

A Novel Packet Aggregation Mechanism for Enhancing VoIP Performance on IEEE 802.11 Wireless Mesh Networks

Guan-Hsiung Liaw^{*}, San-Yuan Wang¹, Tain-Lieng Kao², Tsung-Cheng Chen³

^{*,1,3}Department of Information Engineering, ²Department of Communication Engineering
I-Shou University, Kaohsiung City, Taiwan

*ghliaw@isu.edu.tw (Corresponding author), ¹sywang@isu.edu.tw, ²tlkao@isu.edu.tw, ³makueb@yaho.com.tw

Abstract—In 802.11 Wireless Mesh Network (WMN), bandwidth will be wasted much for transferring VoIP flows since each voice frame must contain relatively large amount of protocol data. “Packet Aggregation” mechanism can be applied to merge the voice data of multiple VoIP flows into one frame for transmission. It reduces the waste on bandwidth and increases the maximum number of successful VoIP calls. In addition, the mechanism “MCF controlled channel access” (MCCA) defined in 802.11 standard can be used to obtain better QoS than adopting default EDCA mechanism. In MCCA, mesh stations which wants to transfer VoIP flows can reserve time intervals of the medium for transmission and this reservation will be advertised to their neighbors. It is why MCCA causes less medium contentions than EDCA. In this paper, a mechanism to transfer VoIP flows in IEEE 802.11 WMN by MCCA with packet aggregation scheme is proposed. The effectiveness of the proposed mechanism is shown by simulation results. In addition, the problem named as Routing-Packet Aggregation / De-aggregation-Scheduling optimization problem (abbr. RPADS problem) derived from the proposed mechanism is also studied. A heuristic algorithm for RPADS problem to maximize the total number of supported calls is also proposed.

Keywords- VoIP, IEEE 802.11 Wireless mesh network, MCF Controlled Channel Access (MCCA), Packet Aggregation

I. INTRODUCTION

In recent years, VoIP has a high growth in application such as Skype [1], that lets people talk over the internet to anyone in the world for free. However, voice traffic is usually sensitive to delay, jitter, and packet loss. When transmitted voice data through IP, traffic is broken into small packets that are sent individually to their destination. Nevertheless, the characteristic of IP is Best-Effort (BE) that does not guarantee the provision of services and may cause to packet loss or packet disorder due to congestion and dynamic routing. It's very important in Quality of Service (QoS) mechanism for VoIP quality. In addition, VoIP is usually compared with the traditional telephone. The VoIP voice quality has not fully caught with traditional telephone voice products. But with the advantages including reducing networking and management cost and supporting new services, such as combining voice communication with other media, VoIP is still considered to be a practical product and encouraged for mass deployment.

We have known that the performance of running VoIP service over IEEE 802.11 WLAN is very low [2] [3]. Table I shows the time needed for transmitting per VoIP frame with speech codec G.729 over 802.11b. Such VoIP stream typically requires about 10kbps. Ideally, the number of simultaneous VoIP streams that can be supported by an 802.11b WLAN is about 11Mbps / 10Kbps = 1100. Actually, the achievable throughput is no more than 6 VoIP sessions (equivalent to 12 VoIP streams) [3].

TABLE I. THE TIME NEEDED PER VOIP FRAME OVER IEEE 802.11B [3]

Delay component	Time (μs)
DCF inter-frame Space (DIFS)	50
Average Contention Windows	310
Voice Frame (G.729)	14.55
RTP/UDP/IP encapsulation	29.09
PLCP preamble and header	192
MAC header and trailer	20.36
Short inter-frame Space (DIFS)	10
Acknowledgement (ACK)	10.18
Total	835.45

In order to increase the number of supported VoIP calls in 802.11-based WMN, in reference [2], [3], and [5]-[10], the concept called packet aggregation is proposed. It aggregates VoIP packet with the same destination or the same next hop to increase the number of calls by reducing the overhead of communication protocol. In addition, some researches proposed other methods to increase the number of calls such as multi-channel [5] and compressed header [2], [5].

In IEEE 802.11 standard, a mechanism called Mesh Controlled Channel Access (MCCA) is defined to provide better QoS service than Enhanced Distributed Channel Access (EDCA) does [4]. In EDCA, after a station waiting for Arbitration Inter-frame Space (AIFS) and Contention Window (CW), it will get a Transmission Opportunity (TXOP) to transmit packet. While in MCCA, a station can reserve a specific time interval to transmit data through advertisement. So it will get fewer contentions than EDCA.

In this paper, a mechanism to transfer VoIP flows in IEEE 802.11 WMN by MCCA with packet aggregation scheme is proposed. The proposed packet aggregation scheme adopts TDMA to divide the channel time as periodical time frames consisting of fixed-size time slots.

We propose a heuristic algorithm called Routing-Packet Aggregation / De-aggregation – Scheduling (abbr. RPADS) algorithm to resolve the problem of arranging the transmission of given VoIP flows. It is trivial that the problem is an NP-hard or NP-complete problem. In the proposed algorithm, in order to avoid the interfere among wireless links during data transmission, a famous heuristic algorithm of resolving edge coloring problem, named Vizing Algorithm [11] is adopted firstly to find out the sets of wireless links that can transmit data concurrently without interfering each other. Then, the routing path of each flow is firstly decided by using the maximum weight priority as the principle. After deciding the routing path, the scheduling of the time slot on each segment of the routing path of the flow is decided. In the decision of time slot, the proposed RPADS algorithm will select the slot with which the total buffering delay of the flow data can be minimum. After deciding the time slot of one segment, the other flows whose routing path contains the same segment are considered to aggregate together to this time slot. In addition, the segments of the other flows whose data transmission can be done at the same time slot are also considered to be scheduled at this time slot. The effectiveness of the proposed RPADS algorithm will be shown by simulation results.

The rest of the paper is organized as follows. In Section 2 the related work, including VoIP, 802.11 WMN, and packet aggregation mechanism are introduced. In Section 3, the proposed system and its operation are described. The proposed algorithm for RPADS problem and its performance evaluation are illustrated in Section 4 and Section 5, respectively. Finally, the conclusion is given in Section 6.

II. RELATED WORKS

A. VoIP

For VoIP [12]-[14], the pulse-code modulation (PCM) or analog voice signals are encoded and compressed into a low-rate packet stream by codecs. Generally, the codecs generate constant bit rate (CBR) audio frames consisting of 40 bytes IP/UDP/RTP headers followed by a relatively small payload. We focus on the G.729 codec in this paper. For G.729, the payload is 20 bytes.

The quality of a VoIP call is usually sensitive to delay, delay jitter, and packet loss. These are determined by the performance of codecs, network protocol, and buffering. The R-Score proposed in [14] is a used to evaluate the quality of a call. R-Score takes into account mouth to ear delay, loss rate, and the type of the encoder. In order to maintain a good call quality, it should provide a value above 70:

$$R = 94.2 - 0.024d - 0.11(d - 177.3)H(d - 177.3) - 11 - 40\log(1 + 10e), \quad (1)$$

Where:

- $d = 25 + d_{\text{jitter buffer}} + d_{\text{network}}$ is the total ear to mouth delay comprising 25 ms vocoder delay, delay in the de-jitter buffer, and network delay
- $e = e_{\text{network}} + (1 - e_{\text{network}})e_{\text{jitter}}$ represents the total loss
- $H(x) = 1$ if $x > 0$; 0 otherwise is the Heaviside function

Mean Opinion Score (MOS), which represents the satisfaction of the VoIP user from network behavior. The R-score to MOS mapping is given in Table II.

TABLE II. R-SCORE TO MOS [2]

R-Score	Quality of Voice Rating	MOS
$90 < R < 100$	Best	4.34 – 4.5
$80 < R < 90$	High	4.03 – 4.34
$70 < R < 80$	Medium	3.60 – 4.03
$60 < R < 70$	Low	3.10 – 3.60
$50 < R < 60$	Poor	2.58 – 3.10

VoIP refers to the diffusion of voice traffic over internet-based networks. Internet Protocol (IP) was originally designed for data networking and following its success, the protocol has been adapted to voice networking system architecture of VoIP over WMNs is shown in Figure 1, where the major devices are explained as follow:

- **Regular Wired Phoned.** These are the wired phone used in PSTN
- **PBX.** A PBX connects the internal telephones within a business and also connects them to the public switched telephone network (PSTN).
- **VoIP Clients.** VoIP clients are abundant on the Internet.
- **VoIP / PSTN Gateway.** The gateway let VoIP calls and PSTN calls coexist in the same network architecture.
- **Mesh Router in WMNs.** A mesh router forwards traffic for other mesh routers.
- **Local SIP Server.** The local SIP server is going to establish the connection for all of the SIP phone call over network.
- **SIP Proxy Server.** When a VoIP call is made among two clients in different WMNs, signaling have to be processed by a SIP proxy server.

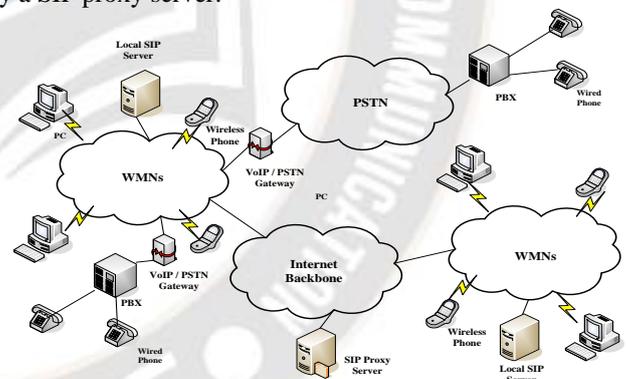


Figure 1: The System of VoIP over WMNs

B. IEEE 802.11 Wireless Mesh Network

Wireless Local Area Networks (WLANs) have become ubiquitous. In 802.11, dense deployment of Access Points (APs) relies on fixed backbone. This limits the deployment of the wireless infrastructure and its coverage. Increase the network coverage by adding APs. But a large number of AP will increase the cost. To overcome the cost barrier, APs need to interconnect wirelessly. So, wireless mesh networks are desirable. And wireless mesh network can enhance the network performance because of the advantages as follows including more flexibility of the network structure, more simple to develop and setup, greater coverage, and less maintenance costs.

In IEEE 802.11 mesh network, the basic mesh device is the Mesh Station (MSTA). MSTAs can exchange frames over multi-hop wireless network. Thus, MSs can communicate with other MSTAs. On the one hand, AP collocated with Mesh Gate (MAP), is functional as an MSTA, collocated with AP which provides BSS services to support communication with Stations (STAs). On the other hand, Mesh Gate (MG) is the point at which MSDUs exit and enter a WLAN Mesh.

In WMNs, its operations mode is shown as Figure 2. STAs are only connected with external network with MAP. And the function of MSTA is packet forwarding or routing to connect with adjacent nodes. Moreover, on purpose of compatibility with IEEE 802.11 network, those WMN must connect with other networks. So, WMN must have the layer-2 function – bridging and layer-3 function – internetworking.

For media access control [4][15][16], MSTAs adopt 802.11e Enhanced Distributed Channel Access (EDCA) as the base standard. The contention-based medium access mechanism relies on carrier sense that does not require synchronization among MSTAs. In opposite, MCF Controlled Channel Access (MCCA) is a contention-free mechanism in which an MSTA must be a synchronizing MSTA.

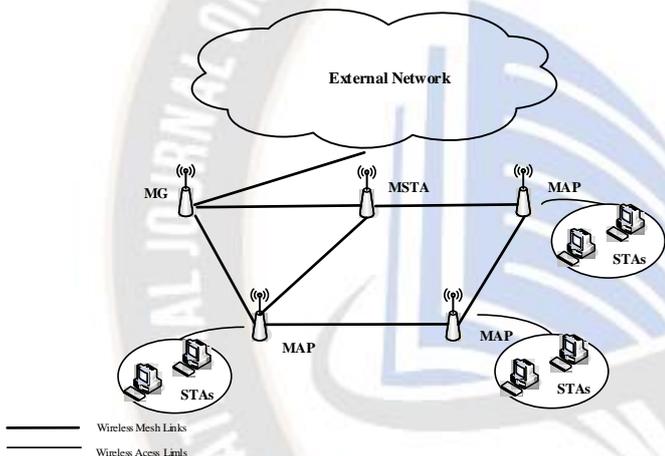


Figure 2: Wireless Mesh Networks Architecture

C. 2.3 Packet Aggregation

Several recent studies have investigated VoIP capacity in WMN. In this subsection, we briefly summarize related work on packet aggregation in wireless mesh networks. Improving of the VoIP capacity in multi-hop networks by aggregation packets was studied in [2], [3], and [5]-[10]

To provision VoIP in multi-hop WMNs is an important service in the feature. However, VoIP service will get some challenges when deployed over a multi-hop WMNs. Packet losses, jitter and delay can significantly degrade the end-to-end VoIP call quality. Moreover, to transmission a small VoIP packet imposes a high MAC layer overhead, which to cause a low capacity for VoIP in WMN.

As shown in Figure 3, it shows the advantages of the packet aggregation. Traditionally, sender transfer a packet to receiver, receiver must returned pass an ACK to sender, then continue next packet transmission. But after use packet aggregation, we can aggregate a lot of packets into a data frame to reduce the waste of bandwidth.

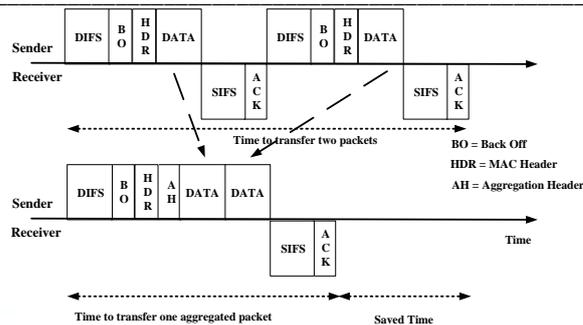


Figure 3: Principle of Packet Aggregation

In several research works have addressed packet aggregation scheme to support VoIP performance. Reference [2] proposed the scheme called Voice Multiplex-Multicast (M-M). The main idea of it is to combine the data from several downlink streams into a single larger packet. The advantage for this scheme is if there are many concurrent calls, with only small additional delay as packet aggregation and result in high efficiency. On the other hand, through 802.11 multicast also reduce some overhead from transmission acknowledgement, because, it does not perform retransmission mechanism.

Through experiments show, M-M scheme than traditional VoIP in the WLAN can send voice calls to enhance the capacity of 80% to 90%. For its shortcomings because of the number of transmission stations more than the size of Back-Off time, it results in high collision probability when a modest number of competing nodes. And because there is not Automatic Repeat Request (ARQ) mechanism in 802.11 MAC layer, so the multicast packet of collision won't retransmission and to make packet lose to increase packet lose rate.

In [3] it to be aimed at has not enough bandwidth in WMN, if not, it will through packet aggregation scheme to reserve bandwidth. Because of the need to wait for the same calls accumulated to a certain extent to arrive at the MAC queue, then to execute packet aggregation action. The wait for this period of time will cause high waste of delay. Therefore, it to carry out the packet aggregation, when the need for bandwidth.

In [5]-[10], their goal is to focus on improving VoIP performance and enhancing the available number of calls. And the main solution to the problem is how to reduce overhead from protocol when transmission VoIP packet. [11] subscribes three packet aggregation algorithm, respective end-to-end aggregation algorithm, hop-by-hop aggregation algorithm and accretion aggregation algorithm.

The advantage of end-to-end aggregation algorithm is only need aggregator located in the ingress node aggregate packets destined to the same destination and intermediate nodes just forward the aggregated packets. It can reduce computational complexity thus the hardware resource requirement can be alleviated at the mesh routers. But it might lead to waste the bandwidth if the size of aggregated packet is small in ingress node.

The shortcoming of hop-by-hop aggregation algorithm is need aggregator located in the all nodes. So it requires computational complexity and hardware resource most. In addition, increase the delay budget although can increase the number of packet to aggregate but also increase the end-to-end delay.

Accretion aggregation algorithm takes the advantage of the previous two algorithms. It only has forced aggregation delay in ingress node and the intermediate nodes have not extra delay

when complete packet aggregation in queuing delay. In this algorithm, packet aggregation ratio might be the same or lower than hop by hop algorithm, but it not only relaxes large jitter drop and the hardware requirement. On the side, it can provide better bandwidth utilization than end-to-end aggregation algorithm.

III. PACKET AGGREGATION OVER WMN

In this section, we detail the proposed packet aggregation mechanism in IEEE 802.11 WMN. As an example, let's see a grid topology of a WMN as in Figure 4. We assume that each mesh point can communicate with its neighbors. Three VoIP flows are placed in the network. It is assumed that all the three flows are CBR (Constant-Bit-Rate) voice streams and their packets are generated with a fixed period. Flow 1 is from the node A to the node B. Flow 2 is from the node C to the node D. Flow 3 is from the node E to node F. The routing of the three VoIP flows are shown in Figure 4. The packets of the three flows will be passed through mesh point M and aggregated into a frame at M. The format of the aggregated frame can be the one shown in Figure 3.

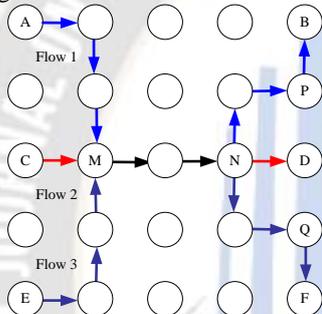


Figure 4: An example of packet aggregation over WMN

As shown in Figure 5, the packets of the three flows are transmitted to mesh point M and mesh point M buffers these packets until the reserved time for forwarding to the next hop. In this example, it is assumed that each mesh point has only one wireless interface and the wireless channel is accessed by TDMA scheme with 10 timeslots in one TDM frame. Note that only the transmission of VoIP flows must obey this TDMA rule while the other packets are still transmitted according to EDCA defined in 802.11. In Figure 5, the incoming packets of the three VoIP flows are transmitted to mesh point M at timeslot 1, 2, and 3, respectively. Then, these packets are aggregated and can be forwarded at one among timeslots 4-10. If reducing the buffering delay is considered, timeslot 4 should be chosen for forwarding the aggregated packet. The aggregated packet is then forwarded in the WMN and will be de-aggregated at the mesh point at which the next hops of the three flows' routes are different. As shown in Figure 4, the packet is de-aggregated into three packets, one for each flow, and forwarded to different next-hop mesh points.

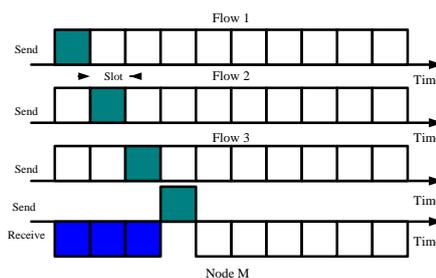


Figure 5: An example to transmit an aggregated packet

To solve the resource allocation problem in the proposed system, we model the WMN as a directed graph $G = (V, E)$, where V is the set of vertices, and E is the set of edges. MSTAs are denoted by vertices, and links are denoted by edge. An example of such a mesh network is also given in Figure 4. In our research, we used a single channel to transmit traffic and adopted TDMA to divide the channel time as periodical time frames consisting of fixed-size time slots as shown in Figure 6. In Figure 6, we can use the front a part of time slot to transmit VoIP data in MCCA mechanism and to transmit VoIP data in EDCA mechanism for spare part of time slot.

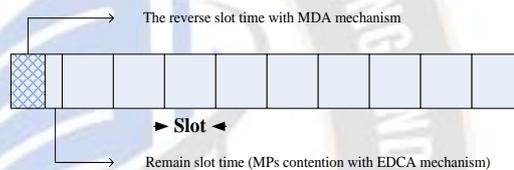


Figure6: Divide channel by TDMA into slots

Based on the above packet aggregation, de-aggregation, routing, and QoS issues, the optimization problem of assigning the routes, the timeslots for forwarding in each segment of the routes, and the decision of aggregation and de-aggregation for a given set of VoIP flows is defined as Routing-Packet Aggregation / De-aggregation - Scheduling (abbr, RPADS) problem. We propose a heuristic algorithm to resolve this problem. Since the previous researches did not adopt the standard 802.11 MCCA, we will not to compare our experiments with past researches. Two algorithms, named as RPADS-1 and RPADS-2, are proposed. Algorithm RPADS-2 is designed as the simplified version of RPADS-1 for the experiments on the effectiveness of the heuristics adopted in RPADS-1.

IV. PROPOSED ALGORITHMS FOR RPADS PROBLEM

The proposed algorithm as named RPADS-1 is shown in Figure 7 and we describe the definition of notations in our scheduling algorithm in Table III. The main idea of our RPADS-1 algorithm is to combine the packet from several paths between the same source destination pair into a single larger packet. In this way, the overheads of multiple VoIP packets can be reduced to the overhead of one packet. For reaching these goals, there have some steps needed to complete.

RPADS-1 Algorithm

```

Input:  $G, N, S, D, f, Slot\_Ni$ 
Output: Flow Scheduling
1. Begin
2. FIND_PARALLEL_TRANSMISSION( $G$ )
3. For int  $i = 1$  To  $N$ 
4. FIND_ROUTE( $i, S, D$ )
5. End For
6. For int  $i = 1$  To  $N$ 
7. FLOW_SORTING( $f$ )
8. PATH_AGGREGATION( $f, S, D, Slot\_Ni$ ) {
9. TIME_SCHEDULE( $f, S, D$ )
10. }
11. goto Step 3
12. End For
13. End
    
```

Figure 7: Algorithm RPADS-1

TABLE III. NOTATIONS

Notation	Meaning
G	a connected Graph $G = (V, E)$, where V is the set of nodes, and E is the set of edges.
$N(v)$	the neighborhood of v in G , $v \in V$
$\Delta(G)$	the largest degree of a vertex in G , $\Delta(G) = \max\{ N(v) \mid v \in V\}$
C	the set of colors, $C = \{1, \dots, n\}$, $n = V $
f	the number of flow, $f = \{1, 2, \dots, M\}$
L	A set of k links of f , $L = \{L_1, L_2, \dots, L_k\}$
D_{uv}^i	distance between node u and node v , $i = \{1, 2, \dots, N\}$
d_w^v	the shortest distance between node u and node v
C_{uv}^i	capacity of link L_i , $i = \{1, 2, \dots, k\}$
$Trans_Set$	The set of each links which can transmit simultaneously, $i = \{1, 2, \dots, T\}$
$Slot_Ni$	the place of slot, $i = \{1, 2, \dots, 13\}$
PA	aggregation packet
PA_Size	size of packets to be aggregated

Figure 8 is the RPADS-2 algorithm and the main difference with RPADS-1 algorithm is below. In RPADS-1 algorithm, it executed the sorting algorithm and time scheduling algorithm but RPADS-2 algorithm does not. We will to compare these two algorithms in the simulation.

When there is packet to be sent, in order not to cause interference with each other, we have adopted the method of the edge coloring. Its rule is that adjacent edges could not give the same color as shown in Figure 9. And we use Vizing theorem, it will be the color of all the edge coloring in $\Delta(G) + 1$. The reason of using Vizing theorem is that if the color more, the set of transmission simultaneous will decrease and cause to reduce opportunities for transmission at the same time.

RPADS-2 Algorithm

```

Input:  $G, N, S, D, f, Slot\_Ni$ 
Output: Flow Scheduling
1. Begin
2. FIND_PARALLEL_TRANSMISSION( $G$ )
3. For int  $i = 1$  To  $N$ 
4. FIND_ROUTE( $i, S, D$ )
5. End For
6. For int  $i = 1$  To  $N$ 
7. PATH_AGGREGATION( $f, S, D, Slot\_Ni$ )
8. End For
9. End
    
```

Figure 8: Algorithm RPADS-2

Algorithm : FIND_PARALLEL_TRANSMISSION

```

Input :  $G$ 
Output :  $Trans\_Set$ 
1. Begin
2. For int  $i = 1$  To  $\Delta(G) + 1$ 
3. Repeat
4. If  $d_{uv} > 1$  and  $d_{vu} > 1$  then
5. return  $Trans\_Set$ 
6. End If
7. until Compare all of the link with the same color
8. End For
9. End
    
```

Figure 9: The algorithm for finding parallel transmission

We use the Dijkstra shortest path algorithm is to have several links between the same source destination pair, aggregated at the time (Figure 10). Nevertheless, each link has its own bandwidth limitations. So, when choose a link has not enough bandwidth, we will choose another path which has enough bandwidth even if it's not a shortest path. But if can't find a path with enough bandwidth to route a flow, we will drop it.

Algorithm : FIND_ROUTE

```

Input :  $S, D$ 
Output :  $f$ 
1. Begin
2. For int  $i = 1$  To  $N$ 
3. If  $d_u^v = d_{uv}^i$  and  $C_{uv}^i > 0$  then
4. return  $f$ 
5. Else If  $d_u^v \neq d_{uv}^i$  and  $C_{uv}^i > 0$  then
6. return  $f$ 
7.  $C_{uv}^i := 1$ ;
8. Else
9. return NULL
10. End If
11. End For
12. End
    
```

Figure 10: Routing Algorithm

And then, we used the Dijkstra shortest path to determine the path of the flow and according to flow scheduling to transmission (Figure 11). Flow scheduling based on three main ways to calculate the priority of them: (1) a completed flow; an executed flow and a non-executed flow; (2) the percentage of completion of the flow; (3) the length of flows. According to the each flows weight priority to implement sorting.

We were division of the flow into each link. At first, we were sorting flow by each weight priority, then to perform the highest weight priority flow to find the link which can transmit simultaneously and the link can aggregate from other flows. Hence the number of packet to be aggregated is limited which could not exceed a maximum transmission unit (MTU) (Figure 12).

Algorithm : FLOW_SORTING

```

Input :  $f$ 
Output : Flows Sorting Algorithm
1. Begin
2. For  $f = 1$  To  $N$ 
3. Sorting by weight priority of each flow
4. End For
5. For  $f = 1$  To  $N$ 
6. Sorting by percentage of completion of each flow
7. End For
8. For  $f = 1$  To  $N$ 
9. Sorting by length of each flow (from short to long)
10. End For
11. End
    
```

Figure 11: Routing Algorithm

Algorithm : PATH_AGGREGATION

Input : $f, Trans_Set, Flows$ Sorting Algorithm

Output : $Flows$ Scheduling

```

1. Begin
2.   repeat
3.     Find the link from  $Trans\_Set$  which can transmit together with  $L$ 
4.     repeat
5.       Find the link with the same  $S, D$  with  $L$ 
6.       If  $PA\_Size < MTU$  then
7.         aggregated the same Link with  $L$  into  $PA$ 
8.       Else
9.         send  $PA$  to destination node
10.    until complete each link of  $f$ 
11.  until All flows complete
12. End
    
```

Figure 12: PATH_AGGREGATION

In order to reduce the MAC and PHY overhead, aggregating many VoIP data in one data frame can increase the number of calls (Figure 13). In addition, appropriate adjustments the time to access channel for each flows, can reduce the buffering delay. For the step of finding the time slot which causes minimum buffering delay, the time complexity is $O(n)$, where n is the number of time slot.

Algorithm : TIME_SCHEDULE

Input : f

Output : f Scheduling

```

1. Begin
2.   For int  $i = 1$  TO  $S$ 
3.     If Buffering Delay is Minimum then
4.       return  $Slot\_N$ 
5.     End If
6.   End For
7. End
    
```

Figure 13: TIME_SCHEDULE

V. SIMULATION RESULTS

In this section, the simulation results of the proposed RPADS-1 algorithm are presented. We develop various scenarios and network topologies in our simulation. The supported number of calls, the total bandwidth consumption, and the buffering delay of the proposed RPADS-1 algorithm are compared with different strategy.

We used C# as the simulation tool. The random network topology in the simulation is generated by the following rules. Firstly, each MSTAs has the equal in distance with adjacent MSTAs. In order to avoid would be unable to find a path between any two nodes, so that the generated network topology must be a connected graph. After create such connected graph, to prevent that a routing bottleneck, the degree of each MSTAs should be restricted and the value are set as 3 and 4 in the simulation. Two examples are shown in Figure 14. In the Figure 14(a), noted that the left side of the node (node A and node B) to send packets to the right side of the node (node E and node F) has to go through the link from node C to node D. In order to avoid a routing bottleneck when the link from node C to node D has broken, we established a connection between any two nodes which degree less than 3 until the degree of the each node equal to 3, such as Figure 23(b).

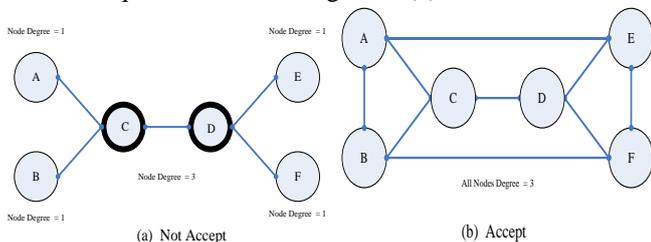


Figure 14: An example of establishing link between each node

The parameters of the network topology including the Maximum Degree of MSs, and Network Size are all can be changed. The detailed parameters related to network topology listed in Table IV.

TABLE IV. PARAMETERS USED IN THE PERFORMANCE EVALUATION

Parameter	Values
Network Size	16 to 25
Maximum degree of MSTAs	3, 4
MAC type	IEEE 802.11s MCCA
Data rate	11Mbps
Transmission Rate	802.11b 11Mbps
RTS-CTS	Disabled
Traffic type	CBR
Voice Codec	G.729
MTU	2312 bytes

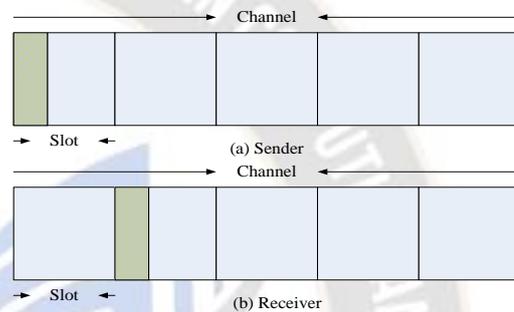


Figure 15: An example of buffering delay

For the simulation results, we mainly to evaluate the following criteria:

- (1) VoIP Capacity: The VoIP capacity is defined as the number of VoIP sessions that can be supported over WMN.
- (2) Buffering Delay: The buffering delay is defined as the waiting time in the queue as shown in Figure 15. In the Figure 15, the time of sender transmits a packet to receiver needs a time slot. We assume the time of redundant frame transmission is 30ms and for a VoIP packet, the header overhead OH_{hdr} consists of the headers of RTP, UDP, IP, and 802.11 MAC layer:

$$OH_{hdr} = H_{RTP} + H_{UDP} + H_{IP} + H_{MAC} \quad (2)$$

Besides, at the MAC layer, the overhead incurred at the sender is:

$$OH_{sender} = DIFS + averageCW + PHY \quad (3)$$

But we do not consider the receiver will return an ACK, so the overhead incurred at the receiver is:

$$OH_{receiver} = SIFS \quad (4)$$

Therefore, we have the transmission time T as below:

$$T = (Payload + OH_{hdr}) * n * 8 / data\ rate + OH_{sender} + OH_{receiver} \quad (5)$$

The values of DIFS, PHY, SIFS, and ACK for 802.11b are listed in Table I. Assuming that G.729 is used, payload is 20 bytes and n packets are aggregated into one packet. In addition to, assume the size of packet aggregation equal to maximum transmission unit. Therefore, a time slot at least requires 2297.3 μ s at 11

Mbps. Then a channel can divide to $30\text{ms} / 2297.3 \approx 13$ slots.

$$\text{Buffering Delay} = \begin{cases} \sum_{i=1}^{k-1} \text{Slot}_{N_{i+1}} - \text{Slot}_{N_i}, & \text{Slot}_{N_{i+1}} > \text{Slot}_{N_i} \\ \sum_{i=1}^{k-1} 13 - \text{Slot}_{N_{i+1}} - \text{Slot}_{N_i}, & \text{Slot}_{N_{i+1}} < \text{Slot}_{N_i} \end{cases} \quad (6)$$

(3) Bandwidth Consumption: The total bandwidth consumption is defined as the produce of all success number of the calls (abbr. SC) and average length per flow (abbr. ALF) in the network.

$$\text{Bandwidth Consumption} = \text{SC} * \text{ALF} \quad (7)$$

In Figure 16 and Figure 17 show the simulation outcome of VoIP capacity with RPADS-1, RPADS-2 and without packet aggregation scheme under different network size. In these figures, the blue bars are the results of RPADS-1 algorithm, purple bars are the results of the RPADS-2 algorithm and the yellow bars are the results of the without packet aggregation scheme. It can be clearly seen that the VoIP capacity with RPADS-1 algorithm is much better than the RPADS-2 algorithm and without packet aggregation scheme in each cases with the maximal degree of MST A equal to 3 and 4. It is inevitable because the packet aggregation merges a large of packets that have the same next hop to reduce MAC and PHY header overhead, so that RPADS-1 can more effective than without packet aggregation scheme. As seen in the numerical results, in the best case the VoIP capacity of the RPADS-1 algorithm is more one thousand of calls than the RPADS-2 algorithm.

In Figure 18 and Figure 19 show the average length of flow in RPADS-1 and RPADS-2 algorithm. In our routing strategy, we have a higher priority to choose the shortest length of VoIP call into the network. The maximum average length difference of flow in RPADS-1 algorithm is less than RPADS-2 about 20%. The difference is very small and can get more number of calls in RPADS-1 algorithm than RPADS-2 algorithm.

In Figure 20 and Figure 21 we have to compare of buffering delay versus the network size with different maximal degree of MP 3 and 4 in RPADS-1 algorithm and RPADS-2 algorithm. In RPADS-1 algorithm, we selected a time slot to access channel which caused a minimum buffering delay. But we selected a random time slot to access channel in RPADS-2 and not to consider the buffering delay of this method.

Figure 22 and Figure 23 show the simulation results of bandwidth consumption versus the network size with different maximal degree of MST A 3 and 4. On the other hand, according The largest bandwidth consumption in RPADS-1 algorithm not more than 10% than RPADS-2 algorithm and supported more number of calls as shown in Figure 25 and Figure 26. So in a limit bandwidth condition network, RPADS-1 algorithm can supported more number of calls than RPADS-2 algorithm.

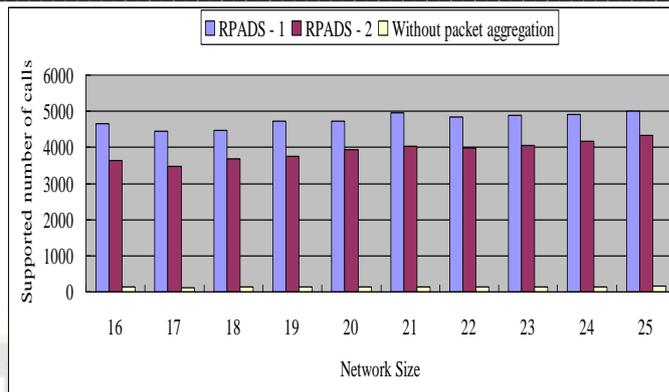


Figure16: VoIP capacity in different network size (max. degree of MST A = 3)

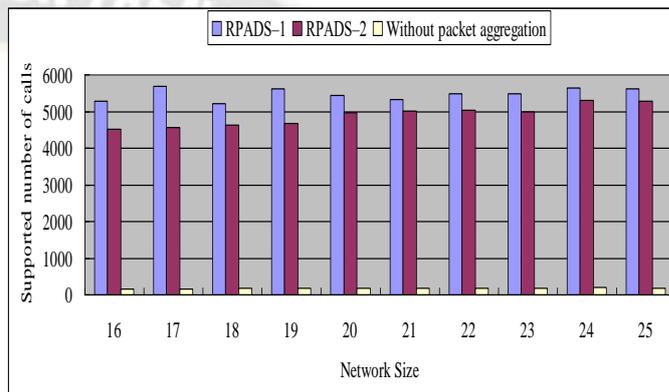


Figure 17: VoIP capacity in different network size (max. degree of MST A = 4)

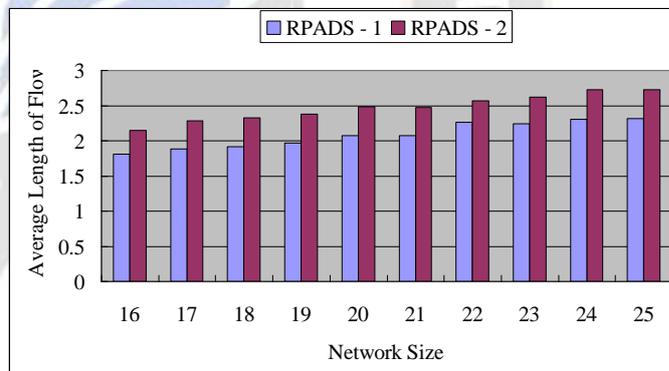


Figure 18: Average flow length in different network size (max. degree of MST A = 3)

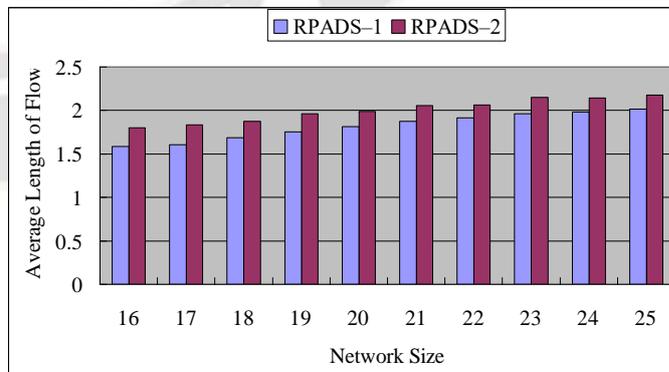


Figure 19: Average flow length in different network size (max. degree of MST A = 4)

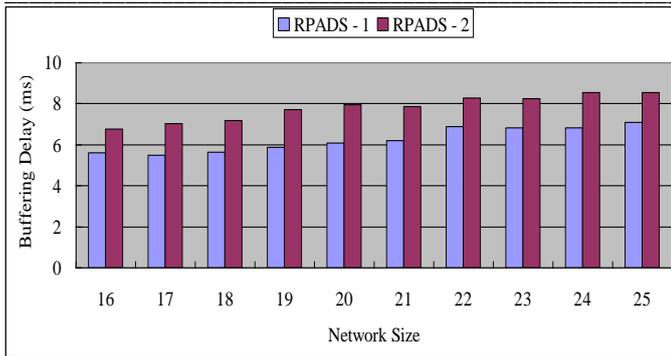


Figure 20: Buffering delay in different network size (max. degree of MSTA = 3)

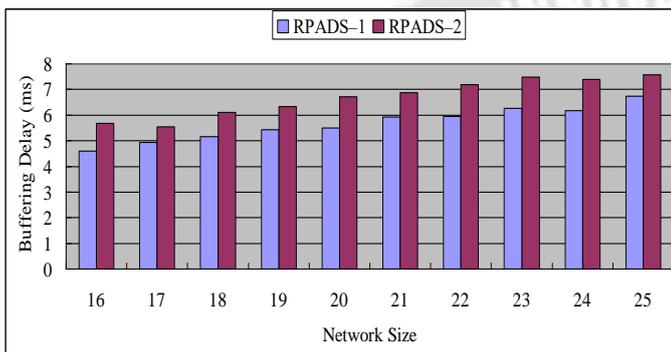


Figure 31: Buffering delay in different network size (max. degree of MSTA = 4)

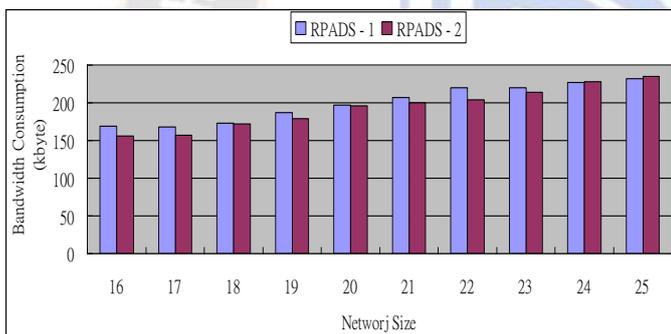


Figure 22: Bandwidth Consumption in different network size (max. degree of MSTA = 3)

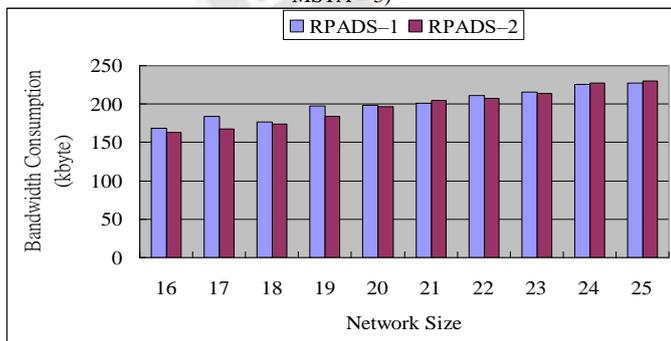


Figure 23: Bandwidth Consumption in different network size (max. degree of MSTA = 4)

VI. CONCLUSION

The paper addresses the VoIP service in multi-hop wireless mesh network with MCCA mechanism and packet aggregation scheme. The effectiveness will be shown by simulation results.

We use 802.11 MCCA mechanism to access channel, it can reserve a fixed time intervals for MSTA transmission. The

quality of a VoIP call is usually sensitive to delay, delay jitter, and packet loss. This method can effectively avoid the significant shake in jitter.

On one hand, we used our proposed RPADS-1 algorithm that supported number of calls has more than without packet aggregation algorithm and RPADS-2 algorithm. On the other hand, whether adopts buffering delay optimization with RPADS-1 algorithm than RPADS-2 algorithm improved about 20% performance. Besides, on our simulation results of bandwidth consumption, in a limit bandwidth condition network, RPADS-1 algorithm can supported more number of calls than RPADS-2 algorithm.

REFERENCES

- [1] Skype, <http://skype.com/intl/zh-Hant/>
- [2] Wei Wang, Soung Chang Liew, and V.O.K. Li, "Solutions to performance problems in VoIP over a 802.11 wireless LAN," Vehicular Technology, IEEE Transactions, Vol. 54, Issue 1, pp. 366-384, Jan 2005.
- [3] David Villacís, Freddy R. Acosta, and Román A. Lara Cueva, "Performance analysis of VoIP services over WiFi-based systems," in 2013 IEEE Colombian Conference on Communications and Computing (COLCOM), 2013.
- [4] IEEE Working Group, "Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE Standard 802.11, 2016.
- [5] D. Niculescu, S. Ganguly, K. Kim, and R. Izmailov, "Performance Evaluation and Delay Modelling of VoIP Traffic over 802.11 Wireless Mesh Network," International Journal of Computer Applications, vol. 21, no. 9, May 2011, pp. 7-13.
- [6] Ji-Young Jung, Hyun-Sik Kang, and Jung-Ryun Lee, "Performance evaluation of packet aggregation scheme for VoIP service in wireless multi-hop network," Ad Hoc Networks, vol. 11, no. 3, May 2013, pp. 1037-1045.
- [7] Houda Zarhouni, Ghizlane Orhanou, Said El Hajji, Redouane Benaini, "Traffic Engineering and Optimization Routing for VoIP Traffic in Wireless Mesh Networks," in proceeding of f the World Congress on Engineering 2012 (WCE2012), July 2012.
- [8] Giovanni Di Stasi, Jonas Karlsson, Stefano Avallone, Roberto Canonico, Andreas Kassler, and Anna Brunstrom, "Combining multi-path forwarding and packet aggregation for improved network performance in wireless mesh networks," Computer Networks, vol. 64, no. 8, May 2014, pp. 26-37..
- [9] Paulo H. Azevedo Filho, Marcos F. Caetano, and Jacir L. Bordim, "A Packet Aggregation Mechanism For Real Time Applications Over Wireless Networks," International Journal of Networking and Computing, vol. 2, no. 1, Jan 2012, pp. 18-40.
- [10] Mosleh M. Abualhaj, Manjur Kolhar, Kefaya Qaddoum1, and Ahmad Adel Abu-Shareha, "Multiplexing VoIP Packets over Wireless Mesh Networks: A Survey," KSII Transactions on Internet and Information Systems, vol. 10, no. 8, Aug. 2016.
- [11] Jonathan Gross, and Jay Yellen, "Graph theory and its Applications," 2nd edition, Chapman and Hall/CRC, 2005.
- [12] Kashyap, A. Ganguly, S. Das, S.R. Banerjee, and S, "VoIP on Wireless Meshes: Models, Algorithms and Evaluation," INFOCOM 2007. 26th IEEE International Conference on Computer Communications, pp. 2036-2044, May 2007.
- [13] Xudong Wang, Abhishek Patil, and Weilin Wang, "VoIP over wireless mesh networks: challenges and approaches," ACM International Conference Proceeding Series; Vol. 220, pp. 9, 2006.
- [14] R. G. Cole and J. H. Rosenbluth, "Voice over IP performance monitoring," ACM Computer Communication Review, Vol. 31, April 2001.
- [15] Guido R Hiertz, Dee Denteneer, Sebastian Max, Rakesh Taori, Javier Cardona, Lars Berleemann, and Bernhard Walke, "IEEE 802.11s: The WLAN Mesh Standard," IEEE Wireless Communications, vol. 17, no. 1, Feb. 2010, pp. 104-111.
- [16] Benveniste, Mathilde, Tao, and Zhifeng, "Performance Evaluation of a Medium Access Control Protocol for IEEE 802.11s Mesh Networks," Sarnoff Symposium, 2006 IEEE, pp. 1-5, March 2006.