International Journal of Applied Information Systems (IJAIS) ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 11 - No. 11, March 2017 - www.ijais.org

Throughput Analysis in an Infrastructure-based Mesh Network using BeeHive Algorithm

Kiran K.
University Visvesvaraya College of Engineering
Bangalore, India

Vaishnavi K.
Viterbi School of Engineering
University of Southern California
Los Angeles, CA, USA

P. Deepa Shenoy
University Visvesvaraya College of Engineering
Bangalore, India

Venugopal K. R.
University Visvesvaraya College of Engineering
Bangalore, India

ABSTRACT

WiMAX and WiFi are among the major wireless technologies escalating today. Both these technologies provide last mile connectivity and differ in their Media Access Control (MAC) layer and physical (PHY) layers. Since most of the devices are equipped with WiFi, and WiMAX is prevailing in today's devices, an efficient way of routing the data is by making use of both these technologies simultaneously. This work presents such a technique based on the Bee-hive routing algorithm for an infrastructure-based mesh network, which promotes interoperability between the WiMAX and WiFi technologies. The network consists of two types of nodes - 1)coordinator node 2)subscriber node, and divides the entire network into regions called foraging regions/zones based on the hop count from the coordinator node. A detailed implementation of this algorithm is presented in this work. The algorithm is tested for various mobile and static topologies by varying the parameters such as the speed of motion of the mobile nodes and the hop limit.

General Terms:

Long Routing Table, Short Routing Table

Keywords

Coordinator Node, Foraging Region, Subscriber Node, WiFi, WiMAX

1. INTRODUCTION

WiMAX (Wireless Interoperability for Microwave Access) is a wireless point-to-multipoint (PMP) standard approved by IEEE 802.16, which defines both the Medium Access Control (MAC) and the physical (PHY) layers. It is a MAN standard providing last mile connectivity. It provides a data rate of upto 376 Mbps[1] and covers a range of 10 Kms to 50 Kms[2]. It operates in the 10 to 66 GHz licensed and unlicensed band[2]. A WiMAX base station can support more than 500 subscriber stations[3].

WiFi (Wireless Fidelity) is a wireless LAN standard based on IEEE 802.11. It provides a data rate of upto 6.933 Gbps[4] and covers a range of upto 1kms[5]. It operates in the 2.44 to 5 GHz unlicensed band[2]. A WiFi router was designed to handle upto 10 subscriber stations at a given instance of time[3]. However, today's WiFi routers can handle a larger number of devices.

An infrastructure-based network is a centralized network consisting of base stations or coordinator nodes and subscriber stations. The overall network is divided into smaller regions with each region consisting of one coordinator node. A message destined for any node within the region is sent to the coordinator node which then forwards it to the destination node. Hence, all communications happen through the coordinator node. In this work, communication across the regions happens through this mechanism. A mesh network consists of peer nodes functioning in a decentralized fashion. The messages are routed directly to the destination node without any centralized control. This is the mode of communication used for nodes within a region in this work.

In this work, an algorithm based on the bee-hive routing protocol is presented for a heterogeneous infrastructure-based mesh network consisting of two kinds of nodes - coordinator node and subscriber node. The bee-hive algorithm divides the entire network into regions called foraging regions or foraging zones. In this work, it is assumed that the coordinator nodes are equipped with both WiMAX and WiFi radios whereas the subscriber node's minimum requirement is a WiFi radio. Routing within the region happens directly in a decentralized mesh mode using the WiFi radio. Inter-foraging zone routing is done through the coordinator node using WiMAX.

The simulations are performed using the OPNET Modeller, a user-friendly network simulator[6].

2. RELATED WORK

A fundamental understanding of WiMAX can be obtained from [7]. The WiMAX and WiFi standards are defined in [8] and [9] respectively. Interoperability between various wireless technologies are exploited in numerous papers. Interoperability between WiMAX and WiFi is observed in [10] and in a testbed environment in [11]. A comparision of WiFi, WiMAX, MBWA and 3G is performed in [12]. A comparative study of WiMAX and WiFi networks is done in [13] and their integration is presented in [14]. [15] discusses the splitting of down-link traffic between WiMAX and WiFi heterogeneous links. Mesh and ad-hoc networks are discussed in [16] whereas a survivability survey in mobile ad-hoc networks is performed in [17]. [18] discusses the network architecture for heterogeneous mobile computing. A proactive end-to-end mobility solution is given for roaming across a heterogeneous wireless network in [19]. One of the fundamental routing algorithms, Ad hoc On-Demand Distance Vector Routing was presented in [20]. [21] provides a mechanism to route data in a multi-radio, multi-hop wireless mesh network. The performance of wireless routing protocols in mobile WiMAX environment is evaluated in [22]. [23] gives a mechanism for double technology routing using WLAN-WiMAX. [24] presents a routing protocol for a hybrid WiFi-WiMAX network. Bio-inspired algorithms are the ones based on the social behaviour of natural beings and are discussed in [25]. Antnet routing[26] is one such example of a bio-inspired algorithm. The algorithm devised in this work finds its basis from the bee-hive routing protocol [27]. [28] performs a survey of different algorithms based on the swarm intelligence of bees.

3. PROBLEM STATEMENT

This section discusses the problem to be addressed. The prime objective of this work is to introduce interoperability between WiMAX and WiFi technologies in a multi-radio infrastructure-based mesh environment, and analyze the performance for both static as well as mobile topologies. Inorder to achieve this, a technique based on the Bee-hive algorithm is used. An in-depth analysis of the algorithm is made by varying the parameter based on which the regions are formed and the speed at which the mobile nodes move.

4. DESIGN

The network considered in this work consists of two types of nodes - coordinator node and subscriber node. The coordinator node is equipped with both WiMAX and WiFi radios. The node model of the coordinator node is as shown in Figure 1.

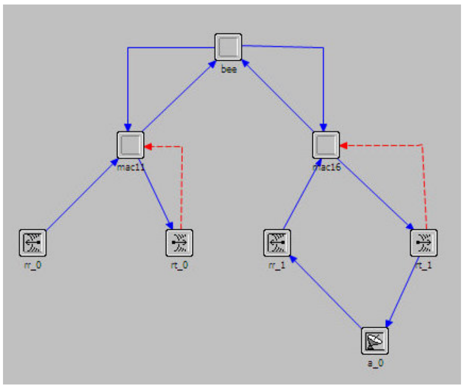

Fig. 1. Node Model of the Coordinator Node

The layers constituting the coordinator node are as follows:

—$beeLayer$: It corresponds to the Network layer of the protocol stack and handles the routing pronouncement.

—$mac11Layer$: This represents the Media Access Control layer in the Data Link layer of the protocol stack. It implements IEEE 802.11, thus represents the WiFi radio of the coordinator node.

—$mac16Layer$: It also corresponds to the Media Access Control layer in the Data Link layer of the protocol stack but implements IEEE 802.16, thus representing the WiMAX radio of the coordinator node.

—Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) algorithm is implemented by both the MAC layers. Not much emphasis is given to the details of its implementation in this work.

—rr_1, rt_1, rr_0 and rt_0: These are the radio receivers and radio transmitters corresponding to the WiMAX and WiFi radios respectively.

—a_0: It is the antenna for the WiMAX radio.

The subscriber nodes whose node model is as shown in Figure 2 have only the WiFi radio.

Its node model has the following layers:

—*src Layer*: It is analogous to the Application layer of the OSI network architecture, generating data packets required for testing the effectiveness of the protocol during the simulation. An inbuilt OPNET process model is made use of for this purpose and the data packets generated do not have any destination address attached to them.

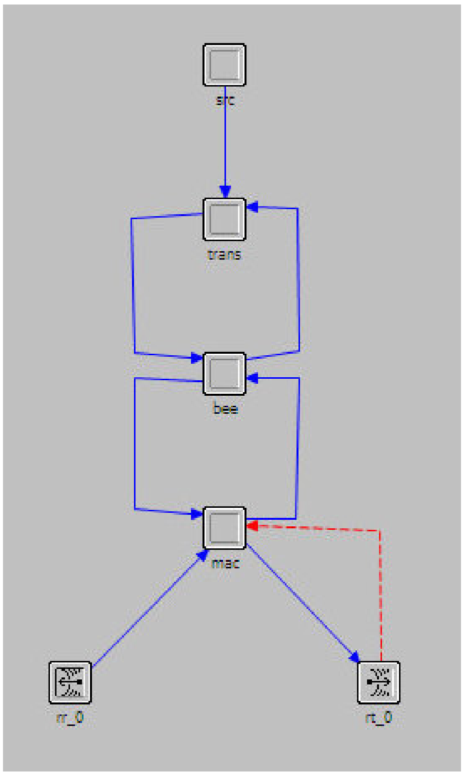

Fig. 2. Node Model of the Subscriber Node

Table 1. Long Routing Table

Coordinator Node ID	**Next Hop Coordinator Node**	**Hop Count**	**List of all subscriber nodes in this region**

—*trans Layer*: It corresponds to the Transport layer of the protocol stack.

—*bee Layer*: It takes care of the routing decisions to be made.

—$macLayer$: This is the Media Access Control layer of the subscriber node and is implemented using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). It employs the IEEE 802.11 and represents the WiFi radio.

—rr_0 and rt_0: Radio receiver and radio transmitter of the WiFi radio respectively.

4.1 Routing Tables

The coordinator node maintains two types of routing tables:

—*long routing table*:
This table maintains the information about the nodes outside its foraging region. It contains the list of other coordinator nodes and subscriber nodes that reside in them. The structure of this table is as shown in Table 1.
The $Coordinator\ Node\ ID$ field represents the coordinator node of the destination node's foraging region. The $Next\ Hop\ Coordinator\ Node$ entry contains the neighbouring coordinator through which the packet must pass to reach the destination given by $Coordinator\ Node\ ID$. The $Hop\ Count$ notifies the number of WiMAX hops from this coordinator to the destination coordinator via the next hop neighbor specified. All the routing actions performed by referring to this table are through the WiMAX radio.

—*short routing table* This table maintains the information about all the nodes within the foraging region. The structure of the short routing table is as shown in Table 2.

Table 2. Short Routing Table

Subscriber Node	Next Hop	Hop Count

Table 3. Routing Table

Node	Next Hop	Hop Count

The $Subscriber\ Node$ field gives the node IDs of the subscriber nodes in this foraging region. The $Next\ Hop$ gives the immediate neighbour to the destination in $Subscriber\ Node$ and $Hop\ Count$ gives the number of hops from this node to the destination through this neighbour. All routing actions specified in this table are performed using the WiFi radio.

Each subscriber node maintains a routing table containing information about nodes in its foraging region. The structure of the subscriber node's routing table is as shown in table 3.
This table is similar to the coordinator node's short routing table and the $Node$ field is the same as the $Subscriber\ Node$ in the coordinator node's short routing table.

4.2 Foraging Regions

The entire network is divided into regions called foraging regions or foraging zones. All nodes that are within n hops from the coordinator node belong to one foraging region. Initially, the coordinator node broadcasts a $register$ packet through its WiFi radio. When the subscriber nodes which do not belong to a region receive this packet, they update their region to that of this coordinator and acknowledge the coordinator. Upon receiving this acknowledgment, the coordinator node adds the subscriber node to its short routing table.

4.2.1 Static Network. In the case of a static network, since the nodes are static, the foraging regions are formed only once, in the beginning when the network is set up. Even if a subscriber node goes down, the foraging regions do not change. Once the failed node comes up again, it will remain to belong to its previous foraging region. Hence, in a static network, formation of the foraging regions is a one-time activity.

4.2.2 Mobile Network. In a mobile network, since the nodes are mobile in nature, the subscriber nodes can go out of one foraging region and enter another. Hence, new foraging regions must be formed at regular intervals so that the routes maintained by the routing tables are optimal and up to date.

4.2.3 Hop Limit. The *hop limit* is specified by the parameter n used for forming the foraging regions. The hop limit controls the size of the foraging regions and hence affects the overall network throughput. An analysis of the performance is made by varying the hop limit.

4.2.4 Speed of mobility of the mobile nodes. Higher the speed of mobility, faster the nodes move out of the foraging regions, which in turn requires the foraging regions to be re-formed at shorter intervals. An analysis of the overall network performance is made by varying the speed of mobility for various topologies.

4.2.5 Unreachable Node. It is common to find a node p which is more than n hops away from any of the coordinator nodes, but is within the WiFi range of some subscriber node. In this case, though the node p is within the WiFi range of a subscriber node, it becomes an unreachable node since no coordinator nodes can reach it. A technique is provided in order to include such nodes for communication. The unreachable node p broadcasts a $region\ request\ bee$ packet. Any node belonging to a region which receives the packet replies with the $region\ request\ bee\ ack1$ consisting of its region number. The unreachable node p acknowledges to the first $region\ request\ bee\ ack1$ packet with a $region\ request\ bee\ ack2$ packet. The respective subscriber node then informs its coordinator about the new node that is in its region and the coordinator adds p to its short routing table.

4.3 Creation and Maintenance of Routing Tables

4.3.1 Coordinator Node's Short Routing Table. The short routing table is created as a by-product of the formation of foraging regions. When the acknowledgment to the register packet is received from the subscriber node, the respective subscriber node is added to the short routing table of the coordinator node.

4.3.2 Coordinator Node's Long Routing Table. Once the initial short routing tables are formed, the coordinator nodes exchange their short routing tables in the form of $long\ bee$ packets. When a coordinator node receives a long bee from another coordinator node, it updates its long routing table entries to the routes with the smallest hop count. The coordinator nodes then exchange their long routing tables in the form of long bee packets and hence update to the optimal routes. All the long bee packets are sent through the WiMAX radio, hence are received only by the other coordinators.

4.3.3 Subscriber Node's Routing Table. Once the coordinator node creates the short routing table, it is broadcast to all the subscriber nodes in the form of $short\ bee$ packets. The subscriber nodes, on receiving this packet, create their initial routing tables and then exchange their routing tables among themselves in the form of short bee packets in order to find the optimal routes. All the short bee packets are exchanged via the WiFi channel.

4.3.4 Static Nodes. In a static network, since the nodes are stationary, the routes are formed only once when the network is set up. There is no updation of the routing tables after their initial stabilization.

4.3.5 Mobile Nodes. Since the updation of the routing tables follows the formation of foraging regions, which occurs at regular intervals in mobile nodes, the routes are regularly updated in a mobile network.

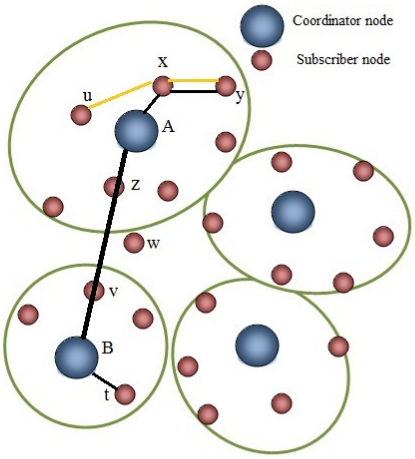

Fig. 3. Example Network

4.4 Exchange of Data Packets

The data packets can be exchanged either within the foraging region or outside the region. Consider the network shown in Figure 3 consisting of four foraging regions each with one coordinator node and a few subscriber nodes.

4.4.1 Intra-foraging Region Routing. Within the region, the data packets are routed in a mesh mode. The source node checks for the destination in its routing table to find the next hop node and forwards the packet to it through the WiFi channel. This process is repeated until the packet reaches the destination node. In Figure 3, node *y* wishes to communicate with node *u*. Node *y* searches its routing table for destination node *u* and finds the next hop node to be *x*. Hence, the packet is forwarded to node *x*. Upon receiving the packet with destination *u*, node *x* searches it routing table for node *u*. The next hop to *u* is *u* itself and the packet is forwarded to *u*.

4.4.2 Inter-foraging Region Routing. The exchange of data packets outside the region happens in an infrastructure-based mode in which the coordinator nodes form the centralized control. Both WiMAX and WiFi radios are used for such a communication. Consider Figure 3 in which node *y* wishes to communicate with node *t*. Node *y* searches for node *t* in its short routing table. Since *t* is not in its region, an entry for *t* is not present in *y*'s routing table. Hence, *y* assumes that *t* is in another region and forwards the packet to its coordinator node *A* through WiFi. The next hop to *A* from *y* is *x*.
The coordinator node *A* searches for node *t* in its long routing table to find that *B* is the destination node's coordinator, and also the next hop coordinator. Hence, the packet is forwarded to *B* through WiMAX. Since the coverage range of WiMAX is larger, this forwarding occurs much faster. In the absence of WiMAX, the coordinator nodes would have communicated through WiFi which incurs many more hops. In the example of Figure 3, the WiFi path from the coordinator *A* to *B* would be from *A* to *z*, *z* to *v* and *v* to *B*, which as expected would be time consuming.
Coordinator node *B* upon receiving the packet, searches its short routing table for *t* and forwards it to *t* through WiFi.

5. IMPLEMENTATION AND ALGORITHM

The implementation consists of two sets of algorithms, the $Coordinator\ Node\ Algorithm$ and $Subscriber\ Node\ Algorithm$, implemented by the coordinator node and the subscriber node respectively. Any node which is equipped with both WiMAX and WiFi radio implements the coordinator node algorithm whereas all other nodes implement the subscriber node algorithm. The two algorithms are subdivided into various functions as explained in $ALGORITHM$ I through VII. Each function can be implemented as a series of interrupts. The coordinator node implements $ALGORITHM$ I, II and III.
The algorithm forms the foraging zones and simultaneously creates the $short\ routing\ table$ containing the list of all nodes in its region. The $register$ packet and the $register\_ack$ packets are as shown in Figures 4 and 5 respectively.
The coordinator creates the $register$ packet. The hop_count is set to the number of hops that define a foraging zone(*n*). In this work, the value of *n* is set to 3. Both the $source\_addr$ and $region\_num$ are set to the coordinator's node ID. The $region\_num$ is also set to the coordinator's node ID for simplicity, since this has to be unique. The $register$ packet is then broadcast to all the subscriber nodes via the WiFi radio. When the coordinator receives the acknowledgment to its $register$ packet from the subscriber node in the form of a $register\_ack$ packet, this subscriber node is added to its *short routing table* which maintains the routes to all nodes in its region. The next hop from the coordinator node to this node is the node which passed on this packet to the coordinator, specified by the $source\_addr$. The number of nodes in its region, implied by the $nodes\_count$ is incremented.

ALGORITHM I: *CoordinatorForagingZone*

```
→ create $register$ packet R
→ set $hop\_count$ → n
→ set $source\_addr$ → coordinator node ID
→ set $region\_num$ → coordinator node ID
→ broadcast R
→ if($register\_ack$ received)
    {
→       increment $nodes\_count$
→       add node address in $actual\_src$ to
            $short\ routing\ table$
→       add $source\_addr$ as the Next Hop in
            $short\ routing\ table$
→       add $hop\_count$ as the Hop Count to that
            subscriber node
    }
→ if(new node to be added)
    {
→       increment $nodes\_count$
→       add node address in $actual\_src$ to
            $short\ routing\ table$
→       add $source\_addr$ as the Next Hop
            in $short\ routing\ table$
→       add $hop\_count$ as the Hop Count to that
            subscriber node
    }
→ if($region\_timer$ has expired)
    {
→       multicast $short\ routing\ table$ via WiFi radio
            as $short\ bee$ to all addresses in its
            $short\ routing\ table$
→       broadcast $short\ routing\ table$ via WiMAX radio
            as $long\ bee$ to all other coordinators
    }
```

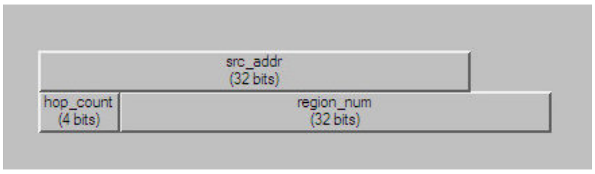

Fig. 4. $register$ Packet Format

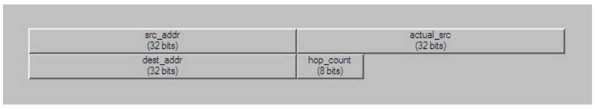

Fig. 5. $register\_ack$ Packet Format

If there is a subscriber node *x* that is not within a hop count of *n* from any of the coordinators, but is reachable from a subscriber node *y* within the coordinator *A*'s region, the subscriber node *y* upon the request from node *x* informs the coordinator *A* about node *x*. The coordinator then adds *x* to its region and increments the $nodes\_count$. Once the predefined time for forming the regions has expired, the coordinator node multicasts the *short routing table* to all nodes in its *short routing table*, i.e; its region, via WiFi in the form of a $short\ bee$ packet. It also broadcasts the $short\ routing\ table$ to all reachable coordinator nodes through the WiMAX radio via the $large\ bee$ packet.

<u>ALGORITHM II</u>: *LongRoutingTable*

```
→ if($large\_bee$ received)
    {
→       add source node into $long\ routing\ table$
→       add data into  List of all subscriber nodes in
          this region
    }
```

When a coordinator node A receives a $large\_bee$ packet from another coordinator *B*, it adds the address of *B* along with the list of all the node IDs sent in the packet into its $long\ routing\ table$.

<u>ALGORITHM III</u>: *HandleData*

```
→ if(data packet received)
    {
→       search for the destination in short routing table
→       if(found)
→          forward the packet to the $Next\ Hop$
             corresponding to this destination
→       else
            {
→           search for the destination in long routing table
→           if(found)
→              forward the packet to the
                 $Next\ hop\ coordinator\ node$
                 corresponding to this destination
→           else
→              discard packet
            }
    }
```

Once a $data$ packet is received, the coordinator node first searches for the destination address in its region, i.e; the $shortroutingtable$. If the destination is in its own region, it forwards the packet to the node through the next hop node through WiFi. In case the destination is not found in its region, it searches for destination in its $longroutingtable$ and forwards the packet to the corresponding coordinator node through WiMAX. If the packet's destination is neither found in its $shortroutingtable$ nor in its $longroutingtable$, the address is invalid and the packet is discarded.

The subscriber node implements $ALGORITHMS$ IV, V, VI and VII. When the subscriber node *x* receives the $register$ packet sent by the coordinator node, if it does not have a region yet, it updates its $region\_number$ to the one sent in the packet and sends an acknowledgment to the sender through the $register\_ack$ packet. The node from which this packet was received is saved as the next hop to the coordinator. The *hop_count* in the packet is then decremented and the packet is further broadcast if the new *hop_count* is greater than zero. This is done to cover a radius of *n* hops around the coordinator into the region. Once a *hop_count* of zero is reached, it is implied that a radius of *n* hops is covered and the packet is discarded. This functionality is implemented in $ALGORITHM$ IV.

Any node that is more than *n* hops away from any of the coordinators cannot be a part of any region using the above *hopcount* mechanism. In the case where such nodes are reachable to other subscriber nodes, they can be drawn into the nearest region. When the $region$ timer, which is started at the beginning times out, if the node *x* does not have a region number, it broadcasts a $region\_request\_bee$. The subscriber node *y* that receives it responds with a $region\_request\_bee\_ack1$ to inform x about its $region\_number$ to which *x* will now belong. Node x acknowledges with $region\_request\_bee\_ack2$ in order to confirm to *y* that its coordinator can be informed to add *x*. In case where there are many such nodes *y* that are reachable to node *x*, *x* sends the $region\_request\_bee\_ack2$ only to the node *y* whose reply was received first. Hence, the other nodes do not inform their coordinators to add *x* to their regions. $ALGORITHM$ IV implements this mechanism.

<u>ALGORITHM IV</u>: *SubscriberForagingZone*

```
→ create routing table
→ start region timer
→ if(register R received)
    {
→      if($region\_number$=-1)
           {
→          update $region\_number \rightarrow region\_num$
             from the received packet R
→          $coord\_next\_hop \rightarrow source\_addr$
→          create $register\_ack$ RA
→          set RA.$dest\_addr \rightarrow source_addr$
             of R
→          set both $source\_addr$ and $actualsource$ to
             $node\_objID$
→          send RA
→          extract $HopCount$ from R and decrement it
→          if(Hop Count → 0)
→             broadcast R with decremented Hop Count
→          else
→             discard R
           }
→      else
→         discard R
    }
```

<u>ALGORITHM V</u>: *Initialize*

```
→ if(region timer has expired and $region\_number$ is
     not yet set)
→    broadcast $region\_request\_bee$ packet
       requesting for a region
→ if($region\_request\_bee$ received)
→    send $region\_number$ via
       $region\_request\_bee\_ack1$ to the source
→ if($region\_request\_bee\_ack1$ received)
     {
→       update $region\_number \rightarrow region\_num$
          of the received packet
→       send $region\_request\_bee\_ack2$ to the sender
     }
→ if($region\_register\_bee\_ack2$ received)
→    inform coordinator about the new node in its region
```

On the receipt of a $short\_bee$ packet from the coordinator node (when no entry for the node IDs are found in its $routingtable$), which contains the list of all nodes in its region along with the $NextHop$ to the coordinator and the $HopCount$, the node creates its $routingtable$ with entries for each node in its region. The *Next Hop* is initialized to the sender of this packet and the $hopCount$ is initialized to the sum of the number of hops to the coordinator and the number of hops from the coordinator to that node. Hence, the initial route to every node goes through the coordinator. It also updates the $Hop\ Count$ against its own node ID in the $routingtable$ as zero. The $routingtable$ is then multicast to all the nodes in its region as a $short\_bee$ packet.

ALGORITHM VI: *RoutingTable*

```
→ if($short\_bee$ S received)
    {
→     for each address in the packet data
        {
→         search for address in the
            routing table
→         if(found)
            {
→             if($Hop\ Count$ +1 $hop\ count$
                in the routing table)
                {
→                 update $hop\_count \rightarrow HopCount$
                    in the received packet
→                 update $next\_hop \rightarrow$ source
                    of the packet
→                 change → true
                }
            }
→         else
            {
→             add all the address in the $data$ of
                S to the routing table
→             update each Hop Count entry as
                the number of hops from this node to the
                coordinator and the number of hops from
                the coordinator to the corresponding
                node(provided in the packet)
→             change → true
            }
        }
→   if(change)
→     exchange routing table via $short\_bee$
    }
→ find its own node ID in the $routingtable$ and set
    $NextHop$ and $HopCount$ to itself and 0
    respectively
→ multicast $routingtable$ as $short\_bee$ to all the nodes
    in its region
```

In the case where the subscriber node x receives a $short\_bee$ packet from another subscriber node y, an entry for the node ID in the packet will already be present in the $routingtable$. Hence, for each of the entries, the node x checks if the hop count specified in the packet plus one is less than the *Hop Count* for that node in its $routingtable$, in which case, the *Next Hop* is updated to node y and the *Hop Count* is updated to the hop count against this node in the packet plus one. One is added to the hop count specified in the packet to account for the one hop from node x to node y which sent the packet. When the node y is an immediate neighbour of x, the hop count specified in the packet against node y is zero. Hence, in the $routingtable$ of x, the $HopCount$ to y becomes one; rightly representing that y is an immediate neighbour of x. The $routingtable$ is then multicast to all the nodes in its region and hence, all nodes update their routing tables to the optimal paths. The above process is explained in algorithms V and VI.

As explained in algorithm VII, the subscriber node, on receiving the $data$ packet, checks if the packet was destined to itself, in which case, it passes on the packet to the higher layers. If the destination is not itself, it searches for the destination in its $routingtable$, and if it is found, the packet is forwarded to the *Next Hop*. If the destination address does not exist in its $routingtable$, the implication is that the packet is destined to a node in another region. Hence, the packet is forwarded to the coordinator node of this region.

ALGORITHM VII: *ReceiveData*

```
→ if($data$ packet received)
    {
→     if($dest$=$node\_objID$)
→       pass the packet to the transport layer
→     else
        {
→         search for $dest\_addr$ in routing table
→         if(found)
→           send the packet to $next\_hop$
→         else
→           send the packet to the coordinator node
        }
    }
```

6. PERFORMANCE ANALYSIS

The algorithm was run for various scenarios by varying the different parameters. The graph shown in Figure 6 was obtained by running the simulation for 2 hour 30 minutes, the data rates of WiMax and WiFi being set to 50 Mbps and 100 Mbps. The coverage range of WiMax was set to 5000 m and that of WiFi to 750 m. The graph shows the comparison of throughputs for static and mobile networks. It can be seen that the throughput of the static network is higher than that of the mobile network. This is because the topology of the static network does not change, hence the routes are formed only once, during the network setup. Once the routes are set, no new routes are to be formed and only data packets are exchanged, increasing the throughput of the network. In the mobile network, the nodes are moving, and routes change. New routes are formed by exchanging the bee packets at a frequency of 20 s. Hence, at an interval of 20 s, along with data packets, bee packets are also exchanged which decreases the overall network throughput. Also, the frequency of bee packet exchange may not be in sync with the time when the nodes move out of the region. In such a case where some of the nodes have gone out of the foraging region, but the new routes are not yet updated, the nodes might be either unreachable or the routes to them may not be optimized, further decreasing the throughput.

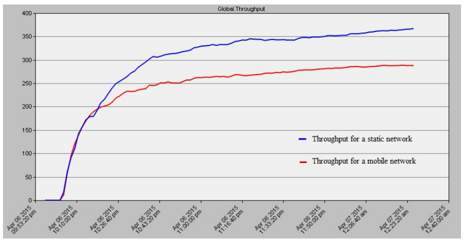

Fig. 6. Throughput of static v/s mobile network

The graph shown in Figure 7 was obtained by varying the average speed of motion of the nodes for a mobile network. The WiMax data rate was set to 50 Mbps and that of WiFi to 100 Mbps. The coverage ranges of WiMax and WiFi were set to 5000 m and 750 m respectively. The frequency at which the routes are formed is 20 s. For easy analysis, the observed maximum throughput was plotted against average speed of the nodes, as shown in Figure 8. As it can be seen from the graph, the overall network throughput initially increases till a threshold after which it drops down.

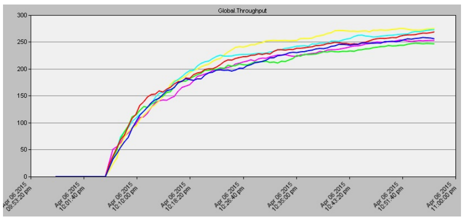

Fig. 7. Throughput for various speeds of motion

Table 4. Distance moved by the node in 8 seconds

Speed of motion (m/s)	**Distance moved (m)**	**Subscriber node's location**
10	**80**	**Neither neighbors' range nor region**
100	**800**	**Neighbors' range**
200	**16000**	**Neighbors' range**
250	**2000**	**Neighbors' range**
275	**2200**	**Neighbors' range and region**
300	**2400**	**Neighbors' range and region**

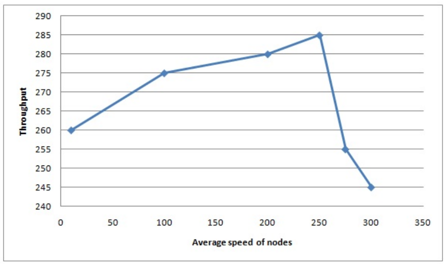

Fig. 8. Average speed v/s maximum throughput

The distance formula is as follows:

$$\textit{distance moved} = \textit{speed} \times \textit{time} \quad (1)$$

Using equation 1 the distance moved by a node in 8 seconds for various speeds is calculated.

The node can either remain in a relatively same position, go out of the WiFi range of its neighbor or go out of its region. Depending on the speed of the node, the bee packets must be exchanged at optimal intervals so that the overall network throughput doesnot drop. In this work, the bee packets are exchanged at a regular interval of 8s for various speeds, and the variation in the overall network throughput is studied. In this work, a hop limit of 3 is considered for the formation of regions. When the speed of the node is 10 m/s, in 8 seconds, the node moves a distance of only 80 metres, which is less than the WiFi coverage range. Hence, the position of the node in the network has not changed much and there is no need to find new routes. But, by the end of 8 seconds, bee packets are exchanged and new routes are formed, which is wasteful of resources. This unnecessary formation of routes results in low overall network throughput.

The nodes moving at a speed of 100 m/s, 200 m/s, 250 m/s and 275 m/s cover distances of 800m, 1600m, 2000m and 2200m respectively in 8 seconds. In such a case, the node moves out of the WiFi range of its neighbors but remains within its region. Hence, the frequency of updation of the routing tables which happens at every 8 seconds is apt for this range of node speeds resulting in higher throughputs.

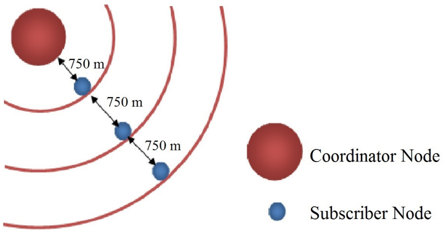

Fig. 9. An example network

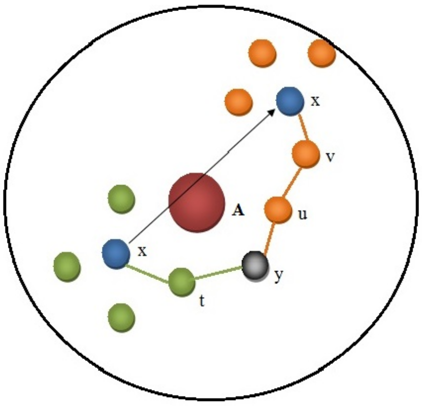

Fig. 10. An example network showing routes before and after a node moves

Once the speed is increased further, the frequency of updation of the routing tables is smaller than that required; i.e, the nodes change their routes much faster than they are updated, resulting in a throughput fall. This can be seen in the graph in Figure 8 for a speed of 300 m/s, where, the node moves a distance of 2400m in 8 seconds. As it can be seen from the Figure 9, the maximum possible radius of 3 WiFi hops is 2250 m since each WiFi hop is 750 m.

In general, it is not expected that all the nodes will traject on a circular path within its region. This assumption results in some nodes moving out of the region, and some moving out of reach from all their neighbors within their region as shown in Figure 10; where node x moves along the diameter of the circle defined by three WiFi hops. Eventhough it is within its region after the movement, all its neighbors have changed. Node y which initially communicated to node x via nodes t now has to communicate via nodes u and v. Hence, even in this case, the routes must be updated. Since the frequency at which new routes are formed is too less compared to the requirement at this speed, the throughput, as expected, falls.

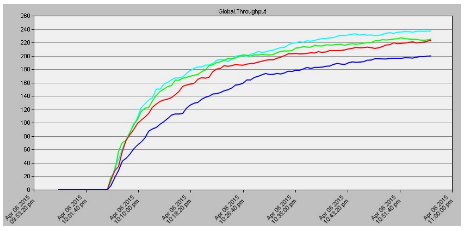

Fig. 11. Throughput for hop limits 3 through 6 for static network 13

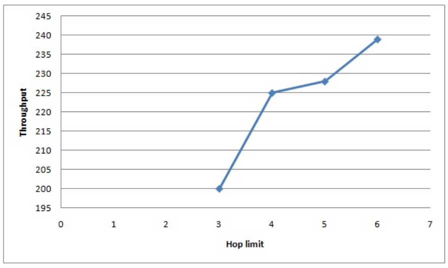

Fig. 12. Hop limit v/s maximum throughput for static network 13

The performance of the network was observed by varying the parameter *hop limit* used for forming the foraging regions. The graph in Figure 11 shows the throughputs for hop limits from 3 to 6 for the static network topology shown in Figure 13. For easier analysis, the maximum throughput is plotted against hop limit and is shown in Figure 12. The coverage range of WiMAX was considered to be 5000 metres and that of WiFi to be 750 metres. It can be seen that the overall network throughput increases gradually. This is because approximately 6.67 WiFi hops equals one WiMAX hop. When the hop limit was considered to be 3, a source node *x*, four WiFi hops away from the destination node *y* on the same side of the coordinator node fall into different foraging regions, resulting in the data packets taking a longer path as shown in the Figure 14. The same reason applies for hop limits of 4 and 5. The throughput is seen to be maximum for hop limit 6 since approximately six WiFi hops make one WiMAX, making it the apt hop limit for the chosen WiMAX and WiFi coverage ranges. It is expected that for a hop limit of 7, the throughput decreases since two WiMAX hops will be required to reach out to this distance against a shorter distance of 7 WiFi hops. From the observation, it can be inferred that the right hop limit can be obtained by equation 2.

$$Hoplimit = floor(\frac{WiMAX_{coveragerange}}{WiFi_{coveragerange}}) \quad (2)$$

7. CONCLUSIONS

Interoperability between WiMAX and WiFi was exploited using an algorithm based on the Bee-hive algorithm. The implementation of the algorithm was discussed in detail. From the observations, it can be concluded that the overall network performance given by throughput and end-to-end delay was higher for a static network since routes do not change frequently. In the case of a mobile network, it was seen that the frequency of formation of new routing tables must depend on the average speed of the nodes. The hop limit must be chosen depending on the available WiMAX and WiFi coverage ranges.

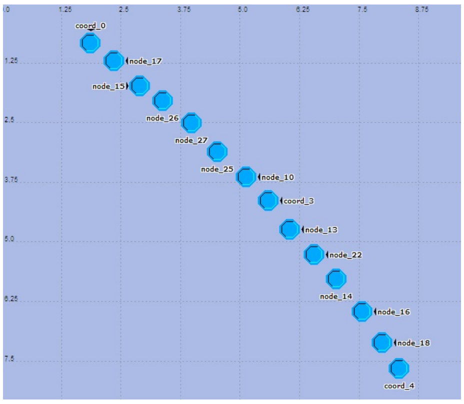

Fig. 13. Static network used to study the dependence of throughput on hop limit

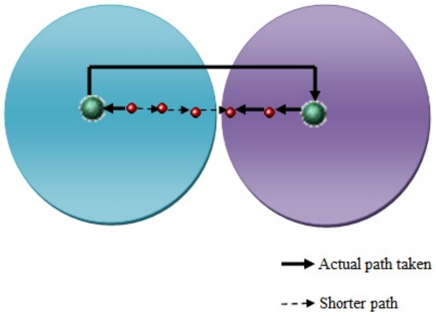

Fig. 14. Path taken when communicating nodes are in different foraging regions

As a part of future work, a mathematical model for the dependency of throughput on various factors such as hop limit, speed of the nodes can be developed. A more detailed, corner-case study of the algorithm's behavior and its ability to tolerate faults in the network nodes and links can be made.

8. REFERENCES

[1] WiMAX and the IEEE 802.16 Air Interface Standard-April 2010, WiMAX Forum

[2] S. Banerji and R. S. Chowdhury, *"Wi-Fi & WiMAX: A Comparative Study"*, arXiv preprint arXiv:1302.2247, 2013.

[3] Abhishek Alfred Singh (2011). *"Traffic Splitting In Hybrid Multi-radio Ad-hoc Networks"* (M.E. Dissertation) Bangalore University.

[4] Proposed TGac Draft Amendment Document, IEEE 802.11-10/1361r3, Jan- uary 18,2011

[5] K. Chebrolu, B. Raman, and S. Sen, *"Long-Distance 802.11b Links: Performance Measurements and Experience"*, in Proceedings of the 12th annual international conference on Mobile computing and networking. ACM, 2006, pp. 74-85.

[6] Opent Modeler, Opnet Modeler Documentation Set, Version 14.5

[7] Jeffrey G. Andrews, Arunabha Ghosh, and Rias Muhamed, *"Fundamentals of WiMAX: Understanding Broadband Wireless Networking"*, Prentice Hall, 2007

[8] IEEE 802.16m standard, IEEE standard for Information Technology Part 16, Amendment 3, 2011

[9] IEEE 802.11g standard, IEEE standard for Information Technology Part 11, Amendment 4, 2003

[10] Silva, H., Figueiredo, L., Rabadao, C., and Pereira, A., *"Wireless networks interoperability Wifi WiMAX Handover"*, Proceedings of the 2009 Fourth International Conference on Systems and Network Communications, pp 100-4, Sept. 2009

[11] Gracias, M., Knezevic, V., and Esmailpour, A., *"Interoperability between WiMAX and WiFi in a testbed environment"*, 24th Canadian Conference on Electrical and Computer Engineering, pp. 1144-48, May 2011

[12] Connie Ribeiro *"Bringing Wireless Access to the Automobile: A Comparison of Wi-Fi, WiMAX, MBWA and 3G"*, Proceedings of the 21st Computer Science Seminar

[13] Sangeetha J., and Kumar, S., A comparative study on WiFi and WiMAX networks, *"IEEE International Conference on Computational Intelligence and Computing Research"*, pp. 1-5, Dec. 2010

[14] Ghazisaidi, N., Kassaei, H., and Bohlooli, M.S., *"Integration of WiFi and WiMAX networks"*, International Conference on Advances in Mesh Networks, pp. 1-6, June 2009

[15] Kim, J.-O., Davis, P., Ueda, T. and Obana, S. (2010), *"Splitting Down- link Multimedia Traffic over Wimax and Wifi Heterogeneous Links Based on Airtime-Balance"*, Wirel. Commun. Mob. Comput.. doi: 10.1002/wcm.999

[16] Bruno, R.; Conti, M.; Gregori, E.; *"Mesh networks: commodity multihop ad hoc networks"*,Communications Magazine, IEEE, vol.43, no.3, pp. 123- 131, March 2005

[17] Lima, M., A. Dos Santos, and Guy Pujolle. *"A survey of survivability in mobile ad hoc networks"*. Communications Surveys & Tutorials, IEEE 11, no. 1 (2009): 66-77.

[18] Brewer, E.A; Katz, R.H.; Chawathe, Y.; Gribble, S.D.; Hodes, T.; Giao Nguyen; Stemm, M.; Henderson, T.; Amir, E.; Balakrishnan, H.; Fox, A; Padmanabhan, V.N.; Seshan, S., *"A network architecture for heterogeneous mobile computing"*, Personal Communications, IEEE, vol.5, no.5, pp.8,24, Oct 1998 doi: 10.1109/98.729719.

[19] Chuanxiong Guo; Zihua Guo; Qian Zhang; Zhu, Wenwu, *"A seamless and proactive end-to-end mobility solution for roaming across heterogeneous wireless networks"*, Selected Areas in Communications, IEEE Journal on vol.22, no.5, pp.834,848, June 2004 doi: 10.1109/JSAC.2004.826921.

[20] Perkins C., Das S., *"Ad hoc On-Demand Distance Vector Routing"*, RFC 3561, July 2003

[21] R. Draves, J. Padhye, and B. Zill, *"Routing in Multi-Radio, Multi-Hop Wire- less Mesh Networks"*, Proc. of MobiCom 04, pp. 114-128, Sept.-Oct. 2004

[22] Azad, Md Saiful, Mohammad Moshee Uddin, Farhat Anwar, and Md Arafatur Rahman, *"Performance Evaluation of Wireless Routing Protocols in Mobile WiMAX Environment"*, IAENG International Conference on Communication Systems and Applications. 2008.

[23] Shafiee, K., Attar, A., and Leung, V.C. M., *"WLAN-WiMAX double technology routing"*, IEEE Vehicular Technology Conference, pp. 1-6, Sept. 2011

[24] Ibanez, S.R., Santos, R.A., Licea, V.R., Block, A.E., Ruiz, M.A.G., *"Hybrid WiFi-WiMAX Network Routing Protocol"*, Electronics, Robotics and Automotive Mechanics Conference, 2008. CERMA 08 , vol., no., pp.87-92, Sept. 30 2008-Oct. 3 2008, doi: 10.1109/CERMA.2008.24

[25] Bonabeau, M. Dorigo, and G. Theraulaz. *"Swarm Intelligence: From Natural to artificial Systems"*, Oxford University Press, 1999.

[26] B.Bar an and R.Sosa. *"A new approach for Antnet Routing"*,Proceedings of the Ninth International Conference on Computer, Communications and Networks, 2000, pp 303-308.E.

[27] Horst F. Wedde, Muddassar Farooq, and Yue Zhang. *"BeeHive:An efficient routing algorithm inspired by Bee behavior"*, Springer-Verlag, 2004, pp-83-94.

[28] Karaboga, Dervis, and Bahriye Akay. *"A survey: algorithms simulating bee swarm intelligence"*. Artificial Intelligence Review 31.1-4 (2009): 61-85.