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

Ramya Raghavendra, Amit P. Jardosh, Elizabeth M. Belding, 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 wireless 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 average backoff, the number of retransmissions, and aggregate energy dissipation per node in a network [1].

The primary cause 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; packets should be large to keep the transmission overhead small [2], [3]. Packet concatenation is a technique in which small packets are aggregated into a large super-packet, thereby allowing a node to send fewer large packets than many small packets. 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.

Before designing a concatenation scheme, one question that needs to to be answered is whether small packets exist in a wireless network. To answer this, we study the traffic traces captured from the 61st IETF held in November 2004 [1]. The IETF network was a IEEE 802.11b-based WLAN consisting of 152 APs (38 physical APs, each supporting four virtual APs) placed on three adjacent floors of the event location and serviced more than 1000 users. Three laptops using Prism2 chipset cards in the RFMon mode were used to sniff traffic and record packet traces on three orthogonal channels 1, 6, and 11. Figure 1 shows the CDF of the packet sizes observed in the network. We observe that a significant fraction of packets are small in size and can be concatenated. We believe that this trace is representative of the various applications that will be run on a wireless network.

Concatenation in multihop networks is not a straightforward extension of the problem of concatenation in single-hop infrastructure LANs. In a single hop network, the sender has complete information about the link characteristics of all the next hop receivers and can use this information to compute the packet sizes. However, in a multihop network, the nodes do not have the complete path information.

To address this problem, 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. 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 reduced. Through the adaptive calculation of an appropriate packet size, packet loss due to transmission of large packets is reduced. To perform adaptive concatenation, a routing metric is used to obtain an indication of the route quality. The packet size is computed based on the value of this routing metric. A high quality route implies that packet errors are low, retransmissions are few and consequently larger packets can be sent. On the other hand, 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 selected 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.

Simulation studies show that with packet concatena-

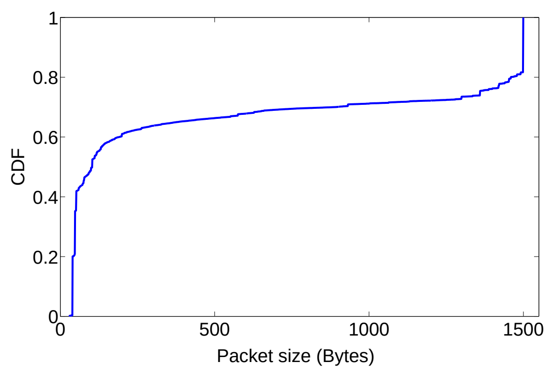

Fig. 1. Packet sizes from traffic analysis of the 61st IETF held in November 2004.

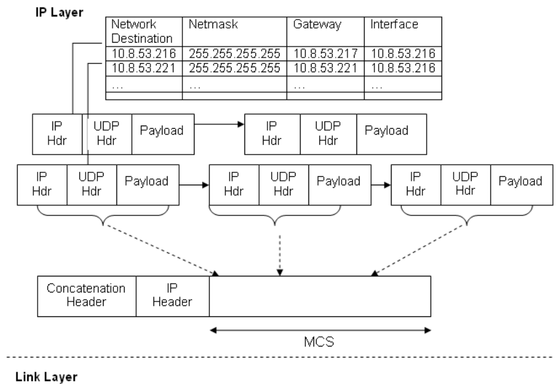

Fig. 2. Queuing and packet concatenation in IPAC.

tion, the average number of times that a node contends before it acquires the medium for transmission 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.

II. Related Work

Previous work on packet concatenation schemes can be categorized based on their target network type: single-hop infrastructure networks and multi-hop ad hoc networks, and the layer at which they operate: link layer or IP layer. One of the first solutions was "Packet Frame Grouping" (PFG) [4]. PFG is a MAC layer scheme for wireless LANs that improves the performance of multimedia traffic by grouping small packets and sharing the performance overhead between the grouped packets.

To date, PAC-IP is the only existing work on IP layer concatenation [5]. 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.

PFG and PAC-IP were developed for WLAN networks that consist of a single hop. Some of the work on concatenation describe link layer protocols that target multihop networks. *PAcket Concatenation* (PAC) is a MAC layer scheme for rate adaptive mobile ad hoc networks [6]. PAC dynamically calculates the number of frames to be concatenated as a ratio of the current data 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 [7].

PAC and APC are link-layer multihop schemes that adapt the payload size to the link conditions. 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 introduce latency at each hop due to queuing and data copy operations. They also slow the intermediate nodes by increasing their processing load.

In this paper, we study the performance of an IP-based concatenation protocol. To the best of our knowledge, this is the first work that studies 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. The trade-off with IP-based schemes is 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. Challenges

The first challenge for the concatenation scheme is to minimize the introduced delay such that the application constraints are met. We decrease this latency by using an end-to-end concatenation scheme. Concatenation takes place only at the source, and there is no concatenation delay introduced at the intermediate routers.

One of the important parameters in our scheme is the time interval for which packets can be queued at the sender before they are concatenated and delivered. This is called the Maximum Concatenation Interval (MCI). Using this parameter, we can control the queuing delay introduced at the source due to IPAC and meet the application constraints. Thus, the delay introduced by the scheme can quantified systematically by varying the MCI parameter, and the effect on the applications can be studied.

The second challenge to be addressed is the potential increase in packet loss due to an increase in bit errors and collisions caused by increase in packet length. Our solution uses route quality as a metric to adaptively compute the maximum size of the concatenated packet. Packets are concatenated only up to an optimal size called the Maximum Concatenation Size (MCS), so as to not increase the packet loss. Each queue is associated with an MCS value that is computed based on the quality of the route to the destination to which the queue corresponds.

IV. 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. The following sections provide details on the protocol operations.

Queuing: The sender maintains one queue for each destination to which it has a packet to send. IP packets are queued based on their destination. 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 V-C.

Dequeuing: At the sender, the packets are dequeued 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 node has to contend for the medium for the single super-packet, instead of several smaller packets. This super-packet is transmitted to the destination, possibly through multiple intermediate nodes. Each of the intermediate nodes have to forward the super-packet instead of several small packets, thus significantly reducing the contention.

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: Computation of the size of the super-packet is a critical aspect of the protocol. A small payload length will increase the medium contention 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 [8]. 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.

To reduce the packet loss, 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.* [9]. 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:

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

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.

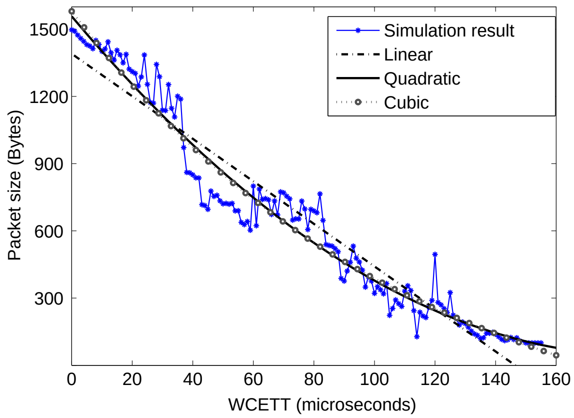

Fig. 3. Mapping of WCETT values to packet sizes, linear, quadratic and cubic fitted curves. (Linear: y = - 9.5x + 1400 Quadratic: y = 0.042x^2 - 16x + 1600 Cubic: y = - 0.00012x^3 + 0.071x^2 - 18x + 1600)

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 V-B.

The WCETT metric was designed for static multihop networks. IPAC leverages the WCETT routing metric as a measure of route quality to dynamically adapt the IPAC leverages the WCETT routing metric as a measure of route quality to dynamically adapt the packet lengths. Hence the concatenation solution is applicable to a static multihop network. IPAC targets planned static multihop networks such as a mesh backbone. However, IPAC is itself independent of any routing solution.

V. 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. This is to give equal weight to total path length and bottleneck links, a value of 0.5 is used for β in Equation 1. 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 IPAC is to map the WCETT values to an appropriate 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 mapping of WCETT to packet size. Simulation results show that for each WCETT value, 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 along with three fitted curves - linear, quadratic and cubic. The x-axis represents WCETT values and y-axis represents the corresponding optimal packet sizes as seen from the experiments. We observe that the linear curve does not fit the graph. The quadratic and cubic curves fit our simulation results, and to avoid overparameterization, we use the equation corresponding to the quadratic curve for the rest of our simulations.

C. Evaluation Results

In this section, we evaluate IPAC to quantify its performance in terms of improvement in throughput and medium utilization and 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. CBR traffic can be used to model voice applications, which are periodic and do not tolerate large delays. HTTP, on the other hand, uses TCP at the transport layer and requires reliability. There is no periodicity of packet transmission and the traffic is more tolerable to delays. The metrics used and the protocol performance are described 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 results in better 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). 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. However, 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 overhead for small packets. With IPAC, the MAC and PLCP headers are added for a single large

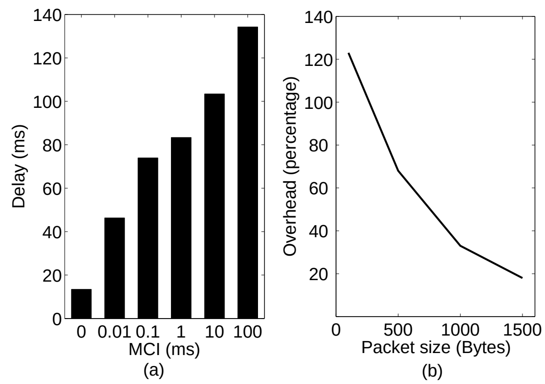

Fig. 4. Evaluation of (a) Delay with varying MCI values (b) Percentage of overhead with varying packet sizes.

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 4(b). 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 Figures 5 and 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 any 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 node has to backoff. As per IEEE 802.11, the backoff counter increases exponentially with each retransmission.

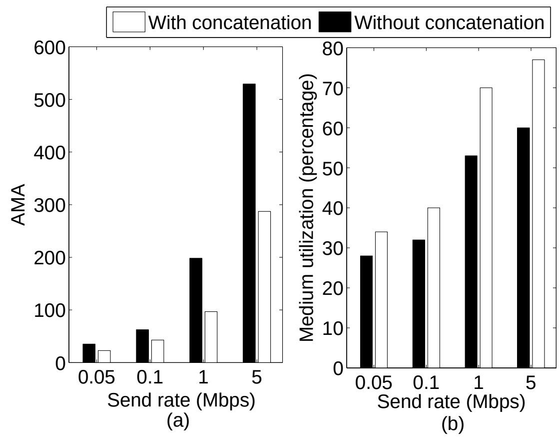

Fig. 5. Comparison of (a) Attempts to Medium Access and (b) Medium utilization with varying rates of CBR traffic.

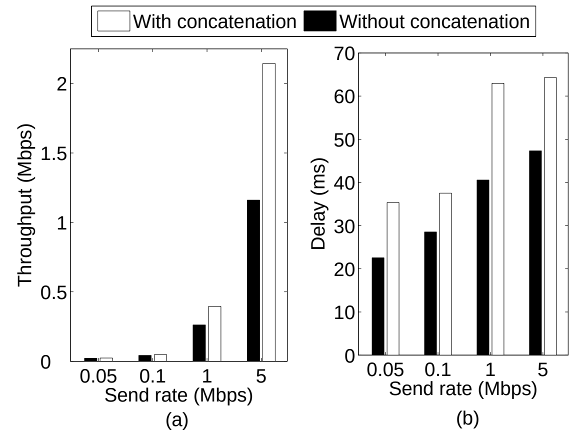

Fig. 6. Simulation results for CBR traffic (a) Throughput (b) Delay.

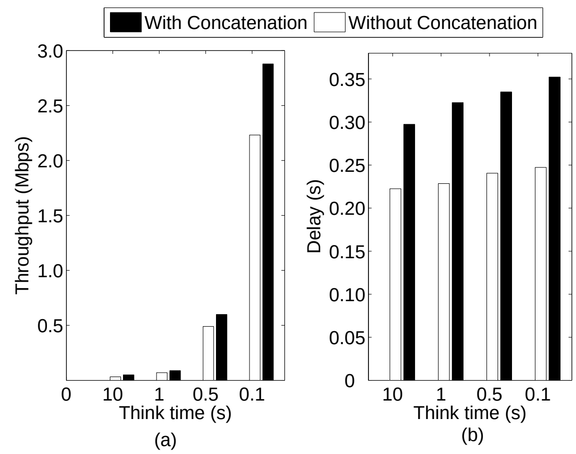

Fig. 7. Results from simulations with HTTP traffic with varying *think times* (a) Throughput measurements (b) Delay measurements

The 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 5(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 5(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(a). 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. The MCI parameter discussed in section IV 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(a). Figure 6(b) shows the delay introduced due to concatenation in the above scenarios. We observe that maximum delay is introduced when the traffic is low. The delay 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.

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. In the next set of experiments, HTTP traffic model is used and the *think time* is varied to obtain different data rates. Think time is the amount of time between HTTP requests, which is often the time spent by a user thinking, remaining idle or deciding what to do next. Figures 7 shows the results for throughput and RTT measurements. As seen in Figure 7(a), 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 7(b) 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.

VI. Conclusion

Wireless network traffic consists of a large number of packets with small payloads. Frequent medium contention for a large number of small 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 packet concatenation 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 that 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] 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.

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

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

[5] 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.

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

[7] 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.

[8] 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.

[9] 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.