Geo-Location Oriented Routing Protocol for Smart Dynamic Mesh Network

Ashish Nanda, Priyadarsi Nanda and Xiangjian He
School of Computing and Communications, Faculty of Engineering and IT
University of Technology, Sydney, Australia
Ashish.Nanda@student.uts.edu.au,
{Priyadarsi.Nanda,Xiangjian.He}@uts.edu.au

Abstract— **Wireless Mesh Network is an emerging technology with great potential to become a Self-Sustained Network. Unlike the traditional networks that dominate the current communication system and rely on a large and expensive setup of wired/wireless access points to provide connection between users, the Wireless Mesh Network is formed by the user devices (referred as Nodes) which connect to each other to form a network. However, due to the use of legacy/traditional network models for mesh networks, there exist various limitations towards its implementation. This paper presents a new approach towards the Wireless Mesh Network, incorporating a new routing scheme based on the Geo-Location of the devices. It puts forward the structure, working principle and its performance during the first implementation.**

Keywords*— Wireless Mesh Network, Mobile Ad-hoc Networks (MANET), Geo-Location Oriented Routing (GLOR), Smart Device Network, Peer to Peer Network.*

I. INTRODUCTION

Mesh networks are known for their reliability as they are formed by several connected devices (nodes) through which the messages are relayed using either a flooding technique or a routing technique. This is achieved by hopping the message from one node to another until it reaches the destination. The mesh network also has the ability of self-healing allowing a routing based network to operate when a node breaks down or when a connection becomes unreliable by automatically creating a new one.

Wireless mesh networks have been around for a while and have been used to build distributed networks, connect satellites for satellite calling and even for data collection from electricity meters. The mesh network topology has been under examination and experimentation to achieve a network model that is self-sustained, secure, scalable and dynamic. In the past few years, it has been identified that mesh networks hold the potential of becoming the network of the future, however only a few attempts have been made to achieve it.

The current versions/implementations face various challenges, amongst which a major concern arises from the fact that the data-packet is re-routed and hopped from one node to another. This results in a limitation to the number of nodes a network can maintain. If the number of nodes increase in the given network, a central controller/access point is required to manage the network that would compromise the connectivity if it fails.

In pursuit of overcoming such limitations, a new network model had to be developed. However, the new network model had several features that could not be achieved using current routing protocols, hence we propose Geo Location Oriented Routing (GLOR) protocol which is a secure, smart and dynamic solution for the new mesh network model. Since devices are becoming smarter and possess higher hardware configurations, GLOR protocol incorporates various new features and a totally remodeled approach towards security, authentication, packet routing, network formation and addressing scheme.

This paper begins with Section II which presents existing approaches, routing models and traditional protocols on mesh routing. It also discusses origin of these protocols their implementation and limitations. Section III presents our proposed scheme with new Smart Approach and Geo-Location Oriented Routing. we present the new network model, the smart packet design and a new addressing scheme in this section along with the functioning of the GLOR protocol and how it is different from traditional protocols. Section IV presents our implementation and the results to validate our model. The paper finally concludes in Section V where we present current progress and future work.

II. RELATED WORKS

The Smart Phone Ad hoc Networks (SPAN) project [1] showcased the first practical implementation of an off-grid network. The routing technique used to implement the network was OLSR (Optimized Link State Routing) which was modified to support the standalone network. The approach showed promising capabilities for off-grid communication using the mesh network. SPAN project also revealed various issues during the testing phase and highlighted a big flaw in the OLSR routing due to which the network self-saturated by 'hello' packets.

A similar approach known as the Several Project [3] was founded in response to the Haiti Earthquake. This project allowed live voice calls whenever the mesh is able to find a route between the participants. It aimed to provide support during disaster relief and recovery operations. A demo of this concept was conducted in an environment which was designed to simulate an after earthquake scenario. Various mobile devices were randomly placed around the complex and a demo rescue mission was showcased. The network was successfully formed by the devices, the trapped victim was able to easily contact the rescue personal and the victim's location was also triangulated using the network.

However, this project is based on Rhizome (Delay Tolerant Networking) system, according to which the data does not have to travel from its origin to its destination instantly, once a device has received data, it may store it and pass it on at another time and place when a connection is available. Also like SPAN, the application built to implement this approach can only support a limited varity of devices. In addition, the approach includes an external hardware called the "Mesh Extender" which is used to extend the range of the network. This makes the network dependent on the hardware and hence less reliable.

Open Garden's FireChat [4] is another such implementation. Its mobile application received over 5 million downloads and became popular during protests when the internet access was disabled. This scheme implements broadcast routing and features various types of features which enable you to choose the proximity, range and number of devices that you wish to communicate with. Despite the popularity, the methodology lacks security as each message will be sent to every device on that network similar to the concept of a chat room.

The BRIAR Project [5] is another open source software for mesh networking technology, designed to provide secure and resilient peer to peer communications with no centralized servers and minimal reliance on external infrastructure. However, the approach once again follows Delay Tolerant Network and in order to implement high levels of security, the devices don't communicate directly unless their owners are common contacts. In other words, it means that a device 'A' can communicate with a device 'C' through another device 'B' only if the device 'A' and device 'C' exist as contacts on device 'B'. This makes it difficult for the network to expand or improve functionality.

A. Routing Models

All the above network implementations are based on two major transmission techniques, namely Flooding/Broadcasting and Unicast/Multicast

Flooding/Broadcast Technique: In the flooding/broadcast technique, each node in the network retransmits the received packet to all connected nodes thereby flooding the network until the packet finally reaches the destination node. This particular method was implemented by Open Garden [4] in their mobile application 'FireChat'. This approach is applicable to a large network but it increases the load on each node as with the increase in the number of nodes, and with each node retransmitting every packet, the traffic on the network increases rapidly. This results in usage of more resources and in some scenarios it could even lead a device to crash. In addition, the communication in the network is open and each packet of data is received and read by every other node in the network thereby compromising the privacy.

Unicast/Multicast Technique: The Unicast/Multicast technique implement a predefined path through which a packet is sent. It supports a limited number of static devices/nodes. Each node broadcasts a "HELLO" message and stores the location of every other node on the network for calculating a route when a packet is to be transmitted. This makes it difficult to upscale the network. The routing protocol also has a major flaw; it was found to saturate the network with "HELLO" packets during normal operation as the number of connected devices increased. In order to support a comparatively larger number of devices, it requires a central node/entity/gateway that stores all information regarding the nodes and controls the network by calculating routs which negates with the very basic principle of a mesh network, its ability to be self-sustained.

B. Legacy Routing Protocols

Since the introduction of the Mesh Network, a few network protocols have been developed and various others have been modified in order to work with mesh topology. Optimized Link State Routing (OLSR) protocol [6, 7, 8, 9] is one such protocol. It is developed using optimization of the classical link state algorithm and modified in accordance to the requirements of a mobile wireless LAN. The key concept used in the protocol is centered on Multi-Point-Relays (MPRs) [11]. MPRs are selected nodes which forward broadcast messages during the flooding process. The protocol was originally designed to work with wired mesh networks, i.e. it is structured to work only on static devices. As mentioned in the SPAN project [1], the protocol is known to over flood the network with "Hello" messages resulting in network saturation.

Ad hoc On-Demand Distance Vector (AODV) routing [10] is another such protocol designed for mobile ad hoc network. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the ad hoc network. The protocol performs well in small networks, but as the number of nodes increases, it starts to fail as it depends upon saving connectivity information of all node data within each node so as to route the packets.

Zone Routing Protocol (ZRP) [14], also referred to as Bordercast Routing Protocol (BRP) [13] is a hybrid routing framework based on various routing protocols, designed to support mobile ad-hoc networks (MANET). Each node maintains a route within a local region (known as the routing zone). Knowledge of the routing zone topology is used by the protocol to improve the efficiency of the routing mechanism. As ZRP/BRP is a combination of various other protocols, it inherits both the merits and demerits of other protocols.

III. GEO-LOCATION ORIENTED ROUTING (GLOR)

Geo Location Oriented Routing (GLOR) is designed as a hybrid routing protocol with the aim of supporting large, dense & dynamic networks without compromising the reliability and security of the network and the devices in it. To achieve the aim, a new network model was created that is unlike any legacy or existing models. One main distinguishing factor of the new approach is that unlike the existing approaches, it utilizes the high performance capabilities of smart devices that possess high hardware configuration. This smart approach provides a new platform for improvements in various aspects.

1) Reverse Network Model:

Unlike the traditional approach, where the network is responsible for the nodes, here nodes are responsible for the network. For example, the node address (geo-location) is calculated and provided by the node itself instead of the network providing one. Similarly, tasks like node registration, node monitoring, packet routing, address allocation etc. are monitored by the nodes.

2) Secure Routing:

The routing protocol uses a simple but strong security measures. A three level security is used, namely Authentication-Encryption-Monitoring. Authentication is done when a device connects to the network. Encryption is achieved by implementing end-to-end encryption using public-private key. Monitoring is used to find unwanted nodes in the network or nodes with a malicious intension.

3) New Addressing Scheme:

Unlike traditional methods, the smart approach uses geo-location of a device as its address (described in Section III A). The geo-location is obtained using GPS or is calculated by nearby nodes. This provides us with the instantaneous position of each node, like dots on a fixed canvas.

4) Smart Packets:

As the protocol uses geo-location for node addressing, the data packet format has been modified. It implies that once a packet is created, a predefined route is not required. The packet knows the destination address, i.e. the geo-location of the destination node as well as its current geo-location. From this information the packet automatically calculates its transmission path (described in Section III C).

The functioning of the protocol is further explained using a network scenario as shown in Fig. 1. The steps of the routing process along with the line of connectivity are presented in Fig. 2. Table 1 defines certain components of the smart approach and GLOR protocol.

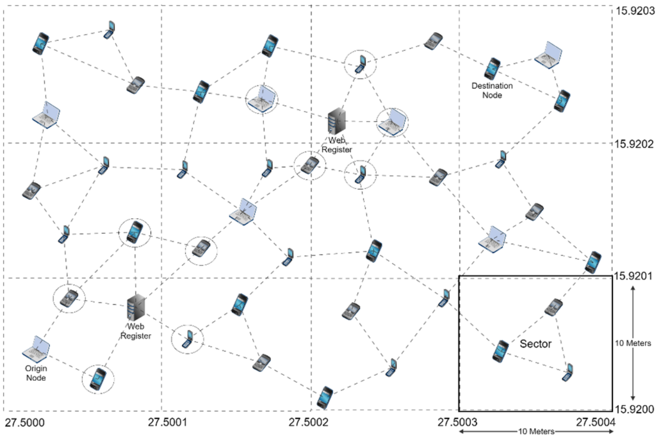


FIG. 1. SCENARIO OUTLINE

TABLE 1. COMPONENTS OF GLOR.

Component	Definition
Node	An electronic device (e.g. Smart-Phone, Laptop, and Tablet) that implements Geo-Location Oriented Routing (GLOR).
Normal Node	A node which has the capability to connect to other devices wirelessly and implements GLOR protocol.
Web Node	A Normal Node with the capability to connect directly to the Web Register.
Neighbor Node	A node X is said to be the neighbor node of Y if there exists a link between the node X and node Y.
Node Location	It is the Geo-Location of the Node, i.e. its latitude and longitude up to 4 decimal places and the node's Unique ID
Unique ID	The Unique ID of the node is a onetime generated Unique Identification number assigned to the Node alongside its MAC address during its first registration on the network.
Web Register	A cloud-based database dedicated for storing vital information about Nodes, including their MAC address, unique ID, Node Location, and Current State.
Sector	The Sector for a particular Node can be defined as a group of its neighboring nodes in a predefined area which improves the accuracy as each node in a sector will know other nodes in that sector. This helps redirect a packet to the destination node in case the node has changed location while the packet is being routed.

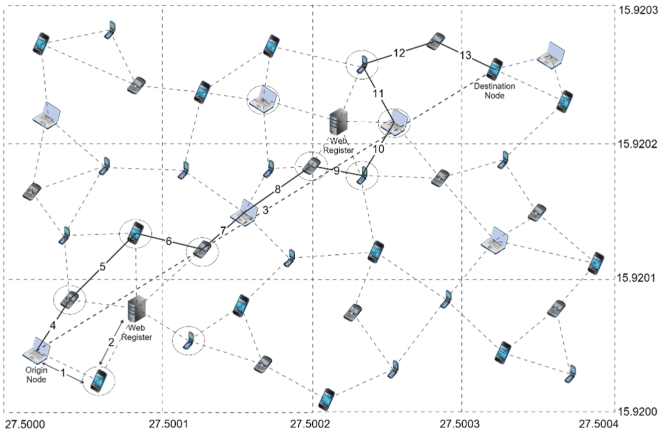


FIG. 2. DIFFERENT STEPS OF ROUTING PROCESS

A. Node Addressing

The GLOR protocol uses the IPv6 addressing format for storing the Geo-Location. IPv6 protocol offers 32 hexadecimal bits, which are further divided into eight groups of 4 hexadecimal bits each. The first 4 groups are used for storing the node location and the last 4 groups store the Sector and Cluster information of the node.

The first 4 groups are subdivided into 2 groups to store the Latitude and Longitude. The first digit represents whether the value of Latitude is positive (denoted by 1) or negative (denoted by 0), while the following 3 digits is the value before the decimal point. The next 4 digits represent the value after the decimal point. The Longitude is represented similarly. The First 8 Hexadecimal bits denote the Latitude and the next 8 bits denote the longitude, both with an accuracy of 10 meters. Fig. 3 explains the structure used to store latitude and longitude.

1	0	3	3	:	8	8	3	9	:	0	1	5	1	:	1	9	9	1
'0' if '+' '1' if '-'	0 to 90 digits before the decimal				0 to 9999 digits after the decimal					'0' if '+' '1' if '-'	0 to 180 digits before the decimal			:	0 to 9999 digits after the decimal			
Latitude										Longitude								

FIG. 3. ADDRESSING SCHEME (PART 1)

The next 4 groups store the cluster number and the sector number. Each sector represents 100 square meters of land and is defined using the Latitude-Longitude system. For example, the area enclosed by latitude 1.0000 to 1.0001 & Longitude 1.0000 to 1.0001 represents a sector as depicted in Fig. 1. The cluster is a combination of predefined sectors. Fig. 4 explains the Sector-Cluster structure used.

0	0	0	1	:	0	0	1	2	:	3	4	5	6	:	7	8	9	0
Cluster					Sector													

FIG. 4. ADDRESSING SCHEME (PART 2)

The Sectors and Clusters are calculated automatically based on the Latitude and Longitude of the node, which is based on International Standard representation of geographic point location by coordinates.

B. Node Registration

The node registration process is initiated when a new device is trying to connect or an existing device is reconnecting to the network. After being powered on, the node scans the surroundings for neighboring nodes. Once the list is populated, it will select the nearest neighbor node to initiate the handshake. On completion of the Handshake the new node requests neighbor node to start registration process explained in Fig. 5.

The process includes collection of various device/user information, its validation and accordingly going through the registration process. The first registration for any node is manual as it requires the user to fill in details manually in order to complete the registration process. If a device is re-connection to the network, it does not have to re-register itself, just pass a simple authentication challenge created by the web server and encrypted using the public key of the device.

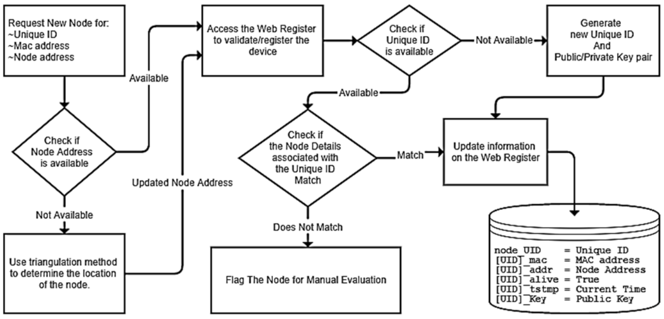

Request New Node for:
~Unique ID
~Mac address
~Node address
Access the Web Register to validate/register the device
Check if Unique ID is available
Not Available
Generate new Unique ID And Public/Private Key pair
Available
Available
Check if Node Address is available
Updated Node Address
Check if the Node Details associated with the Unique ID Match
Match
Update information on the Web Register
Not Available
Does Not Match
Use triangulation method to determine the location of the node.
Flag The Node for Manual Evaluation
node_UID = Unique ID
[UID]_mac = MAC address
[UID]_addr = Node Address
[UID]_alive = True
[UID]_tstmp = Current Time
[UID]_Key = Public Key

FIG. 5. NODE REGISTRATION PROCESS

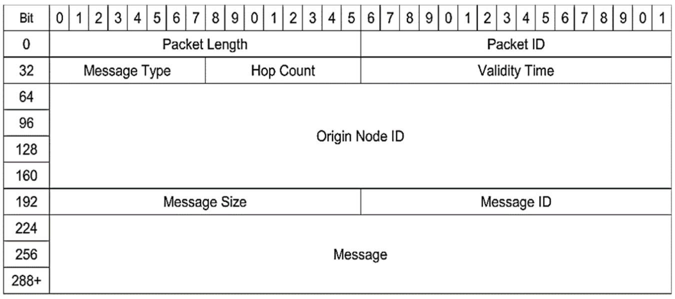
Web Register: As described before, the web register is a cloud-based database that stores vital information about the nodes. It is an application that runs on the network and its sole purpose is to improve the performance and accuracy of the network. The web register also takes the role of monitoring devices. This helps preventing un-authenticated nodes or impersonating nodes from entering the network.

However, the network is not dependent on it and can still function on a sector-broadcast mode in the absence of the web register. In the sector-broadcast mode, instead of forwarding the data packet to each node, it is forwarded to just one node in each sector. This minimizes the overhead and can be easily progressed as the sectors are defined using geo-location. If the destination device is present in that sector, it will receive the packet and can then keep updating the origin node about its location and encryption key, else the packet will be forwarded to the next sector/s.

C. Smart Packets

The GLOR protocol defines the functioning of a node in the network. This includes the universal specifications of GLOR messages, Node Registration, Packet Format & Transmission, Neighbor discovery and Routing.

Packet Format: GLOR protocol communicates using a modified packet format. The purpose is to keep it simple in order to reduce the load on the network. It helps incorporate different types of information in a single transmission which optimizes the use of max framesize. The basic layout of the packet has been updated to include the new addressing scheme and is represented in Fig. 6.



<table>
<tr><td>Bit</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td colspan="16">Packet Length</td><td colspan="16">Packet ID</td></tr>
<tr><td>32</td><td colspan="8">Message Type</td><td colspan="8">Hop Count</td><td colspan="16">Validity Time</td></tr>
<tr><td>64</td><td colspan="32" rowspan="4">Origin Node ID</td></tr>
<tr><td>96</td></tr>
<tr><td>128</td></tr>
<tr><td>160</td></tr>
<tr><td>192</td><td colspan="16">Message Size</td><td colspan="16">Message ID</td></tr>
<tr><td>224</td><td colspan="32" rowspan="3">Message</td></tr>
<tr><td>256</td></tr>
<tr><td>288+</td></tr>
</table>

FIG. 6. PACKET FORMAT (OMITTING TCP/IP HEADERS)

The simple design and minimized header size helps the packets carry more data and reduce overhead. Various components of the packet are described below.

- Packet Length - It is the length of the packet (in bytes).
- Packet ID - The Packet ID or PID must be incremented by one each time a new GLOR packet is transmitted
- Message Type - It indicates the type of the message that is being transmitted.
- Hop Count - It is the number of hops a message has attained. It is incremented every time the packet is retransmitted.
- Validity Time - It is the maximum time during which the information of the packet is considered valid. If a node receives a packet with Validity Time = 0, the packet is discarded.
- Origin Node ID - This is the ID of the node that originally generated the packet. It is not to be confused with the Source Node ID in the IP header as it is updated each time to the address on the intermediate node.
- Message Size - It is the total size in bytes measured from the beginning of "Message Type" till the end of the message.
- Message ID - A unique ID is provided to each message by the Origin Node. It is incremented by one for each message.

Packet Formation: This process defines how a packet is generated. Once the origin node is ready to send a packet, it requests the address of the destination node by providing the destination node's unique id to the web node. The web node initiates a request to accesses the web register to retrieve the information represented as step 1 & 2 in Fig. 2. Once it gains access to the web register, it checks if the unique id exists in the registry. If the unique id is not linked to a node, a 'not_found' response is then sent to the origin node.

If the unique id is found, the next step is to check if the node is still connected to the network or not. This is done by accessing the destination nodes [UID]_alive parameter. If the destination node is still connected to the network, the origin node receives the destination node's address. However, if the destination node is currently offline, the origin node receives a 'not_alive' message.

Once the Origin Node receives destination node address and the public key, it creates the packet with the appropriate information and encrypts the message part (which also includes its own public key to ensure any reply to be encrypted as well) using the destination node's public key. The packet is then processed according to the type of the messages defined in the next section.

Next Hop Calculation: Once the origin node has gathered the required information and the packet is created, the next hop is calculated according to the method depicted in Fig. 7. The same procedure is also used at every hop for the calculation of the next hop. The various parameters and math involved calculation are as follows.

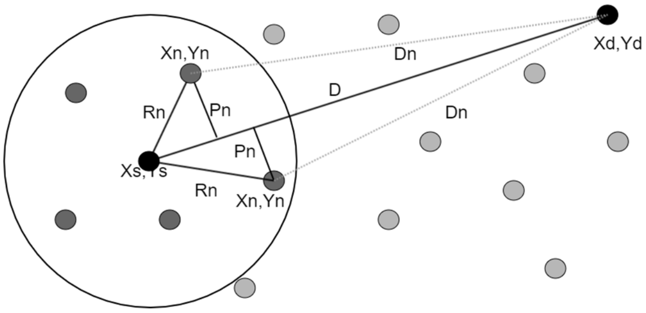


FIG. 7. NEXT HOP CALCULATION

TABLE 2. KEY FOR FIG. 7.

Variable	Description
Xs,Ys	Geo-location of the source node
Xd,Yd	Geo-location of the destination node
Xn,Yn	Geo-location of the neighboring node/s
Rn	Distance of neighbor node from source node
Dn	Distance of neighbor node from destination node
D	Distance of source node from destination node
Line D	A straight line from source node to destination node
Pn	Distance of neighbor node from line D

The distance Pn is calculated using the following equation:

$$\text{Pn} = \frac{|(Xd-Xs)(Ys-Yn)-(Xs-Xn)(Yd-Ys)|}{\sqrt{(Xd-Xs)^2+(Yd-Ys)^2}} \qquad (1)$$

The neighbor node is selected using the geo-location of the source node and the destination node. Using these two location details as two points on a graph, a straight line is plotted and then the neighbor node closest to the line and farthest to the source node is selected and the packet is transmitted to it as shown in Fig. 7. A neighbor node can be selected as the next hop if it satisfies the following conditions:

- The node should be alive and in the neighbor of the source node
- The node's distance from the destination node (Dn) should be less than or equal to the distance from source node to destination node (D).
- If there are two or more nodes that satisfy the above conditions, then a node is given preference based on the following.

— Its distance from source node (Rn) is greater
— Its distance from destination node (Dn) is less
— Its distance from line D (Pn) is less

If two or more nodes satisfy the conditions, the one with less load is selected. This process repeats itself until the packet reaches its destination.

*Packet Processing and Forwarding***:** Once a node receives a packet, it examines the header and its contents based on the message type. The process that takes place once a packet is received is presented in Fig. 8.

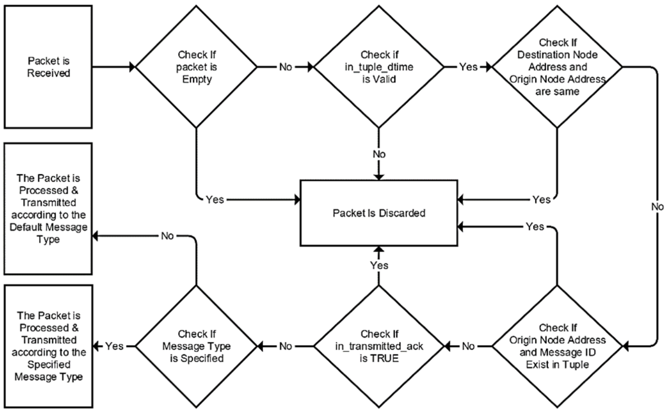

Packet is Received
Check If packet is Empty
No
Check if in_tuple_dtime is Valid
Yes
Check If Destination Node Address and Origin Node Address are same
No
Yes
Packet Is Discarded
Yes
Yes
No
Yes
The Packet is Processed & Transmitted according to the Default Message Type
No
The Packet is Processed & Transmitted according to the Specified Message Type
Yes
Check If Message Type is Specified
No
Check If in_transmitted_ack is TRUE
No
Check If Origin Node Address and Message ID Exist in Tuple

FIG. 8. PACKET PROCESSING AND FORWARDING

In order to make the system robust, the origin node can transmit the same packet multiple times or to multiple nodes. This can result in each node receiving the same packet multiple times. To prevent retransmission of the same packet, each node creates a duplicate tuple with details about the packet (in_origin_address, in_message_id, in_transmitted, in_transmitted_ack, in_tuple_dtime). In_origin_address is the origin node address, in_message_id is the message id of the packet, in_transmitted is a boolean represents if the packet was transmitted further or not, in_transmitted_ack is also a boolean representing if the acknowledgment was received or not after the packet was transmitted, in_tuple_dtime is the time after which the data expires and the tuple will be discarded.

Default Packet Forwarding: Once a packet has been processed, it is checked if it has been transmitted before. If so, the acknowledgement is checked. If an acknowledgement has been received, the packet is discarded, else the packet details are updated and is transmitted, as shown in Fig. 9.

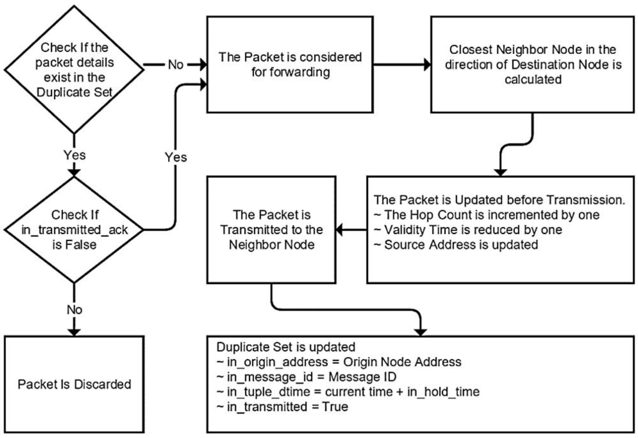

Check If the packet details exist in the Duplicate Set
No
The Packet is considered for forwarding
Closest Neighbor Node in the direction of Destination Node is calculated
Yes
Yes
Check If in_transmitted_ack is False
The Packet is Transmitted to the Neighbor Node
The Packet is Updated before Transmission.
~ The Hop Count is incremented by one
~ Validity Time is reduced by one
~ Source Address is updated
No
Packet Is Discarded
Duplicate Set is updated
~ in_origin_address = Origin Node Address
~ in_message_id = Message ID
~ in_tuple_dtime = current time + in_hold_time
~ in_transmitted = True

FIG. 9. DEFAULT PACKET FORWARDING

Node updating: Each node connected to the network will send an update to the web register informing it about its location change or to acknowledge that it is still connected to the network. The 'node_check' process handles this task and is repeated at regular intervals. The process first checks if the node has changed its location; if yes, then it checks if the change in location is more than 10 meters. If the change in location is more than 10 meters, the web register is sent a request to update the location and [UID]_alive status. If the location change is less than 10 meters, only a [UID]_alive message is sent.

The GLOR protocol along with the smart approach create a new platform for the wireless mesh network. Unlike the legacy/traditional network models, this approach offers a new method for achieving a self-sustained network. Its implementation and results are discussed in Section IV.

IV. SIMULATION AND RESULTS

As the structure of GLOR protocol vastly differs from the legacy protocols due to its unique working parameters and design. Hence, the proposed routing protocol has been developed using C# in Visual Studio Enterprise 2015 IDE and the required basic libraries were created from scratch for the new network model. The machine used for simulation is an Alienware 13 powered by a 6th Gen. Intel i7 (3.1 Ghz.) CPU and 16GB DDR3L RAM.

A. Environment Setup

The devices/nodes are represented using objects available in the Visual Studio 2015 IDE, and each object (referred to as device or node) runs GLOR protocol by default. The geo-location is calculated using the X-Y coordinates of the node according to its placement on the 2D plane. The devices are randomly distributed over the plane during testing. The web register is designed using a localized database that stores the node information. Other components such as the data packet design and various variables being used in the routing process are also defined in the library files.

The test bed has been created for simulation with the following assumptions:

- The nodes are uniformly distributed across the plane.
- The nodes have already been authenticated and have a unique id.
- None of the nodes fail during the operation.
- All nodes have the capability to calculate their location.
- No packet is dropped during the transmission process.
- Each node has a direct/indirect connection to the web register.

B. *Simulation and Observation*

Once the simulation starts, the nodes first calculate their geo-location (in this case it's X-Y coordinates on the plane). The next step is to search and connect with neighboring nodes. This step also involves creating the neighbor table which helps with the next-hop selection during the routing.

In addition to the above steps, each node also updates its information on the web register. Once the devices have connected and the network is formed, two random devices are manually selected to start an exchange of a predefined send-acknowledge packet. The scenario also traces the path taken by the packets as shown in Fig. 10. The preliminary test conducted using 20 nodes provided vital firsthand information about the setup. It also helps update the next-hop calculation method as it was found to go into an infinite loop in some scenarios.

Once the appropriate modifications were made, the final tests were conducted with 72 nodes. The test shows promising results as the GLOR protocol is able to route packets through multiple devices. It was also observed that different packets form the same device may take a different route based on its calculation of the next hop and the availability of neighboring nodes.

C. *Results and Discussion*

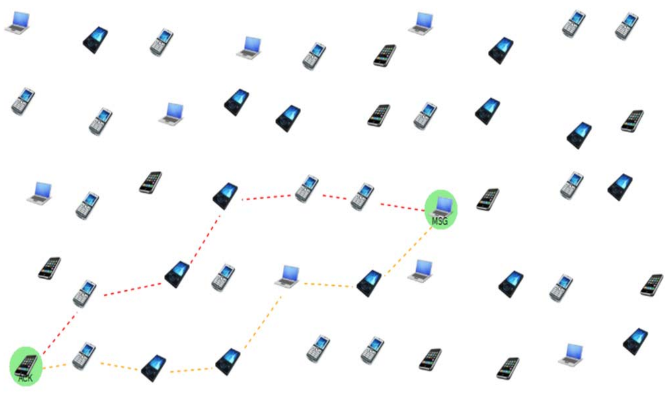


FIG. 10. INSTANCE SHOWING PACKET ROUTE TRACE

As shown in Fig. 10, the acknowledgment packet from the destination node (depicted with light dotted line) did not take the same path as the original packet (depicted with a dark dotted line). Current analysis proves that the routing can easily and efficiently adapt to a dynamic mesh network. The graph in Fig.11 shows the time taken for a complete sent-acknowledge cycle.

In addition to the time, the simulation result also shows that hardware utilization for a node to forward a packet is 11% of CPU usage with 5MB of additional RAM for a duration of 10 milliseconds.

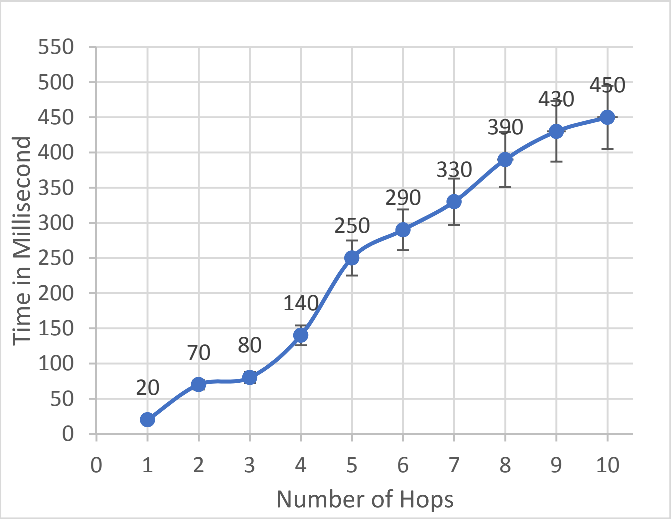


FIG. 11. MESSAGE ROUNDTRIP TIME (+/- 10%)

The results obtained from the simulation will further help to improve the performance and also reveal various scenarios which might not have been addressed so far. As the simulation moves forward, it will be expanded to include more nodes and observe the behavior and performance in such scenarios.

Since then first testing, GLOR protocol has been under constant modification to increase the efficiency, enable better security and be able to handle unique/exceptional scenarios that might arise in real world scenarios such as the dead loop or the "V" tip. Further development of the GLOR protocol will enable it to identify such exceptional scenarios and take appropriate measure to avoid them or find a way around it.

V. CONCLUSION & FUTURE WORK

The new network model with a new addressing scheme, GLOR protocol and security framework together form a very robust communication network providing end to end security. The innovative model also provides a new platform for further development of this routing technique. As it is not governed by the limitations of legacy protocols, the GLOR model also opens the doors for development of various applications that can perform considerably better as compared to the legacy network model.

The GLOR protocol is currently under modification to include an all new security framework that provides enhanced authentication for devices and end-to-end data encryption on the network. The next test-bed will also include a mobility model and the recorded results will accordingly be compared to other existing models. It will also provide more information about the performance, network load and resource requirement.

The next phase of the research will take the network model into the practical world. This will be achieved by adding the GLOR protocol to smartphones and observing the performance in real world scenarios. The valuable information received from real world implementations will be used to further improve the protocol and add more applications to it.

ACKNOWLEDGMENTS

The authors acknowledge Pulkit Rohilla for his contribution and technical assistance in implementation of GLOR protocol and setting up the scenario. We also thank Deepak Puthal and Alina Rakhi Ajayan for their comments that helped improve this paper.

REFERENCES

[1] Josh Thomas, Jeff Robble, and Nick Modly. "Off Grid communications with Android." In 2012 IEEE Conference on Technologies for Homeland Security (HST). 2012.

[2] Paul Wong, Vijay Varikota, Duong Nguyen, and Ahmed Abukmail. "Automatic android-based wireless mesh networks." Informatica 38, no. 4 (2014): 313.

[3] Paul Gardner-Stephen, "The serval project: Practical wireless ad-hoc mobile telecommunications." Flinders University, Adelaide, South Australia, Tech. Rep (2011).

[4] ‘Opengarden’. [Online]. Available: https://opengarden.com. [Accessed : 19-May-2015].

[5] Michael Rogers, Eleanor Saitta and Bernard Tyers, ‘The briar project’, [Online]. Available: https://code.briarproject.org [Accessed : 1-June-2015]

[6] Thomas Clausen, and Philippe Jacquet. Optimized link state routing protocol (OLSR). No. RFC 3626. 2003.

[7] Thomas Clausen, Justin W. Dean, and Christopher Dearlove. Mobile ad hoc network (MANET) neighborhood discovery protocol (NHDP), No. RFC 6130. 2011

[8] Thomas Clausen, Christopher Dearlove, Philippe Jacquet, and Ulrich Herberg. The optimized link state routing protocol version 2. No. RFC 7181. 2014.

[9] Christopher Dearlove and Thomas Clausen. An Optimization for the Mobile Ad Hoc Network (MANET) Neighborhood Discovery Protocol (NHDP). No. RFC 7466. 2015

[10] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (AODV) routing. No. RFC 3561. 2003.

[11] Thomas Clausen, Philippe Jacquet, Dang-Quan Nguyen, and Emmanuel Baccelli. OSPF multipoint relay (MPR) extension for ad hoc networks. No. RFC 5449. 2009.

[12] Scott Corson and Joseph Macker. Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations. No. RFC 2501. 1998.

[13] Zygmunt J. Haas, Marc R. Pearlman, and P. Samar. The Bordercast Resolution Protocol (BRP) for Ad Hoc Networks, June 2001. IETF Internet Draft, draft-ietf-manet-brp-01. txt, 2001.

[14] Zygmunt J. Haas, Marc R. Pearlman, and Prince Samar. "The zone routing protocol (ZRP) for ad hoc networks." (2002).

[15] Muhammad Shoaib Siddiqui. "Security issues in wireless mesh networks." In 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07), pp. 717-722. IEEE, 2007.

[16] Ian F. Akyildiz and Xudong Wang. "A survey on wireless mesh networks." IEEE Communications magazine 43, no. 9 (2005): S23-S30.

[17] Ivan Stojmenovic. "Position-based routing in ad hoc networks." IEEE communications magazine 40, no. 7 (2002): 128-134.

[18] Bose, Prosenjit, Pat Morin, Ivan Stojmenović, and Jorge Urrutia. "Routing with guaranteed delivery in ad hoc wireless networks." Wireless networks 7, no. 6 (2001): 609-616