash during transmission when sent at maximum power from a hidden node, even though the likelihood of this collision is far lower than for transmitting data packets. As a result, the standard protocol has a greater retransmission rate than the energy-efficient routing methods. MTRTP undervalues the connection cost and frequently uses a way with more hops since it does not account for energy used in signalling, which is important given the signalling protocol's high signalling overhead , S. Sugumaran et al. [29]. This will also make it more likely that RTS packets may be lost, leading to more retransmissions. As a result, under all simulated situations, the PEER protocol has the least RTS retransmission rate out of the three. It's noteworthy to note that all protocols' RTS retransmission rates in Fig. 11 rise with node density, although there is no such trend in Fig. 10's energy usage per packet. A higher quantity of nodes as well as a lower hop distance may be located on more energy-efficient paths in a highly dense network, which would offset the increased energy used by more retransmissions.

Protocol in mobile scenario

By adjusting the packet size, average node speed, and connection arrival rate for all three protocols in the mobile scenario, we were successfully compared the same metrics as in static settings. Figures 16, 17, 18, 19, 20, and 21 show the simulation's outcomes. The PEER protocol achieves the best for all three groups of trials in regards to Energy Usage per Packet and Average RTS Retransmission per Data Packet. MTRTP consistently consumes the most energy, as might be predicted. The least energy path chosen at the point of route establishment may no more be power saving and may now waste even further power than the standard routing protocol since its path typically has much more hops because it could not adjust properly to the mobility.

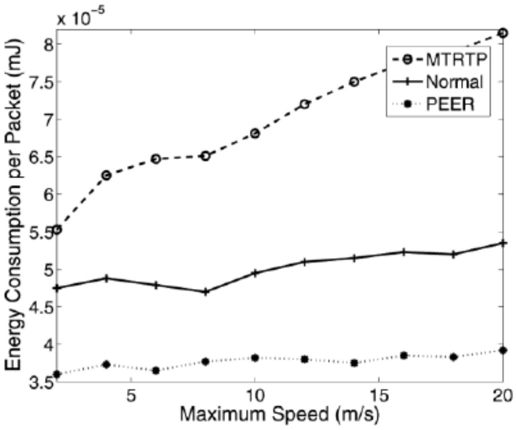

Fig 16

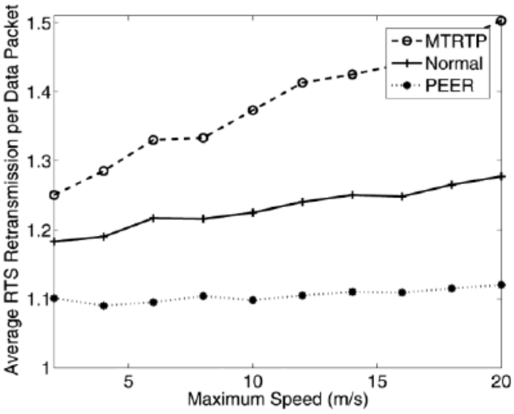

Fig 17

The outcomes of the simulation support this. MTRTP has been seen to use considerably more power than the standard routing protocol. With an effective route maintenance plan, PEER may change its course with the flexibility to continuously maintain an energy-efficient path. Hence, in all test cases, PEER outperforms the standard on-demand routing protocol as well as uses significantly less energy than MTRTP. When changing the average connection, packet size and movement speed arrival rate, PEER can cut energy consumption by up to 27, 25, and 25%, respectively, when compared to the standard routing protocol (which is enhanced with power management capabilities as described in the performance setup). PEER can cut energy use by up to 51, 40, and 40%, respectively, as compared to MTRTP.

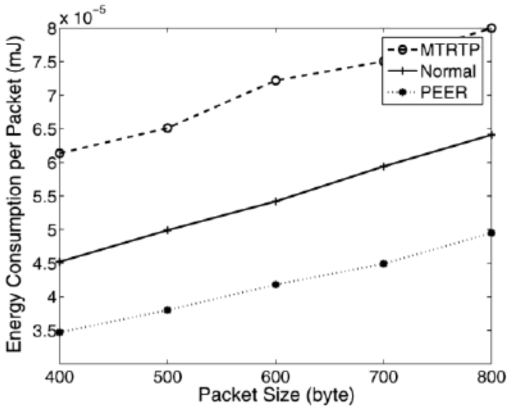

Fig 18

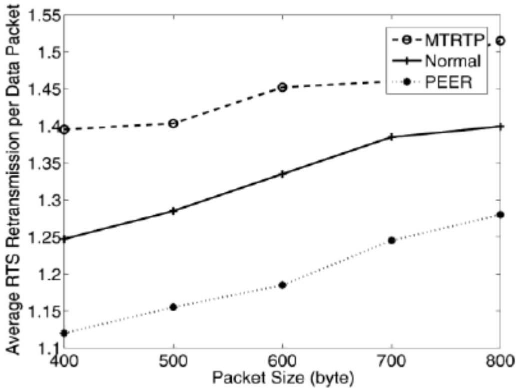

Fig 19

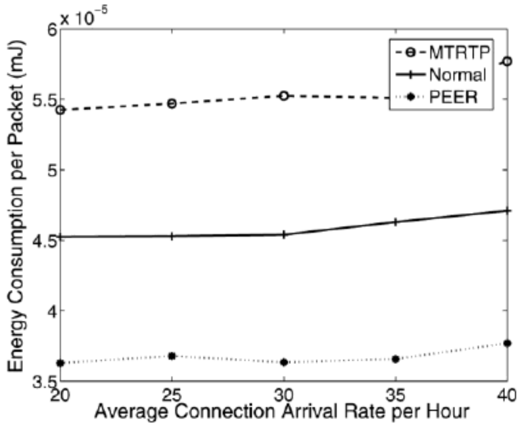

Fig 20

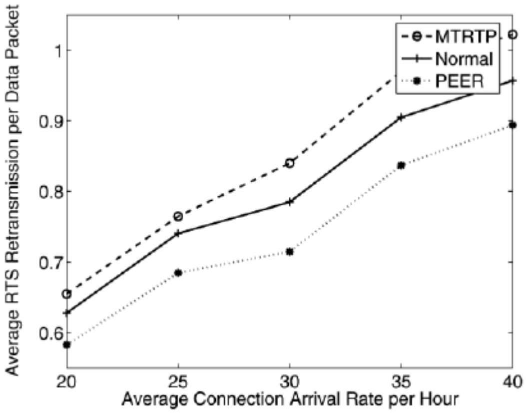

Fig 21

As was noted in the static case, asymmetric energy is primarily responsible for the RTS retransmission. The separation of two nodes in MTRTP can rise to the transmission range due to the mobility of the node. This further results in asymmetric power issues similar to those in the standard protocol. Also, RTS retransmission rate in MTRTP is higher than in standard protocol because of its utilisation of more hops. The PEER protocol, on the other hand, could modify the path when nodes moved, allowing it to continue on an energy-efficient path despite node mobility S M, B et al. [30]. As a result, it uses less power than the standard protocol and has fewer RTS retransmissions than usual.

Conclusion

Designing energy-efficient routing systems for mobile ad hoc networks is crucial. A routing protocol highly efficient in using power might, however, perform significantly worse than a standard routing protocol if it is not carefully designed. Moreover, as shown by our experiments, an energy-efficient routing protocol might result in much greater control overhead and route setup latency, and spend far more power than a typical routing protocol in a mobile context. In order to more precisely monitor the energy usage owing to various causes, we first developed a new connection cost model in this research. The problems with path discovery and route management related to the minimal energy routing techniques were then covered.

Based on these results and our innovative link cost measurement, we propose a PEER protocol with a speedy and low overhead route discovery approach and an efficient path management strategy for reducing energy consumption, especially in mobile contexts. Our performance evaluations demonstrate that the PEER protocol is highly flexible to environmental changes and decreases routing overhead and route setup latency by around 2/3 when compared to a standard energy-efficient routing strategy. With respect to network density, load and node mobility, network density, and load, PEER performs significantly better than the average energy-efficient protocol, both in static as well as in mobile scenarios. Comparing PEER to the traditional energy-efficient routing protocol MTRTP in mobile situations, PEER can cut transmission energy usage by up to 50% in all simulated cases.

References

[1] Jamali, S.; Rezaei, L.; Gudakahriz, S.J. An Energy-Efficient Routing Protocol for MANETs: A Particle Swarm Optimization Approach. J. Appl. Res. Technol. 2013, 11, 803–812.

[2] Devi, M.; Gill, N.S. Mobile ad hoc networks and routing protocols in IoT enabled. J. Eng. Appl. Sci. 2019, 14, 802–811.

[3] Kumar Debnath, S.; Saha, M.; Islam, M.; Sarker, P.K.; Pramanik, I. Evaluation of Multicast and Unicast Routing Protocols Performance for Group Communication with QoS Constraints in 802.11 Mobile Ad-Hoc Networks. IJCNIS 2021, 13, 1–15.

[4] Hu, Z.; Odarchenko, R.; Gnatyuk, S.; Zaliskyi, M.; Chaplits, A.; Bondar, S.; Borovik, V. Statistical Techniques for Detecting Cyberattacks on Computer Networks Based on an Analysis of Abnormal Traffic Behavior. IJCNIS 2021, 12, 1–13.

[5] Khokhlachova, Y.; Hu, Z.; Sydorenko, V.; Opirskyy, I. Method for Optimization of Information Security Systems Behavior under Conditions of Influences. IJISA 2017, 9, 46–58.

[6] Ray, S.; Mishra, K.N.; Dutta, S. Sensitive Data Identification and Security Assurance in Cloud and IoT Based Networks. IJCNIS 2022, 14, 11–27.

[7] Jayalakshmi, D.S.; Hemanand, D.; Kumar, G.M.; Rani, M.M. An Efficient Route Failure Detection Mechanism with Energy Efficient Routing (EER) Protocol in MANET. IJCNIS 2021, 13, 16–28.

[8] Jesudurai, S.A.; Senthilkumar, A. An improved energy efficient cluster head selection protocol using the double cluster heads and data fusion methods for IoT applications. Cogn. Syst. Res. 2019, 57, 101–106.

[9] Rahman, T.; Ullah, I.; Rehman, A.U.; Naqvi, R.A. Notice of violation of IEEE publication principles: Clustering schemes in MANETs: Performance evaluation, open challenges, and proposed solutions. IEEE Access 2020, 8, 25135–25158.

[10] Hou, Y.T.; Shi, Y.; Sherali, H.D. Rate Allocation and Network Lifetime Problems for Wireless Sensor Networks. IEEE/ACM Trans. Netw. 2008, 16, 321–334.

[11] Alam, T.; Rababah, B. Convergence of MANET in Communication among Smart Devices in IoT. IJWMT 2019, 9, 1–10.

[12] Niu, Z.; Li, Q.; Ma, C.; Li, H.; Shan, H.; Yang, F. Identification of Critical Nodes for Enhanced Network Defense in MANET-IoT Networks. IEEE Access 2020, 8, 183571–183582.

[13] Narayandas, V.; Archana, M.; Raman, D. The Role of MANET in Collaborating IoT End Devices: A New Era of Smart Communication. Int. J. Interact. Mob. Technol. 2021, 15, 80–92.

[14] Aroulanandam, V.; Latchoumi, T.; Balamurugan, K.; Yookesh, T. Improving the Energy Efficiency in Mobile Ad-Hoc Network Using Learning-Based Routing. RIA 2020, 34, 337–343.

[15] Amiri, I.S.; Prakash, J.; Balasaraswathi, M.; Sivasankaran, V.; Sundararajan, T.V.P.; Hindia, M.N.; Tilwari, V.; Dimyati, K.; Henry, O. DABPR: A Large-Scale Internet of Things-Based Data Aggregation Back Pressure Routing for Disaster Management. Wirel. Netw. 2020, 26, 2353–2374.

[16] Kathiroli, P.; Selvadurai, K. Energy Efficient Cluster Head Selection Using Improved Sparrow Search Algorithm in Wireless Sensor Networks. J. King Saud Univ.-Comput. Inf. Sci. 2021, 34, 8564–8575.

[17] Maheshwari, P.; Sharma, A.K.; Verma, K. Energy Efficient Cluster Based Routing Protocol for WSN Using Butterfly Optimization Algorithm and Ant Colony Optimization. Ad Hoc Netw. 2021, 110, 102317.

[18] Rajpoot, P.; Dwivedi, P. Multiple Parameter Based Energy Balanced and Optimized Clustering for WSN to Enhance the Lifetime Using MADM Approaches. Wirel. Pers. Commun. 2019, 106, 829–877.

[19] Li, Y.; Xiong, W.; Sullivan, N.; Chen, G.; Hadynski, G.; Banner, C.; Xu, Y.; Tian, X.; Shen, D. Energy Efficient Routing Algorithm for Wireless MANET. In Proceedings of the 2019 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2019; pp. 1–9.

[20] Er-rouidi, M.; Moudni, H.; Mouncif, H.; Merbouha, A. A Balanced Energy Consumption in Mobile Ad Hoc Network. Procedia Comput. Sci. 2019, 151, 1182–1187.

[21] Vu, Q.K.; Le, N.A. An Energy-Efficient Routing Protocol for MANET in Internet of Things Environment. Int. J. Onl. Eng. 2021, 17, 88–99.

[22] Bruzgiene, R.; Narbutaite, L.; Adomkus, T. MANET Network in Internet of Things System. In Ad Hoc Networks; Ortiz, J.H., de la Cruz, A.P., Eds.; InTech: London, UK, 2017; ISBN 978-953-51-3109-0.

[23] Chatterjee, B.; Saha, H.N. Parameter Training in MANET Using Artificial Neural Network. IJCNIS 2019, 11, 1–8.

[24] Sugumaran, S.; Venkatesan, P. Attacks Reduction in MANET with Cluster Supported Trusted Routing Protocol. J. Adv. Res. Dyn. Control. Syst. 2017, 6, 2023–2032.

[25] Kanthe, A.M.; Simunic, D.; Prasad, R. Comparison of AODV and DSR on-demand routing protocols in mobile ad hoc networks. In Proceedings of the 2012 1st International Conference on Emerging Technology Trends in Electronics, Communication & Networking, Surat, India, 19–21 December 2012; pp. 1–5.

[26] Sugumaran, S.; Venkatesan, P. Optimized Trust Path for control the Packet dropping and collusion attack using Ant Colony in MANET. Int. J. Eng. Adv. Technol. 2019, 8, 4833–4841.

[27] Lorincz, J.; Ukic, N.; Begusic, D. Throughput comparison of AODV-UU and DSR-UU protocol implementations in multi-hop static environments. In Proceedings of the 2007 9th International Conference on Telecommunications, Zagreb, Croatia, 13–15 June 2007; pp. 195–202.

[28] Venkatasubramanian, S.; Suhasini, A.; Vennila, C. Cluster Head Selection and Optimal Multipath Detection Using Coral Reef Optimization in MANET Environment. IJCNIS 2022, 14, 88–99.

[29] Sugumaran, S.; Venkatesan, P.; Chitra, M.G.; Sivasakthiselvan, S.; Jayarajan, P. Best Optimized Route in MANET using Token Economy Management System. In Proceedings of the 2022 IEEE 4th International Conference on Advances in Electronics, Computers and Communications (ICAECC), Bengaluru, India, 10–11 January 2022.

[30] Benakappa, S.M.; Kiran, M. Energy Aware Stable Multipath Disjoint Routing Based on Accumulated Trust Value in MANETs. IJCNIS 2022, 14, 14–26.