Dynamic Routing Using Petal Ant Colony Optimization for Mobile Ad-hoc Networks

Sathyaprapaksh B.P.\(^1\), Manjunath Kotari\(^2\)

School of Electronics and Communication Engineering, REVA University, Bengaluru-560064, India\(^1\)
Research Scholar, Visvesveraya Technological University, Belagavi-590018, India\(^1\)
Dept. of CSE, Alva’s Institute of Engineering and Technology, Moodbidri, Mangalore-574225, India\(^2\)
Visvesveraya Technological University, Belagavi-590018, India\(^2\)

Abstract—A Mobile Ad-hoc Network (MANET) is a temporary wireless network that configures itself as needed. Each MANET node has a finite number of resources and serves as both a node and a router at the same time. MANET nodes are mobile and move from one location to another. Because MANET nodes are dynamic, choosing an optimal node for data transfer is a difficult issue. Because packets must propagate in a multi-hop manner, they take a longer path and may endure a longer delay, causing them to become lost in the network. The network’s overall performance suffers as a result of the re-transmission of those lost packets. We propose a modified version of a nature-inspired algorithm called Petal Ant based Dynamic Routing (PADR) in this research study, which reconstructs data packets to traverse inside a given region and achieves minimal delay during data transmission. The PADR is simulated in Network Simulator (NS2) and compared against nature-inspired routing protocols like PAR and SARA, as well as traditional routing protocols like AODV.

Keywords—Petal ant routing; dynamic petal ant routing; MANET; ant colony optimization

I. INTRODUCTION

In our day-to-day lives, wireless communication networks are becoming increasingly crucial [3] [12]. Wireless devices connect, exchange, and transfer data wirelessly, giving them a substantial level of mobility [20]. Infrastructure-based networks and infrastructure-less networks are the two types of wireless networks available [1] [19]. Wireless devices communicate in infrastructure-based networks through a centralized administration, such as a base station or an access point [28]. The wireless device communicates directly with no pre-existing communication infrastructure in an infrastructure-less network, which is known as an Ad-hoc Network. Mobile Ad-hoc Network is an example of an ad-hoc network (MANET) [5].

The Mobile Ad-hoc Network is a transitory wireless network made up of a collection of wireless devices such as laptops, PDAs, and mobile phones that may dynamically configure, communicate, and react with no central administration [6][7]. Because MANET’s nodes are mobile, they connect to other networks using a regular Wi-Fi connection [23]. Because MANET mobile nodes communicate with one another over a wireless medium, the devices must establish a communication link and send data to others even when they are not in direct transmission range [14][30]. MANETs, in general, forward data in a multi-hop fashion and serve as routers even when no infrastructure equipment is present [16][31]. Congestion, routing, security, and other challenges arise as a result of a MANET’s absence of infrastructure equipment [18] [29]. One of the most important challenges is routing, because when determining the best way for data transmission, complications like delay, performance, throughput, and overheads can generate a slew of problems and reduce network efficiency [4] [15]. As a result, additional strategies are required to overcome such difficulties with high node mobility, according to [25] [17]. A Routing in a MANET is a common theme that has resulted in numerous routing protocols in the literature and has remained a difficult issue for the past few decades [21] [22].

The key contributions and originality aspects of the proposed routing are as follows:

1) To identify the shortfalls of existing PAR route discovery procedure. We provide a cutting-edge technique for obtaining petal width from static to dynamic petal between end nodes.

2) Our approach aims to maximize the data delivery rate between end-to-end nodes during data transmission by discriminating the mobile node during the process. We propose an innovative technique of electing the nearest neighboring node so that delay and routing overheads between end nodes is minimum as quoted in the algorithm. This impacts the performance and user experience of various applications and services in the real world.

3) For each of the proposed tests, 20 simulation runs were conducted ranging from 15 to 150 mobile nodes, and an average value was calculated at the end of the simulation. The suggested routing algorithm is evaluated for various metrics such as packet delivery fraction, throughput, overhead and delay, and the results are compared with PAR routing.

Overall, this article proposes an extension of the ant colony optimization technique, a nature-inspired algorithm that optimizes both the route discovery and route maintenance mechanisms for data delivery from groups of mobile nodes in MANETs. The proposed technique evaluation results demonstrate its effectiveness in discriminating node during route discovery and provides better performance than PAR routing strategy. The following section contains various related works, our contributions, and an analysis of the results.
II. RELATED WORK

Hua Wang, Z ao Shi, et al. (2008) [2] proposed a modified Ant Colony algorithm for ad-hoc networks that included a progressive changing orientation component and used it to solve multi-constraint routing problems. The gradual changing orientation factor’s directionality allows the ants to consider direction when searching for a path, allowing them to proceed fast to the end node and so improve convergence speed by overcoming the slow convergence setback. According to the author, an ant in the improved algorithm not only uses prior search findings, but also lowers the misleading influence of pheromones on improper paths, overcoming the sluggish convergence problem. The results of the experiments show that the ant colony algorithm with progressive shifting orientation factor outperforms the standard ant colony algorithm in terms of convergence time while also avoiding the risk of a local best solution.

Saptarshi Benerjee et al., [9] (2015) suggested a new MANET routing strategy based on on-demand multipath power balanced routing. The routing protocol in [9] is a modified version of the ACO architecture, and it consists of three phases: Route Discovery, Route Maintenance, and Route Failure. The routing algorithm updates the pheromone value in the routing table using the battery charge of a mobile node as the major parameter. During the route discovery phase, the routing algorithm uses the identical Forward Ant (FANT) and Backward Ant (BANT) generated by the source and destination. The source establishes the communication channel to the destination based on the pheromone value obtained by the probability of battery charge of mobile nodes. In OMNET++ version 4.5, the routing algorithm is simulated for a small-scale network. The simulation findings show that the ACO algorithms give better packet delivery count convergence than the existing ACO algorithms.

B Jayalalitha et al., (2016) [11] For MANET, the ANT Colony Optimization Routing (ACOR) algorithm was presented as a variation of the ant routing technique. The ACOR is proactive routing, similar to AntHocNet, which sends out control packets regularly. The author of ACOR modified the Ant-route proto-header packet and successfully compiled for execution, but the packets were not transmitted from source to destination node. In NS2, the ACOR is modeled and compared to AntHocNet, AODV, and DSR. According to ACOR, AntHocNet, AODV, and DSR simulation results, the ACOR builds the routing table but fails to transport packets between end nodes.

Ajay Kumar et al., (2017) [15] the importance of MANETs is discussed, as well as the disadvantages of routing for temporary networks. The author of [10] introduced the Efficient Fuzzy based Multi-constraint Multicast Routing protocol (EFM- MRP), which uses fuzzy logic to supervise uncertainty issues. The algorithm calculates the fuzzy cost and chooses the best path between end nodes with the lowest fuzzy cost, then sends the data to a group of receivers. In NS 2.35, the EFMMRP is simulated for 15 to 300 mobile nodes. For mobility, the EFMMRP employs the random waypoint network model, while the propagation model is based on the free space propagation model. The EFMMRP is compared to the On-demand Multicast Routing Protocol (ODMRP) and the Multicast Ad-hoc On-Demand Vector Routing Protocol (MAODV). The simulation results of EFMMRP, ODMRP, and MAODV show that EFMMRP outperforms ODMRP and MAODV in terms of delay, bandwidth, and energy metrics when faced with uncertainty.

Dipika Sarkar et al. (2018) [20] proposed the Enhanced Ant-AODV routing protocol for MANET, which is based on Quality of Service (QOS). The Enhanced-Ant-AODV is a reactive technique for MANET route selection that combines AODV with ANT. In the frame format of the AODV route request and route reply packet, an extra field called Pheromone count has been added. Based on the pheromone value, hop count, congestion, and reliability of the path between the end nodes, the Enhanced-Ant-AODV chooses a finite route for data delivery. The path with the highest pheromone value is chosen as the optimum data delivery path. In NS 2.35, the Enhanced-Ant-AODV is simulated and tested with the AODV and DSR MANET routing protocols.

Valanto Alappatt et al. (2020) [18] to solve the issue of routing, the authors proposed Ant Colony Optimization and Binary Particle Swarm Optimization (ACO: BPSO), a modified version of the ant colony optimization technique for mobile ad hoc networks. The ACO: BPSO is a hybrid approach for enhancing the network’s lifetime that combines Ant Colony Optimization and Binary Particle Swarm Optimization techniques. In NS2, the ACO: BPSO routing protocol is simulated for 50 nodes and compared to the regular AODV routing system. Based on the simulation results, the author of [18] claims that the ACO: BPSO hybrid strategy outperforms the AODV routing protocol in terms of packet delivery rate, throughputs, and residual energy.

III. PROPOSED ALGORITHM: PETAL ANT DYNAMIC ROUTING (PADR)

As described in several research proposals published in the literature, real ants are able to find the fastest and most efficient road to get from their food source to their nest in response to their environment [26] [13]. Petal Ant based Dynamic Routing (PADR) is a reactive routing system that is a variant of the Ant Colony Optimization (ACO) Algorithm. When a wireless node is out of transmission range, the PADR divides the network into a Petal Region (PR), which makes it easier for the node to choose a destination from the source. The detailed description of the proposed algorithm is as follows:

The Petal Ant based Dynamic Routing (PADR) algorithm has three phases namely

- Route Discovery,
- Route Maintenance, and
- Route Repair

A. Route Discovery

Consider the example of a small network of 10 nodes shown in Fig. 1. The wireless mobile nodes are represented by the alphabets A, B, C, D, E, F, G, H, and I, with the node S denoting the Source, D denoting the Destination, and the other nodes denoting the Intermediate node in the network. If any
source node seeks to construct a data path to a destination node during the route discovery phase, the PADR divides the network into petal regions ($P_w$). If S does not have a routing table entry for destination D in its routing table, node S divides the network into petal regions as illustrated in the Fig. 1 using equations (1, 2, 3), where $(x_s, x_d)$ and $(y_s, y_d)$ are the longitudes and latitudes in real-time measurements of node S and D, $d$ is the distance of S & D, "c" is a variable (minor axis), "a" is 50% of d and "b" is 50% of c.

$$\sqrt{(x_d - x_s)^2 + (y_d - y_s)^2}$$ \hspace{1cm} (1)

$$h = (x_s + x_d)/2, \quad k = (y_s + y_d)/2$$ \hspace{1cm} (2)

$$P_w = \pi ab$$ \hspace{1cm} (3)

Once the petal width between end nodes has been determined, the source propagates $P_{FANT}$ across the network. The rebroadcasting of the $P_{FANT}$ packet is delayed until it reaches D, which response with $P_{BANT}$ as an acknowledgment to the $P_{FANT}$ packet. The route discovery procedure of PADR is depicted in Fig. 2 and Fig. 3.

B. Route Maintenance

By balancing the pheromone concentration, the second phase of the PADR algorithm includes a route management procedure that keeps track of active and updated routes during data communications [24]. A link quality and activity are determined by its pheromone concentration. The link with the most data traffic has the highest pheromone levels, whereas the link with the least data traffic has the lowest.

$$\frac{(x-k)^2}{a^2} + \frac{(y-k)^2}{b^2} \leq 1$$ \hspace{1cm} (4)

The analysis of the pheromone difference provides considerable information about the network status, as the authors have already investigated in [2] [6], the data path with the highest pheromone level is the one that draws the most traffic [27]. The routing processing component of the PAR [8] algorithm has been significantly improved to model the PADR. Assume the data flow between the source and destination is via S-A-F-G-D and S-A-F-I-D in the scenario depicted in Fig. 4. PADR proposes an alternative path when a node travels outside the ($P_w$) region, as shown in Fig. 5.
On receiving APacket()
if recvPADRPkt(Packet* p) then
  if (PADR_DATA_Packet) then
    Increase the pheromone intensity between the intermediate link (i,j)
    if intermediate node observe the link breakage between node link (i,j) then
      Change the pheromone value from i – j to zero
      Broadcast link failure message to all nodes
      Source node tries to obtain alternative path to destination
    if (Source found alternative path) then
      Start sending data packet on the new path
    else
      Source initiate new route discovery process
  else
    if (Intermediate node lies inside $P_w$) then
      Intermediate node increase the pheromone intensity in the link (i,j)
      Source continues to transmit the data packet on the same path
    else
      Change the pheromone value from i – j to zero
      Intermediate node drops the packet and does not forward the packet to
      its neighbor node
  end
end
end

Algorithm 1: PADR Route Maintenance

The PADR checks whether the mobile node is inside the $P_w$ region using Eq. (4). PADR’s data transmission is shown in Algorithm 1.

![Algorithm 1: PADR Route Maintenance](image)

IV. RESULTS AND ANALYSIS

This section compares and evaluates the performance of the PAR, SARA, AODV, and proposed PADR algorithms in Network Simulator (NS2) running on Ubuntu Linux version 18.04. The ACO framework method is used in the PAR, SARA, and suggested routing algorithms, but the AODV is a normal traditional-based routing algorithm. The PAR coding implementation in NS2 version 2.35 has been improved to behave like PADR. A network of nodes ranging from 15 to 150 nodes dispersed across a 1000 x 1000 m geographic region was used to test the proposed PADR. Table I lists the essential parameters and exact values used during the simulation investigation.

A. Study of Packet Generation

For the PADR, PAR, SARA, and AODV routing algorithms, Fig. 6 illustrates the total number of packets generated and sent by the source node to the destination across the network. The nature-inspired method creates more data packets than the conventional based routing technique, according to the simulation analysis of PADR, PAR, SARA, and AODV. When compared to the PAR, SARA, and AODV algorithms, the number of packets generated by the source node of PADR is higher because the source node receives a faster response from the destination for establishing a path through intermediate nodes, so the source starts transmitting
packets in the network right away. According to the graph in figure 6, the PADR creates 6.16% more packets on average than PAR, 9.61% more packets than SARA, and 26.5% more packets than AODV routing approaches.

### TABLE I. SIMULATION PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface queue type</td>
<td>Queue/DropTail</td>
</tr>
<tr>
<td>Radio-propagation model</td>
<td>Propagation/TwoRayGround</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>15, 30, 50, 75, 100 &amp; 150</td>
</tr>
<tr>
<td>Channel</td>
<td>Channel/WirelessChannel</td>
</tr>
<tr>
<td>Simulation area Size</td>
<td>1000 x 1000</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>160s</td>
</tr>
<tr>
<td>Packet size</td>
<td>512 bytes</td>
</tr>
<tr>
<td>Mac type</td>
<td>802.11</td>
</tr>
<tr>
<td>Node speed</td>
<td>0 m/s to 10 m/s</td>
</tr>
<tr>
<td>Transport protocol</td>
<td>TCP</td>
</tr>
<tr>
<td>Routing protocol</td>
<td>PAR and PADR</td>
</tr>
<tr>
<td>link layer type</td>
<td>LL</td>
</tr>
<tr>
<td>Network interface</td>
<td>Phy/WirelessPhy</td>
</tr>
</tbody>
</table>

### B. Study of Packet Received

For the PADR, PAR, SARA, and AODV routing algorithms, Fig. 7 depicts the total number of packets received by the destination node across the network. The routing algorithms PADR, PAR, SARA, and AODV are used. The nature inspired algorithm outperforms the classical based routing technique even at packet acceptance by the destination, according to the simulation analysis of PADR, PAR, SARA, and AODV. This is because in the ACO algorithm, the path with higher pheromone value attracts more data packets in the network. According to the graph in Fig. 7, the PADR’s destination receives 11.12% more packets on average than PAR, 14.66% more packets than SARA, and 47.42% more packets than the AODV routing technique.

### C. Study of Packet Delivery Fraction (PDF)

The analysis for packet delivery fraction for PADR in comparison to PAR, SARA, and AODV approaches is shown in Fig. 8. There are two main reasons why an intermediate node cannot participate in data transmission in PADR. If the node is outside the Pw, the pheromone value of that nearby links is zero, and if node moves outside the Pw, the pheromone value of the neighboring links is zero. In the latter case, data packets from intermediate (upstream and downstream) nodes with the connection value zero are deleted. The delivery rate of data packets is high in PADR because each node traverses all of the data packets within Pw. According to the graph in figure 8, PADR performs 0.103% better than PAR, 0.055% better than SARA, and 11.42% better than AODV routing technique.

### D. Study of End-to-End Delay

The end-to-end latency of PADR is shown in Fig. 9 in comparison to PAR, SARA, and AODV routing techniques.
Because the path with a higher pheromone value attracts more data packets, PADR has better end-to-end delay outcomes than PAR and SARA, but is slower than AODV. Pheromone concentration, on the other hand, lowers as time passes owing to pheromone evaporation. As a result, the ACO algorithm uses a restricted number of paths, which attracts an excess of traffic to the same routes, resulting in a nearly constant high end-to-end delay [2]. Because, in PADR, when a node with a data path travels outside of the $P_w$, the upstream node discovers an alternative path to downstream node within $\pi * a * b$ as shown in Fig. 5. As a result, PADR has a lower delay and latency than PAR and SARA routing approaches. The end-to-end delay of PADR is 8.65% less than PAR, 11.06% less than SARA, and 47.45% more than AODV routing approach, according to the graph in Fig. 9.

**E. Study of Throughput**

The throughput represents the path’s quality and the number of data packets delivered to the destination during a given period of time. The throughput of PADR is shown in Fig. 10 in comparison to PAR, SARA, and AODV routing techniques. In comparison to PAR, SARA, and AODV routing, PADR gives superior throughput simulation time, according to the simulation data. PADR gives an average of 6.17% more throughput than PAR, 9.69% more than SARA, and 32.50% more than AODV routing approach, according to the graph in Fig. 10.

**F. Study of Routing Overheads**

The routing overheads of PADR is shown in Fig. 11 in comparison to PAR, SARA, and AODV routing algorithms. PADR has slightly greater overhead than PAR and SARA, but less than AODV, according to the simulation results. Because nodes move beyond the $P_w$, PADR considers extra $P_w$ computation during route maintenance sessions, the overhead in PADR is slightly higher. Despite the fact that PADR consumes more overhead than PAR and SARA, it outperforms them in all of the listed measures.

**V. CONCLUSION**

This research article introduces Petal Ant Dynamic Routing (PADR), an upgraded version of a nature-inspired method. The proposed work’s major goal is to improve Quality of Service (QoS) in terms of performance, throughput, and networking latency. The proposed routing is simulated in NS 2.35 and compared to MANET routing domain related work. According to the simulation results, PADR outperforms AODV, SARA, and PAR in terms of packet generation, packet delivery fraction (PDF), end-to-end delay, and throughput, while using PADR slightly higher routing overheads than PAR.

**REFERENCES**


