IPAC: IP-based Adaptive Packet Concatenation for Multihop Wireless Networks

Ramya Raghavendra, Amit P. Jardosh, Elizabeth M. Belding-Royer, Haitao Zheng
{ramya, amitj, ebelding, htzheng}@cs.ucsb.edu

***Abstract*— Because medium contention occurs for each packet that is transmitted in a IEEE 802.11 wireless network, transmission of a large number of small packets can be particularly detrimental to performance. As a result of contention overhead, end-to-end delay and energy dissipation increase and the medium utilization decreases. In this paper, our goal is to reduce contention through concatenation of several small packets into a single large packet, and subsequently transmit this large packet. We propose IPAC, an IP-based packet concatenation protocol that adaptively selects an appropriate packet size based on the route quality. Simulation results show that with IPAC, contention is reduced by a factor of two, resulting in a throughput increase by a factor of two to three.**

I. INTRODUCTION

In a multihop network, packets are relayed by intermediate nodes between a source and destination. Each node in a collision domain contends for medium access for every transmitted packet. The transmission of a large number of packets may significantly deteriorate performance due to overhead imposed by medium contention. An increase in contention results in an increase in the MAC overhead, the number of retransmissions, average backoff and aggregate energy dissipation per node in a network that is highly utilized [1].

Previous research has shown that medium contention can be reduced by a number of approaches. These include:

1) Reduction of Retransmission Limit: Medium contention is lowered by limiting the number of retransmissions at the MAC layer [2]. This approach is shown to work well for applications that are resilient to packet losses, such as multimedia applications. However, the approach is detrimental for TCP applications, such as FTP, which cannot tolerate losses. In this case, a MAC layer loss leads to retransmissions at the upper layer resulting in additional delay and bandwidth utilization.

2) Admission Control: Limiting the number of flows in the network reduces the number of contending nodes and results in higher medium utilization [3]. Admission control schemes suffer from the drawback of not being scalable as the delay experienced by a flow waiting for admission increases with the increase in number of contending flows. While this delay is tolerated by certain applications, such as file transfer, delay sensitive applications will suffer performance degradation.

3) Reservation-based Schemes: These schemes exploit the application layer characteristics in medium access control. They utilize the periodicity of transmissions at the application layer to reserve time slots, thereby decreasing medium contention. However, since these applications rely on the periodic nature of applications, they are applicable only to a class of applications, such as VoIP.

One of the primary causes of severe contention and congestion in a multihop network is the transmission of a large number of small packets at each intermediate node. Studies have shown that there is more overhead and power utilized in medium contention than is needed to transmit longer packets and packets should be fairly large to keep the transmission overhead small [4], [5]. Concatenation of small packets is likely to result in reduced medium contention and improved performance. Therefore, the aim of this paper is to reduce contention by transmitting larger packets. In this technique, small packets are aggregated into a large super-packet. MAC contention takes place only once for the single super-packet instead of multiple times for the smaller packets. As a result, a node spends less time in contention and backoff, which leads to better medium utilization, and consequently higher throughput.

In the design of a concatenation scheme, some of the questions that need to be answered are: (1) Is there a significant number of small packets in a wireless network? (2) Does transmission of a large packet significantly increase the likelihood of errors? (3) What is the delay introduced due to queuing and concatenation operations? The discovery of an answer to these questions is non-trivial, however it is essential for a well-designed and practical concatenation protocol.

As an answer to the first question, we observe the packet size distribution from a sample of network traffic traces captured from a large conference wireless LAN. This is depicted in Figure 1. The figure shows that a significant fraction of packets

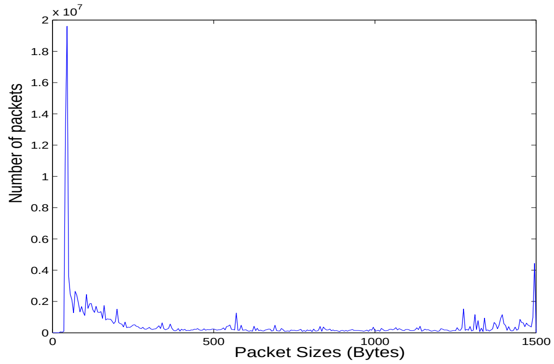


Fig. 1. Packet sizes from traffic analysis of the 61st IETF held in November 2004.

in a large wireless network are small in size. This demonstrates the availability of packets for concatenation.

The second challenge to be addressed by the concatenation scheme is the potential increase in packet loss due to an increase in bit errors and collisions. Our solution uses route quality as a metric to adaptively calculate the maximum size of the concatenated packet. Packets are concatenated only up to an optimal size so as to not increase the packet loss. The third challenge for the concatenation scheme is to minimize the delay introduced by the scheme such that the application constraints can be met. Any concatenation scheme comes with a tradeoff that it introduces latency. We minimize this latency by using an end-to-end concatenation scheme. The delay introduced by the scheme is quantified systematically.

In this paper, we propose IPAC, an IP-based adaptive PAcket Concatenation scheme. IPAC is distinct from previous work in that it is both an IP layer scheme and dynamically adaptive. Through the adaptive calculation of an appropriate packet size, packet loss due to transmission of large packets is reduced. As an IP-based scheme, queuing and concatenation is performed only once at the source, as a result of which the end-to-end delay is minimized. To perform adaptive concatenation, a routing metric is used to obtain an indication of the route quality. The packet size is calculated based on the value of this routing metric. A high quality route implies that larger packets can be sent, whereas a low quality route indicates that larger packets will likely suffer a high loss rate. For this reason, the packet size computation is closely tied to the route selection. The routing protocol chooses the best route based on the routing metric, which is also used to compute the packet size that will be used on that route. The packet size determination is described in detail in Section III.

Simulation studies show that with packet concatenation, the average number of times that a node contends before it acquires the medium for transmission, which we call "Attempts to Medium Access" (AMA), decreases by a factor of two. Consequently, an improvement in throughput by a factor of two to three is observed. A systematic study of the delay introduced by the concatenation scheme in a 3-hop network shows that the end-to-end packet delay increases by only 1.3 to 1.6 times due to buffering prior to concatenation, which is deemed as an acceptable increase in delay for most applications.

The rest of the paper is organized as follows. Section II discusses the related work on packet concatenation. The details of the protocol proposed are discussed in Section III. Section IV provides a detailed evaluation using simulations. Finally, the summary and conclusion are presented in Section V.

II. Related Work

Previous work on packet concatenation schemes can be categorized based on their target network type and the layer at which they operate. One of the first solutions was "Packet Frame Grouping" (PFG) [6]. PFG is a MAC layer scheme for wireless LANs that improves the performance of MAC protocols for multimedia traffic and short packets. The principle behind PFG is to group small packets and share the performance overhead between the grouped packets. This is done by bursting the packets with an SIFS interval between each, once the medium is acquired.

PAC-IP is to date the only existing work on IP layer concatenation [7]. This work notes that the main reason aggregation is not currently implemented at the link layer is because modifying the link layer involves a significant effort for standardization and modification of firmware. Hence, it proposes concatenation at the IP layer in a wireless LAN. The main idea is to concatenate IP packets into a single large "Concatenated Collection", which is considered as an ordinary payload at the link layer. The receiver separates the original IP packets by using the information stored in the MAC header and IP headers.

PFG and PAC-IP were developed for WLAN networks that consist of a single hop. Some of the later work on concatenation describes link layer protocols that target multihop networks. *PAcket Concatenation* (PAC) is a MAC layer scheme for rate adaptive mobile ad hoc networks [8]. PAC dynamically calculates the number of frames to be concatenated as a ratio of the current data

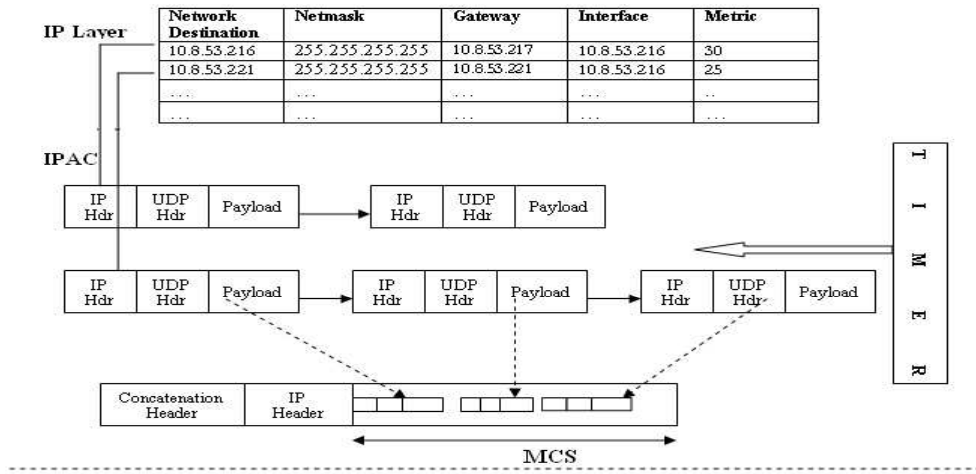


Fig. 2. Queuing and packet concatenation in IPAC.

rate to the lowest supported data rate. *Adaptive Packet Concatenation* (APC) is a distributed MAC layer packet concatenation scheme for multihop sensor and ad hoc networks. APC adaptively concatenates packets using the current transmission rate [9]. The transmission rate is determined by observing the received power of the CTS frame from the next hop.

Concatenation in multihop networks is not a straightforward extension of the single hop case because the link characteristics can change significantly in a multihop path. Hence adaptivity becomes an important requirement of the concatenation protocol. PAC and APC are multihop schemes that adapt the payload size to the link conditions. These schemes function at the link layer. Link layer schemes provide a higher granularity of adaptation to the link dynamics since they can adapt to the link quality at each hop on the path. However, they come with the trade-off that they introduce latency at each hop due to queuing and data copy operations, which in turn affects the performance of delay-sensitive applications. They also slow down the intermediate nodes by increasing the processing load on them.

In this paper, we study the performance of an IP-based concatenation protocol. As per our knowledge, this is the first work that looks at adaptive concatenation at the IP layer. IP-based schemes concatenate packets at the source and deconcatenate at the destination, thus eliminating hop-by-hop packet concatenation delay. However, they come with the trade-off that the payload size calculation is performed once, at the source, as a result of which the adaptation to link variation is more coarse-grained than a link layer approach.

III. PROTOCOL DESCRIPTION

The IPAC protocol functions at the IP layer. The underlying principle of IPAC is that packets that are addressed to a common destination are concatenated before being passed to the link layer. This process is shown in Figure 2. The link layer contends for the medium for this single large packet. Once the IP destination receives this packet, the packet is deconcatenated.

There are multiple important parameters in the design of this protocol. The first parameter is the maximum size of a concatenated packet, which we call the Maximum Concatenation Size (MCS). Each queue is associated with a MCS value that is calculated based on the quality of the route to the destination to which the queue corresponds. The second parameter is the time interval for which packets can be queued at the sender before they are concatenated and delivered, called the Maximum Concatenation Interval (MCI). Using this parameter, we can control the queuing delay introduced at the source and ensure a maximum delay bound due to IPAC.

The following sections provide details on the protocol operations.

Queuing: The sender maintains one queue for each destination that it has a packet to send to. IP packets are queued based on their destination. To reduce the overhead of maintaining queuing information at the source, the queues can be deleted after a specified period of time. Each queue is associated with a MCS value depending on the route quality to the corresponding destination. The timer module controls the maximum packet queuing interval. The timer is set to the desired MCI value based on the delay that can be tolerated by the receiver. A study of MCI values and the associated delay is described in Section IV-C.

Dequeuing: The packets are dequeued at the sender for transmission in one of the two cases:

- The number of queued bytes exceeds the MCS.
- The timer expires. In this case, the packets are dequeued for delivery regardless of the queue size.

After dequeuing, the packets are aggregated into a single super-packet and a four-byte header is added to indicate the number of concatenated packets and the size of each concatenated packet. When there is an incoming packet that cannot be queued because the MCS will be exceeded, the queue is flushed, the dequeued packets are transmitted and the incoming packet is then queued. If the incoming packet size is larger than the MCS, the queue is flushed and the packet is transmitted without being queued. This prevents re-ordering of packets.

After dequeuing, a packet is passed to the link layer, where it is processed as a single IP packet. The MAC protocol now contends for the medium for the super-packet instead of several smaller packets. This super-packet is transmitted to the destination, possibly through multiple intermediate nodes.

Deconcatenation: On receiving a packet, the destination examines the incoming packet and checks for a concatenation header. If this header is not present, the packet is delivered to the transport layer. If a concatenation header is present, then the packet is deconcatented to obtain the smaller packets and the individual packets are delivered to the transport layer. The concatenation header provides the destination with the information needed for deconcatenation.

Adaptive MCS Determination: Calculation of the size of the super-packet is a critical aspect of the protocol. A small payload length will increase the contention rate and consequently decrease the throughput and medium utilization. Large packets reduce contention and increase medium utilization in the presence of a high quality route. However, large packets are prone to bit errors and collisions, thus increasing packet loss if the route is lossy [2]. The transmission of large packets, without considering the channel quality, increases the bit error rate and packet loss. Packet losses result in retransmissions, which further decrease the throughput. Throughput is shown to be a function of packet payload length, as well as the number of retransmissions. It is hence important that the concatenation scheme does not increase the number of retransmissions.

The concatenation size should be adaptive to the channel quality. The sender should determine the optimal payload length that can be transmitted without increasing the packet loss. To compute this, we use the the routing metric "Weighted Cumulative Expected Transmission Time" (WCETT) described by Draves *et al.* [10]. For the sake of completeness, ETT and WCETT calculations are briefly described below. The Expected Transmission Time (ETT) is calculated using the formula:

$$ETT = ETX * \frac{S}{B} \tag{1}$$

where ETX is defined as the Expected Transmission Count, which estimates the number of retransmissions required to send unicast packets by measuring the loss rate of broadcast packets between pairs of neighboring nodes. This is a measure of the link's loss rate. S denotes the size of the packet and B is the bandwidth, so the fraction $\frac{S}{B}$ measures the link bandwidth. Bandwidth is measured using the technique of packet pairs as described by Draves *et al.* [11].

WCETT is a path metric that is calculated as the sum of the ETT's of all the hops on the path. This gives an estimate of the end-to-end delay experienced by a packet traveling along the path based on the loss rate and bandwidth. Thus WCETT for a path with n hops is given by:

$$WCETT = (1-\beta) * \sum_{i=1}^{n} ETT_i + \beta * \max_{1 \leq j \leq k} X_j \tag{2}$$

where k is the number of channels in the network and X_j is the sum of transmission times of hops on channel j. The authors note that the metric is a tradeoff between delay and throughput of the path. The first term gives a measure of latency and the second term represents the impact of bottleneck links. The weighted average strikes a balance between the two.

The WCETT is a measure of the quality of a path and hence it serves as a suitable metric for choosing packet sizes. In a set of paths between a source and destination, the path with the lowest WCETT value is most likely to deliver the maximum number of packets with least delay. Because the path is of high quality, it is likely that large packets, perhaps up to some maximum size, can be sent over such a link without increasing the packet loss rate. We perform empirical evaluations to extract the mapping from WCETT value to packet sizes. This is described in Section IV-B.

The WCETT metric was designed for static multihop networks. IPAC leverages the WCETT routing metric as an indication of route quality to dynamically adapt the packet lengths. Hence the concatenation solution is applicable to a static multihop network. IPAC targets

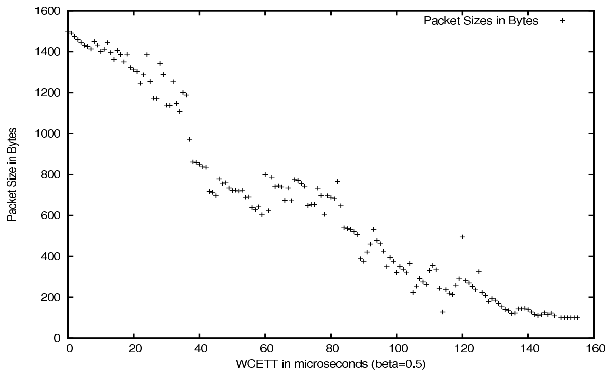


Fig. 3. Mapping of WCETT values to packet sizes.

planned static multihop networks such as a mesh backbone. However, IPAC is itself orthogonal to any routing solution, and is oblivious to mobility in the network.

IV. Performance Evaluation

IPAC has been implemented on the Qualnet simulator. We extended the OLSR-INRIA implementation in Qualnet to incorporate the WCETT metric. The packet concatenation protocol has been implemented to use these WCETT values and adapt the packet size to the routing metric. The performance of IPAC has been evaluated through extensive simulations using Qualnet. The evaluation methodology, simulation environment and results are described in the sections that follow.

A. Evaluation Methodology

The simulations consist of 100 nodes in a 1000m X 1000m area, of which there are 10 pairs of sender-receiver nodes. The nodes are placed in a uniform random topology in the simulation area. The results are an average of five seed values. Each simulation run is for a duration of 200 seconds. Each of the nodes is equipped with a single IEEE 802.11b radio. The RTS/CTS mechanism is turned off in all the simulations expect when mentioned otherwise. The routing protocol used is OLSR, extended to select routes based on the WCETT metric. A value of 0.5 is used for β in Equation 2. This is to give equal weight to total path length and bottleneck links. A β value of 1 will pick a path with the least bottleneck but will not factor in the path length. On the other hand, β=0 will randomly select one path from a set of equivalent paths without considering whether there is a bottleneck link in the path; the throughput will suffer if there is a bottleneck link.

B. Mapping WCETT to Packet Size

A critical aspect of the protocol is to map the WCETT values obtained to a packet size. The efficacy of the concatenation protocol depends on this mapping. An optimal packet size is one that maximizes medium utilization and throughput while avoiding an increase in packet drops. The optimal packet size is directly correlated with the link quality and data rate. Packet sizes greater than the optimal size result in a higher loss rate.

We performed empirical evaluations to obtain this mapping. The application type used is CBR traffic. In each run of the simulation, the application packet size was varied from 100 bytes to 1500 bytes, incremented in steps of 100 bytes. For each of these packet sizes, the WCETT value that is computed is recorded. A data rate of 64 Kbps, typical of voice applications, was used. Packet concatenation was not performed in this experiment.

The goal of the experiment was to determine a possible mapping of WCETT to packet size. Interestingly, for each WCETT value, it was observed that there was a particular packet size above which the throughput decreased due to an increase in packet drops. This confirms that there is a threshold packet size above which the bit error rate increases. This maximum packet size above which a throughput decrease was seen was recorded for each WCETT value. The 90^{th} percentile value of all the simulation runs was computed and mapped to the corresponding WCETT value. Figure 3 shows the results from the simulations. The WCETT values are plotted on the x-axis and the corresponding optimal packet sizes on the y-axis. The packet size decreases approximately linearly with the increase in WCETT. With WCETT under 0.02 ms, packet lengths close to the MTU (1500 bytes) provide maximum throughput. At WCETT=1ms, this optimal packet size decreases to 400 bytes. The mapping extracted from these empirical evaluations is used in the remainder of the simulations.

C. Evaluation Results

In this section, the protocol performance is evaluated using various traffic patterns. Through these evaluations, we can quantify IPAC performance in terms of the benefits it offers and quantify the drawbacks such as delay and overhead. Evaluations are performed with two traffic patterns: CBR and HTTP applications. These applications represent two classes of traffic with different characteristics. CBR, using UDP as the transport protocol, is sent as best effort. It can be used to model voice applications, which are periodic and do not tolerate large delays. HTTP, on the other hand, uses TCP at

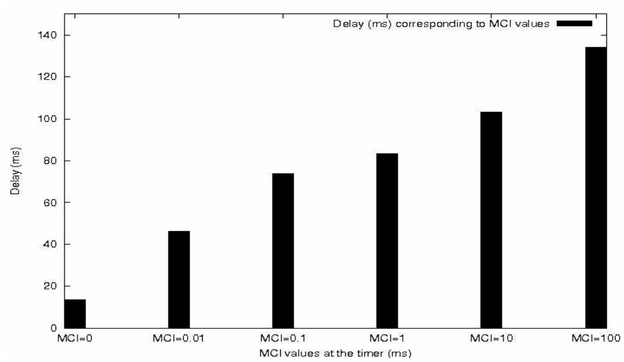


Fig. 4. Effect of varying MCI values.

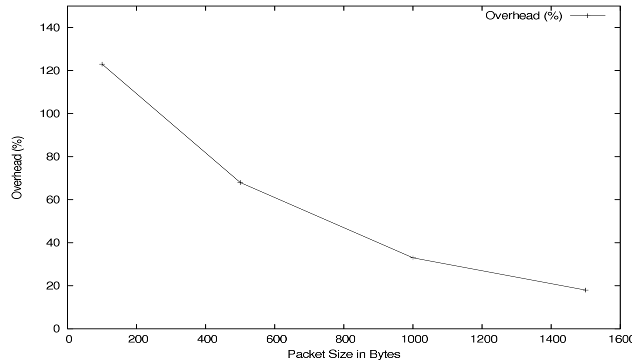


Fig. 5. Evaluation of overhead with packet concatenation.

the transport layer and requires reliability. There is no periodicity of packet transmission and it is more tolerable to delays. The metrics used and the protocol performance are described below. Simulations were performed using the setup described in Section IV-A to evaluate the protocol performance in terms of throughput and delay for both UDP and TCP traffic. The results for both these traffic patterns are shown below.

Effect of varying MCI value: In this experiment, the MCI values are varied to study the end-to-end delay they introduce in the system. This will result in understanding of the delay caused by the timer, so that the timer value can be tuned based on the delay tolerance of the application.

In this experiment, ten random senders transmit CBR packets to ten random receivers. The sending rate is set to 64 Kbps and the packet size is 160 bytes, which is typical of voice applications. The MCI values are varied from 1 μs to 100 ms and the resulting end-to-end delay is plotted. The results are shown in Figure 4. A MCI value of 1 ms results in an end-to-end delay of 83 ms. The delays obtained up to MCI values of 10 ms are within the tolerable delay limits for voice applications using the popular ITU-T G.711 codec.

Although the delay values seen depend on the network topology, the results indicate that for a given topology, a MCI value can be chosen such that the delay caused by IPAC does not adversely impact the application performance. A higher value of MCI will, however, increase the benefits of concatenation.

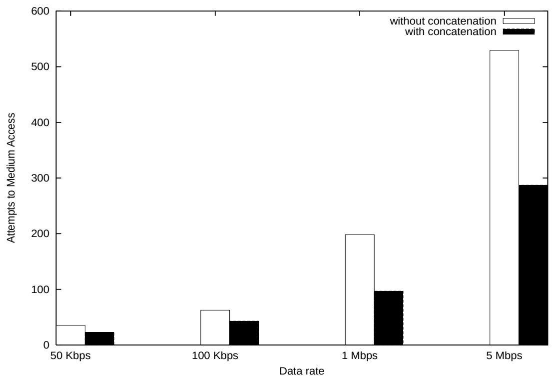
Overhead: A potential drawback of IPAC is the overhead introduced by the concatenation header. Even though a four byte header is added by the protocol, overall there is a significant reduction of MAC layer overhead. IEEE 802.11b adds a 30 byte header at the MAC layer, a 4 byte FCS and a 24 byte PLCP header. This results in 58 bytes of MAC overhead which can be considerable for small packets. With IPAC, the MAC and PLCP headers are added for a single large packet. The MAC overhead without IPAC is even more substantial if RTC/CTS is enabled. With IPAC, RTS/CTS takes place for the single super-packet instead of several small packets. This reduction in overhead becomes significant when the traffic rate is high and a large number of small packets are available for concatenation.

The overhead reduction can be observed from Figure 5. The simulations consist of ten random CBR flows with 160 byte packets and a data rate of 64 Kbps and RTS/CTS disabled. The overhead is calculated as a fraction of the payload and expressed as a percentage.

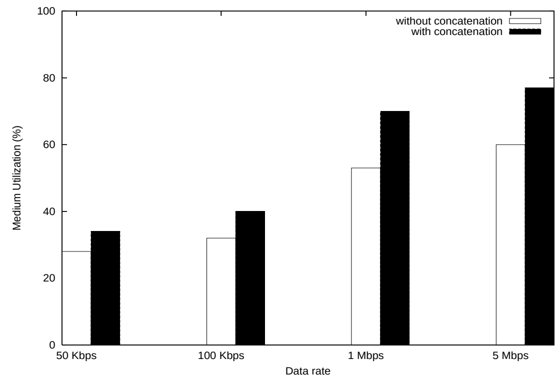
UDP Performance

To study the performance of UDP traffic, ten random senders transmit CBR packets to ten random receivers. The data rates are varied as 50 Kbps, 100 Kbps, 1 Mbps and 5 Mbps. The 50 Kbps data rate results in an underloaded network (27% utilization). The 100 Kbps and 1 Mbps results in a moderate utilization (30-50%) while the 5 Mbps data rate the network is heavily utilized (77%). The results from these experiments are shown in Figure 6. Each of the graphs is further explained below:

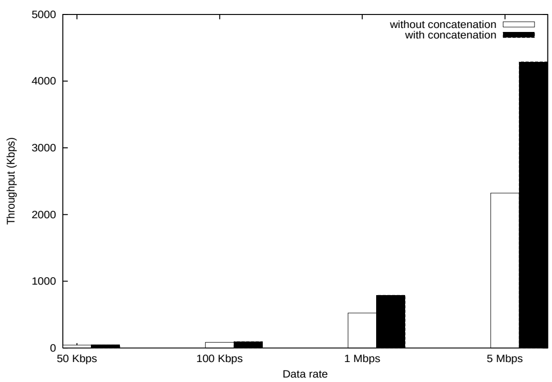
Attempts to Medium Access (AMA): A node contends for the medium for each packet transmission. The “Attempts to Medium Access” metric is a count of the number of times a node contends for the medium for the successful transmission of a packet. This count includes the retransmission attempts. The AMA is an important metric as it translates to the amount of time a node spends in backoff. A higher AMA count indicates that a node attempted a greater number of transmissions. With each unsuccessful transmission attempt, a



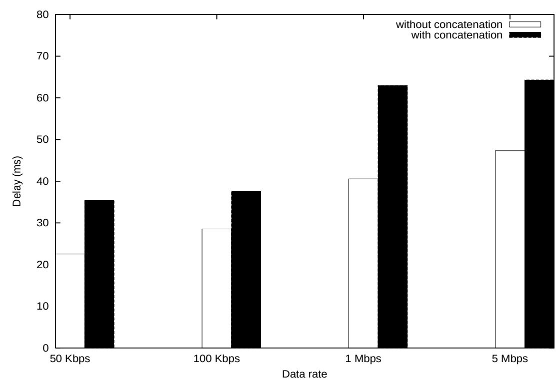
(a) Attempts to medium access



(b) Medium utilization



(c) Throughput



(d) Delay

Fig. 6. Evaluation results of IPAC with CBR traffic over a uniform random static network.

node has to backoff. As per IEEE 802.11, the backoff counter increases exponentially with each retransmission.

AMA has been calculated as the average number of times a node contends for the medium during the entire length of the simulation. As shown in Figure 6(a), there is a decrease in AMA with packet concatenation. When the traffic load is high, as in the 1 Mbps and 5 Mbps cases, the decrease in AMA is significant, approximately 50%. Because of concatenation of small packets into a single super-packet, there are fewer packets to send and the node contends fewer times. This reduces the time spent by a node in contention and backoff. With low traffic loads, there are fewer packets contending and hence the reduction in AMA is not as large.

Medium Utilization: This is measured as a ratio of the time spent by a node transmitting a packet, against the total time spent in transmission and backoff. An increase in medium utilization usually results in an increase in the throughput (unless the medium is congested). Figure 6(b) shows that medium utilization increases due to packet concatenation. This increase is because, with fewer packets to send, the node spends less time in medium contention and backoff. When the medium is acquired, the node transmits as large a packet as can be sent without increasing the bit error rate.

Throughput: The increase in end-to-end application layer throughput is shown in Figure 6(c). Under higher load, a throughput improvement up to a factor of two is obtained. Under low traffic conditions, the nodes are sending fewer packets, the utilization is low and hence there is only a moderate improvement in throughput.

Delay: A potential drawback to packet concatenation schemes is the end-to-end delay. Concatenation involves queuing and data copy operations which introduce delay.

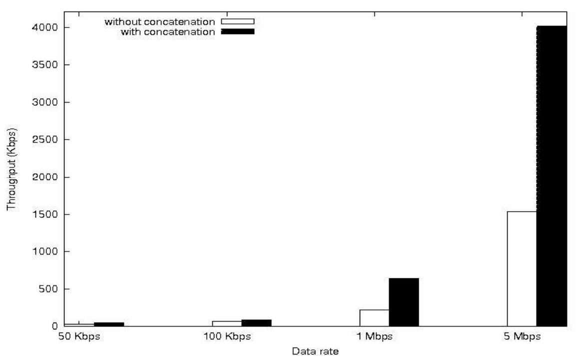


Fig. 7. Throughput comparison with RTS/CTS enabled.

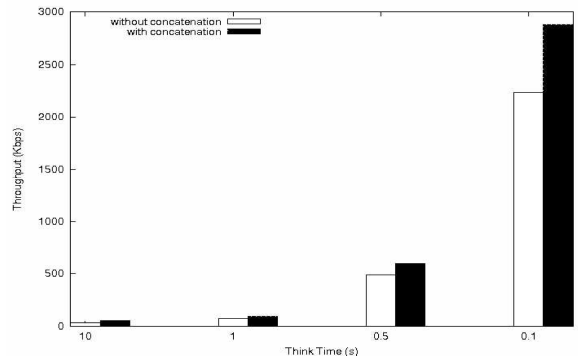


Fig. 8. Throughput measurements with HTTP traffic.

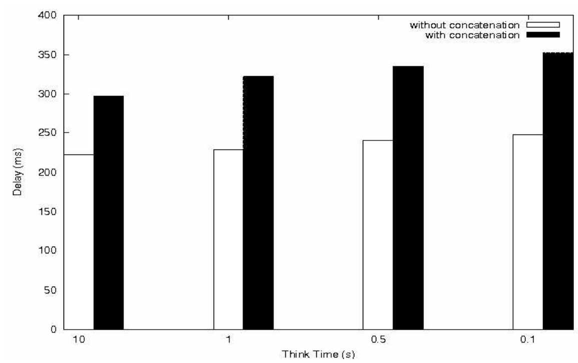


Fig. 9. RTT measurements with HTTP traffic.

The MCI parameter discussed in section III is used to control the queuing delay. In the above experiments the MCI value was set to 10 μs, which was selected based on the results from Figure 4. Figure 6(d) shows the delay introduced due to concatenation in the above scenarios. It can be seen that maximum delay is introduced when the traffic is low. The delay seen under high traffic loads is between 1.3 to 1.6 times the delay seen when there is no packet concatenation. The maximum throughput benefit is also seen at higher traffic loads, implying that concatenation is most beneficial under high traffic conditions.

Effect of RTS/CTS: In this set of simulations, the RTS/CTS mechanism of IEEE 802.11 was enabled to determine its effect on the concatenation mechanism. As can be seen from Figure 7, the throughput improvement with RTS/CTS is greater than the increase observed without concatenation, by approximately a factor of three. With a large number of small packets, there is a substantial overhead due to the RTS/CTS for each packet. With concatenation, this overhead is reduced significantly since virtual carrier sensing is now performed only once, for the single super-packet.

TCP Performance

Evaluating the protocol performance in the presence of TCP traffic is important as the delay can affect the TCP timers, potentially resulting in timeouts. Simulations were conducted using the setup described in Section IV-A. The HTTP traffic model is used, and the *think time* is varied to obtain different data rates. The think time is the amount of time between HTTP requests, which is often the spent by a user thinking, remaining idle or deciding what to do next. Figures 8 and 9 show the results for throughput and RTT measurements. As seen in Figure 8, with concatenation, a throughput increase up to 1.2 times is obtained. The queuing and concatenation delays do not affect the TCP timers adversely. Figure 9 compares the Round Trip Times (RTT) with and without concatenation. The delay is comparable to the delay with UDP traffic, approximately 1.4 times the delay seen when concatenation is not performed.

The results show that the throughput increase with HTTP traffic is modest as compared to the increase with CBR. This is because HTTP traffic does not have a large number of small sized packets available for concatenation; TCP typically transmits MTU sized packets. As voice and video applications become more widespread, we anticipate the transmission of packets smaller than the MTU, in which case IPAC will become increasingly beneficial.

V. Conclusion

Wireless network traffic consists of a large number of packets with small payloads. Frequent medium contention for a large number of small sized packets is expensive, and the medium can be better utilized by sending large packets once the medium is acquired. This paper studies the benefits of concatenating packets at the IP layer and proposes a solution to adapt packet concatenation size based on the route quality. We have shown that there is an optimal packet size corresponding to a route quality which results in maximum throughput. Above this packet size, the packet loss due to bit errors increases. With concatenation, an increase in medium utilization and consequently an increase in throughput is observed. This improvement becomes increasingly significant under high traffic loads. Simulation results show that the throughput and medium utilization can increase by a factor of two to three. As the number of deployed multihop wireless networks increases and voice and video applications become widely used, packet concatenation will be increasingly beneficial to network performance.

ACKNOWLEDGMENTS

This work was supported in part by NSF Career Award CNS-0347886 and by NSF NeTS Award CNS-0435527.

REFERENCES

[1] A. P. Jardosh, K. N. Ramachandran, K. C. Almeroth, and E. M. Belding-Royer, "Understanding Link-Layer Behavior in Highly Congested IEEE 802.11b Wireless Networks," in *Proceedings of ACM SIGCOMM Workshop on EWIND*, Philadelphia, PA, August 2005, pp. 11–16.

[2] S. Choudhury, I. Sheriff, J. Gibson, and E. Belding-Royer, "Effect of Payload Length Variation and Retransmissions on Multimedia in WLANs," in *Proceedings of IWCMC*, Vancouver, Canada, July 2006.

[3] I. D. Chakeres and E. M. Belding-Royer, "PAC: Perceptive Admission Control for Mobile Wireless Networks," in *Proceedings of First International Conference on QSHINE*, Orlando, FL, October 2004, pp. 18–26.

[4] Y.-C. Hu and D. B. Johnson, "Exploiting Congestion Information in Network and Higher Layer Protocols in Multihop Wireless Ad Hoc Networks," in *Proceedings of ICDCS*, Washington, DC, March 2004, pp. 301–310.

[5] J.-P. Ebert, B. Burns, and A. Wolisz, "A Trace-Based Approach for Determining the Energy Consumption of a WLAN Network Interface," Florence, Italy, February 2002, pp. 230–236.

[6] J. Tourrilhes, "Packet Frame Grouping," HP Labs Technical Reports, Bristol, UK, Tech. Rep. HPL-97-132, October 1997.

[7] D. Kliazovich and F. Granelli, "On Packet Concatenation with QoS support for Wireless Local Area Networks," in *Proceedings of ICC 2005*, Seoul, Korea, May 2005, pp. 1395–1399.

[8] K. Yeung, "802.11a Modeling and MAC Enhancements for High Speed Rate Adaptive Networks," Techical Report UCLA, Tech. Rep., 2002.

[9] H. Zhai and Y. Fang, "A Distributed Adaptive Packet Concatenation Scheme for Sensor and Ad Hoc Networks," in *Proceedings of Milcom*, Atlantic City, NJ, October 2005.

[10] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in *Proceedings of the ACM MOBICOM*, Philadelphia, PA, September 2004, pp. 114–128.

[11] R. Draves, J. Padhye and B. Zill, "Comparison of Routing Metrics for Static Multi-Hop Wireless Networks," *SIGCOMM Comput. Commun. Rev.*, pp. 133–144, August 2004.