IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 21, 2018, accepted October 7, 2018, date of publication October 22, 2018, date of current version December 3, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2875952

A Hybrid Routing Protocol for Wireless Distributed Networks

MALIK NAJMUS SIRAJ[1], ZAHEER AHMED[1], MUHAMMAD KASHIF HANIF[2], MUHAMMAD HASANAIN CHAUDARY[3], SHOAB AHMED KHAN[4], AND NADEEM JAVAID[5]

[1]Center for Advanced Studies in Engineering, Islamabad 44000, Pakistan
[2]Government College University, Faisalabad 38000, Pakistan
[3]COMSATS University Islamabad, Lahore Campus, Lahore 54000, Pakistan
[4]College of Electrical and Mechanical Engineering, Rawalpindi 46000, Pakistan
[5]COMSATS University Islamabad, Islamabad Campus, Islamabad 44000, Pakistan

Corresponding author: Malik Najmus Siraj (siraj@case.edu.pk)

This work was supported by the Center for Advanced Studies in Engineering, Islamabad, Pakistan.

ABSTRACT Trends in networking are shifting from distributed to logically centralized networks. Software defined networking (SDN) is the key technology behind this shift. This new paradigm not only proves the ease of network management in wired networks but also results into graceful evolution of networking protocols. Moreover, wireless distributed networks (WDNs) and software defined radios (SDRs) are also shifting towards the centralized approach. However, due to distributed control nature of WDNs, already prevailing routing algorithms cannot cope the design principle of centralized routing algorithms. Hence, a new routing algorithm that incorporates SDN in WDNs is required. In this paper, we propose a logically centralized approach for WDNs called centralized approach to mobile ad hoc network (CATMAN). It is a self-healing and hybrid routing protocol. It combines distributed and logically centralized network control along with the flavors of reactive, proactive, and opportunistic routing protocols. CATMAN is designed to automatically switch between logically centralized routing protocol and distributed routing protocol based on the types of available network nodes. The protocol significantly reduces control overhead and hence improves bandwidth utilization. While route computations, this protocol deliberately avoids the nodes with weaker batteries, optimizing the effective life of whole network. The simulation results showed that the CATMAN protocol outperform better approach to mobile ad hoc network (BATMAN) protocol by signaling overhead reduction up to 570%.

INDEX TERMS Software defined networking, software defined radios, wireless distributed networks, routing protocols.

I. INTRODUCTION

Internet is evolved as a distributed network with routing and management protocols that facilitate distributed control, management and fault tolerance. As a result, network devices independently made routing and management decision; thereby making the network difficult to control and debug. Moreover, internet architecture also poses challenges to the deployment and testing of new protocols.

Mobile Ad hoc NETworks (MANETs) and wireless sensor networks collectively called WDNs, uses either reactive or proactive routing protocols. In distributed network control, each node makes independent routing decisions that results into poor network management. SDN has emerged as a new innovative networking paradigm that has a logically centralized network control. SDN decouples the control plane from the data plane. This decoupling enables the centralized control to take coordinated decisions that directly guide the network to desired operating conditions. Moreover, decoupling the control plane enables graceful evolution of protocols, and the deployment of new protocols without replacing data plane switches. As a proof of concept, SDN is tested in a campus network [1], home networking [2] and in wide area networks [3].

In this paper, we propose a framework for WDNs along with a novel hybrid routing protocol called Centralized Approach To Mobile Ad-hoc Network (CATMAN). The CATMAN protocol features are the amalgamation of proactive routing protocol BATMAN [4] and OLSR [5], reactive routing protocol AODV [6], opportunistic routing protocol JOKER [7] and southbound protocol in SDN (openflow). The proposed protocol has a potential of identifying multiple paths towards the destination and on the same hand, for some cases, only identifies single path for each destination. In this way, it combines the advantages of both methods.

CATMAN protocol is a self-healing, multi-hop routing protocol for WDNs. It is a hybrid routing protocol i.e. it inherits reactive and proactive routing protocol properties. This protocol is designed in order to incorporate SDN concepts to WDNs. WDNs demand distributed control whereas SDN demands centralized control. For logically centralized approach in WDNs; ad hoc network is divided into two types of nodes i.e. fog nodes and normal nodes. Fog node has short and long range communication interfaces in order to communicate with normal nodes, and centralized control and management layer/fog nodes respectively as shown in Fig 3. Normal node has only short range communication interface. Here, fog node is similar to those nodes in BATMAN protocol that has internet connectivity. CATMAN protocol allows participating normal nodes to maintain 1 hop and 2 hop nodes information through originator message. Fog nodes maintain larger network footprint and help to have a logically centralized network control. CATMAN protocol allows each normal node to maintain best next hop address towards the destination instead of maintaining a complete path.

The proposed protocol allows nodes to quickly respond topology changes in the network. This protocol is designed in such a way that it works in the presence of fog node(s) or without fog node(s). Once there is no fog node in the network, the propose protocol works as a traditional distributed routing protocol.

The rest of the paper is organized as follows. Section II motivates the adaptation of logically centralized control for WDNs. In section III, we briefly described the related work. The proposed framework is outlined in section IV and proposed protocol is described in section V. Simulation results are presented in section VI and finally, conclusion is drawn in section VII.

II. MOTIVATION FOR LOGICALLY NETWORK CENTRALIZED CONTROL

Incorporation of SDN concepts in WDNs is due to many fold.

First, most of the nodes in WDNs are battery powered and the drainage of battery is directly proportional to nodes transmission power. This means that for efficient battery utilization in WDNs, nodes need to transmit less control messages. Consider an example presented in Fig. 1. In this topology, there are 9 nodes and the dash line shows that two nodes can directly communicate with each other. In this topology, the number mentioned on the nodes represents the battery level of the node. There is one transmitter node represented as *Tx* and one receiver node represented as *Rx*. Now let's assume that critical battery level be less than 25%. In this network, if the *Tx* node sends a data to *Rx* node it uses a path 3, however, in this path one of the node has a battery level of 20%. If this path is adopted then this node will deplete its battery very quickly as compared to other network nodes. Using a logically centralized network control approach, path 1 is adopted because in this path all nodes have higher battery level. This example demonstrates that centralized network control manages network better than the distributed network control.

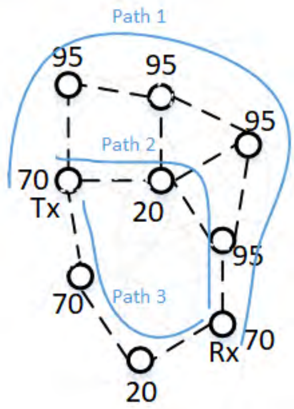


FIGURE 1. Simple network.

Second, adding a logically centralized network control in WDNs is due to SDRs. Battle field networks that use SDRs have multiple narrow band waveforms that may run Orthogonal Frequency Division Multiple Access (OFDMA). However, the dynamic switching in these waveforms, effectively selecting the access frequency bands and dynamic selection of radio parameters require logical centralized algorithm to decrease the control overhead and uniformly assigns the frequencies. Moreover, SDR radios are using different frequencies or using same frequency, however, are located on geographically distributed locations as shown in below Fig. 2. In Fig. 2, three networks are shown with rectangle boxes. The dotted line shows the communication, on same frequency; within a network, whereas the solid line shows the communication in different frequencies; outside a network. For a limited BW network, sending the routing information of network C to network A using network B may over utilized the resources using traditional approach as compared to SDN enabled network.

Furthermore, SDR used in tactical networks has narrow band and wide band waveforms. The narrow band waveforms are used for long distance communication and can range up to 50 km. These waveforms use a band of 30 MHz to 400 MHz [8] and can support data rates up to 82 kbps in sophisticated environment. Usually data rate of less than 20kpbs is achieved in practical scenarios. Wide band waveforms use a band of 225 MHz to 2000 MHz and can reach up to 8 Mbps. At this point it is noteworthy that NATO has only standardized narrow band waveforms. Now in order to create a network using Narrow band waveforms that contains

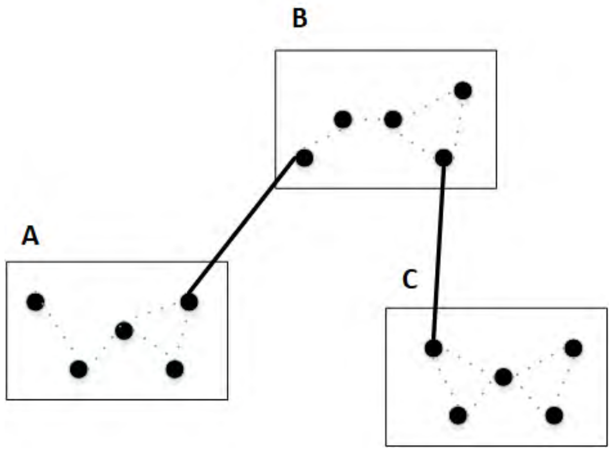


FIGURE 2. SDR based WDNs.

at least 30 nodes with 2 VOIP calls, a less than 10kbps data rate is available for network signaling. This signaling involves getting the destination nodes information and selecting the channel for sending data/voice messages in a fully collision environment. It means that a routing protocol with less overhead is required.

Third, SDN easily eliminates the compromised node during battlefield. Fourth, nodes are becoming little intelligent, however, the routing decision is performed by logically centralized network control. This decreases the number of control messages transmitted and the processing on the nodes and resulted into energy saving.

III. RELATED WORK

Routing protocols in WDNs are categorized in four domains i.e. reactive routing protocol, proactive routing protocol, opportunistic routing protocol and SDN based routing protocol. Furthermore, these protocols are categorized into geographical forwarding, link state forwarding and distance vector forwarding. In recent years, more focus is towards SDN based routing protocols.

The authors presented Greedy Perimeter Stateless Routing (GPSR) protocol [9] that uses both router position and packet destination to forward a packet. It uses greedy approach by using only router's immediate neighbors and route around the perimeter of a region if unable to follow greedy approach. Simulation results showed that GPSR achieved better scale as compared to shortest path routing algorithm in WDNs. Moreover, this technique outperformed in highly mobile WDNs.

In recent years, many researchers are integrating SDN with Vehicular Ah hoc NETwork (VANET) [10]–[12] in order to resolve issues of scalability, control overhead, delay reduction and network management. In [10], SDN based routing protocol is proposed for VANET. In SDN environment, controller gathers all link-state information from switches and runs the global optimal routing algorithm to find the shortest path between the source and the destination. Soua *et al.* [11] addressed the content dissemination in VANET and combined floating content and content centric network using SDN in order to optimize the packet forwarding process. Yaqoob *et al.* [12] outlined the recent studies related to the requirements for software defined vehicular networks.

Authors in [13], [16], [18], [19] proposed energy aware optimization algorithm for wireless mobile networks. In this [13] paper, authors adopted multicast routing technique to decreases the load in a network and increases the battery life of network. A Fuzzy logic based routing algorithm is proposed in this [16] paper that uses node energy consumption, residual energy, nodes density as decision making parameters in order to reduce the energy consumption. Hu *et al.* [18] proposed SDN based routing algorithm that also protects wireless nodes against attacks. Simulation results showed that this algorithm performed well in terms of packet delivery ratio, energy efficiency and throughput. Wei *et al.* [19] paper adopted SDN based energy harvesting and increases the network life. To address the scalability in WDNs, Abolhasan *et al.* [20] proposed the new architecture and protocol that eliminates multi-hop flooding and resulted into scalable architecture. It used separate frequency bands for control messages and data to reduce the overhead in data spectrum. Moreover, for route discovery, a differential computation algorithm is adopted that splits routing complexities between forwarding nodes and SDN controller. SDN controller gathers link state information and sends preprocessed weights to forwarding nodes for computing routing paths. Table 1 summarizes literature review with categorization in routing protocols and optimization algorithms.

Wide range of applications in wireless sensor networks requires scalability. Moreover, network also requires excessive Monitoring [21] and good quality of information [25] extraction from sensor nodes. In this [29] paper, author discusses the challenges in routing protocols that hinders the massive deployment of such networks.

IV. PROPOSED FRAMEWORK

The framework of the proposed work is depicted in Fig. 3. It has two layers: hierarchical ad hoc layer and control/management layer. Both are discussed as follows:

A. HIERARCHICAL AD HOC LAYER

This is the base layer that contains nodes forming a wireless ad hoc network. The motivation of adding this layer are:

- This layer helps to incorporate logically centralized control plane in WDNs by incorporating two types of nodes.
- This layer further add the distributed control in logical centralized WDNs. In this layer, nodes can directly communicate with each other even if there is no communication between both layers.

1) NORMAL NODES

Nodes that have only short range communication interfaces are termed as normal nodes. These nodes can connect to each other and also with fog nodes through a light weight proposed protocol that is discussed in section V.

TABLE 1. Related work.

Name	Methodology	Routing protocol or optimization algorithm	Pros
AODV [6]	Reactive protocol and uses destination sequence number	Routing protocol	Less control overhead
OLSR v1 rfc 2326 [5]	proactive protocol and uses hop count	Routing protocol	Use MPRs to reduce control messages
OLSR v2 rfc 7181 [17]	Proactive protocol and uses link state information	Routing protocol	Each router selects two MPRs. MPR routers transmit only link state information
BATMAN [4]	Proactive protocol	Routing protocol	Nodes only maintain best hop towards destination and lost packets are used to find best hop towards destination
JOKER [7]	Opportunistic protocol	routing protocol	Use dynamic control message sending interval
Dynamic source routing [26]	Reactive protocol and uses hop count	Routing protocol	On demand route discovery and route maintenance and maintain multiple routes
Zone routing protocol [27]	Combines reactive and proactive protocols	Routing protocol	Network is divided into multiple zone in order to reduce network control overhead
Greedy perimeter stateless routing [9]	Geographical position	optimization algorithm	Use position of router and packet destination to forward packets
SDN based VANET routing [10]	Centralized	optimization algorithm	Reduce routing overhead due to centralized approach
Hybrid SDN [20]	Hybrid network control	both	Split the network control and data into separate frequencies
SDN geographical routing [28]	Centralized	optimization algorithm	Identifies routing path based on node location, traffic density and network map

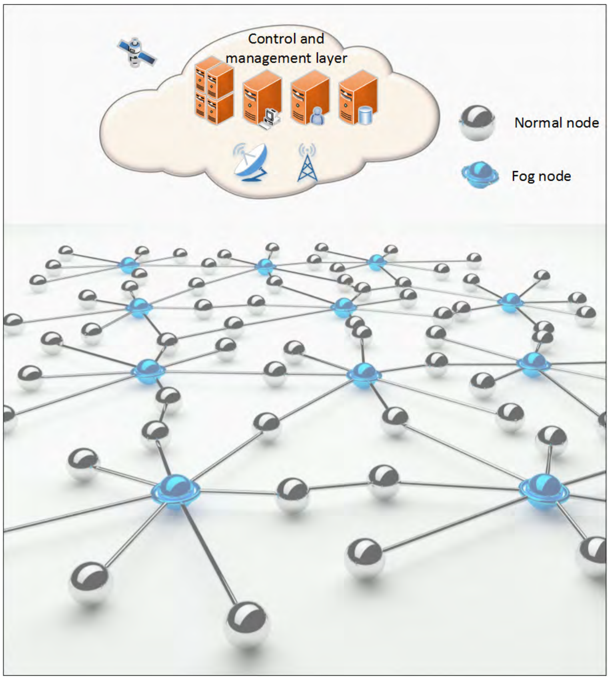


FIGURE 3. SDN based framework for WDNs.

2) FOG NODES

Nodes that have both short and long range communication interfaces are termed as fog nodes. These nodes use short range communication interface to form an ad hoc network with each other or with normal nodes. Long rang communication interface in fog nodes is used to connect control and management layer and other fog nodes. Fog nodes have a small network footprint. Two long range communication interfaces, satellite and cellular, are shown in Fig. 3. The advantage of fog layer is that every node in the network is not required to maintain the information of every other node. Nodes that have short range communication interface only maintain limited network information. Fog node maintains bigger network footprint of nodes connected to it at any hop location.

B. CONTROL AND MANAGEMENT LAYER

Control and management layer has complete network picture of all the geographically distributed nodes that form a complete network. Now once the complete network picture is stitched at control layer, different types of routing and network management can be performed. This layer has datacenter for processing and running complex control and management applications. The control and management layer has multiple long range communication interfaces for communication with hierarchical ad hoc layer.

V. PROPOSED PROTOCOL: CATMAN

In this section, we discuss detailed description of CATMAN protocol. It is a self-healing routing protocol that uses logically centralized control in WDNs. The proposed protocol is an amalgamation of different routing protocols in order to combines their positives.

A. OVERVIEW

CATMAN protocol is a hybrid routing protocol that uses originator, far node, route request, route reply, status update, flow add and configuration update messages. The purpose of originator message is to inform first hop nodes regarding the presence of originator node. Similarly, route request and route reply messages are used to get the routing information of the destination node. Far node, stats update and flow add messages help to create a logically centralized network control, where far node message and stats update message increase the network visibility of fog node and flow add message adds the routing information in the node. Configuration update message configures the physical layer parameters of SDR nodes in the network. CATMAN protocol proactively uses originator message and it uses far node, flow add, configuration update, stats update route request and route reply messages reactively.

The CATMAN protocol is designed in such a way that it can work in the presence of fog node and without fog node. This combines the advantages of centralized and decentralized routing protocol.

There are three types of interface in SDN

- South bound interface i.e. between control plane and data plane. A de facto protocol for this interface is openflow.
- East west interface for controller to controller communication.
- Northbound interface for control plane to application plane communication.

The proposed protocol runs between normal nodes and between fog node and normal node as a southbound protocol in SDN based WDNs. Table 2 enlists messages used in CATMAN and their descriptions.

In CATMAN protocol, every node transmits an originator message and only the first hop nodes broadcast this originator message. The nodes receiving this message extract useful information about other nodes and add it in its table. In this way, every node has an information about its 1 and 2 hop nodes. In this protocol, fog node does not broadcast any received message, however, it also extracts useful information from the received message.

After transmitting 10 originator messages, if a normal node has no information about a fog node then it generates a far node message. The purpose of far node message is to increase network visibility of fog node. In this way fog node maintains bigger network footprint.

In WDNs, if a node wants to send a message to other node, first it checks the presence of destination node in its table. If no information found, it creates a route request message (if know about fog node it sets next hop address otherwise next hop address $= -1$).

Wireless link quality is measured through SNR. A normal node sends a stats update message when the received SNR value of any first hop link is less than 40% or battery life of the node is less than the threshold value. The purpose of stats update message is two fold:

- Informs the link breakage to neighbor nodes and fog nodes.
- Updates the battery life to its neighbor node and fog node. This helps the originating node of stats update message to stay alive for longer time be generating less messages.

Configuration update message is a special type of message designed for SDR nodes. It is used to configure the physical layer and MAC layer parameters of TCP/IP model. This configuration includes but not limited to selection of modulation scheme, selection of radio access network, narrow band waveform selection etc. All these parameters help to efficiently use the frequency spectrum to increase the performance of the complete network. In this paper, we only focus on routing using CATMAN.

TABLE 2. CATMAN messages description.

Message	Description
Originator message	Informs other nodes regarding the presence of this node and it also selects 1st hop MPRs nodes of sender node.
Far node message	Informs the presence of a node and its neighbors to fog node.
Route request message	A node that does not know the presence of destination node will generate this message.
Route reply message	A message is generated in response of route request message and it contains next best hop towards destination.
Flow add message	This message is generated in response to route request message or fog node generates this message in order to update the configuration.
Stats update message	Informs fog node that either SNR of neighbor node or battery level is changed
Configuration update message	Centralized control and management layer or fog node updates the configuration of other nodes using this message.

B. INHERITED PROPERTIES

CATMAN runs between transport layer and application layer in TCP/IP stack. As previously mentioned, this protocol is inherited from Openflow, AODV, BATMAN and JOKER. It takes the advantages of centralized control from Openflow, route request and route reply messages from AODV, small size of originator messages from BATMAN, selection of MPRs from OLSR and adoptive Control Message Sending Interval (CMSI) from JOKER. However, for centralized control, it has far node message, stats update message, configuration update message and flow add message. Out of these messages stats update message, configuration update message and flow add message are similar to openflow whereas far node message is a totally new message for

WDNs that helps to increase the network visibility of logically centralized controller. Moreover, the purpose of route request message and route reply message is the same as in AODV. However, it has a different message headers as shown in Fig. 6c and Fig. 6d. Similarly, the originator message in BATMAN is broadcasted until all the nodes receive at least 1 copy. But in CATMAN only first hop nodes broadcast this message. Finally, CMSI [7] for JOKER and CATMAN is the same as shown in equation 1. However, the number of control messages in CATMAN protocol are less as compared to JOKER protocol.

$$CMSI = 0.006 \times TP + 1.5 \tag{1}$$

where TP represents the throughput.

Now consider simple case of torus Networks with only 2 dimensions called Manhattan network [34]. Here, N is the number of nodes in the network and that depends on the internal parameter and is defined as $N = n^2$ having edges $E = 2n^2$. Consider an ideal case of successful transmission probabilities i.e. p = 1. In case of BATMAN and JOKER the total number of control packet broadcasted by each node after considering for even n is modeled as

$$CPB = 2 + \sum_{s=1}^{\frac{n}{2}-1} 4s + \sum_{s=\frac{n}{2}+1}^{n-1} 4(n-s) \tag{2}$$

where CPB is the Control Packets Broadcast by each node in JOKER and BATMAN.

In bidirectional Manhattan network every node is connected with 4 other nodes in a network. As previously mentioned in section V-A, in CATMAN only first hop nodes broadcast the originator message. This means that every node transmits 5 originator messages. Now considering that maximum distance between normal node and fog node is 3 hops and only 30% of nodes are far nodes. Then, total number of CPB in case of CATMAN protocol is shown below

$$CPB_{\text{CATMAN}} = 5 \times N + (0.3 \times N) \times 9 \tag{3}$$

Now measuring the control overhead because in wireless network very less bandwidth is available. So, the total Control Packet Overhead (CPO) in case of BATMAN, JOKER [7] and CATMAN is defined as

$$CPO_{\text{BATMAN}} = CPB \times N \tag{4}$$

$$CPO_{\text{JOKER}} = \frac{(CPB \times N)}{(0.006 \times TP + 1.5)} \tag{5}$$

$$CPO_{\text{CATMAN}} = \frac{CPB_{\text{CATMAN}}}{(0.006 \times TP + 1.5)} \tag{6}$$

Fig. 4 shows the comparison of CPO by varying the throughput and number of nodes in case of BATMAN, JOKER and CATMAN. The result shows that in case of JOKER and BATMAN as the number of nodes increases, there is an uncontrolled increase in number of control packets. However, CATMAN results in reduced number of control messages. This is because, in CATMAN only first hop nodes flood the originator message. Although six other types of messages are transmitted but overall control overhead is very less as compared to JOKER and BATMAN.

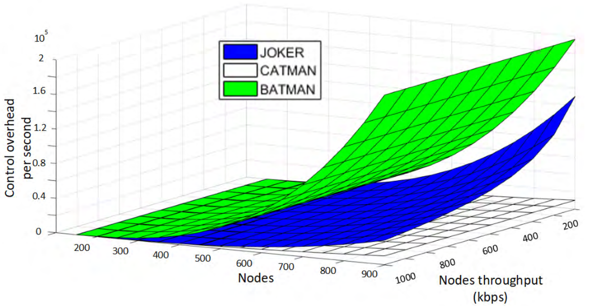


FIGURE 4. Comparison of control packets overhead of JOKER, BATMAN and CATMAN.

Probability of successful transmission has a great impact on control overhead in case of BATMAN, JOKER and CATMAN. In real scenarios, the probability of successful transmission is less than 1. We tested all the three protocols by varying the successful transmission probabilities as shown in Fig. 5. In case of CATMAN when the successful transmission probability is close to zero, number of control packets broadcast by CATMAN are high because a greater number of nodes has no information about fog node and they are generating far node messages. Similarly, with the successful transmission probability increases from 70%, CATMAN beats the other two protocols. Although we can say that overall, CATMAN results into control flooding in network and saves a lot of BW.

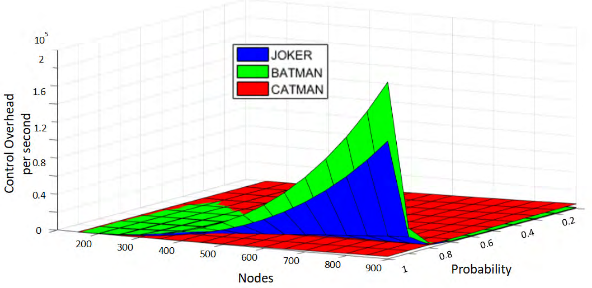


FIGURE 5. Overhead comparison by varying transmission probabilities and number of nodes for JOKER, BATMAN and CATMAN.

C. MESSAGE FORMATS

This section discusses the message structure of CATMAN protocol. Table 3 describes message fields for different messages in CATMAN protocol.

1) ORIGINATOR MESSAGE

The purpose of originator message is of two fold. It is used for link discovery and reduces the control overhead in a network by selecting MPR nodes. Only the selected MPR

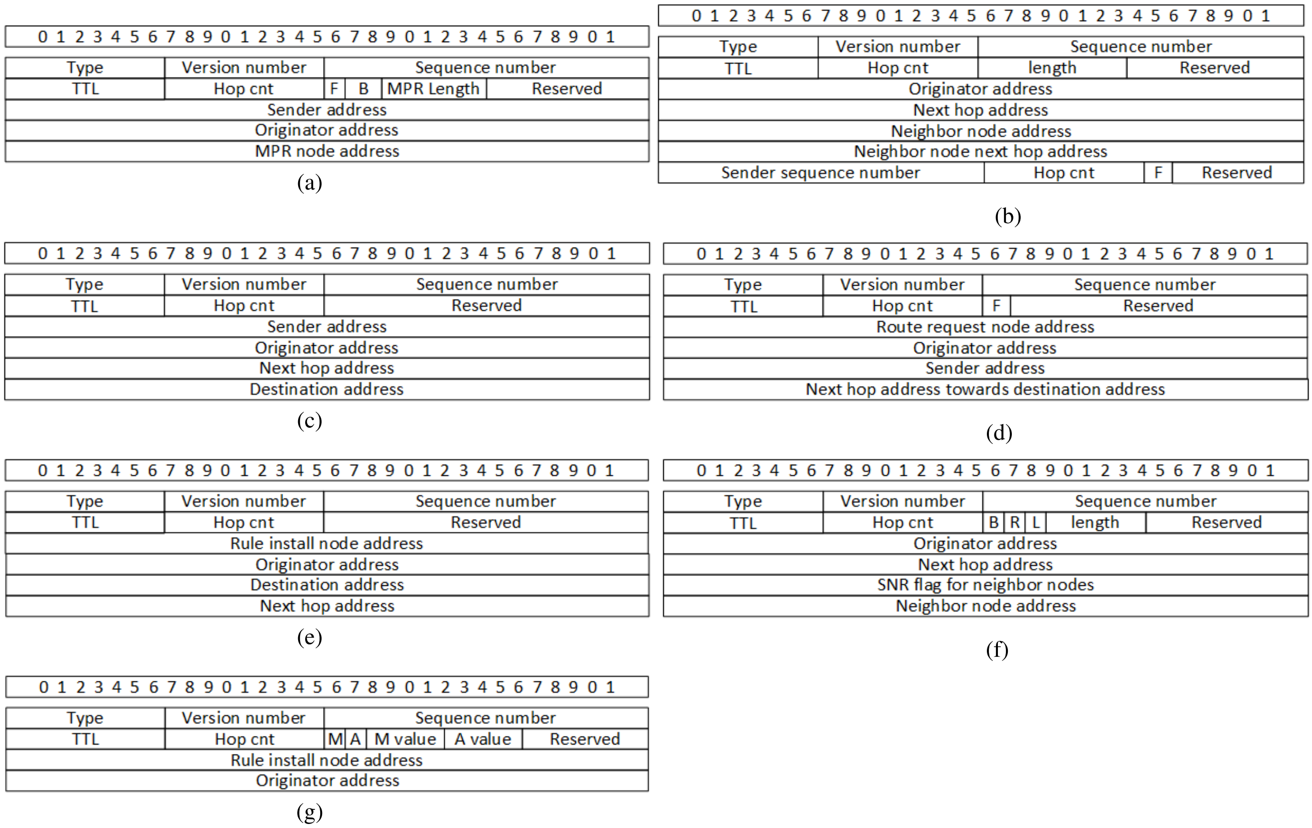
TABLE 3. Description of different fields in CATMAN messages.

Field	Description
Type	Defines the message type in proposed protocol. For originator message type =1, far node message type =2, route request message type = 3, route reply message type = 4, flow add message type = 5, stats update message type = 6, configuration update message type = 7.
Version number	MUST be set to VERSION. Each packet received with a version field different from VERSION MUST be ignored.
Sequence number	The originator node consecutive numbers for each new originator message with an incremented (by one) sequence number.
TTL	TTL = Time to live: The TTL can be used to define an upper limit on the number of hops originator message can travel.
Hop cnt	Hop cnt = Hop count: The number of hops from the originator IP Address to the node handling this message.
Length	Length = Number of neighbor nodes in FN message node: Tells number of neighbors of a node generating far node message.
Originator address	Address of node that creates this message.
Sender address	Address of node that broadcasts this message or unicasts this message.
Next hop address	Address of node that is next best hop towards fog node. If node has no information about next hop towards destination than this field is set to -1.
Neighbor node address	Address of node that is first hop or second hop neighbors of a node generating this far node message.
Neighbor node next hop address	Next hop node that is first hop or second hop neighbor of a node.
Sender sequence number	Sequence number stored in far node table for respective neighbor node.
Destination address	The address of the destination for which a route is required.
Route request node address	Address of node that requests the route of a destination node.
Next hop address towards destination	Address of node that is next best hop towards destination node if route reply message is generated by fog node. If this message is generated by normal node then this field is -1.
Rule install node address	Address of node on which flow add rule is going to be installed.
SNR flag for neighbor nodes	This field is valid only if stats update message is generated due to change in SNR of neighbor node/s. It represents 1 bit flag for each neighbor node/s. If respective bit is ÑśÒăthis means that SNR is greater than 40% and if bit is zero than SNR is less than 40%.
MPR node address	list of nodes that are MPRs of sender node
F	F = Fog flag: To represent that a node is a fog node or not. F = 1 originator node is a fog node. F = 0 originator node is a normal node.
B	B = Battery level flag: Represents battery level of a node. B = -1 node cannot monitor battery level. B = 0 node battery level is less than 20%. B = 1 node battery level is greater than 20%.
R	R = SNR flag: R = 1 Stats update message is generated because of change in SNR of neighbor node/s. R = 0 There are other reasons of generating this message.
M	M = Modulation scheme flag: M = 1 configuration update message is generated because of update in modulation scheme is required and M value contains modulation scheme number. M = 0 There are other reasons of generating configuration update message.
A	A = Access frequency flag: A = 1 configuration update message is generated because of update in access frequency is required and A value contains access frequency number. A = 0 There are other reasons of generating this message.
Reserved	Sent as 0; ignored on reception.
MPR length	No. of MPRs selected by node generating this message.
R length	No. of first hop neighbors whose SNR is increased/decreased from threshold.

nodes broadcast the received message. In CATMAN protocol, each node transmits originator message after every CMSI time as represented in equation 1. This time has a lot of impact on quality of service (QoS). Network control overhead is directly proportional to CSMI. If network is static then this time can be decreased and if network is highly dynamic then this time can be increased. In this version of CATMAN protocol every node transmits an originator message after every 60 seconds (as we used 60 seconds time in our simulation). The details of originator message format is shown in Fig. 6a.

2) FAR NODE MESSAGE

The purpose of far node message is to increase the network visibility of fog node that ultimately increases the network footprint of controller. After generating 10 originator



<table>
<tr><td colspan="6">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td colspan="4">Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>F</td><td>B</td><td>MPR Length</td><td>Reserved</td></tr>
<tr><td colspan="6">Sender address</td></tr>
<tr><td colspan="6">Originator address</td></tr>
<tr><td colspan="6">MPR node address</td></tr>
</table>

(a)

<table>
<tr><td colspan="5">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td colspan="3">Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>length</td><td colspan="2">Reserved</td></tr>
<tr><td colspan="5">Originator address</td></tr>
<tr><td colspan="5">Next hop address</td></tr>
<tr><td colspan="5">Neighbor node address</td></tr>
<tr><td colspan="5">Neighbor node next hop address</td></tr>
<tr><td colspan="2">Sender sequence number</td><td>Hop cnt</td><td>F</td><td>Reserved</td></tr>
</table>

(b)

<table>
<tr><td colspan="3">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td>Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>Reserved</td></tr>
<tr><td colspan="3">Sender address</td></tr>
<tr><td colspan="3">Originator address</td></tr>
<tr><td colspan="3">Next hop address</td></tr>
<tr><td colspan="3">Destination address</td></tr>
</table>

(c)

<table>
<tr><td colspan="4">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td colspan="2">Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>F</td><td>Reserved</td></tr>
<tr><td colspan="4">Route request node address</td></tr>
<tr><td colspan="4">Originator address</td></tr>
<tr><td colspan="4">Sender address</td></tr>
<tr><td colspan="4">Next hop address towards destination address</td></tr>
</table>

(d)

<table>
<tr><td colspan="3">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td>Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>Reserved</td></tr>
<tr><td colspan="3">Rule install node address</td></tr>
<tr><td colspan="3">Originator address</td></tr>
<tr><td colspan="3">Destination address</td></tr>
<tr><td colspan="3">Next hop address</td></tr>
</table>

(e)

<table>
<tr><td colspan="7">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td colspan="5">Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>B</td><td>R</td><td>L</td><td>length</td><td>Reserved</td></tr>
<tr><td colspan="7">Originator address</td></tr>
<tr><td colspan="7">Next hop address</td></tr>
<tr><td colspan="7">SNR flag for neighbor nodes</td></tr>
<tr><td colspan="7">Neighbor node address</td></tr>
</table>

(f)

<table>
<tr><td colspan="7">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Type</td><td>Version number</td><td colspan="5">Sequence number</td></tr>
<tr><td>TTL</td><td>Hop cnt</td><td>M</td><td>A</td><td>M value</td><td>A value</td><td>Reserved</td></tr>
<tr><td colspan="7">Rule install node address</td></tr>
<tr><td colspan="7">Originator address</td></tr>
</table>

(g)

FIGURE 6. Messages format. (a) Originator message. (b) Far node message. (c) Route request message. (d) Route reply message. (e) Flow add message. (f) Stats update message. (g) Configuration update message.

messages, if a node has no information about the fog node it generates a far node message. This message has information about node generating this message and its two hop neighbor nodes. Far node message structure is shown in Fig. 6b.

3) ROUTE REQUEST MESSAGE

A node generates a route request message if it has no information about the destination node to which it wants to send a message. If node generating this message has an information about the fog node then it unicasts this message towards next best hop towards fog node. If it has no information about the fog node then it broadcasts this message. The message structure of route request message is shown in Fig. 6c.

4) ROUTE REPLY MESSAGE

The purpose of route reply message is to inform route requesting node to get the information about the destination node. This message informs route requesting node to get the best next hop address towards the destination. The message structure of route reply message is shown in Fig. 6d.

5) FLOW ADD MESSAGE

The purpose of this message is to update the network configuration. This adds a great flexibility to the logically centralized network control. This message is generated by controller or fog node. This message can be generated against route request message or controller proactively generates this message in order to apply any specific rule in the network e.g. if any node in the network is compromised then controller can add rule in the network so that this node cannot participate in the network or use configuration update messages to change to physical layer parameters of the compromised node in the network. The message structure of flow add message is shown in Fig. 6e.

6) STATS UPDATE MESSAGE

Stats update message helps fog nodes, normal nodes and controller to intelligently route traffic in the network. It informs the neighboring nodes regarding battery life and 1 hop neighbor node link connectivity of node generating the message. Stats update message structure is shown in Fig. 6f.

7) CONFIGURATION UPDATE MESSAGE

The purpose of configuration update message is to increase the spectrum efficiency of a network. It is used to configure the physical layer parameters of a node in a network. These parameters are narrow band frequency selection and modulation scheme etc. The message structure of configuration update message is shown in Fig. 6g.

D. CATMAN OPERATION

CATMAN uses both centralized and decentralized approaches to decrease the number of control messages. If a node has no information about destination node in its table, it sends route request message. A node replies to this message using route reply message. In this section, we discussed the working of CATMAN in WDNs where multiple scenarios are taken to explain the working of proposed protocol.

1) SCENARIO 1

In scenario 1, we considered three nodes namely node 1, node 2 and node 3 as shown in Fig. 7. Node 1 is a special node having both short and long range communication interfaces and is termed as fog node and node 2 and 3 have only short range communication interface and are termed as normal nodes. There is one controller shown in Fig. 7. Fog node has a similarity with those nodes in BATMAN protocol that has internet connectivity or other network connectivity. In CATMAN protocol, long range communication interface of fog nodes connects it with centralized control and management. Now assume that only node 1 is present and when node 2 arrives, it broadcasts its originator message. When node 1 receives this originator message it adds an entry in its table regarding the presence of node 2, however, node 1 belongs to fog layer, it only broadcasts its originator message. Node 2 and node 3 are normal nodes, they also broadcast node's 1 originator message. Since all the nodes have information of every other node in the network so there are no far node messages. Total number of messages sent are: $1 + 2 + 2 = 5$.

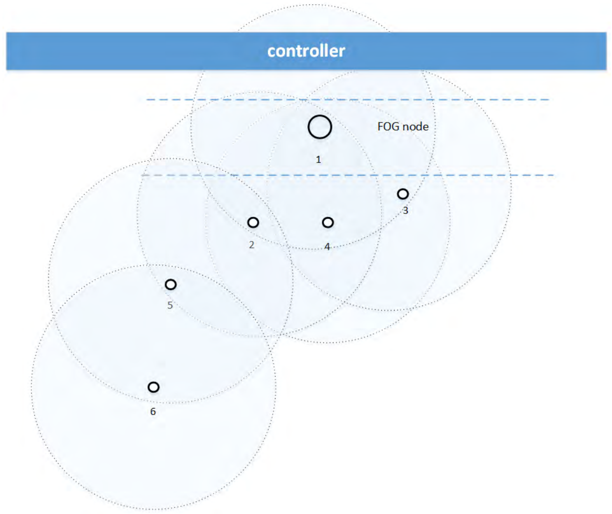


FIGURE 7. Network for scenario 1 and 2 having only 1 fog node.

2) SCENARIO 2

In scenario 2, we have considered 6 nodes as depicted in Fig. 7. Here, node 1 is a fog node and remaining nodes are normal nodes. In this scenario when node 6 broadcasts its originator message, only node 5 receives it, updates its table, and broadcasts it further. Now node 2 receives it and only updates its table. At this point node 2 does not broadcast it further as we have stated that only first hop nodes broadcast the originator message. After some time, when node 6 has no information about fog node, it broadcasts far node message. Far node message has the same information as the originator message plus the number of addresses that are accessible from this node as illustrated in Fig. 6b. Any node that has no information of fog node broadcasts this message and a node that has an information of fog node unicasts it to the next hop towards the fog node. Since, node 5 has an information of fog node so it unicasts it to node 2 that unicasts it to fog node i.e. node 1. Now at this point if a node 3 wants to send a message to node 6, it first creates a route request message and sends it to a fog node i.e. node 1. Node 1 replies to this message with a route reply message and also generates flow add message for node 4. In this way, node 1 broadcasts one message and unicast 2 messages, node 2 broadcasts four messages and unicasts one far node message, node 3 broadcasts three messages and unicast one message, node 4 broadcasts four messages, node 5 broadcasts three messages and unicasts one far node message, node 6 broadcasts two originator messages and one far node message. Total number of messages = $(1+2)+(4+1)+(3+1)+4+(3+1)+(2+1) = 23$. Using far node message, fog node and ultimately centralized control and management layer has an information of node 6 and its neighbors.

3) SCENARIO 3

In this scenario, we consider 12 nodes and out of which node 1 and node 12 are fog nodes as shown in Fig. 8. In this network, if node 6 wants to send a message to node 11, since, node 6 has no information about the fog node it broadcasts route request message. Node 5 after receiving this message unicasts it to node 2. Fog node after receiving route request message finds no information about the destination node in its table. It sends this message to the controller. Since all fog nodes are sending their network information to the controller. It identifies that node 11 is connected with fog node 12. After identifying the optimal path between node 6 and node 11, it creates a route reply message and sends it to node 6. Controller also generates flow add message and update the intermediate nodes flow table accordingly.

Now let us assume that if both fog nodes die even then communication within this network is possible. In that case, every node is transmitting a far node message and this message is re-broadcasted by every other node until all nodes in the network received this message. In that case, overall overhead due to exchange of control messages in the network is increased. This is the extreme case where we are unable to take the advantages of centralized control and management.

VI. SIMULATION RESULTS

In this section, we showed simulation results of CATMAN protocol and compares its performance with BATMAN protocol. Initially, we discuss the simulation parameters

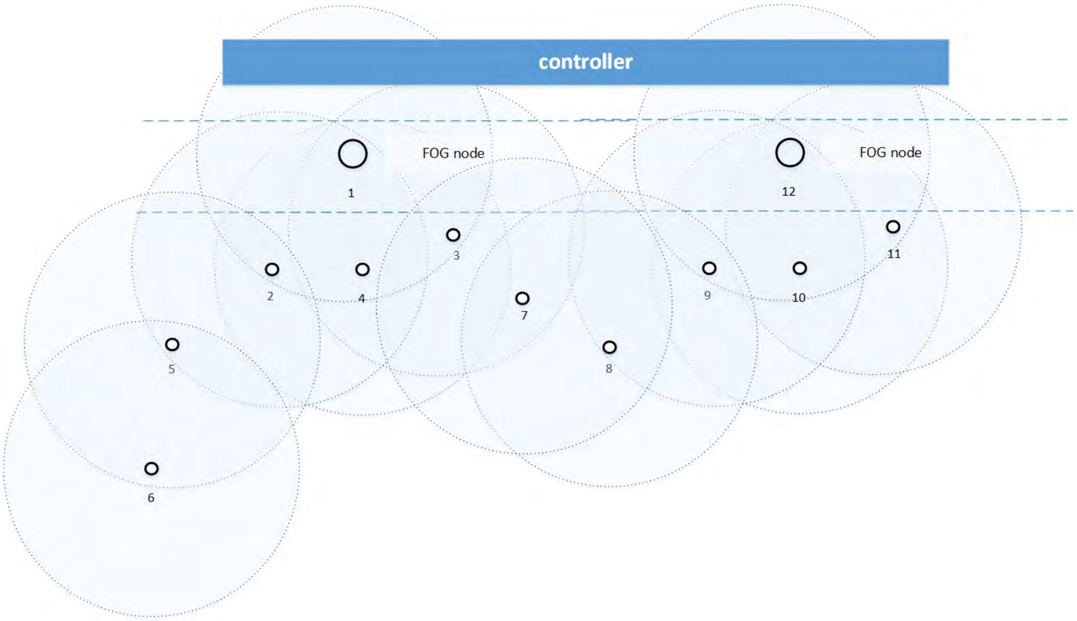


FIGURE 8. Multiple fog nodes: normal nodes and fog nodes are randomly distributed in the network.

in section VI-A. And then we study the impact of each CATMAN protocol message towards total control overhead as described in section VI-B. We compare the performance of CATMAN protocol and BATMAN protocol by increasing the number of nodes in a network and measure the control traffic overhead and end to end delay as described in section VI-C. We also tested CATMAN protocol by varying its internal parameters for different network nodes density. In section VI-D CATMAN protocol is tested on the basis of fog nodes. In section VI-E the impact of zone radius on CATMAN protocol is described and in section VI-F the impact of adding MPRs in CATMAN protocol is presented.

A. SIMULATION PARAMETERS

In this section, we discuss the simulation parameters that are used to evaluate CATMAN protocol. We characterized the simulation environment by zone radius (ρ), node density (λ), number of nodes (N), fog nodes (ψ) and relative node velocity (V) as mentioned in Table 4. Pearlman *et al.* [35] performed performance analysis of zone routing protocol and described average node density as a measure of average number of neighbors/node. Although, this indirectly relates the actual density of a network as the numbers of neighbors are increasing in a limited boundary, network density increases. Similarly, relative node velocity is measure of new neighbor acquisition instead distance/time. These parameters are enough to measure the connectivity of network.

TABLE 4. Simulation parameters.

Parameter Name	Symbol	Range of values
Zone radius	ρ	1-4
Number of nodes	N	1-900
Node Density	λ	3-9
Node velocity	V	1.0-2.0
Fog nodes	ψ	1-36

Each node moves at a constant speed and assigned initial direction from 0 to 2π (radian) within a network. When a node reaches an edge of simulation it is reverted back by changing its direction to $-\theta$. In this way, all the nodes in the network moves within the boundaries of simulation network with a constant speed.

We have made some assumptions in order to reduce the complexities in simulation and increase the understanding of CATMAN protocol. These assumption simplifies lower layer network behavior. In CATMAN protocol each node broadcast originator message for neighbor discovery. This short message of 20 bytes is transmitted by nodes at a random interval having a mean of $T_{\text{originator}} = 1$ second. A link failure is reported to centralized control and management layer using a stats update message. A link failure occurs if a node fails to receive originator message after 2*$T_{\text{originator}}$of most recent originator message. We further eliminate the inaccurate link failure by assuming that originator messages are given highest priority and they are not destroyed because of collision.

The collision free Media Access Control (MAC) protocol means that the Signal To Interference (SIR) ratio of received packet is only limited by receiver noise and ambient background noise. This means that Bit Error Rate (BER) is very low within a $d_{\max}$distance. This resulted into a simplified path loss having following behavior: within $d_{\max}$distance a packet

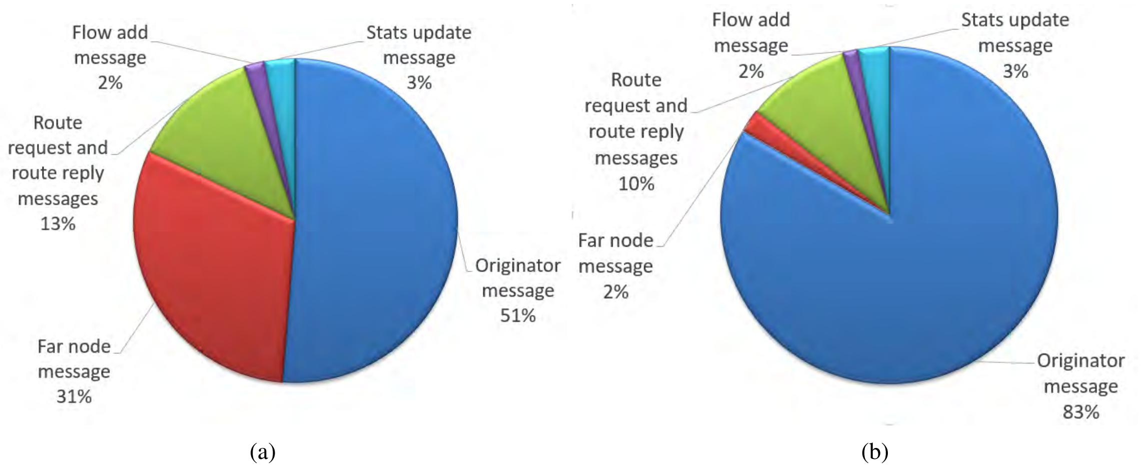


FIGURE 9. Percentage share of messages generated in CATMAN protocol for N = 300. (a) Percentage of messages in CATMAN protocol for λ = 3. (b) Percentage of messages in CATMAN protocol for λ = 6.

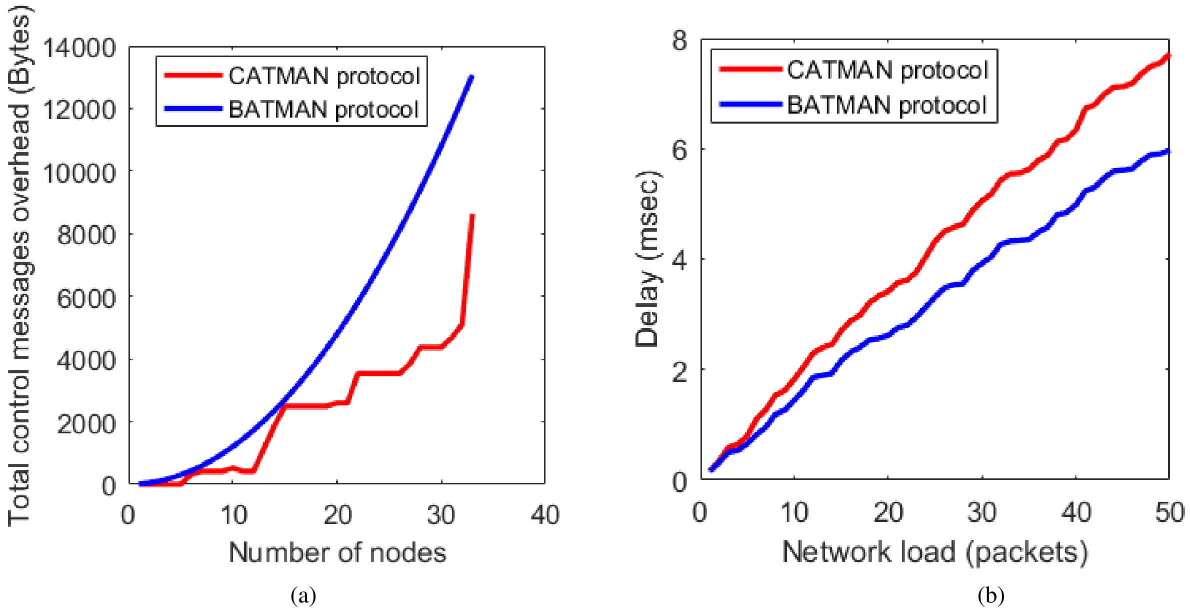


FIGURE 10. Comparison of performance parameter between BATMAN protocol and CATMAN protocol for random topology. (a) Control messages overhead for 33 nodes random topology. (b) Network delay for 33 nodes random network.

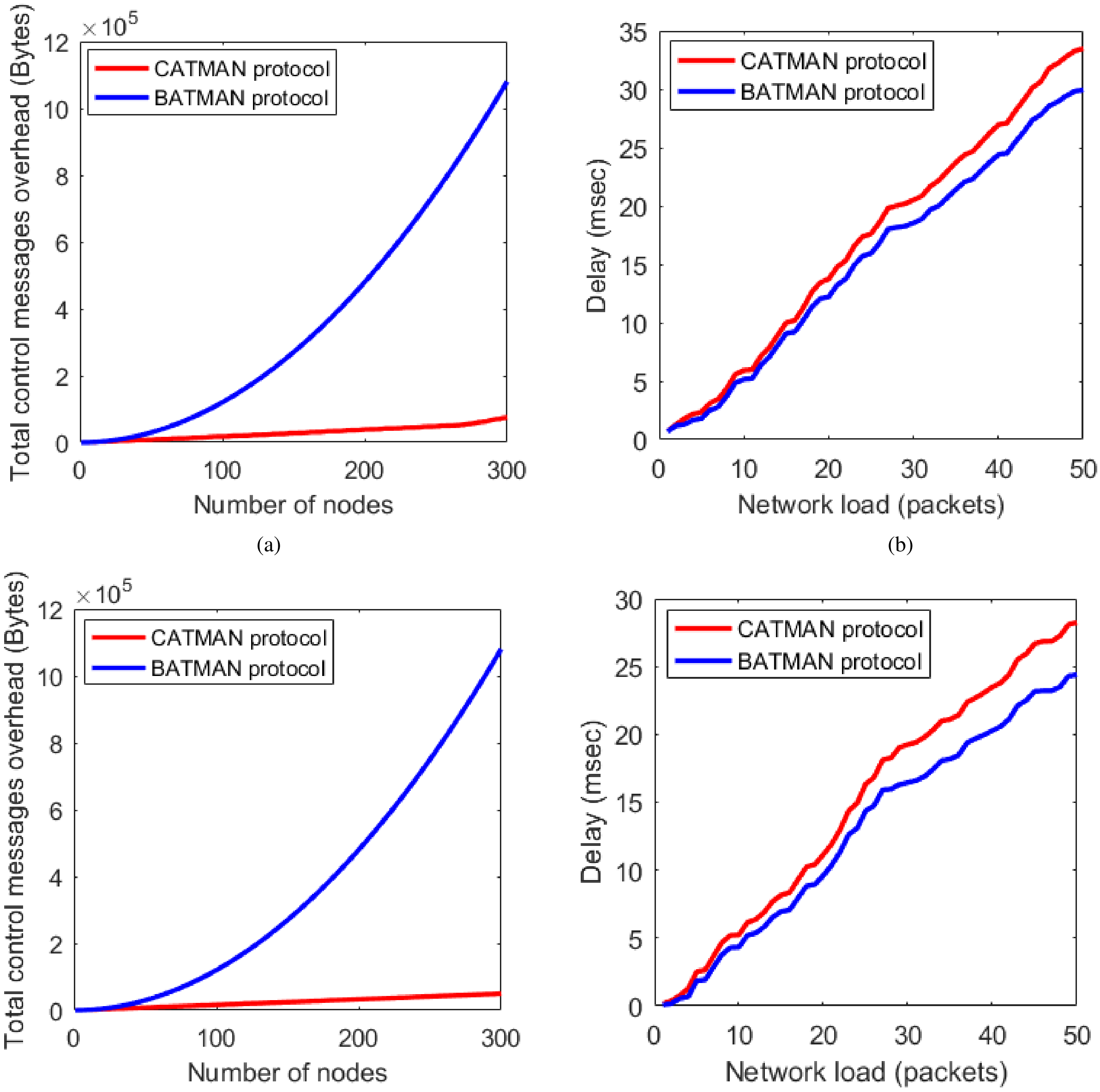
is received error free and after $\mathrm{d_{max}}$distance packet is lost as mentioned in eq. 7.

$$P(d) = \begin{cases} \propto |dB| & \text{for } d \leq d_{max} \\ 0|dB| & \text{for } d \geq d_{max} \end{cases} \tag{7}$$

The selected path loss model helps us to simulate a very large scale densely packed network.

B. IMPACT OF DIFFERENT MESSAGE IN CATMAN PROTOCOL

In CATMAN protocol, there are seven types of messages. Out of these messages, originator message and far node message shows proactive behavior while route request message, route reply message, flow add message, stats update message and configuration update message shows reactive behavior. We considered two different network topologies i.e. $N = 300$, $\lambda = 3$ and $N = 300$, $\lambda = 6$. Fig. 9a and Fig. 9b showed the percentage of each message generated in CATMAN protocol with different node density. For $\lambda = 3$, we observed that the originator message contributed to 51%, far node message represents the 31% of overall control overhead. As the node density increases ($\lambda = 6$) the percentage of originator message is increased to 83% and far node message is decreased to 2%. This is because now nodes are densely packed and far node messages decreases because now more node has information about fog node. We have assumed that only 33% of nodes are sending message to other nodes at



(a)

(b)

(c)

(d) I

FIGURE 11. **Comparison of performance parameter between BATMAN protocol and CATMAN protocol for 3 and 6 neighbors per node. (a) Control messages overhead for 3 neighbors/node. (b) Network delay for 3 neighbors/node. (c) Control messages overhead for 6 neighbors/node. (d) Network delay for 6 neighbors/node.**

one time and out of these 20% of nodes has information of destination node. This resulted into 13%, 2% and 3% of route request/route reply messages, flow add message and stats update message for $\lambda = 3$ and 10%, 2% and 3% of route request/route reply messages, flow add message and stats update message for $\lambda = 6$. From Fig. 9a and Fig. 9b we observed that as the node density increases, there is a slight change in request/route reply messages, flow add message and stats update message. However, there is a drastic change in originator message and far node message. This means that originator message and far node message greatly depends on network size and its topology. The route request and route reply message directly proportional to nodes that have no information of destination nodes. It means that if all nodes have information about destination node then there is no route request message, flow add message and route reply message. Moreover, stats update message depends on nodes mobility and battery level of node and configuration update message involve changes in physical layer parameters of node.

C. CATMAN PROTOCOL COMPARISON WITH BATMAN PROTOCOL

As a proof of concept, we simulate the CATMAN protocol and evaluate its performance. We considered a battlefield

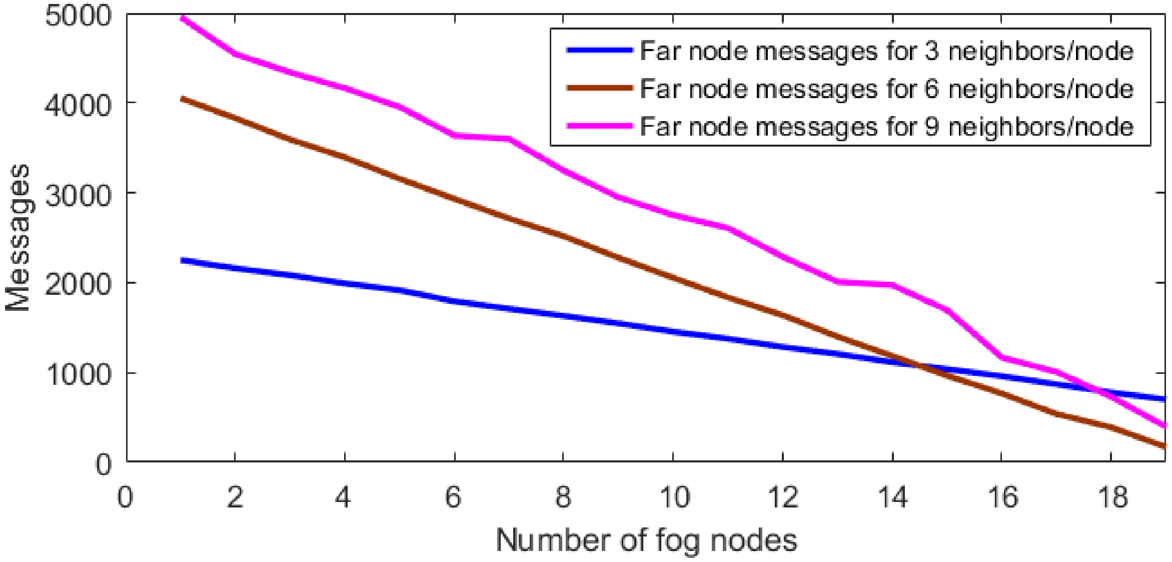


FIGURE 12. **Impact of Fog nodes on 300 nodes topology.**

network where radios are using narrow band waveforms having a bit rate of 96kbps. Coordinated movement in battlefield network results in less mobility. This means that a static network is considered. We compare the performance of CATMAN protocol with BATMAN protocol.

We first test the control signaling overhead. For this, we increase number of nodes and measure the total number of messages sent in both the protocols. In CATMAN protocol, we restrict the number of control messages transmitted by every node. As we assumed that only first hop nodes broadcast the received originator message and fog node do not broadcast any message. This helps to decrease the number of originator messages. After a convergence time of 3 minutes, if a node do not have an information about the fog node it will transmit a far node message. In Fig. 11c, we increased number of nodes and measures the control overhead. We observed that for number of nodes between 12 to 15 there is a sharp increase in number of control messages. This is due to the fact that node 12, 13, 14 and 15 are two hops away from the fog node. This means that these nodes are also broadcasting far node message that results in sharp increase in control messages. Similarly, for node 33, there is again a sharp increase in number of control messages. Since this node is approximately middle of topology and is 4 hops away from three fog nodes and 3 hops away from one fog node. In Fig 11d, we measured the latency of our proposed protocol. For this test we fixed a topology to 33 nodes and increase number of flow in a network and measure the delay. It is evident from the results that the increase in network traffic also increases the delay because the traffic is generated by those source nodes that had no information about the destination node. So, they first request the fog node for destination route and then they send message. In case of BATMAN protocol, all nodes have information about other nodes. So, the time required to get the information about the destination node is negligible.

We also analyzed both protocol with two other network i.e. $N = 300$, $\lambda = 3$ and $N = 300$, $\lambda = 6$. For both the network scenarios we measured the number of bytes transmitted and delay as shown in 11.

D. IMPACT OF FOG NODES

In this section, we evaluate CATMAN protocol by varying the number of fog nodes in 6 different scenarios. During our first three simulation scenarios, we fixed the number of nodes to 300. For 300 nodes we generate three scenarios having 3 neighbors/node ($\lambda = 3$), 6 neighbors/node ($\lambda = 6$) and 9 neighbors/node ($\lambda = 9$). For each different scenarios we increased number of fog nodes and measured the impact on far node messages as shown in Fig. 12. The number of originator messages are approximately fixed for each different scenario and are 1100, 1750 and 2100 respectively. In Fig. 12, we have noticed that as the number of fog nodes increases, the number of far node messages decreases, however, we have seen a noticeable difference in the slop of each far node message graph. The slop increases as the λ increases. Initially, with only 1 far node the number of far node messages are 2350, 4050 and 5000 respectively. However, for $N = 300$ as the fog node reaches 18 the number of far node messages in case $\lambda = 9$ is less as compared with $\lambda = 6$. This is because for higher density more nodes has information about fog node.

In order to test the scalability of CATMAN protocol we increased the number of nodes to 900 and study the impact of increase in fog nodes as shown in Fig. 13. We have observed that for *1* fog node in a network the number of far node messages are high in case of highly dense network, however,

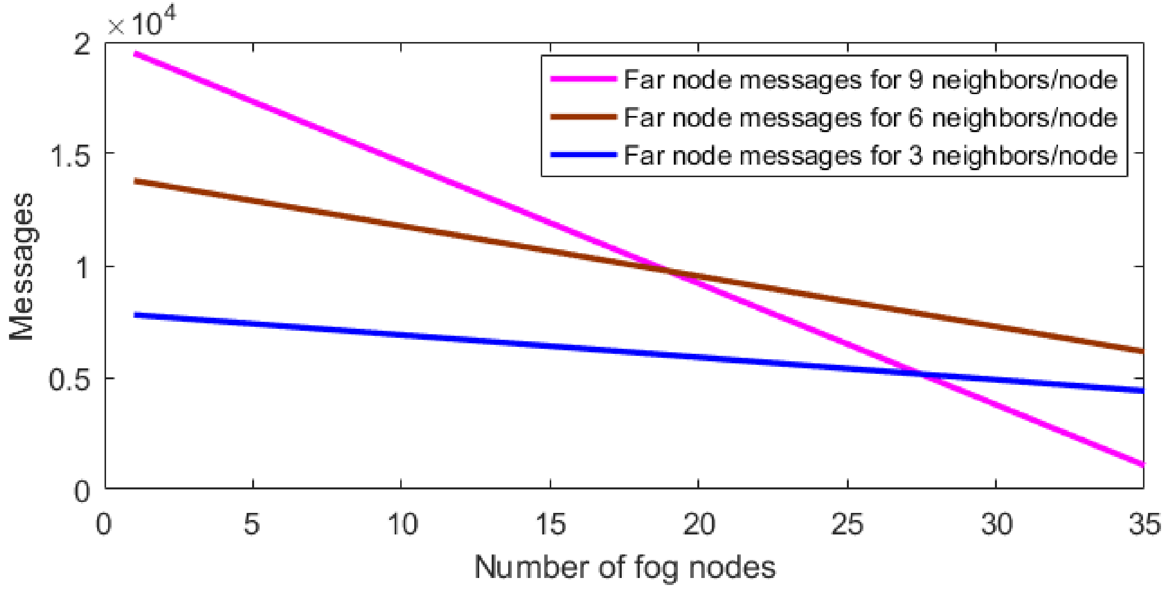


FIGURE 13. Impact of Fog nodes on 900 nodes topology.

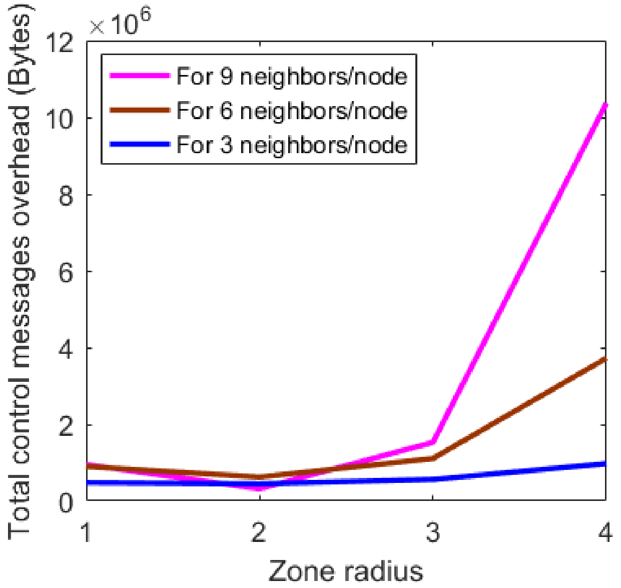


FIGURE 14. Impact of zone radius on 900 nodes topology.

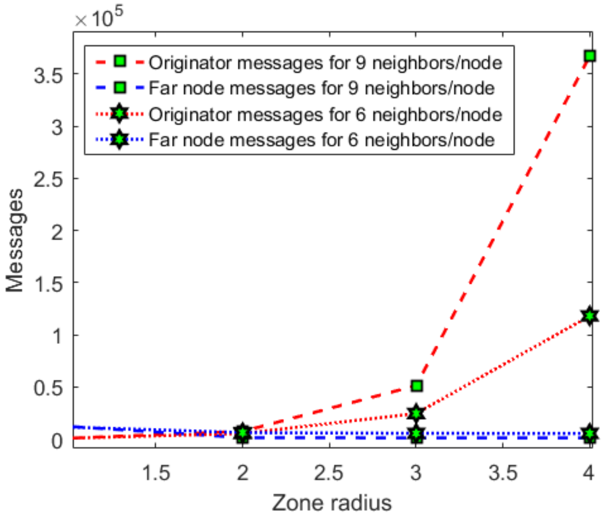


FIGURE 15. Impact of zone radius on 900 nodes topology.

as the fog node reaches *26* the number of far nodes messages for highly dense network are less as compared with other two scenarios. As, the number of far node messages reaches *36* there are few far node messages in case of highly dense network.

In these set of simulation we identified that for 6% fog nodes in a network the number of far node messages are reasonable. Although fog nodes can be increased from 6%. This means that it 6% fog nodes is the lower bound in order to get the optimum performance.

E. IMPACT ON VARIATION OF ZONE RADIUS

In this section, we study the impact of zone radius on network performance. We vary the zone radius by changing the TTL value in CATMAN protocol. Fig. 14 and Fig. 15 demonstrate the behavior of CATMAN protocol by varying the TTL value in originator and far node messages. From Fig. 15, we have seen that as ρ increases the number of originator messages for low density and high density network increases. However, there is a sharp increase in case of high density network. Similarly, we have seen that far node messages decreases with the increase in ρ for both the networks. The behavior is well understood because as ρ increases, number of

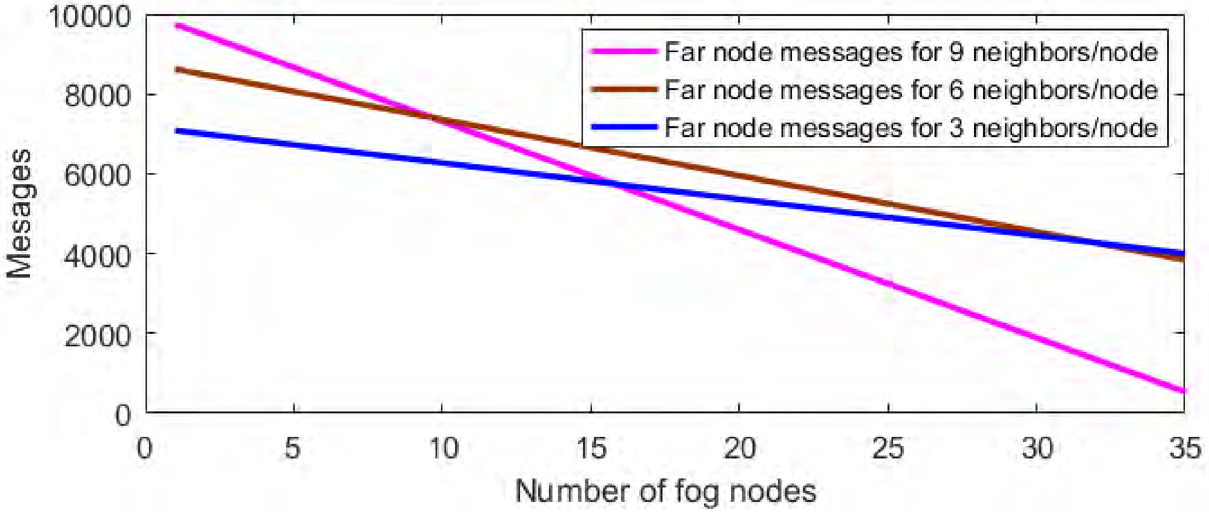


FIGURE 16. **Impact of MPRs on CATMAN protocol for 900 nodes network.**

originator messages increases resulted into more node that have information of fog node. We have observed that when $\rho = 2$ there are less number of far nodes messages in case of higher λ. Fig. 14 showed the number of bytes transmitted on the network. We have observed that as the ρ increases the number of bytes transmitted by CATMAN protocol increases for high density network. For $\rho = 2$, number of bytes transmitted for high density network is less than others two network scenarios. We conclude that to get the optimal performance $\rho = 2$.

F. CATMAN PERFORMANCE WITH MPRs

In this section we added MPRs selection in CATMAN protocol in order to reduce the number of control messages. Each node selects its MPRs node [5] and add it in originator message. Once MPRs are decided, only selected first hop nodes acting as MPRs of originator message broadcast CATMAN messages. Fig 16 show the behavior of CATMAN protocol by adding MPRs selection criteria to individual nodes. This is one of the distributed control part of CATMAN protocol where each node decides its MPRs. The selection criteria of MPRs node is taken from OLSR protocol [5]. The Fig. 16 shows the impact of MPRs on far nodes messages by increasing the fog node. We observed that MPRs has greater effect for higher density network. The result are obvious as with higher λ the number of neighbors to a node are high.

VII. CONCLUSION

In this paper, we first discussed the importance of incorporating the centralized control in WDNs and then proposed a framework and novel routing protocol called CATMAN protocol that adds a logically centralized network control in WDNs. To the best of our knowledge, this is the first attempt of a new generation routing protocol that combines centralized control and distributed control in WDNs. CATMAN combines the advantages of proactive routing protocol, reactive routing protocol, zone routing protocol using a logically centralized network approach. There are 7 messages in CATMAN protocol. The CATMAN protocol run with and without the presence of logically centralized control. An extensive simulation is performed in order to test the internal parameters of this protocol and its comparison with BATMAN protocol. The results shows that the CATMAN protocol generates less signaling overhead as compared with BATMAN protocol and optimal performance of proposed protocol is attained by setting zone radius = 2. While computation of routing path, the proposed protocol also takes care of low battery life of node and increases the overall life of a network. The future work includes the design of a centralized control plane that incorporates geographically distributed nodes, testing of configuration update message in case of SDR nodes.

REFERENCES

[1] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, 2008.

[2] J. Jo, S. Lee, and J. W. Kim, "Software-defined home networking devices for multi-home visual sharing," *IEEE Trans. Consum. Electron.*, vol. 60, no. 3, pp. 534–539, Aug. 2014.

[3] S. Jain *et al.*, "B4: Experience with a globally-deployed software defined WAN," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 3–14, 2013.

[4] A. Neumann, C. Aichele, M. Lindner, and S. Wunderlich, *Better Approach to Mobile Ad-Hoc Networking (BATMAN)*, IETF draft, 2008, pp. 1–24.

[5] T. Clausen and P. Jacquet. *Optimized Link State Routing Protocol (OLSR)*, document RFC 3626, 2003.

[6] C. Perkins, E. Belding-Royer, and S. Das, *Ad Hoc on-Demand Distance Vector (AODV) Routing*, document RFC 3561, 2003.

[7] R. Sanchez-Iborra and M.-D. Cano, "JOKER: A novel opportunistic routing protocol," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1690–1703, May 2016.

[8] M. Wisniewski, A. Dobkowski, G. Pater, R. Matyszkiel, P. Kaniewski, and B. Grochowina, "Test results of polish SDR narrowband radio," in *Proc. Commun. Inf. Technol. (KIT)*, Oct. 2017, pp. 1–6.

[9] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 243–254.

[10] M. Zhu, J. Cao, D. Pang, Z. He, and M. Xu, "SDN-based routing for efficient message propagation in VANET," in *Proc. Int. Conf. Wireless Algorithms, Syst., Appl.*, 2015, pp. 788–797.

[11] R. Soua *et al.*, "SDN coordination for CCN and FC content dissemination in VANETs," in *Ad Hoc Networks*. Cham, Switzerland: Springer, 2017, pp. 221–233.

[12] I. Yaqoob, I. Ahmad, E. Ahmed, A. Gani, M. Imran, and N. Guizani, "Overcoming the key challenges to establishing vehicular communication: Is SDN the answer?" *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 128–134, Jul. 2017.

[13] U. Shaukat and Z. Anwar, "A fast and scalable technique for constructing multicast routing trees with optimized quality of service using a firefly based genetic algorithm," *Multimedia Tools Appl.*, vol. 75, no. 4, pp. 2275–2301, 2016.

[14] M. Mueck *et al.*, "ETSI reconfigurable radio systems: Status and future directions on software defined radio and cognitive radio standards," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 78–86, Sep. 2010.

[15] C.-X. Wang *et al.*, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[16] B. M. Khan, R. Bilal, and R. Young, "Fuzzy-TOPSIS based cluster head selection in mobile wireless sensor networks," *J. Elect. Syst. Inf. Technol.*, to be published, doi: 10.1016/j.jesit.2016.12.004.

[17] T. Clausen, C. Dearlove, P. Jacquet, and U. Herberg. *The Optimized Link State Routing Protocol Version 2*, document RFC 7181, 2014.

[18] J. Hu, L. Xu, Y. Tian, L. Liu, and S. Blakeway, "A trustworthy and energy-aware routing protocol in software-defined wireless mesh networks," *Comput. Elect. Eng.*, vol. 64, pp. 407–419, Nov. 2016.

[19] Y. Wei, M. Song, and X. Wang, "Flow-table updating strategy for efficient use of renewable energy in software defined wireless relay networks," *J. Commun. Netw.*, vol. 19, no. 6, pp. 605–617, 2017.

[20] M. Abolhasan, J. Lipman, W. Ni, and B. Hagelstein, "Software-defined wireless networking: Centralized, distributed, or hybrid?" *IEEE Netw.*, vol. 29, no. 4, pp. 32–38, Jul. 2015.

[21] M. Polychronakis, E. P. Markatos, K. G. Anagnostakis, and A. Oslebo, "Design of an application programming interface for ip network monitoring," in *Proc. IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, vol. 1, Apr. 2004, pp. 483–496.

[22] W. Roh *et al.*, "Millimeter-wave beamforming as an enabling technology for 5G cellular communications: Theoretical feasibility and prototype results," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 106–113, Feb. 2014.

[23] S. Sun, L. Gong, B. Rong, and K. Lu, "An intelligent SDN framework for 5G heterogeneous networks," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 142–147, Nov. 2015.

[24] S. Sun, M. Kadoch, L. Gong, and B. Rong, "Integrating network function virtualization with SDR and SDN for 4G/5G networks," *IEEE Netw.*, vol. 29, no. 3, pp. 54–59, May/Jun. 2015.

[25] M. Mathew, N. Weng, and L. J. Vespa, "Quality-of-information modeling and adapting for delay-sensitive sensor network applications," in *Proc. IEEE 31st Int. Perform. Comput. Commun. Conf. (IPCCC)*, Dec. 2012, pp. 471–477.

[26] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Boston, MA, USA: Springer, 1996, pp. 153–181.

[27] N. Beijar, "Zone routing protocol (ZRP)," Netw. Lab., Helsinki Univ. Technol., Espoo, Finland, Tech. Rep., 2002, vol. 9, pp. 1–12.

[28] X. Ji, H. Yu, G. Fan, and W. Fu, "SDGR: An SDN-based geographic routing protocol for VANET," in *Proc. IEEE Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Dec. 2016, pp. 276–281.

[29] M. Hammoudeh, "Applying wireless sensor networks to solve real-world problems," in *Proc. Int. Conf. Intell. Inf. Process., Secur. Adv. Commun.*, 2015, p. 1.

[30] H.-H. Cho, C.-F. Lai, T. K. Shih, and H.-C. Chao, "Integration of SDR and SDN for 5G," *IEEE Access*, vol. 2, pp. 1196–1204, Sep. 2014.

[31] W. Wang, G. Yu, and A. Huang, "Cognitive radio enhanced interference coordination for femtocell networks," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 37–43, Jun. 2013.

[32] X. Xu, G. He, S. Zhang, Y. Chen, and S. Xu, "On functionality separation for green mobile networks: Concept study over LTE," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 82–90, May 2013.

[33] K. I. Pedersen, P. H. Michaelsen, C. Rosa, and S. Barbera, "Mobility enhancements for LTE-advanced multilayer networks with inter-site carrier aggregation," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 64–71, May 2013.

[34] R. Sanchez-Iborra and M.-D. Cano, "JOKER: A novel opportunistic routing protocol," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1690–1703, May 2016.

[35] M. R. Pearlman and J. Zygmunt Haas, "Determining the optimal configuration for the zone routing protocol," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 8, pp. 1395–1414, Aug. 1999.

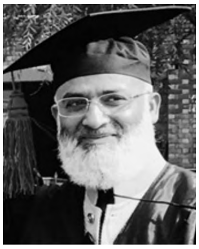
MALIK NAJMUS SIRAJ received the B.Sc. degree in electrical engineering and the M.S. degree in computer engineering from the University of Engineering and Technology Taxila in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree in heterogeneous SDN enabled networks. He has over seven years of academic experience at the Center for Advanced Studies in Engineering and one year of industry experience in CARE (Pvt) Ltd, Islamabad, Pakistan. His areas of expertise are digital system, software defined networks, and wireless networks.

ZAHEER AHMED is currently serving as the Director at CARE (Pvt.) Ltd, Islamabad, Pakistan. He also serves as an Associate Professor with the Faculty of the Center for Advanced Studies in Engineering. His work experience spans over more than 25 years in public and private sector organizations. He also possesses 16 years of academic experience as a visiting faculty member in a number of public/private sector institutions both at undergraduate and graduate level. He has one U.S. patent to his credit. His major areas of expertise are software defined networks, heterogeneous network design, customer experience management for 2G/3G/4G networks, network protocol analysis/design, DPI-based system design, cyber Security, VOIP, big data analysis, enterprise software design, FPGA/ASIC design, and embedded systems.

MUHAMMAD KASHIF HANIF received the Ph.D. degree from the Hamburg University of Technology. He is currently an Assistant Professor at the Department of Computer Science, Government College University, Faisalabad. His research interests cover big data analytic, computer network, and scientific computing.

MUHAMMAD HASANAIN CHAUDARY received the Ph.D. degree from the Asian Institute of Technology, Thailand, in 2012. He has several years of teaching and research experience. He is currently an Assistant Professor with the COMSATS University Islamabad, Lahore, Pakistan.

SHOAB AHMED KHAN received the Ph.D. degree from the Georgia Institute of Technology in 1995. He worked with companies like Scientific Atlanta, Ingersoll Rand, and Cisco Systems in the area of signal processing and communication systems. He is currently a Founder of a garage base startup Avaz Networks, now with $ 17 Million in venture funding and the Center for Advanced Research in Engineering (CARE Pvt Ltd). The company has successfully developed a satellite burst modem for STM wireless and seven first time right ASICs for NEC, Scientific Atlanta, ITEX, and Nortel. Recently, the company has delivered the industry highest density Media Processor for Voice over Packet market, a complex System on Chip with 12 processors on it. He is a CTO and a lead designer of this chip. He has authored *Digital Design of Signal Processing System: A network Approach* (John Wiley and Sons, Ltd). He is also a Professor with the College of Electrical and Mechanical Engineering, National University of Sciences and Technology. He is an Inventor of five awarded U.S. patents and has over 260 international publications. He received Tamgh-e-Imtiaz (Civil), the National Education Award 2001, and the NCR National Excellence Award in Engineering Education.



NADEEM JAVAID received the bachelor's degree in computer science from Gomal University, D. I. Khan, KPK, in 1995, the master's degree in electronics from Quid-i-Azam University, Islamabad, Pakistan, in 1999, and the Ph.D. degree from the University of Paris-Est, France, in 2010. He is currently an Associate Professor and the Founding Director of the ComSens (Communications over Sensors) Research Lab, Department of Computer Science, COMSATS University Islamabad, Islamabad, Pakistan. He has supervised 12 Ph.D. and 85 master's theses. He has authored over 650 articles in technical journals and international conferences. His research interests include: energy optimization in smart grid clouds and in IoT enabled wireless sensor networks, and big data analytics in smart grids. He was a recipient of the Best University Teacher Award from the Higher Education Commission of Pakistan in 2016 and the Research Productivity Award from the Pakistan Council for Science and Technology in 2017. He is also an Associate Editor of the IEEE Access Journal and an Editor of the *International Journal of Space Based and Situated Computing*.