Increasing Supported VoIP Flows in WMNs through Link-Based Aggregation

J. Okech, Y. Hamam, A. Kurien
F'SATIE
TUT
Pretoria, South Africa
okechjr@gmail.com

T. Olwal
Meraka Institute
Council of Scientific and Industrial Research (CSIR)
Pretoria, South Africa

M. Odhiambo
Electrical and Mining Engineering
University of South Africa (UNISA)
Pretoria, South Africa

***Abstract*—As Voice over IP (VoIP) becomes a reality, service providers will be able to offer the service to remote and over populated areas that currently are not or are only partially reached by available Public Switched Telephone Network (PSTN). The combination of wireless mesh networks (WMNs) with VoIP is an attractive solution for enterprise infrastructures; presenting availability and reduced cost for both consumers and service providers. The large number of clients in WMNs leads to increased number of concurrent flows. However, only a handful of these flows reaches their destination while still within the quality of service (QoS) bound for VoIP. This performance degradation can be attributed to protocol overhead, packet collision and interferences. This paper introduces VoIP over WMNs and uses a link based packet aggregation scheme to improve VoIP performance in IEEE 802.11 based WMNs operating under distributed coordinate function (DCF). Simulation results show that the proposed aggregation scheme increases the number of supported flow while also reducing end-to-end delay, jitter, and packet loss of VoIP in WMNs.**

Keywords-component; SNIR; VoIP; WMNs; QoS.

I. INTRODUCTION

Voice over Internet Protocol VoIP) refers to the transmission of voice using IP technologies over packet switched networks. Internet telephony is one of the typical applications of VoIP. As compared to the traditional resource dedicated PSTNs, VoIP provides for resource sharing. Thus, IP based VoIP applications presents a cost effective means of facilitating voice communication. The increasing popularity of IEEE 802.11 based networks in homes and offices also provides a motivation to use wireless VoIP. For example, with wireless local area networks (WLANs) it becomes easier for users to access telephone services anywhere anytime through portable handsets.

Wireless mesh networks (*WMNs*) provide an attractive solution in areas where networks are not easy to install or uneconomical to set up. Lack of proper network structures creates alienated areas called dead zones where there are limited or no network coverage. Thus, *WMNs* technology presents a viable alternative to create an enterprise-scale or community-scale wireless backbone with multiuser wireless VoIP connectivity. However, a major challenge is that as the number of VoIP flows increase in a network so does the number of supported calls drops.

The number of supported client capacity is affected by the network forwarding performance, shared contention and self interference [1]. For IEEE 802.11 based *WMNs*, the main challenge in providing higher packet transfer ratio lies on management of the medium access control (*MAC)* protocol overhead. This overhead is attached to every packet transmitted and therefore consumes significant portion of network bandwidth that can be used to carry additional packets. Thus, the dismal performance associated with channel access protocol and transmission overhead magnifies for small packets such as *VoIP*. This work proposes a link based packet aggregation mechanism that adjusts aggregation packet size based on local link quality to provide guaranteed *QoS* for *VoIP* packets in *WMNs*.

The remaining part of this paper is organized as follows. Section II discusses related work. In section III details of the impact of protocol overhead on *VoIP* call capacity for *VoIP* over *WMNs* are presented. The aggregation algorithms are analysed in section IV. Obtained simulation results are presented and discussed in section V. Finally, section VI concludes the work.

II. RELATED WORK

The problem of transmitting small sized packets in IEEE 802.11 based network has existed for quite sometime. Authors such as Hole and Tobagi [2] found that each Access Point (*AP*) can only support a few *VoIP* flows due to the large overhead of IEEE 802.11 *MAC* in processing small packets. Studies conducted to understand the capacity of *WMNs* in [1] show that the throughput of each node decreases at order $O(1/n)$, where n is the number of hops.

To improve performance of in such networks, several approaches have been proposed for both single and multi-hop wireless networks. However, this work narrows to only the literature in sync with the proposed methodology. The use of packet aggregation to improve performance of *VoIP* application on the network is presented in [3], [4], [5] and [6]. The basic decision for an aggregation algorithm in *WMNs* is the placement of de-aggregation capability. This choice defines the applied packet aggregation approach. There are two basic approaches to packet aggregation: end-to-end aggregation and hop-by-hop aggregation. In *end-to-end* aggregation, packet aggregation takes place at the ingress nodes while the egress nodes do the de-aggregation. The *hop-by-hop aggregation* does

Tshwane University of Technology (TUT) and French South Africa Technical Institute in Electronics (F'SATIE)

aggregation at every node from source to destination. Important parameters for implementing packet aggregation are maximum aggregation packet size and maximum aggregation delay. These parameters can be implemented as fixed, dynamic or a combination of fixed and dynamic at various stages in the network. Thus, suitable mix of these parameters can be made to diversify basic aggregation mechanisms and achieve maximum benefits.

In [3], the use of concatenation mechanism to reduce protocol overhead is proposed. It assumes a network with homogeneous nodes. This assumption presents an inefficient usage of bandwidth. In [4], IP based adaptive packet concatenation algorithm for multi-hop *WLANs* is proposed and simulated. The simulation results reveal that more than double the throughput can be achieved in highly loaded networks but at the expense of increased end-to-end delay. The authors in [5] describe IEEE 802.11 overhead and the importance of packet aggregation in Ad Hoc networks. Two aggregation algorithms are proposed: forced algorithm and adaptive algorithm. The forced algorithm introduces additional delay at every hop from source to destination. The algorithm can result in higher cumulative delay which is not suitable for real-time application. On the other hand, the adaptive algorithm proposed in [5] does not usually have sufficiently enough packets to aggregate to provide good bandwidth savings. The authors in [6] investigate the impacts of aggregating multiple small VoIP streams in wireless networks. The results of the experiment reveal the existence of relationship between number of VoIP calls, output link rate and certain teletraffic metrics. However, the aggregation algorithm used a link rate which is not adjustable to the network situation.

Frame aggregation and optimal frame size adaptation for IEEE 802.11 WLANs are presented in [7] and [8]. In [8], a model for calculating the successful transmission probability of a frame of a certain length is proposed. The results of this experiment show that the levels of network contention only has a minor influence on transmission and that the proposed aggregation outperforms fixed frame aggregation. However, the paper fails to detail out how the frames are delayed. It was developed and verified for single-hop where only self interference is more prominent. These situations do not apply to WMNs. In [7], a method to adapt the frame size dynamically to the channel quality and network contention is presented. By intermarrying end-to-end and hop-by-hop aggregation algorithms, the proposed accretion algorithm exploits the advantages of the two while also routing out their shortcomings. The accretion algorithm uses forced delay at the ingress to collect packets of the same flow and natural media access delay for intermediate nodes. The paper shows that for higher offered load, the optimum frame size increases up to a dropping point. Thus, it is beneficial to reduce the channel rate and packet size to minimize the interference.

III. VOIP OVER WIRELESS MESH NETWORK

When rolling out VoIP services in IEEE 802.11 based WMNs, the main challenge is the satisfaction of users who are already accustomed to high qualities provided by PSTN. Such a quality in WMNs is compromised by the architecture of the IEEE 802.11 and the transmission overhead for small sized *VoIP* calls.

A. *WMNs Architecture*

The general architecture of *WMNs* is a mix of fixed backhaul mesh routers and fixed or mobile mesh clients as shown in Figure 1. Mesh clients can be Wi-Fi enabled *VoIP* handsets, laptops or any other wireless handheld devices and have connections across the *WMNs* to other wireless devices. Communication from these mesh clients go through the local mesh network to other wired or wireless *VoIP* phones, out to the Internet with the help of gateways, or to PSTN through local Private Bag Exchange (PBX) [7]. Wired or wireless phones that extend network coverage are called mesh routers. These routers provide backhaul connectivity at the link level or network layer.

Typical IEEE 802.11 nodes use two main *MAC* access protocols; Distributed Coordination Function (*DCF*) and Point Coordination Function (*PCF*). Although *PCF* offers adequate support for *QoS* needs of real-time traffic, it is uncommon and is almost never deployed. In this work, all wireless nodes are based on IEEE 802.11b Wi-Fi interfaces and running on *DCF* channel access mechanism. Both wired and wireless nodes are uses IP level addressing so as to exclude the problems resulting from routing at the link level. However, the work can be tailored for link layer routing.

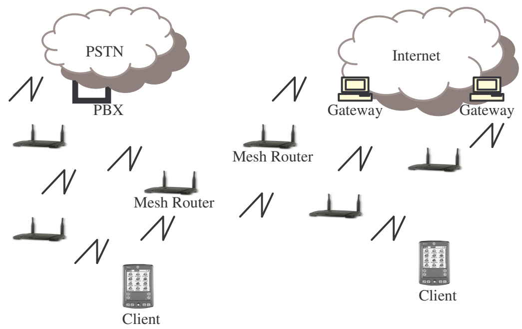

Figure 1. Voice over WMNs. Communication paths are maintained among wireless mesh routers. Each mesh router has enough interfaces to connect to clients and backhaul. Clients can connect to fixed wireless client, internet or to PSTN through the PBX.

B. *Overhead in IEEE 802.11 based WMNs*

VoIP systems use codecs to harmonise interactions between the digital and analogue worlds. The codec interface receives analogue voice, converts it to packets and releases them at a defined rate. To date, there are several vocoders available in the market such as G.711, G.723, GSM and G.729A each coming with its pros and cons. Notably, G.729A is increasingly becoming more popular.

For correctness, this study uses the behaviour of G.729A codec for the generation of VoIP packets. However, the general issues addressed in this paper are also applicable to other codecs. When using G.729A [9], a voice payload of 20 bytes is generated at a rate of 50 packets every second. Therefore, after

40 bytes IP/UDP/RTP header is added, the minimum channel capacity needed to support a voice stream in one direction is 24 Kbps for 11 Mbps channel. This capacity is equivalent to about 229 VoIP calls. However, experimental and analytical results indicate that there is low VoIP call capacity. The decrease in capacity can be attributed to the larger aggregate time spent by network in sending headers and acknowledgements, waiting for inter-frame separations, and contending for the medium. For example, 20 bytes VoIP payload contributes 14.5 μs at 11 Mbps but IP/UDP/RTP header, MAC headers and physical headers, trailers, inter-frame periods, Back-off and acknowledgements (ACK) need a total of 818 μs [7]. The contribution of the VoIP payload increases the transmission time to 832.5 μs. The number of supported calls is calculated using the formulae below.

$$(2.\beta.\alpha)^{-1}, \qquad (1)$$

where β, is the number of packets generated by a coder per second and α is the total transmission time for *VoIP* payload overheads. This yields only about 12 *VoIP* calls supported per hop. The calculations above reveal that per-frame overhead in the IEEE802.11 standard significantly limits the capacity of *VoIP* over *WMNs*.

Apart from high protocol overhead, providing voice services over *WMNs* faces other technical challenges based on the nature of VoIP traffics and behaviour of WMNs. VoIP has strict *QoS* requirements and this gets threatened in WMNs as chances for packet loss is more profound in channels with more interference. Channel interferences increase with increase in number of flows, a characteristic common in *WMNs*. Because packet aggregation reduces packet overhead, it is imperative to note that it can be used to improve the performance of *VoIP* over *WMNs*.

IV. PACKET AGGREGATION ALGORITHMS

Aggregation algorithms entail the process of assembling and forwarding of packets with similar destination called aggregation target and eventual recovery of the original packets at the target as shown in Figure 2.

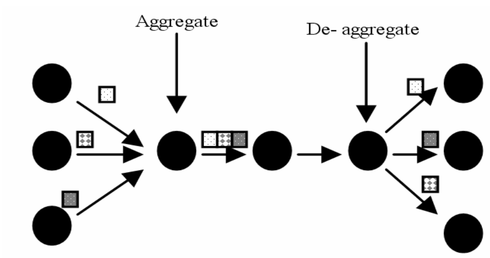

Figure 2. Packet Aggregation

The process of assembling multiple small packets into a single packet is called packet aggregation when it operates at IP-level and frame aggregation or frame concatenation when it operates at *MAC* level. Packet assembly is usually done closer to the source of traffic with the aggregate packet forwarded to an aggregation target. Upon arrival at the target, the original small *VoIP* packets are recovered from the aggregate packet. This recovery process is known as fragmentation or de-aggregation depending on the layer in which aggregation is done.

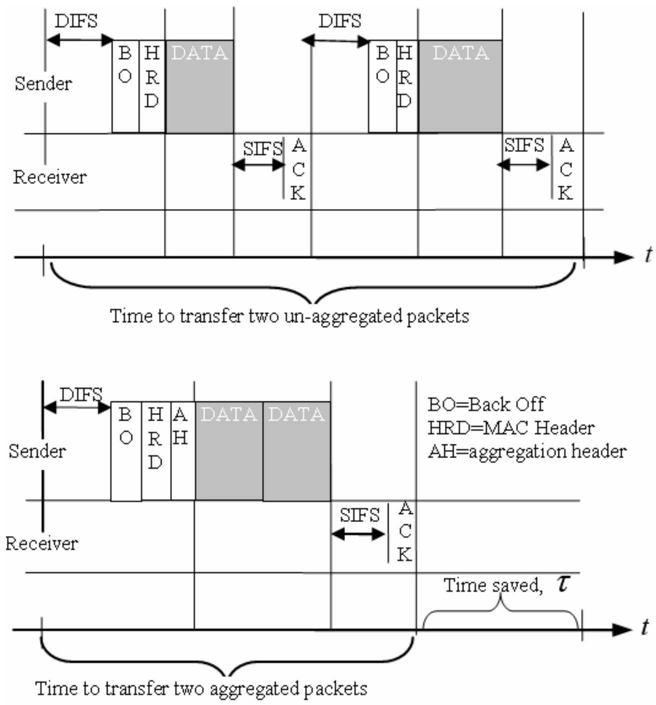

Figure 3. Aggregation of two packets

Packet aggregation can be adopted to boost the throughput of IEEE 802.11 based *WMNs*. Figure 3, shows the transfer of two packets with and without aggregation. It is found that *VoIP* packet takes *832.5 μs* and *1665 μs* transfer times for one and two packets respectively. When the two packets are aggregated, it takes only 84 *μs* to transfer them together, which is about 50% time saving. Thus, only a small number of *VoIP* packets can be supported in *WMNs* since a good portion of the bandwidth is taken by the protocol overheads.

To illustrate the benefit of packet aggregation, assume that packets of the same size ρ *bytes* are transmitted at a channel rate of are transmitted at a channel rate of λ *Mbps*. The benefit of aggregating κ *packets* during transmission can be determined by calculating the difference between transmission with aggregation and without aggregation. The saved time τ (*seconds*), can then be expressed as follows.

$$\tau = \tau_0.(\kappa - 1) - \frac{8.\gamma}{\lambda}, \qquad (2)$$

where γ denotes the size of aggregation header and τ_0 is the channel time. Since λ and γ may be assumed constant for IEEE 802.11b based *WMNs*, by inspecting equation (2), it can be noted that the aggregation benefit, τ, increases with increase in the number of packets. Although this implies that "the larger the aggregation size the better", the implementation prompts for further considerations on end-to-end delay, delay variance and packet loss parameters which are crucial for quality *VoIP*.

When aggregating, an extra overhead of *20ms* is usually added to the first packet. This makes it illogical to use aggregation in lightly loaded networks. However, under a heavily loaded network, which usually happens in *WMNs*, the small packets experience heavy contention. The increased contention causes voice packets to drop or be retransmitted resulting into increased network traffic. In such networks, packets have to be queued while waiting for media access.

A. The Fixed Packet Aggregation Algorithm

This is also called forced-delay aggregation algorithm. The algorithm marks arriving packets with a timestamp. The marked packets are then delayed for a pre-defined time called maximum delay period (δ). After the expiry of δ packets destined to same next hop are aggregated. The size of the aggregated packet is however limited by the maximum transmission unit (*MTU*), which is 2300 bytes for IEEE 802.11 standard [10]. The right choice of δ is important. Higher delays yield a higher aggregation rate, but also a higher end-to-end delay. In this work, MTU and δ has been fixed at 1500 bytes and 10 millisecond respectively.

Packet aggregation is done by first collecting all packets having same next hop. This is implemented at the outbound queue in the *MAC* layer. Nodes capable of aggregation maintain virtual queues; each for one out-links. These queues temporarily keep packets as they wait to be aggregated. When a node is idle, it checks each link's queue in a round-robin manner if it's ready for aggregation. The decision is influenced by two parameters: maximum queue size φ_l, and delay time χ_l. If a link has a queue size greater than φ_l or a head-of-line packet timestamp indicates it is χ_l old, then the packets in the queue are aggregated. During this time, *VoIP* packets are packed together until the size of the new packet becomes larger than *MTU* or the queue becomes empty. If no queue satisfies the conditions, the node stays idle. This releases the wireless channel to be used by other nodes. The two parameters, φ_l and χ_l, are related by equation

$$\varphi_l = \beta.\chi_l , \tag{3}$$

where β, is the average input rate of link l. When l is given, the primary problem is to determine how to choose χ_l for each wireless link. The packet aggregation rate of link l is defined as

$$\psi_l \equiv 1/\chi_l . \tag{4}$$

Here, the optimal value of equation (4) minimizes packet delay in *WMNs*. However, the optimal value for ψ is constrained by flow conservation (*FC*), Capacity limit (*CL*) and *MTU* size properties. The *FC* property emphasizes that the incoming data rate of a link is equal to the outgoing data rate. This data rate is also the aggregation rate. The capacity constraint ensures that the utilized capacity is no more than the capacity that the channel can offer. As for the *MTU* size, the aggregated packet size should not exceed *MTU.*

B. The Proposed Packet Aggregation Algorithm

VoIP call capacity is determined by the packet that meets VoIP QoS constraint. By reducing packet loss occasioned by bit errors while transmitting aggregated packet, the VoIP call capacity can be improved. This algorithm aims at dynamically readjusting maximum aggregation packet size to maximize the number of flows accommodated in WMNs.

Since aggregation aims at achieving higher capacity by combining smaller packets, in the proposed algorithm, the packet rate formulation narrows down to determining the maximum packet size that would optimize equation (4). For a given channel quality, contention level and traffic injection rate, different packet sizes produce different packet loss ratios. To minimize this loss, it is desirable to determine the optimal frame size. Packet loss in *WMNs* is dependent on the bit error (*BE*), queue overflows, and collisions. Packet loss due to collision and queue overflows can be reduced by increasing packet sizes. However, larger packets increase packet loss due to *BE.*

The *BE* occurs when a received signal cannot be decoded properly. The extent of *BE* called bit error rate *(BER)* is dependent on the modulation scheme, signal-to-noise and Interference ratio (*SNIR*) of the received signal, the coding scheme and data rate [11]. Here, apart from *SNIR,* other factors are usually constant in IEEE 802.11b based networks. The *BER* is therefore only dependent on *SNIR.* According to [12], *SNIR* can be defined as

$$SNIR = 10\log_{10}\left(\frac{P_s}{P_n}\right) , \tag{5}$$

where P_s, is the strength of the signal and P_n is the strength of noise produced by thermal noise and interference. Therefore, by defining the following variables: a relationship $D_i = (1-\alpha(\beta,R_i))^{L_i}$, $D_j = (1-\alpha(\beta,R_j))^{L_j}$ and $D_k = (1-\alpha(\beta,R_k))^{8.L_k}$, between frame error rate (*FER*) and *BER* may be expressed as follows.

$$FER = 1 - D_i.D_j.D_k , \tag{6}$$

where, α is the *BER*, β is the *SNIR* value, R_j is the transmission rate of preamble, R_i is the transmission rate of physical layer control protocol (*PLCP*) header, R_k is the transmission rate of *MAC* frame, L_j is the length of the preamble bits, L_i is the length of *PCLP* header in bits and L_k is the length of *MAC* frame in bytes.

If the lengths of the preamble and header, and transmission rates are considered to be constant, the *FER* is a function of *SNIR* and the frame length. For any network, as the *SNIR* goes to infinity the average error rate goes to zero. This means that the network becomes more accommodative to larger packets as the *SNIR* gets higher. Figure 4 illustrates the relationship between packet size and *SNIR* assuming IEEE 802.11 standard overheads

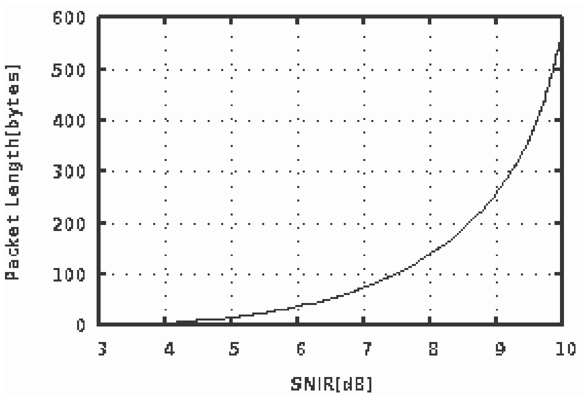

Figure 4. Correct Packet length for a given SNIR [13]

With these arguments, an optimal packet determination scheme can be developed as a function of *SNIR*. The scheme should incorporate the sender and the receiver handshake. The receiving node measures the *SNIR* of the coming packets, calculates the maximum tolerable packet size based on the current *SNIR* and transmits the calculated value to the sender. The current *SNIR* value (S_k) for each link is calculated and stored in the routing table. The formula used is

$$S_{k+1} = S_k + \alpha(S_m - S_k) \tag{7}$$

where S_k defines *SNIR* value before receiving the current packet, S_m is the *SNIR* of the incoming packet and α is the smoothing factor defined by the equation $0 < \alpha < 1$. Since static *WMNs* are stable, the value of α is adequate. In this work, α is chosen as 0.1.

V. PERFORMANCE EVALUATION

In this section, the performance of the DA is evaluated in terms of end-to-end delay, jitter and packet loss rate of VoIP packets under different number of concomitant flows. The results are compared with those obtained without aggregation and under fixed aggregation scheme. The ns-2 simulation environment is used. The variation of the number of parallel flows by use of injected flows model different degrees of network contention and interference. This aids in understanding the performance of the proposed algorithm over real mesh network deployments.

The ns-2 simulator does not come with an already developed VoIP traffic agent. In this paper, a bidirectional VoIP conversation with silence suppression is modelled as an on-off Markov process. The traffic flow is assigned a talk spurt of 35% and silent periods of 65% as typical with G.729A vocoder. VoIP is transmitted over UDP/RTP/IP protocols to form a total packet size of 60 bytes.

Figure 5 illustrates the network topology used in the simulation. It comprises of mesh clients that are either wired or wireless, access points (AP) that provides access to the Internet and wireless mesh routers to extend the coverage of APs. This arrangement of nodes replicates the current single radio networks where the closest gateway is usually no more than two hops. The network assumes that there is only one AP in the network. All wireless nodes are based on IEEE 802.11b with DCF channel access mechanism and RTS/CTS are disabled since they reduce network performance for small packets. Nodes in the network are configured for hierarchical routing.

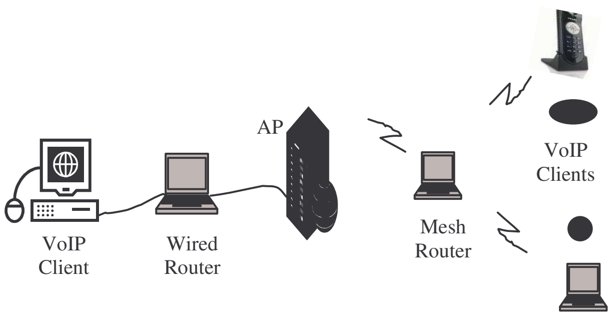

Figure 5. Simulation topology

Simulation results are reported in Figures 6, 7, 8 and 9. The plotted values are obtained by varying concomitant flows per simulation that lasts for 150 seconds, then the average end-to-end delay, jitter and packet loss for the current simulation are calculated and plotted against the injected flows. For each performance metric, a maximum value is seen beyond which performance begin to degrade rapidly. These values correspond to the threshold for supported concomitant flows.

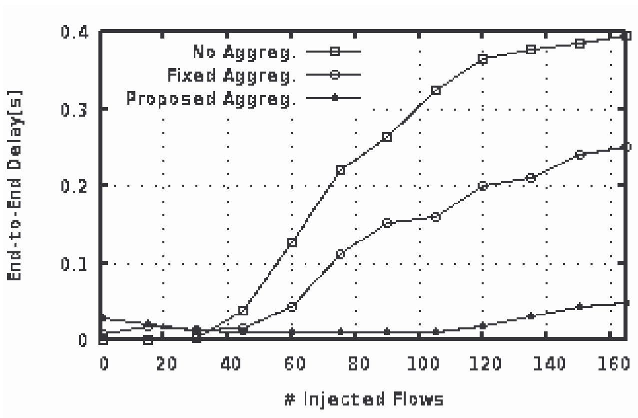

Figure 6. End-to-end delay for VoIP in WMNs

Figure 6 illustrates the end-to-end delay characteristics for three scenarios. Looking at the figure, it can be seen that for low traffic, aggregation algorithms have higher traffic end-to-end delay compared to no aggregation. However, as the number of injected flows increases, more packets get aggregated and thus reducing the average packet delay. The proposed algorithm presents superior performances with a brink experienced from 105 flows compared to 45 and 30 for fixed and no aggregation.

In Figure 7, the relationship between packet end to end jitter and injected flows is presented. From the figure, it can be seen that the use of packet aggregation reduces delay variation. By sending larger blocks of packets, aggregation algorithms reduce chances of having unnecessarily longer queues that causes jitter in the network. The proposed aggregation experiences a brink after 105 flows while fixed aggregation and no aggregation have their jitter rising from 30 and 25 flows respectively

However, for flows less than 20, no aggregation scenario has superior jitter and end-to-end delay values compared to aggregation techniques as shown in Figures 6 and 7. This is because, for lower traffic some packets are delayed due to the δ delay parameter and queuing. As a result packets require different time to be transferred. If δ is small, most packets will be sent without aggregation thereby demystifying the use of aggregation.

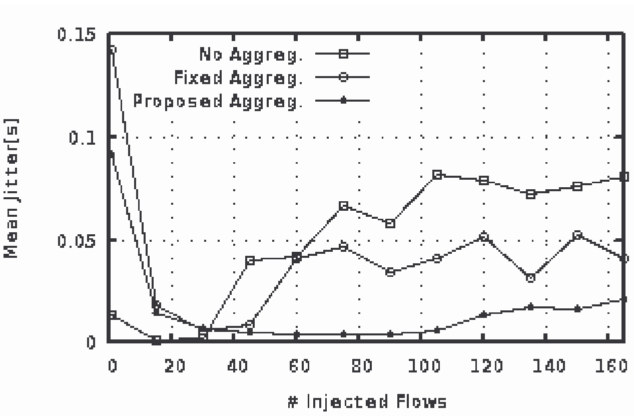

Figure 7. Average delay variation for VoIP packets

Apart from end-to-end delay and jitter, packet loss rate is also a crucial parameter in evaluating network performance. Packet loss includes both packets that do not reach the destination at all or reaches with unacceptably longer delay. Although aggregation techniques uses the media well by transmitting larger blocks of packet thereby reducing contention and overhead, the lager packets have higher chances of being dropped due to frame errors conditions. As illustrated in Figure 8, fixed aggregation that uses an invariable aggregation packet size experiences larger packet loss compared to other techniques. The use of no aggregation experiences higher packet loss as a result of jitter buffer being overwhelmed by large number of packets.

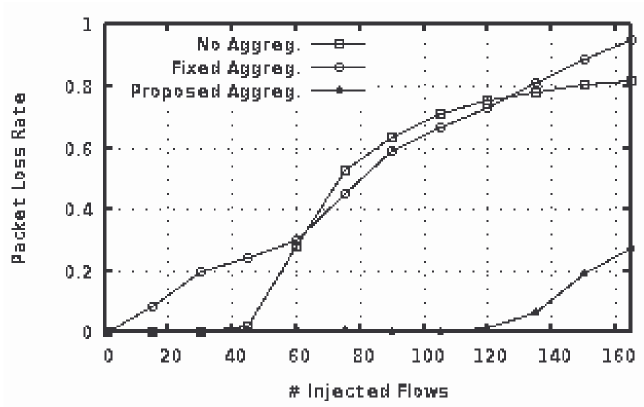

Figure 8. VoIP packet loss rate in WMNs

Figure 9 shows the number of supported flows for each scenario when the number of concomitant flows is varied. Fixed aggregation schemes support the least number of flows and above 30 flows it supports almost null. The proposed aggregation however shows remarkable performance with nearly 90% support for injected flows.

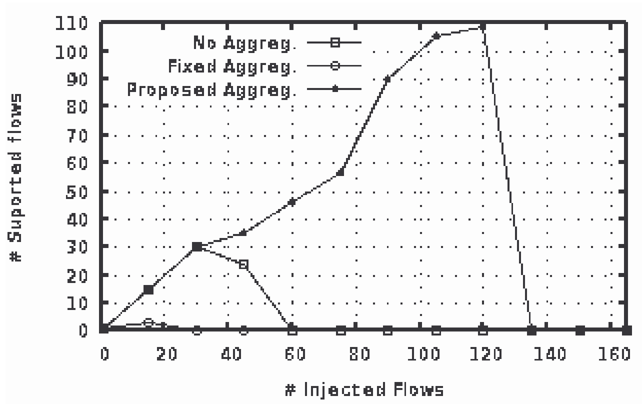

Figure 9. Support Flows versus Injected Flows

The better performance realized by the proposed algorithm is attributed to the ability of the algorithm vary packet size in response to link characteristics. The fixed aggregation algorithm may create packets that are too large to be accommodated in a channel leading to a drop to packet loss. However, even below the capacity threshold it happens that some flows have bad quality. Ideally all flows below threshold are to be supported and this divergence can only be attributed to the difference in confidence levels between flows.

VI. CONCLUSION

This paper has shown that VoIP performs poorly in WMNs. It further proposed a link based aggregation algorithm that adjusts aggregation packet size based on local link characteristics. The proposed algorithm has been simulated and its performances compared with no aggregation and fixed aggregation approaches. The simulation results show that the proposed aggregation scheme yields superior VoIP QoS performance compared to other approaches by increasing the number of supported flows while also reducing end-to-end delay, jitter and packet loss ratio of VoIP packets. Thus, the results have proven that by considering link quality parameters to adjust aggregation packet size, packet *VoIP* performance in *WMNs* can be enhanced.

REFERENCES

[1] J. Jun and M. L. Sichitiu. "The nominal capacity of wireless mesh networks." IEEE Wireless Communications, Oct 2003.

[2] D. P. Hole and F. A. Tobagi, "Capacity of an IEEE 802.11b wireless LAN supporting VoIP," In Proceedings of IEEE Int. Conference on Communications (ICC), 2004.

[3] Y. Xiao, "Concatenation and Piggyback Mechanisms for the IEEE 802.11 MAC," in IEEE WCNC, 2004.

[4] Raghavendra, R., et al., "IPAC-An IP-based Adaptive Packet Concatenation for Multihop Wireless Networks." in Proceedings of Asilomar Conference on Systems, Signals and Computing. 2006.

[5] Jain, M. Gruteser, M. Neufeld, and D. Grunwald, "Benefits of packet aggregation in ad-hoc wireless network," Dept. Comput. Sci., Univ. Colorado, Boulder, CO, Tech. Rep. CU-CS-960-03, 2003.

[6] R. Komolafe, O. Gardner, "Aggregation of VoIP Streams in a 3G mobile network: A Teletraffic Perspective," in European Personal Mobile Communications Conference (EPMCC), 2003.

[7] Ganguly, S., et al., "Performance Optimizations for Deploying VoIP Services in Mesh Networks." Appeared in IEEE Journal on Selected Areas in Communications, Vol. 24, no. 11, Nov. 2006. p. 2147-2158.

[8] Lin, Y. and W.S. Wong, V." Frame Aggregation and Optimal Frame Size Adaptation for IEEE 802.11n WLANs." in Proceedings of IEEE Global Telecommunications Conference. 2006. San Francisco, CA.

[9] ITUT Rec. G.729A (11/96): Reduced Complexity 8kbit/s CS-ACELP Speech Codec.

[10] J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in Proceedings of the 7th ACM International Conference on Mobile Computing and Networking, Rome, Italy, July 2001, pp. 61-69.

[11] S. Mangold, S. Choi, and N. Esseling, "An Error Model for Radio Transmissions of Wireless LANs at 5GHz." in Proceedings of 10th Aachen Symposium on Signal Theory. 2001.

[12] Xiuchao, W., "Simulate 802.11b channel within ns2," National University of Singapore: Singapore, 2004.

[13] W. Liu, Y.Fang, "Courtesy Piggybacking: Supporting Differentiated services in Multihop Mobile Ad Hoc Networks", WINET, University of Florida, 2004