On Designing Bee Inspired Routing Algorithm for Device-to-Device Communication in the Internet of Things

Asmaa Mohammed Almazmoomi[1], Muhammad Mostafa Monowar[2]

Department of Information Technology, Faculty of Computing and Information Technology
King AbdulAziz University, Jeddah, 21589, Kingdom of Saudi Arabia

Abstract—**Device-to-device communication is popular research trend that presents ubiquitous information exchange on the Internet of Things. D2D communication provides data exchange without transiting to a base station using direct communication between two devices. For such environment, successful delivery of data to the receiver is needed. In this paper, we suggest a Bee-Inspired Routing Algorithm (BIRA) for D2D communication in IoT exploits the multiple interfaces of a "thing" in IoT having different wireless standards. BIRA is on demand routing algorithm simulates the bee's foraging behavior model to find optimal path between source and destination for multi-hop communication. The performance of BIRA is assessed through extensive simulations that concludes BIRA realizes better packet delivery ratio as well as it performs lower average end-to-end delay in different traffic load compared to the conventional AODV protocol. Also, BIRA achieves least energy consumption than AODV and increases network lifetime.**

Keywords—*Device-to-device communication; internet-of-things; Bee-Inspire Algorithm; routing protocol*

I. INTRODUCTION

The Internet of Things (IoT) is ubiquitous computing that introduces the concept of connecting several physical objects to generate, share and consume information with minimal human intervention. It changes traditional system to be smart and available at any time and position [1], [2]. In such environment, different type of devices can collaborate with each other to ensure quality communication and to make more efficient utilization of information. With the increasing demands for diverse applications as well as boost in micro-electro-mechanical systems, the vast proliferation has been observed in the use of these devices (also known as "things") which demands more spectrums supplementary to the radio frequency (RF) for future generation. The Device-To-Device Communication (D2D) is one of the communication models that is implemented in IoT ecosystem which applies direct connection between devices to exchange data in a distributed fashion without traversing a core network. D2D communication is an emerging research phenomenon which benefits ultra-low latency [3] in communication due to a shorter signal traversal path. D2D communication architecture consists of heterogenous computing devices with different capabilities connected together [4], [5]. In addition, due to the limited energy of wireless sender and receiver, it is required for the devices to transfer data through multihop communication.

However, devices characteristics in such network are constrained in terms of battery life and memory size [6]. Due to the device constraints, User Equipments (UEs) should be connected using different short-range wireless networks such as wireless sensor networks (WSNs), wireless fidelity (WiFi), Bluetooth, radio-frequency identification (RFID) networks, and ZigBee [7]. A routing protocol considering these limitations is required to provide efficient data delivery in D2D network.

To date, a number of researches are presented that use bio-inspired algorithm to decrease memory consumption with effective data delivery [8-21]. Biology inspired algorithm is a class of algorithm that mimic specific phenomena from nature [7], and it used to solve problems which resembles the way nature performs. D2D communication in IoT network possesses some unique challenges including autonomous and complex architecture, non-homogeneous nature of devices, resource constraints and distributed control and infrastructure. We argue that routing protocols that adopt bio-inspired algorithm can effectively address these challenges since this genre of algorithms have distinctive features as follows. First, they are appropriate for optimization processes in terms of throughput, energy consumption, and packet delivery ratio. Second, they can handle the heterogeneity of devices and communication technologies in D2D communications. Third, they support device collaboration and self-configuration. Fourth, they have the ability to repair failures. Finally, they provide efficient management for limited resources.

In [22], the authors presented the process of adopting bio-inspired engineering as shown in Fig. 1. First, analogies need to be pointed out between biological and technical systems such as Information and Communication Technology (ICT). Second, model to be created for realistic biological behavior to use it later for developing a technical solution. Finally, simplify and tune the model for the technical application.

Bio-inspired algorithms can be categorized into Swarm Intelligence (SI), Evolutionary algorithms (EAs), and bacterial foraging. Swarm Intelligence (SI) algorithm is one of the bio-inspired methods that is inspired by the collective behavior of distributed and self-organized models e.g. bees swarm. Routing protocol that adopts bee-inspired principle utilizes collective foraging behavior to find source of food. as discussed in detail in Section III.
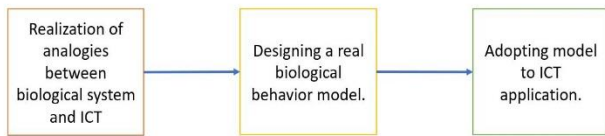
Fig. 1.   Bio-Inspired Engineering.

State-of-the-art research efforts are introduced to present a routing protocol inspired by several biological models in the IoT ecosystem. In [11-14], energy-aware ant routing algorithms are proposed based on ant colonies foraging behavior in network. In WSN, many artificial bee colony-based algorithms are proposed to minimize energy consumption [16-18]. However, these protocols mainly optimized energy efficiency and designed in the context of WSNs, and are not directly applicable to D2D communication in the IoT environment due to its unique features. Several studies exist that use Ant Colony method to meet some D2D challenges the IoT framework [19-21]. However, the aforementioned studies are not suitable for D2D communication in IoT environment as they did not address the heterogeneity of devices along with the coexistence of different wireless standards. In this study, an energy-aware Bee-Inspired Routing Algorithm (BIRA) for D2D communication in IoT is introduced.

BIRA is a reactive routing protocol that is inspired by bee's behavior for searching the food source. BIRA is an on-demand, unicast, self-configured and distributed routing approach. Considering the existence of multiples interfaces in today's devices, BIRA attempts to find optimal path between a source and a destination utilizing multiple interfaces having different wireless standards. In routing discovery, BIRA chooses selective neighbors having acceptable residual energy to decrease the number of control packets. It generates a two types of control packet for route discovery namely, forward and backward scout packets and delivers data packet to destination using foragers. As a routing metric, expected transmission count (ETX) is utilized to measure link cost of different type of Wi-Fi communication. BIRA is simulated using ns3 [23] to assess the efficiency and compare the performance of BIRA with the traditional AODV protocol.

The paper is outlined as follows: Section II introduce the related works. The overview of Bee inspired algorithm in biological system is presented in Section III. The protocol description is discussed in detail in Section IV. Section V explains simulation results, and finally Section VI concludes the paper.

II.   RELATED WORK

As of now, a number of studies presented routing protocols in D2D communication [8-10]. Laha A. et al. proposed a quasi-group routing protocol (QGRP), a distributed cluster-based routing scheme focusing energy efficiency [8]. In [9], the authors presented a D2D routing protocol to avoid interference to cellular users. Their attempt aimed to minimize delay, hop count, power consumption, and computational complexity. Utilizing location information, Park J. proposed fast and energy-efficient multihop D2D routing algorithm focusing to improve the end-to-end packet delivery ratio while increasing

network lifetime [10]. However, all aforementioned approaches are devised considering smartphone networks, and not appropriately applicable in IoT environment. Also, these protocols cannot provide efficient packet delivery.

Biology inspired algorithm exploits the behavior from nature to solve problems [7]. Ant colony optimization (ACO) attempts to devise solutions motivated by the self-organizing collective behavior of social ant colonies [7]. In [11], an Energy-aware ant routing algorithm (EARA) is proposed utilizing the foraging behavior of ants in multihop communication. EARA introduced energy information with the pheromone value in the route discovery phase and performed energy information maintenance process to update energy values in the routing table in case of change the residual energy of node. The results revealed that EARA performed better than ant routing algorithm. Energy-aware routing protocol and gradient-based routing (GBR) protocol also exploits ACO [12] to minimize energy consumption. Some solutions presented schemes to address mobility, energy consumption, self-optimization and reliability through modifying the traditional AC technique [13], [14]. Enhanced Ant Colony algorithm (EAC) is proposed focusing to improve reliability [13]. The proposed algorithm categorized the node states into active node, sleep node, and critical node. Here, nodes having less residual energy are considered as critical nodes. The path discovery is done before critical packet transmission. The scheme shows negligible effect on energy consumption than AODV. In [14], Bio-inspired optimization for sensor network (BiO4SeL) is presented aiming to reduce energy consumption of sensor nodes. It uses battery power information to create and update the routing table in the case of mobility. However, the approach ignored other performance metrics such as delay and packet delivery ratio. In [15], Ismail et al. presented a modified version of AODV for low-power wireless personal area networks (6LoWPAN). They intended to repair the link failure by proposing a local repair using bio-inspired artificial bee colony routing protocol. The protocol is inspired from the bee foraging process to find the food source. The study can optimize the route performance, decrease route maintenance delay and save energy. However, the authors did not present detailed study including implementation. Few other studies are proposed to minimize energy consumption in WSN exploiting artificial bee colony algorithm [16-18]. Kumar R. et al. proposed artificial bee colony based energy-efficient clustering mechanism to improve network lifetime and cluster head selection [16]. The proposed algorithm is designed to find an optimal shortest path with less energy consumption. It was analyzed and compared with LEACH and PSO and ABC-based routing protocol. It concluded that the proposed method could maximize the network lifetime. Okdem S. et al. provided performance testing and complexity analysis of Cluster-based Wireless Sensor Network routing protocol utilizing artificial bee colony algorithm (CWA) [17]. CWA method consists of four main steps including initialization, setup, broadcast setup configuration, and data gathering. The performance test result shows that the CWA algorithm enhances the network lifetime by saving more energy. In [18], Zheng W. et al. proposed intelligent routing protocol based on ABC algorithm to solve delay-energy trade-off challenge in WSNs. They concluded that the algorithm exhibits good performance in balancing

power consumption, maximizing network lifetime, and optimize delivery ratio. However, these algorithms mainly optimized energy efficiency and designed in the context of WSNs and cannot be applied directly to D2D communication in the IoT environment due to its unique features.

In the IoT environment, few studies exist exploiting AC method to address some D2D characteristics [19-21]. Y. Lu et al. proposed routing algorithm model to address the mobility of IoT network topology as well as the scalability [19]. They concluded that the time for establishing route decreased with the increasing number of nodes and it reduced broadcast storm. Another routing scheme is proposed using AC algorithm considering the node mobility, self-organizing network and node energy [20]. It reduced the time for establishing a route with the increasing number of nodes, and decreased broadcast storm. In [21], researchers presented an ant colony foraging routing scheme utilizing Markov decision model to find an optimal route. The study reduced network overhead and energy consumption while increasing network lifetime. However, the proposed algorithms are not suitable for D2D communication in IoT environment as they did not address device heterogeneity as well as the coexistence of different wireless standards.

III. BEE COLONY OPTIMIZATION ALGORITHM: AN OVERVIEW

This section presents detailed overview of bee's foraging principle in biological system and how it is tuned to a realistic biological model in routing protocol.

A. Bee Algorithm in Biology

Artificial bee colony algorithm is a type of swarm intelligence models that simulates intelligent foraging behavior of bees. In such algorithm, food source searching is the main uses of bee communication [24]. To perform a food source searching process, the model has several types of bee groups: scouts and foragers. In particular, the bee foraging behavior begins with the "scouts". Scout bees fly and explore the beehive neighborhood to discover a food source. They carry out a random search with n dimensions in the area. Moreover, they are able to fly around to a radius of up to three kilometers from the hive to discover the food sources. Whenever food source is discovered, the scout bees return to the hive with their information to inform other bees about detected food source using dance language. This information related to the quality of food source is exchanged among bees to build collective knowledge through "waggle dance" in the dance area. In bee colony algorithm, a possible solution to the problem is represented by the position of a food source and the quality of the related solution is corresponded to the nectar amount of a food source. Some bee "foragers" obtain the collected information and associate to a particular food source to start forage. The number of foragers is proportional to the quantity of food information shared by the scouts with their nectars. Foragers assess the nectar information taken from all scouts and selects a food source with a probability related to its nectar amount according to (1).

$$P_a = \frac{fit_a}{\sum_{n=1}^{SN} fit_n} \qquad (1)$$

where, fit_a is the fitness rate of individual in a population and SN is the number of individuals in the population.

This step is termed as exploration phase followed by the exploitation step [25]. Here, the forager bee collects food and estimates its quantity to make a new decision. The forager either memorizes the food source location or marks it as exhausted food source and returns to the hive as a scout to start finding a new food source [26][27].

B. Bee Inspired Routing Algorithm

In Bee inspired routing protocol, the mapping of bee behavior to routing function is achieved as per follows: Source node in the network is considered as beehive that consists of forward scouts that can work exactly as real forward scout do to discover food source which is represented as destination node. Relay node has both forward and backward scouts which are sent to assist beehive to find path to a food source. The forward/backward scouts provide the neighbors with information about the route they discovered. A routing table represents the dance floor where the forward/backward scouts deliver the information about the path quality. Application data packet is considered as forager which access the information of routing table to deliver the data in efficient way.

IV. PROTOCOL DESCRIPTION

BIRA is inspired by bee colony algorithm in which source node is represented as beehive and the food source resembles destination node. In such model, relay nodes are simulated as worker bees which are neither in beehive nor on the food, called workers. Worker bees might be aware the path to the food at one point. BIRA is an on-demand hop-by-hop routing algorithm. Each node maintains two tables namely routing table and neighbors table. The proposed protocol stores only next-hop and previous-hop in node's routing table rather caching the whole path. The following subsections discuss the protocol operations in detail including packet types, neighbor discovery, route discovery and route maintenance.

A. Routing Packet Types

Forward scout: Forward scout is used to discover the path toward the destination on-demand. This packet consists of some fields initialized by source node as it is shown in Fig. 2. It includes scout identifier (Scout ID) which is a unique incremental value. Scout ID is used to ensure the uniqueness of the route to be discovered. It is generated by source node only and its value is unchangeable. Moreover, Forward scout packet includes Beehive identifier (Beehive ID) that represents the source node ID, and food ID referred to as destination ID. Beehive ID and food ID cannot be manipulated by other nodes. The combination of scout ID and beehive ID preserves the uniqueness of the route request.The hop-count field caches the hop count the scout traversed from the source node to the node it currently passes. Next-hop ID refers to the address of the next neighbor and it changes from hop-to-hop. Expected Transmission Count (ETX) is used to store the path cost information whenever the scout traverses toward destination, the value of ETX is accumulated from source node to the node it current passes. Estimation of ETX is explained in Section IV-b.

Forward scout packet

Scout ID	Beehive ID	Food ID	Hop count	Next-hop ID	ETX

Fig. 2. Packet Structure of Forward Scout.

a) Backward scout: Once the path is discovered, the destination node propagates the scout as backward scout packet toward the source node along the reverse path.

Similar to forward scout, the backward scout uses the same field value for scout ID, the beehive ID, and the food identifier as shown in Fig. 3. Backward scout uses the hop count field to refer the number of hops from the source node to the destination. When the desired route is found, this field is initialized using the hop-count field of the forward scout. Through the ETX field, the source node knows about the path cost to the destination. Upon receiving the forward scout, the Destination node initiates a unique path identification, and embed the value in the path ID field.

b) Forager: Application data is delivered to the destination using forager packet. Forager packet includes several fields which help to direct the waiting data packets to the intended node. Forager ID is a unique incremental value generated by beehive. The Beehive, Food ID, Hop-count and Path ID represents the same as scout packets. The Path ID helps intermediate node to forward the forager packet to the next hop associated with that particular path ID stored in the routing table. Data field carries the application data sent by source node to intended node. Forager packet structure is shown in Fig. 4.

B. Neighbor Discovery

In BIRA, every node broadcasts a periodic refresh packet to update its immediate neighbors about the active connections. The refresh packet also includes the residual energy of the node. To periodically measure the link quality, each node unicast a "probe" packet to its "eligible" neighbors through multiple interfaces (in our scheme we consider two interfaces) to maintain the link quality of the neighbors. We define the "eligible" neighbors are the nodes for which residual energy exceeds some threshold. Thus, BIRA preserves the information of energy efficient neighbors in its neighbor table in a distributed fashion.

Backward scout packet

Scout ID	Beehive ID	Food ID	Hop count	Path ID	ETX

Fig. 3. Packet Structure of Backward Scout.

Forager packet

Forager ID	Beehive ID	Food ID	Hop count	Path ID	Data

Fig. 4. Packet Structure of Forager.

C. Link Cost Estimation

BIRA exploits ETX as a link cost metric for the eligible neighbors, and stores the value in the neighbor table. ETX is defined as the average number of transmissions including retransmissions a node expects to make to successfully forward a packet to the destination [28]. The successful delivery is guaranteed through the reception of link-layer acknowledgment. ETX over the link (a_i,a_j) for node a_i can be estimated as:

$$ETX_{a_i} = \frac{N_{Tx}^{a_i,a_j}}{N_{ACK_{rx}}^{a_i,a_j}} \tag{2}$$

Where, $N_{Tx}^{a_i,a_j}$ denotes the total number of transmissions (including retransmission) from a_i to a_j , and $N_{ACK_{rx}}^{a_i,a_j}$ designates total number of ACK receptions by node a_i from a_j which also signifies the total number of successful transmission from a_i to a_j.

BIRA utilizes Exponential weighted moving averages (EWMA) for the smooth estimation of ETX_{a_i}, to address the abrupt changes in link condition. The running average of this ETX is estimated as:

$$ETX_{tr}^{a_i} = \alpha \left(ETX_{tr}^{a_i}(curr) \right) + (1 - \alpha)ETX_{tr}^{a_i} \tag{3}$$

Where, $ETX_{tr}^{a_i}(curr)$ refers to the current observation of ETX and α is the tuning parameter that satisfy $0 < \alpha < 1$.

D. Route Discovery

When a node requires to transmit data, it first checks the availability of a valid path information of the destination in its routing table. If a valid route is found with sufficient foragers, the source node forwards the data. Otherwise, it launches the route discovery procedure. Here, the source node generates a forward scout packet with unique scout ID, and replicates a number of forward scouts with the same scout ID and transmits to its eligible neighbors. The number of forward scouts depends on its eligible neighbors. The forward scout is transmitted through the interface having minimum ETX value. Thus the path traversed by forward scout toward the destination node comprises different interfaces with optimal link cost as illustrated in Fig. 5. Forward scout marks temporarily its path in the routing tables of the visited nodes which will be utilized later for the return packet (i.e., backward scout).
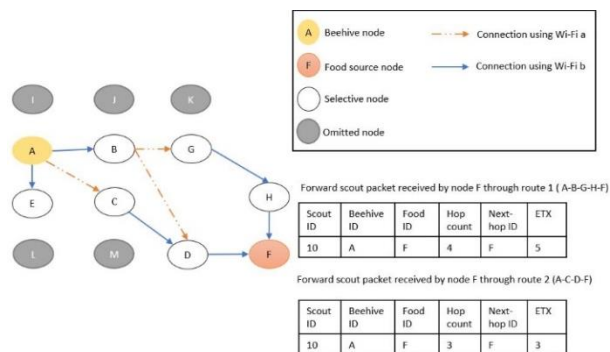

Fig. 5. Forward Scout Delivery.

When the forward scout passes an eligible neighbor, the ETX value of the traversed link is accumulated in the ETX field of the forward scout. The intermediate node verifies whether it has already received a forward scout with the same scout ID and beehive ID. In this case, the intermediate node compares the ETX value of the currently received forward scout packet with the previous one and drops the current forward scout if its ETX value exceeds than that of the previous one. Afterwards, the node forwards the forward scout packet to its own eligible neighbor following the same manner toward the Beehive. Every intermediate node updates its routing table fields whenever it received scout packets. Fig. 5 illustrates how node A finds a route to node F if it doesn't have a valid route. Node A creates forward scouts and transmits to only selective nodes B, C and E based on acceptable percentage of remaining energy. Then node S communicates with selective nodes through optimal interface. Node A communicates with node B and E through wi-fi b link, but it communicates with node C through wi-fi a interface since the selected links have lower link cost. After receiving forward scout, node B, C and E check if the received forward scout is duplicate or not and then update the routing table and change hop count and ETX fields of the received packet, and forward it in a similar way as done by node A. In this example, node D receives duplicated forward scout packet; one from node C and the other from node B. But node D drops the forward scout packet from node B as it has higher ETX value than that of node C. Node C communicates with its neighbor node D through Wi-Fi b link while node B communicates with its neighbors, node G and D through Wi-Fi a link. Here, node F which is the food source (i.e., destination) receives forward scout packets from both node H and D through path A-B-G-H and A-C-D respectively as shown in Fig. 5.

Upon receiving a forward scout, the intermediate node may generate a backward scout and transmits it along the reverse path toward the source node, if the path information is available in its routing table. This is done by looking at the previous-hop field at the routing table. Otherwise, the node continues forwarding the forward scout. When the forward scout packet reaches to the destination, a backward scout packet is created and transmitted along the reverse path to the source node. When a node receives backward scout, it permanently stores the routing information, and utilizes the next hop field to determine a path to the destination. After the arrival of the backward scout at the beehive node, the path cost is retrieved from the ETX field of backward scout packet, and is stored as a routing table entry. As mentioned earlier, the destination node initializes the Path ID in the backward scout packet to indicate the traversed route. Fig. 6 shows the delivery of backward scout at the beehive node which is node A. Backward scout packets generated by node F are sent along reverse path through similar interfaces the forward scout has been received. Node A received two backward scout packets from different path (F-H-G-B and F-D-C) as depicted in Fig. 6. Node A updates ETX field of received backward scout and records the information carried in routing table. Eventually, the beehive node recruits forager to deliver data to the destination as depicted in Fig. 7. Node A select optimal path based on ETX value stored in the routing table. It selects path 2 because the ETX value is 3 which is less than path 1. Using path 2, the

forager packet is forwarded from node A to node C through wi-fi an interface then node C communicates with node D through wi-fi b link and forward the received forager packet. Also, node D sends the forager to the intended node F through Wi-Fi b interface.

The flowcharts of BIRA in routing discovery process for Beehive, intermediate node and food source are illustrated in Fig. 8, 9 and 10, respectively.

E. Route Maintenance

In order to maintain route, BIRA utilizes two types of timer: route validity timer and route request waiting timer. After transmitting the forward scout the beehive node waits for the timeout period of route request waiting timer and retransmits the forward scout after timeout occurs. On the other hand, a node purges the expired entry from the routing table upon the expiry of route validity timer.
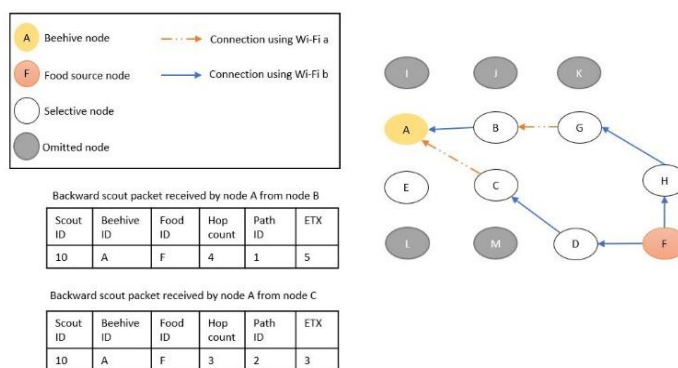

Backward scout packet received by node A from node B

Scout ID	Beehive ID	Food ID	Hop count	Path ID	ETX
10	A	F	4	1	5

Backward scout packet received by node A from node C

Scout ID	Beehive ID	Food ID	Hop count	Path ID	ETX
10	A	F	3	2	3

Fig. 6.   Backward Scout Delivery.

Forager packet received by node F through path 2

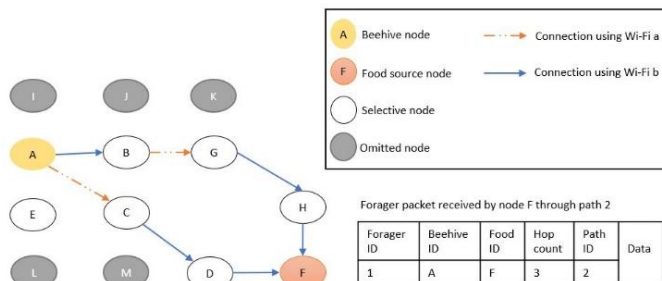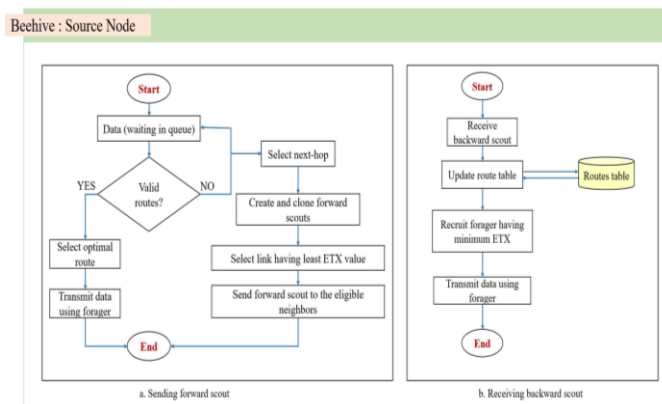| Forager ID	Beehive ID	Food ID	Hop count	Path ID	Data
1	A	F	3	2	

Fig. 7.   Data Delivery using Forager.

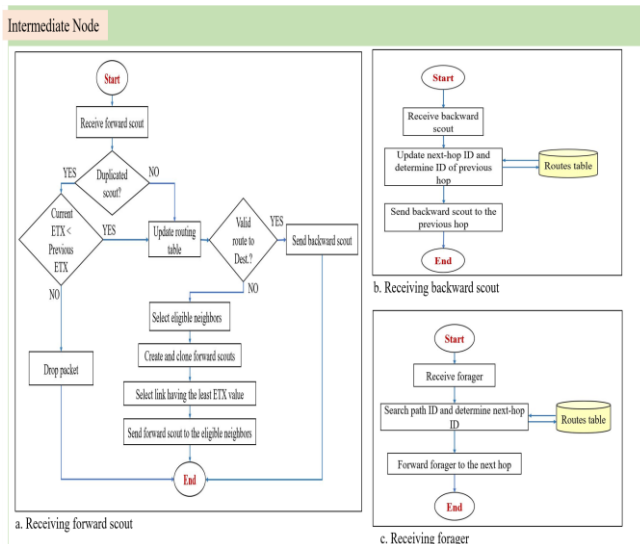Fig. 8.   Beehive's Route Discovery Flowchart.

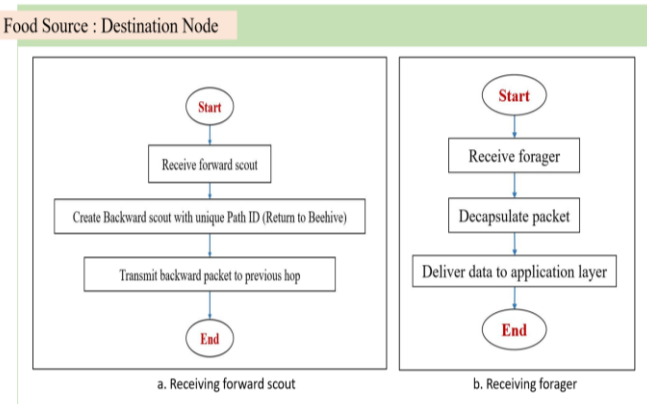Fig. 9. Intermediate Node's Route Discovery Flowchart.

Fig. 10. Food Source's Route Discovery Flowchart.

In an IoT network, a node broadcast periodic refresh packet to know about the active neighbors. A node marks the routing table entry of its neighbor as invalid if it does not receive the refresh packet from its neighbor for a specified period of time. Then, an error scout packet is transmitted to notify the other nodes regarding the broken link. Upon receiving the error scout packet, all the nodes eventually delete the corresponding routing table entry from their routing table. The beehive node might initiate the route discovery procedure again if required.

V. PERFORMANCE EVALUATION

In this section, the performance of BIRA is assessed and compared its outcomes with AODV using simulation.

A. Simulation Parameters

BIRA is simulated using NS3 version 3.28.1 on the Ubuntu 16.04 LTS Linux Operating System. Varying number of nodes are deployed in a 1000x1000 m² area. The simulation is carried out for 800 seconds. Two MAC layer protocols are implemented in each node with two different interfaces: IEEE 802.11a and IEEE 802.11b. In the simulation scenarios, a log distance propagation model is utilized and Constant Bit Rate

generators (CBR) is used over (User Datagram Protocol) UDP in the application layer with different traffic rate (packet/sec). Varying number of source and destination pairs is used in different experiments of the simulation. The payload size that considered is 512 bytes and initial energy of node is set to 100 joules. In each trial, the outcomes are acquired by taking the average over 10 random runs. Table I depicts the simulation parameters. The performance of the protocol is evaluated using three metrics: average end-to-end delay, packet delivery ratio (PDR) and average energy consumption. The performance of BIRA is compared with AODV, one of the most widely used routing protocols for IoT. To comprehensively assess the performance of BIRA, we conduct three types of experiment. In the first experiment, the results of all performance metrics are obtained using different numbers of nodes range from 10 to 50 nodes with one source/destination pair and 1packet/sec application traffic rate. Second experiment results are extracted from 50 nodes and source/destination pairs vary from 1 to 5 pairs with 1 packet/sec application traffic rate. In the last experiment, different traffic rate is implemented ranging from 2 to 10 packets/sec with fixed 50 nodes and 4 source/destination pairs.

B. Performance Metrics

Three performance parameters have been used in the experiment to analyze the performance of BIRA protocol [29]. These metrics are explained as follows:

- Average end-to-end delay comprises all possible delays including route discovery latency, queuing at the interface queue, retransmission delays at the MAC, propagation and transmission delay. It is measured from the time a packet is queued to the time the acknowledgment of the packet is received.

- Packet delivery ratio (PDR) is measured by dividing the number of received packets at the destination node to the number of packets created by the source nodes.

- Average energy consumption parameter of nodes which caused by event transmitting, receiving, checking the channel for incoming transmission and idle. It can be measured as follows [30].

$$E_{rx} = P_{rx} * d_{size} / S \tag{4}$$

$$E_{tx} = P_{tx} * d_{size} / S \tag{5}$$

$$E_{min} = E_{check} + E_{idle} \tag{6}$$

$$E_{cons} = E_{min} + d_{rx} * E_{rx} + d_{tx} * E_{tx} \tag{7}$$

$$E_{re} = E_{init} - E_{cons} \tag{8}$$

Where E_{tx} is energy consumed to transmit a data packet (J), P_{tx} is Tx power (W), E_{rx} is energy consumed to receive a data packet (J), P_{rx} is Rx power (W), d_{size} is data packet size (bit), d_{tx} is number of data packet transmitted (packet), d_{rx} is number of data packet received (packet), S is communication speed (bits per second), E_{min} is minimum consumed energy (J), E_{check} is energy for checking channel (J), E_{idle} is energy in idle time (J), E_{cons} is total energy consumed (J), E_{init} is initial energy (J) and E_{re} is remaining energy (J).

TABLE. I.   SIMULATION CONFIGURATIONS

Parameter	Value
Routing protocols	BIRA and AODV
Payload size	512 bytes
Simulation time	800 sec
Application data traffic	CBR
Application traffic rate	2,4,6,8 and10 packet/sec
Wireless network	Wi-Fi IEEE802.11
MAC layer protocols	IEEE 802.11a and IEEE802.11b
Propagation loss model	Log distance propagation model
Number of nodes	10,20,30,40 and50 nodes
Number of source and destination pairs	1,2,3,4 and 5
Initial energy	100 Joule

C. Experimental Results

Packet delivery Ratio: As shown in Fig. 11, 12 and 13, BIRA achieves better PDR than AODV in all the three experiments. Fig. 11 shows that BIRA deliver data successfully ranging from 89% to 100% packets but the value for AODV ranges from 75% to approximately 91%. Although the experiment is conducted having 1 source/destination pair, but with the increasing number of nodes, the PDR slightly decreases for BIRA as shown in Fig. 11. This is due to the fact that, in larger network size, the hop-count increases, and packet retransmissions also increases due to varying link quality for each hop. In the first experiment, it is observed that the PDR difference in both the protocol is about 14%. Due to pure flooding of route request packet in AODV, it shows poorer performance as compared to BIRA. BIRA achieved better PDR because it restricts the number of forward scouts which speeds up the convergence.

Despite increasing sending packets from different pairs of source and destination as illustrated in Fig. 12, BIRA outperforms significantly AODV because it unicasts forward scout to selective neighbors and communicates with them through interface having the least ETX value. Fig. 13 shows the delivery ratio of both protocols while increasing the number of packets per second; in the best case, AODV attains 94% PDR at 6 packets/sec traffic rate while BIRA, at that traffic rate, shows 99% PDR, and the best performance of BIRA achieves at 2 packets/sec traffic rate which is 100% PDR.
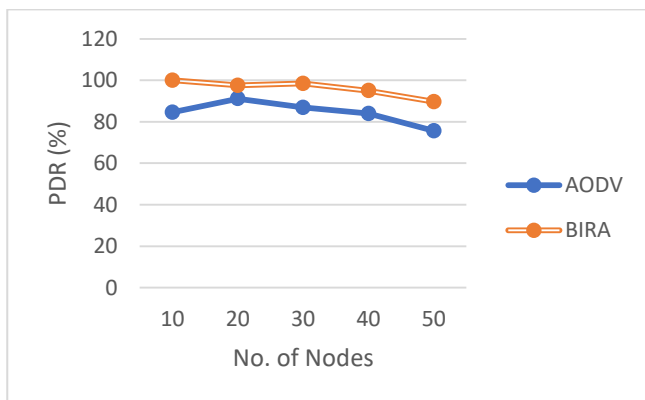

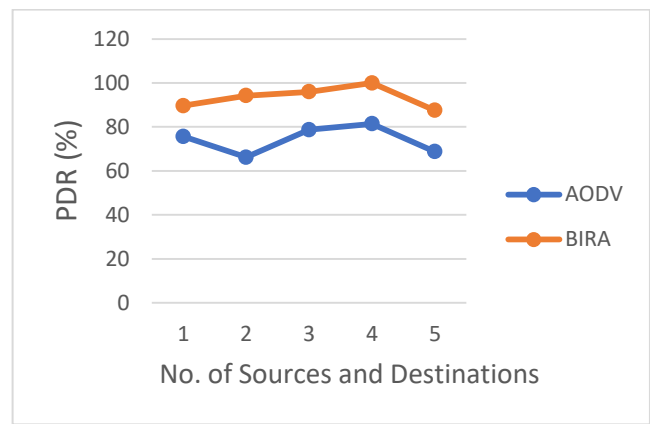Fig. 11. Packet Delivery Ratio of Varied no. of Nodes.

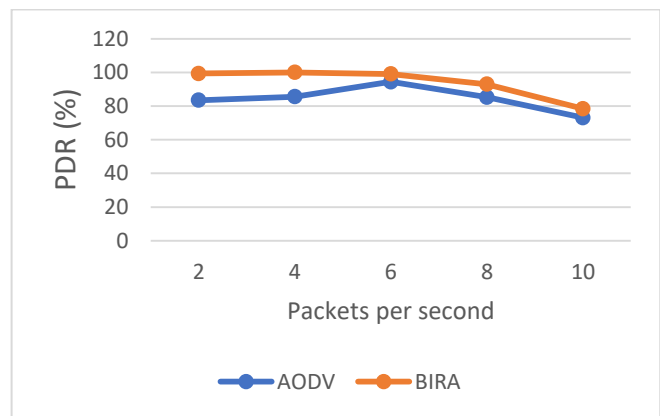Fig. 12. Packet Delivery Ratio of different Source/Destination Pairs.

Fig. 13. Packet Delivery Ratio of different Traffic Rate.

a) End-to-End Delay: Fig. 14, 15 and 16 illustrate that the BIRA can deliver the data to the destination faster than AODV. As shown in Fig. 14, the average end-to-end delay increases as the number of nodes increases due to the increase in hop-counts, and BIRA outperforms AODV in all cases. This is self-evident since, BIRA chooses ETX as routing metric as well as select the interface having optimal link cost. Hence, the total path cost of BIRA is always lower than AODV in terms of end-to-end delay. Moreover, BIRA does not drop all duplicated forward scout packet rather it drops only if duplicated packet has highest ETX value than the earlier one and thus chooses a route having lower path cost which in turn reduces the end-to-end latency. The better delay performance of BIRA is also observed in second experiment with varying source/destination pairs. When the number of source/destination pairs are 5 (i.e., more sources are generating traffic), the end-to-end latency of AODV is around 0.06 seconds while BIRA achieves a delay of 0.03 seconds as illustrated in Fig. 15. BIRA also outperforms AODV in varying traffic load as shown in Fig. 16. In all the traffic loads, AODV shows poorer delay performance than BIRA since AODV routes the packet depending on hop count metric and does not exploit the heterogeneity of different wireless standards for choosing optimal route.

b) Energy Consumption: The energy consumption performance of BIRA is evaluated in all the three experiments

as illustrated in Fig. 17, 18 and 19. Fig. 17 indicates that BIRA is energy efficient and can increase the network lifetime as compared to AODV in different network size. This is because, BIRA always chooses eligible neighbors having higher residual energy. Moreover, it selects the links with least ETX value that signifies reduced retransmissions which in turn minimizes energy consumption. The lower energy consumption of BIRA has also been observed while varying source/destination pairs as depicted in Fig. 18, and with different traffic rate as shown in Fig. 19. AODV has poor energy performance than BIRA because it consumes a lot of power when broadcasting route request packets in the route discovery phase, also AODV cannot choose link with lower retransmission value.
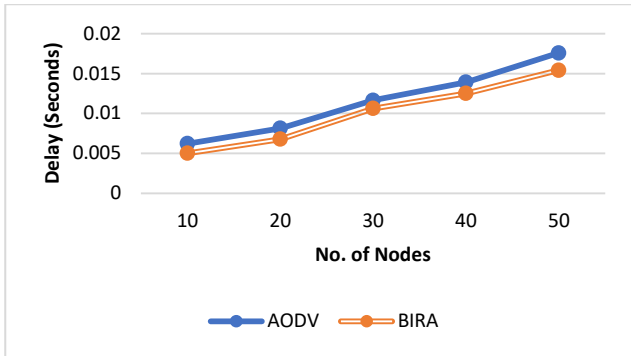

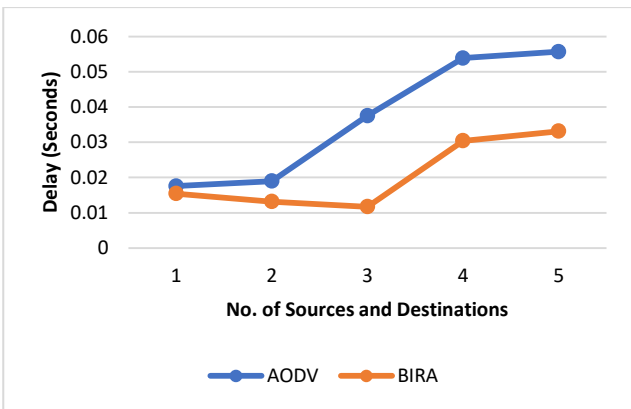Fig. 14. Average End-to-End Delay of different no. of Nodes.

Fig. 15. Average End-to-End Delay of different no. Source/Destination Pairs.
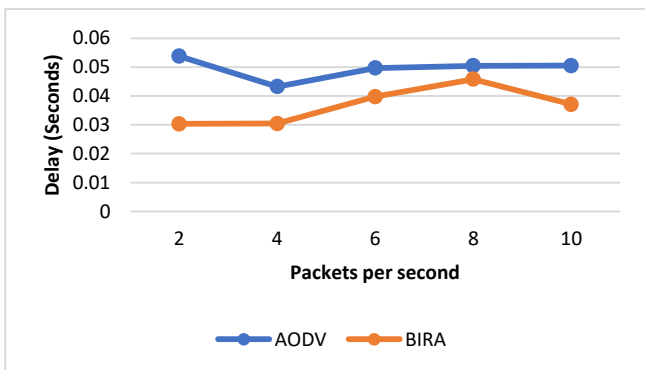

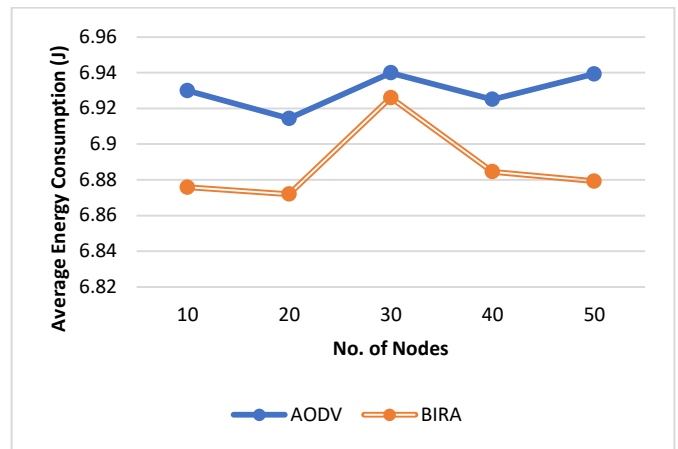Fig. 16. Average End-to-End Delay of different Traffic Rate.

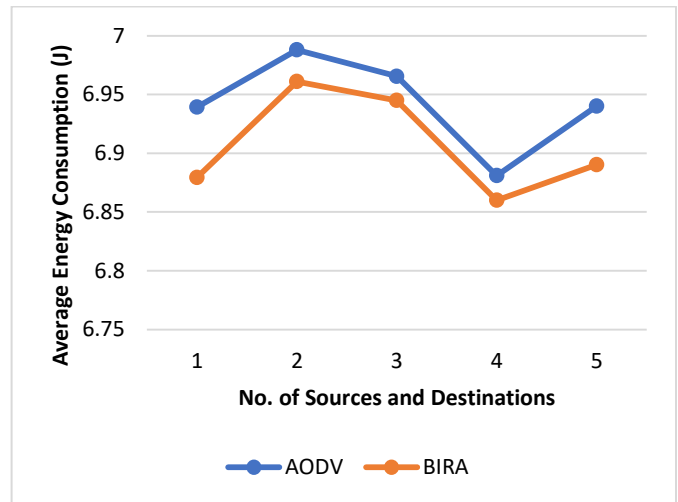Fig. 17. Average Energy Consumption of different no. of Nodes.

Fig. 18. Average Energy Consumption of different no. of Source/Destination Pairs.
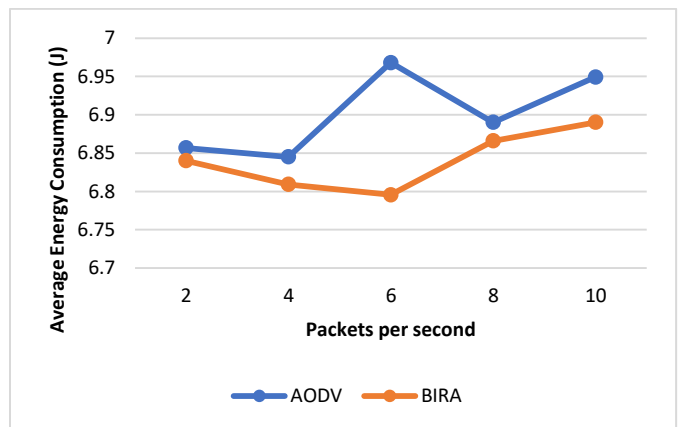

Fig. 19. Average Energy Consumption of different Traffic Rate.

VI. CONCLUSION

In this paper, we introduced BIRA routing protocol inspired from the bee communication principle to meet the challenges of Device-to-device communication in IoT environment. BIRA is an on-demand hop-by-hop routing

algorithm that exploits the multiple interfaces of a node having different wireless standards to find an optimal path between a source and destination. The performance of BIRA is realized through extensive simulations using ns-3 simulator. The outcomes deduced that as compared to AODV, the BIRA obtains least energy consumption and longer network lifetime. In addition, BIRA performs better packet delivery ratio than AODV as well as it achieves lower average end-to-end delay in different traffic loads. In future work, authors aim to consider dynamic network topology and interference.

REFERENCES

[1] Majeed, "Internet of Things (IoT): A verification framework," in 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–3, 2017,.

[2] P. Barnaghi and A. Sheth, "On searching the internet of things: Requirements and challenges," IEEE Intell. Syst., vol. 31, no. 6, pp. 71–75, 2016.

[3] R. M. Mare, C. Luiz Marte, and C. E. Cugnasca, "Visible Light Communication Applied to Intelligent Transport Systems: An Overview," IEEE Lat. Am. Trans., vol. 14, no. 7, pp. 2101–2106, 2016.

[4] F. Deicke, W. Fisher, and M. Faulwasser, "Optical wireless communication to eco-system," Futur. Netw. Mob. Submit, pp. 1–8, 2012.

[5] P. E. Hockberger, "A history of ultraviolet photobiology for humans, animals and microorganisms.," Photochem.Photobiol., vol. 76, no. 6, pp. 561–579, 2002.

[6] M. Frey, F. Grose, and M. Gunes, "Energy-aware Ant Routing in Wireless Multi-hop Networks", IEEE International Conference on Communications (ICC), pp. 190-196, 2014.

[7] O. Bello and S. Zeadally, "Intelligent Device-to-Device Communication in the Internet of Things," IEEE Systems Journal, vol. 10, no. 3, pp. 1172–1182, Sep. 2016.

[8] A. Laha, X. Cao, W. Shen, X. Tian, and Y. Cheng, "An energy efficient routing protocol for device-to-device based multihop smartphone networks," in Communications (ICC), 2015 IEEE International Conference on, 2015, pp. 5448–5453, 2015.

[9] P. Ren, Q. Du, and L. Sun, "Interference-aware routing for hop-count minimization in wireless D2D networks," in Communications in China-Workshops (CIC/ICCC), 2013 IEEE/CIC International Conference, pp. 65–70, 2013.

[10] J. Park, "Fast and Energy Efficient Multihop D2D Routing Scheme," International Journal of Distributed Sensor Networks, vol. 12, no. 5, p. 2148734, May 2016.

[11] M. Frey, F. Grose, and M. Gunes, "Energy-aware Ant Routing in Wireless Multi-hop Networks", IEEE International Conference on Communications (ICC), pp. 190-196, 2014.

[12] R. Arya and S. C. Sharma, "Analysis and Optimization of Energy of Sensor Node Using ACO in Wireless Sensor Network," Procedia Computer Science, vol. 45, pp. 681–686, 2015.

[13] M. Umadevi and M. Devapriya, "An Enhanced Ant Colony Based Approach to Optimize the Usage of Critical Node in Wireless Sensor Networks," Procedia Computer Science, vol. 47, pp. 452–459, 2015.

[14] M. F. de Castro, L. B. Ribeiro, and C. H. S. Oliveira, "An autonomic bio-inspired algorithm for wireless sensor network self-organization and efficient routing," Journal of Network and Computer Applications, vol. 35, no. 6, pp. 2003–2015, Nov. 2012.

[15] N. H. A. Ismail and R. Hassan, "6LoWPAN Local Repair Using Bio Inspired Artificial Bee Colony Routing Protocol," Procedia Technology, vol. 11, pp. 281–287, 2013.

[16] R. Kumar and D. Kumar, "Multi-objective fractional artificial bee colony algorithm to energy aware routing protocol in wireless sensor network," Wireless Networks, vol. 22, no. 5, pp. 1461–1474, Jul. 2016.

[17] S. Okdem, D. Karaboga, and C. Ozturk, "An application of wireless sensor network routing based on artificial bee colony algorithm," in Evolutionary Computation (CEC), 2011 IEEE Congress on, 2011, pp. 326–330, 2011.

[18] W. Zheng and D. Luo, "Routing in Wireless Sensor Network Using Artificial Bee Colony Algorithm," pp. 280–284, 2014.

[19] Y. Lu and W. Hu, "A Research on the dynamic routing of Internet of Things Based on Ant Colony Algorithm," The Proceeding of 2nd International Conference on Computer and Applications (CCA), 2013.

[20] Y. Lu and W. Hu, "Study on the application of ant colony algorithm in the route of internet of things," International Journal of Smart Home, vol. 7, no. 3, pp. 365–374, 2013.

[21] C. Cheng, Z. Qian, and G. Ji, "An IoT Ant Colony Foraging Routing Algorithm Based on Markov Decision Model," The Proceeding of International Conference on Soft Computing in Information Communication Technology (SCICT), 2014.

[22] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," Computer Networks, vol. 54, no. 6, pp. 881–900, Apr. 2010.

[23] "ns-3", ns-3, 2019. [Online]. Available: https://www.nsnam.org/. [Accessed: 21- Aug- 2019].

[24] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, "A comprehensive survey: Artificial bee colony (ABC) algorithm and applications", Artif. Intell. Rev., pp. 1-37, 2012.

[25] H. F. Wedde et al., "BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior," in Proceedings of the 2005 conference on Genetic and evolutionary computation - GECCO '05, Washington DC, USA, p. 153, 2005.

[26] H. F. Wedde, M. Farooq, and Y. Zhang, "BeeHive: An Efficient Fault-Tolerant Routing Algorithm Inspired by Honey Bee Behavior," in Ant Colony Optimization and Swarm Intelligence, vol. 3172, M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, and T. Stützle, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 83–94, 2004.

[27] S. Bitam, M. Batouche, and A. Mellouk, "QoSBeeManet: A new QoS multipath routing protocol for mobile ad-hoc networks," in 2010 IEEE Globecom Workshops, Miami, FL, USA, pp. 1648–1652, 2010.

[28] D. Couto, D. Aguayo, J. Bicket and R. Morris, "a high-throughput path metric for multi-hop wireless routing", Wireless Networks, vol. 11, no. 4, pp. 419-434, 2005.

[29] C.E. Perkins, E.M. Royer, S.R. Das, M.K. Marina, "Performance comparison of two on-demand routing protocols for ad hoc networks", IEEE Personal Communications, vol. 8, no. 1, pp. 16-28, 2001.

[30] D. Sharma and K. Sharma, "Grid based multipath energy aware routing protocol for WSNs", International Journal of Computer Trends and Technology (IJCTT), Vol. 4, Issue 4, pp. 729-733, 2013.